Guida per l'utente per la Versione 2

AWS Command Line Interface



Copyright © 2026 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

AWS Command Line Interface: Guida per l’utente per la Versione 2

Copyright © 2026 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

I marchi e l’immagine commerciale di Amazon non possono essere utilizzati in relazione a prodotti o servizi che non siano di Amazon, in una qualsiasi modalità che possa causare confusione tra i clienti o in una qualsiasi modalità che denigri o discrediti Amazon. Tutti gli altri marchi non di proprietà di Amazon sono di proprietà delle rispettive aziende, che possono o meno essere associate, collegate o sponsorizzate da Amazon.

Table of Contents

..... xv
Informazioni su AWS CLI 1
Informazioni sulla versione 2 della AWS CLI 1
Manutenzione e supporto per le versioni principali dell'SDK 2
Informazioni su Amazon Web Services 2
Informazioni sugli esempi 2
Documentazione e risorse aggiuntive 4
Documentazione e risorse per la AWS CLI 4
Altri SDK e strumenti AWS 4
Nozioni di base 6
Prerequisiti 7
Creare un account amministrativo IAM o Centro identità IAM 7
Fasi successive 8
Installazione o aggiornamento 9
Istruzioni per l'installazione e l'aggiornamento di AWS CLI 9
Risoluzione degli errori di installazione e disinstallazione della AWS CLI 29
Passaggi successivi 30
Rilasci precedenti 30
Risoluzione degli errori di installazione e disinstallazione della AWS CLI 46
Passaggi successivi 46
Compilazione e installazione dall'origine 46
Perché compilare dall'origine? 47
Passaggi rapidi 48
Passaggio 1: configurare tutti i requisiti 50
Passaggio 2: configurare l'installazione dell'origine di AWS CLI 55
Passaggio 3: compilare AWS CLI 61
Passaggio 4: installare AWS CLI 62
Passaggio 5: verificare l'installazione di AWS CLI 64
Esempi di flussi di lavoro 64
Risoluzione degli errori di installazione e disinstallazione della AWS CLI 67
Passaggi successivi 67
Amazon ECR Public/Docker 67
Prerequisiti 68
Decidere tra Amazon ECR Public e Docker Hub 68

Esegui le immagini ufficiali ... 68
Note sulle interfacce e sulla retrocompatibilità delle immagini ufficiali ... 70
Utilizzo di versioni e tag specifici ... 70
Aggiornamento all'ultima immagine ufficiale ... 71
Condivisione di file host, credenziali, variabili d'ambiente e configurazione ... 72
Abbreviazione del comando di esecuzione Docker ... 77
Configurazione ... 81
Raccogliere le informazioni sulle credenziali per l'accesso programmatico ... 81
Configurazione di nuove configurazioni e credenziali ... 83
Utilizzo di file di configurazione e credenziali esistenti ... 92
Configurazione di AWS CLI ... 93
Configurazione e precedenza delle credenziali ... 93
Argomenti aggiuntivi in questa sezione ... 94
Impostazioni di configurazione ... 95
Formato dei file di configurazione e delle credenziali. ... 95
Dove vengono archiviate le impostazioni di configurazione? ... 104
Utilizzo di profili denominati ... 104
Definizione e visualizzazione delle impostazioni di configurazione utilizzando i comandi ... 105
Esempi di comandi per impostare nuove configurazioni e credenziali ... 110
Impostazioni supportate del file `config` ... 114
Variabili di ambiente ... 135
Come impostare le variabili di ambiente ... 136
AWS CLI variabili di ambiente supportate ... 137
Opzioni della riga di comando ... 149
Come utilizzare le opzioni della riga di comando ... 149
AWS CLI opzioni della riga di comando globali supportate ... 150
Utilizzi comuni delle opzioni della riga di comando ... 155
Completamento dei comandi ... 155
Come funziona ... 156
Configurazione del completamento dei comandi su Linux o macOS ... 157
Configurazione del completamento dei comandi su Windows ... 160
Tentativi ... 161
Modalità di ripetizione disponibili ... 162
Configurazione di una modalità di riprova ... 165
Visualizzazione dei log dei tentativi di riprova ... 166
Proxy HTTP ... 167

Utilizzo degli esempi ... 167
Autenticazione in un proxy ... 168
Utilizzo di un proxy sulle istanze Amazon EC2 ... 169
Risoluzione dei problemi ... 169
Endpoints ... 170
Impostare l'endpoint per un unico comando ... 170
Imposta un endpoint globale per tutti i Servizi AWS ... 170
Imposta per utilizzare gli endpoint FIP per tutti i Servizi AWS ... 172
Impostato per utilizzare gli endpoint dual-stack per tutti i Servizi AWS ... 173
Imposta endpoint specifici del servizio ... 174
Endpoint basati sull'account ... 177
Configurazione dell'endpoint e precedenza delle impostazioni ... 179
Credenziali di autenticazione e accesso ... 181
Configurazione e precedenza delle credenziali ... 182
Argomenti aggiuntivi in questa sezione ... 184
Credenziali della console ... 184
Prerequisiti ... 185
Accedi alla AWS CLI con il `aws login` comando. ... 185
Esegui un comando con il tuo profilo ... 187
Esci dalla sessione utilizzando il comando aws logout ... 188
Risoluzione dei problemi ... 189
Risorse correlate ... 191
Autenticazione in Centro identità IAM ... 191
Prerequisiti ... 192
Configura il tuo profilo con la procedura guidata `aws configure sso` ... 194
Configura solo la sezione `sso-session` con la procedura guidata `aws configure sso-session` ... 198
Configurazione manuale utilizzando il file `config` ... 199
Accedere a una sessione di Centro identità IAM ... 203
Eseguire un comando con il profilo Centro identità IAM ... 205
Disconnettersi dalle sessioni in Centro identità IAM ... 205
Risoluzione dei problemi ... 205
Risorse correlate ... 205
Concetti sul Centro identità IAM ... 206
Tutorial: AWS IAM Identity Center e Amazon S3 ... 211
Credenziali a breve termine ... 217

Ruoli IAM ... 219
Prerequisiti ... 219
Panoramica sull'utilizzo dei ruoli IAM ... 219
Configurazione e utilizzo di un ruolo ... 221
Utilizzo di MFA ... 223
Ruoli per più account e ID esterno ... 225
Specifica di un nome di sessione del ruolo per semplificare l'audit ... 225
Assunzione di un ruolo con un'identità Web ... 226
Cancellazione delle credenziali nella cache ... 227
Utenti IAM ... 228
Fase 1: creare un utente IAM ... 228
Fase 2: ottenere le chiavi di accesso ... 229
Passaggio 3: configura il AWS CLI ... 229
(Facoltativo) Utilizzo dell'autenticazione a più fattori con le credenziali utente IAM ... 232
Metadati Amazon EC2 ... 233
Prerequisiti ... 233
Configurazione di un profilo per i metadati Amazon EC2 ... 234
Credenziali esterne ... 235
Utilizzo di AWS CLI ... 238
Chiedere aiuto ... 239
Comando di aiuto per la AWS CLI integrato ... 239
AWS CLI guida di riferimento ... 244
Documentazione API ... 244
Risoluzione degli errori ... 245
Aiuto aggiuntivo ... 245
Struttura di comando ... 245
Struttura di comando ... 246
Comandi wait ... 247
Specifica dei valori di parametro ... 248
Tipi di parametri comuni ... 249
Virgolette con stringhe ... 255
Parametri dei file ... 259
Generazione di un modello dello skeleton della CLI ... 263
Sintassi abbreviata ... 277
Prompt automatico ... 280
Come funziona ... 280

Funzionalità di prompt automatico ... 281
Modalità di prompt automatico ... 284
Configurare il prompt automatico ... 284
Controllo dell'output dei comandi ... 285
Output sensibili ... 285
Opzioni di output lato server o lato client ... 286
Formato di output ... 286
Paginazione ... 296
Uscita del filtro ... 302
Codici restituiti ... 326
Procedure guidate ... 328
Come funziona ... 329
Alias ... 330
Prerequisiti ... 330
Fase 1: creazione del file alias ... 331
Fase 2: creazione di un alias ... 332
Fase 3: chiamata di un alias ... 335
Esempi di repository di alias ... 337
Risorse ... 338
Risoluzione degli errori ... 339
Risoluzione generale dei problemi come primo tentativo ... 339
Errori di comando non trovato ... 348
Il comando `aws --version` restituisce una versione diversa da quella installata ... 351
Il comando "`aws --version`" restituisce una versione dopo la disinstallazione di AWS CLI ... 352
Hanno AWS CLI elaborato un comando con un nome di parametro incompleto ... 353
Errori di accesso negato ... 354
Credenziali non valide ed errori delle chiavi ... 355
Errori di firma non corrispondente ... 357
Errori di certificato SSL ... 358
Errori JSON non validi ... 360
Risorse aggiuntive ... 362
Esempi di AWS CLI ... 363
Esempi di comando guidati ... 363
DynamoDB ... 364
Amazon EC2 ... 368

Amazon Glacier 388
IAM 394
Amazon S3 398
Amazon SNS 419
Esempi di comando 421
ACM 428
Gateway API 439
Gateway API HTTP e WebSocket API 506
API di gestione Gateway API 558
App Mesh 560
App Runner 605
AWS AppConfig 640
Application Auto Scaling 674
Application Discovery Service 692
Segnali applicativi 698
AppRegistry 719
Athena 730
Auto Scaling 764
Piani di Auto Scaling 834
AWS Backup 842
AWS Batch 847
Budget AWS 862
Amazon Chime 873
Cloud Control API 918
AWS Cloud Map 925
AWS Cloud9 961
CloudFormation 970
CloudFront 1033
Amazon CloudSearch 1140
CloudTrail 1141
CloudWatch 1159
CloudWatch Registri 1193
CloudWatch Monitoraggio della rete 1199
CloudWatch Observability Access Monitor 1212
CloudWatch Observability Admin 1224
CloudWatch Synthetics 1230

CodeArtifact 1248
CodeBuild 1277
CodeCommit 1341
CodeDeploy 1415
CodeGuru Revisore 1456
CodePipeline 1474
AWS CodeStar Notifiche 1507
CodeConnections 1518
Amazon Cognito Identity 1526
Gestore dell'identità per Amazon Cognito 1532
Amazon Comprehend 1679
Amazon Comprehend Medical 1816
AWS Config 1852
Amazon Connect 1876
AWS Cost and Usage Report 1893
Servizio Cost Explorer 1896
Firehose 1904
Amazon Data Lifecycle Manager 1907
AWS Data Pipeline 1913
DataSync 1923
DAX 1927
Test 1946
Device Farm 1958
Direct Connect 1963
Directory Service 2014
Directory Service Dati 2016
AWS DMS 2042
Amazon DocumentDB 2085
DynamoDB 2143
DynamoDB Streams 2240
Amazon EC2 2247
Amazon EC2 Instance Connect 2927
Amazon ECR 2928
Amazon ECR Public 2960
Amazon ECS 2989
Amazon EFS 3115

Amazon EKS 3124
Elastic Beanstalk 3222
Elastic Load Balancing - Versione 1 3252
Elastic Load Balancing - Versione 2 3280
ElastiCache 3334
MediaStore 3440
Amazon EMR 3457
Amazon EMR su EKS 3506
EventBridge 3507
EventBridge Tubi 3514
Firewall Manager 3522
AWS FIS 3532
GameLift Server Amazon 3550
Amazon Glacier 3584
Global Accelerator 3606
AWS Glue 3645
GuardDuty 3667
AWS Health 3688
HealthImaging 3695
HealthLake 3723
HealthOmics 3736
IAM 3806
Sistema di analisi degli accessi IAM 3948
Image Builder 3984
Strumento di gestione degli incidenti 4026
Contatti di Incident Manager 4049
Amazon Inspector 4073
AWS IoT 4121
AWS IoT Analytics 4303
Device Advisor 4330
AWS IoT data 4345
AWS IoT Events 4348
AWS IoT Events-Data 4374
AWS IoT Greengrass 4399
AWS IoT Greengrass V2 4485
AWS IoT Processi SDK rilascio 4511

AWS IoT SiteWise 4514
AWS IoT Things Graph 4564
Wireless AWS IoT 4591
Amazon IVS 4628
Chat Amazon IVS 4673
Streaming in tempo reale di Amazon IVS 4687
Amazon Kendra 4740
Kinesis 4749
AWS KMS 4768
Lake Formation 4842
Lambda 4894
License Manager 4937
Lightsail 4950
Macie 5075
Grafana gestito da Amazon 5081
MediaConnect 5082
MediaConvert 5099
MediaLive 5123
MediaPackage 5130
MediaPackage VOD 5145
MediaStore Piano dati 5157
MediaTailor 5163
MemoryDB 5168
MSK Amazon 5205
Monitoraggio flusso di rete 5214
Network Manager 5230
OpenSearch Servizio 5268
Organizations 5282
AWS Outposts 5320
AWS Payment Cryptography 5324
AWS Payment Cryptography Piano dati 5345
Amazon Pinpoint 5354
Amazon Polly 5378
Listino prezzi AWS 5384
AWS Private CA 5389
AWS Proton 5396

Amazon RDS 5409
Servizi di dati di Amazon RDS 5607
Approfondimenti sulle prestazioni di Amazon RDS 5611
Amazon Redshift 5626
Amazon Rekognition 5706
AWS RAM 5783
Esploratore di risorse 5807
Gruppi di risorse 5829
Resource Groups Tagging API 5842
Route 53 5846
Registrazione dei domini Route 53 5860
Profili di Route 53 5887
Risolutore Route 53 5899
Simple Storage Service (Amazon S3) 5944
Amazon S3 Control 6037
Secrets Manager 6053
Security Hub CSPM 6082
Security Lake 6161
AWS Serverless Application Repository 6196
Service Catalog 6198
Service Quotas 6229
Amazon SES 6239
Scudo 6252
Signer 6269
Snowball Edge 6278
Amazon SNS 6280
Amazon SQS 6302
Storage Gateway 6323
AWS STS 6326
Supporto 6336
Amazon SWF 6349
Systems Manager 6365
Amazon Textract 6542
Amazon Transcribe 6553
Amazon Translate 6596
Trusted Advisor 6598

Autorizzazioni verificate da Amazon 6618
Reticolo in VPC 6643
AWS WAF Classic 6679
AWS WAF Classic regionale 6685
AWS WAFV2 6691
WorkDocs 6736
Amazon WorkMail 6769
Flusso di WorkMail messaggi Amazon 6793
WorkSpaces 6795
X-Ray 6810
Esempi di script Bash 6828
AWS Batch 6829
AWS Cloud Map 6840
CloudFront 6863
DynamoDB 6872
Amazon EC2 6971
HealthImaging 7128
IAM 7137
AWS KMS 7202
Lightsail 7204
Simple Storage Service (Amazon S3) 7212
AWS STS 7243
Sicurezza 7247
Protezione dei dati 7247
Crittografia dei dati 7248
Identity and Access Management 7249
Destinatari 7249
Autenticazione con identità 7250
Gestione dell'accesso tramite policy 7251
Come Servizi AWS lavorare con IAM 7253
Risoluzione dei problemi di AWS identità e accesso 7254
Convalida della conformità 7255
Resilienza 7256
Sicurezza dell'infrastruttura 7256
Applicazione di una versione minima di TLS 7257
Guida alla migrazione 7259

Utilizzo degli strumenti di migrazione AWS CLI per mitigare le interruzioni 7259
Aggiorna la modalità di debug 7260
AWS Strumento di migrazione CLI da v1 a v2 7260
Scelta tra la modalità di debug dell'aggiornamento e AWS lo strumento di migrazione CLI da v1 a v2 7261
Argomenti aggiuntivi in questa sezione 7264
Nuove funzionalità e modifiche 7264
AWS CLI nuove funzionalità della versione 2 7264
Interruzione delle modifiche tra AWS CLI la versione 1 e la AWS CLI versione 2 7266
Istruzioni di migrazione 7274
Sostituzione della versione 1 con la versione 2 7274
ide-by-sideInstallazione S 7275
Utilizzo della modalità di debug di aggiornamento 7276
Prerequisiti 7276
Come funziona 7276
Limitazioni 7280
Configura la modalità di debug per l'aggiornamento 7282
Utilizzo dello strumento di AWS migrazione CLI da v1 a v2 7282
Come funziona 7283
Prerequisiti 7283
Installazione 7284
Utilizzo 7284
Limitazioni 7286
Disinstallazione 7290
Risoluzione degli errori di installazione e disinstallazione della AWS CLI 7294
Cronologia dei documenti 7295

Le traduzioni sono generate tramite traduzione automatica. In caso di conflitto tra il contenuto di una traduzione e la versione originale in Inglese, quest'ultima prevarrà.

Cos'è la AWS Command Line Interface?

AWS Command Line Interface (AWS CLI) è uno strumento open source che consente di interagire con i servizi AWS utilizzando i comandi nella shell a riga di comando. Con una configurazione minima, l'AWS CLI consente di iniziare a eseguire i comandi che implementano funzionalità equivalenti a quelle fornite dalla Console di gestione AWS basata su browser dal prompt di comando nel programma di terminale:

- Shell di Linux: utilizza programmi shell comuni come `bash`, `zsh` e `tcsh` per eseguire i comandi in Linux o macOS.
- Riga di comando Windows: in Microsoft Windows esegui i comandi al prompt dei comandi di Windows o in PowerShell.
- In remoto: esegui comandi nelle istanze Amazon Elastic Compute Cloud (Amazon EC2) tramite un programma terminale remoto quali PuTTY o SSH oppure con AWS Systems Manager.

Tutte le funzioni di amministrazione, gestione e accesso IaaS (Infrastructure as a Service) di AWS nella Console di gestione AWS sono disponibili nell'API AWS e nella AWS CLI. I nuovi servizi e caratteristiche IaaS di AWS forniscono la funzionalità completa della Console di gestione AWS mediante API e CLI all'avvio o entro 180 giorni dall'avvio.

AWS CLI fornisce accesso diretto alle API pubbliche dei servizi AWS. Puoi esplorare le capacità di un servizio con AWS CLI e sviluppare script di shell per gestire le risorse. Oltre ai comandi di basso livello equivalenti alle API, diversi servizi AWS offrono personalizzazioni per AWS CLI. Le personalizzazioni possono includere comandi di livello più elevato che semplificano l'utilizzo di un servizio con un'API complessa.

Informazioni sulla versione 2 della AWS CLI

La versione 2 è la versione principale più recente della AWS CLI e supporta tutte le funzionalità più recenti. Alcune funzionalità introdotte nella versione 2 non sono compatibili con la versione 1 ed è necessario eseguire l'aggiornamento per accedere a tali funzionalità. Sono state apportate alcune modifiche "importanti" rispetto alla versione 1 che potrebbero richiedere la modifica degli script. Per un elenco delle modifiche importanti nella versione 2, consulta Guida alla migrazione per la AWS CLI versione 2.

La AWS CLI versione 2 è disponibile per l'installazione solo come un programma di installazione in bundle. Sebbene sia possibile trovarla nei gestori di pacchetti, si tratta di pacchetti non supportati e

non ufficiali che non sono prodotti o gestiti da AWS. Si consiglia di installare l'AWS CLI solo da punti di distribuzione AWS ufficiali come documentato in questa guida.

Per installare la versione 2 della AWS CLI, consulta the section called "Installazione o aggiornamento".

Per controllare la versione attualmente installata utilizzare il comando seguente:

```
$ aws --version
aws-cli/2.27.41 Python/3.11.6 Linux/5.10.205-195.807.amzn2.x86_64 botocore/1.18.6
```

Per la cronologia delle versioni, consulta AWS CLI version 2 Changelog su GitHub.

Manutenzione e supporto per le versioni principali dell'SDK

Per informazioni sulla manutenzione e sul supporto per le versioni principali dell'SDK e le relative dipendenze sottostanti, consulta quanto segue nella Guida di riferimento degli strumenti e degli SDK AWS:

- Policy di manutenzione degli SDK AWS e degli strumenti
- Matrice di supporto delle versioni degli SDK AWS e degli strumenti

Informazioni su Amazon Web Services

Amazon Web Services (AWS) è una raccolta di servizi infrastrutturali digitali utilizzabile dagli sviluppatori per sviluppare le proprie applicazioni. I servizi includono le risorse di elaborazione, storage, database e sincronizzazione delle applicazioni (messaggistica e accodamento). AWS utilizza un modello di servizio con pagamento in base al consumo. Ti vengono addebitati solo i servizi utilizzati da te o dalle tue applicazioni. Inoltre, al fine di rendere AWS più accessibile per realizzare prototipi e sperimentare, AWS offre un piano di utilizzo gratuito. In questo piano, i servizi sono gratuiti al di sotto di un determinato livello di utilizzo. Per ulteriori informazioni sui costi di AWS e sul piano gratuito, consulta Utilizzo del piano gratuito di AWS. Per ottenere un account AWS, accedi alla home page di AWS e scegli Crea un account AWS.

Informazioni sugli esempi nella Guida per utente della AWS CLI

Gli esempi di AWS Command Line Interface (AWS CLI) in questa guida sono formattati utilizzando le seguenti convenzioni:

- Prompt: il prompt dei comandi utilizza il prompt di Linux e viene visualizzato come ($). Per i comandi specifici di Windows, `C:\>` viene utilizzato come prompt. Non includere il prompt quando digiti i comandi.
- Directory: quando i comandi devono essere eseguiti da una directory specifica, il nome della directory viene visualizzato prima del simbolo del prompt.
- Input utente: il testo del comando da immettere nella riga di comando viene formattato come **`user input`**.
- Testo sostituibile: il testo variabile, che include i nomi delle risorse scelte o gli ID generati dai servizi AWS che devi includere nei comandi, è formattato come *`testo sostituibile`*. Nelle righe di comando multiple o nei comandi in cui è richiesto input da tastiera specifico, anche i comandi da tastiera possono essere mostrati come testo sostituibile.
- Output: l'output restituito dai servizi AWS viene mostrato sotto l'input utente e viene formattato come `computer output`.

Ad esempio, il comando **`aws configure`** seguente include l'input dell'utente, il testo sostituibile e l'output:

1. Nella riga di comando, immetti **`aws configure`**, quindi premi Invio.
2. L'AWS CLI genera righe di testo che richiedono di immettere ulteriori informazioni.
3. Inserisci tutte le chiavi di accesso consecutivamente e quindi premi Invio.
4. Quindi, immetti un nome di regione AWS nel formato mostrato, premi Invio, quindi premi Invio per l'ultima volta per ignorare l'impostazione del formato di output.
5. Il comando Invio finale viene mostrato come testo sostituibile in quanto per quella riga non esiste alcun input utente.

```
$ aws configure
AWS Access Key ID [None]: AKIAIOSFODNN7EXAMPLE
AWS Secret Access Key [None]: wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY
Default region name [None]: us-west-2
Default output format [None]: ENTER
```

L'esempio seguente mostra un semplice comando con l'output. Per usare questo esempio, immetti il testo completo del comando (il testo evidenziato dopo la richiesta) e quindi premi Invio. Il nome del gruppo di sicurezza, *`my-sg`*, è sostituibile con il nome del gruppo di sicurezza desiderato.
Il documento JSON, incluse le parentesi graffe, è l'output. Se si configura l'interfaccia a riga di

comando per l'output in formato testo o tabella, l'output viene formattato in modo diverso. JSON è il formato di output predefinito.

```
$ aws ec2 create-security-group --group-name my-sg --description "My security group"
{
    "GroupId": "sg-903004f8"
}
```

Documentazione e risorse aggiuntive per la AWS CLI

Documentazione e risorse per la AWS CLI

Oltre a questa guida, di seguito sono elencate altre risorse online preziose per l'utilizzo della AWS CLI.

- Guida di riferimento alla versione 2 della AWS CLI
- Repository di esempi di codice di script Bash pe la AWS CLI. Esempi di script Bash open source. Gli esempi di script Bash sono ospitati nel repository degli esempi di codice AWS su GitHub.
- Repository GitHub per la AWS CLI. È anche possibile visualizzare e creare una fork del codice sorgente per la AWS CLI su GitHub. Unisciti alla community di utenti su GitHub per fornire un feedback, richiedere funzionalità e inviare i tuoi contributi. Ciò include la visualizzazione e la disponibilità di esempi di comandi per la documentazione della AWS CLI.
- Repository di esempi di alias per la AWS CLI È possibile visualizzare e creare una fork degli esempi di alias per la AWS CLI su GitHub.
- Changelog della versione 2 della AWS CLI

Altri SDK e strumenti AWS

A seconda dei casi d'uso, potrebbe essere necessario scegliere uno degli SDK AWS o degli strumenti che meglio si adattano alle tue esigenze:

- Guida di riferimento per strumenti e SDK AWS
- AWS SDK per C++
- AWS SDK per Go
- AWS SDK per Java
- AWS SDK per JavaScript

- AWS SDK per Kotlin
- AWS SDK per .NET
- AWS SDK per Python (Boto)
- AWS SDK per PHP
- AWS Strumenti per PowerShell
- AWS SDK per Ruby
- AWS SDK per Rust
- AWS SDK per SAP ABAP
- AWS SDK per Swift
- AWS Amplify

Nozioni di base su AWS CLI

Questo capitolo fornisce le fasi per iniziare con la versione 2 della AWS Command Line Interface (AWS CLI) e fornisce collegamenti alle istruzioni pertinenti.

1. Completa tutti i prerequisiti - Per accedere ai servizi AWS con la AWS CLI, sono necessari almeno un Account AWS e le credenziali IAM. Per aumentare la sicurezza del tuo AWS, ti consigliamo vivamente di non utilizzare le credenziali di accesso associate al tuo account root. È necessario creare un utente con il privilegio minimo per fornire le credenziali di accesso alle attività in cui verranno eseguite in AWS.
2. Installa o accedi a AWS CLI utilizzando uno dei seguenti metodi:
   - (Consigliato) the section called "Installazione o aggiornamento".
   - the section called "Rilasci precedenti". L'installazione di una versione specifica viene utilizzata principalmente se il team allinea i propri strumenti a una versione specifica.
   - the section called "Compilazione e installazione dall'origine". La creazione della AWS CLI da un codice sorgente GitHub è un metodo più approfondito utilizzato principalmente dai clienti che lavorano su piattaforme che non supportiamo direttamente con i nostri programmi di installazione predefiniti.
   - the section called "Amazon ECR Public/Docker".
   - Accedi alla versione 2 della AWS CLI nella console AWS dal tuo browser utilizzando AWS CloudShell. Per ulteriori informazioni, consulta la Guida per l'utente AWS CloudShell.
3. Dopo aver avuto accesso alla AWS CLI, configurala con le tue credenziali IAM per il primo utilizzo.

> **Risoluzione dei problemi relativi all'installazione o alla configurazione**
>
> In caso di problemi dopo l'installazione, la disinstallazione o la configurazione della AWS CLI, consulta la procedura the section called "Risoluzione degli errori" per la risoluzione dei problemi.

Argomenti

- Prerequisiti per utilizzare la AWS CLI versione 2
- Installazione o aggiornamento dell'ultima versione di AWS CLI
- Installazione dei rilasci precedenti della AWS CLI versione 2

- Compilazione e installazione di AWS CLI dall'origine
- Esecuzione delle immagini ufficiali di Amazon ECR Public o Docker per la AWS CLI
- Configurazione del AWS CLI

Prerequisiti per utilizzare la AWS CLI versione 2

Per accedere ai AWS servizi con AWS CLI, è necessario disporre di credenziali IAM o IAM Identity Center. Account AWS Quando si eseguono AWS CLI i comandi, è AWS CLI necessario avere accesso a tali AWS credenziali. Per aumentare la sicurezza del tuo AWS account, ti consigliamo di utilizzare solo credenziali di breve durata quando usi utenti root o IAM. Dovresti creare un utente con i privilegi minimi per fornire le credenziali di accesso alle attività che eseguirai. AWS Per informazioni sulle best practice, consulta le best practice di sicurezza nella IAM User Guide.

Argomenti

- Creare un account amministrativo IAM o Centro identità IAM
- Fasi successive

Creare un account amministrativo IAM o Centro identità IAM

Prima di poter configurare AWS CLI, devi creare un account IAM o IAM Identity Center.

Opzioni di creazione dell'account

Scegli un modo per gestire le tue credenziali	Per	Come creare un account	Come configurare l'accesso programmatico all'account
AWS Credenziali della console di gestione	Utilizza credenziali a breve termine corrispondenti all'utente root creato durante la configurazione iniziale dell'account, un utente IAM o un'identità federata	Apri la https://portal.aws.amazon.com/billing/registrazione e segui le istruzioni online.	Non sono necessari passaggi aggiuntivi. Per accedere al AWS CLI con il tuo AWS account, vedi Accedi per lo sviluppo AWS locale utilizzando le credenziali della console

Scegli un modo per gestire le tue credenziali	Per	Come creare un account	Come configurare l'accesso programmatico all'account
	del tuo provider di identità.		
Centro identità IAM	Utilizza credenziali a breve termine per accedere ai AWS servizi.	Segui le istruzioni contenute nella Guida introduttiva di AWS IAM Identity Center User Guide.	Configura l'accesso programmatico configurando l'opzione da utilizzare nella Guida per l'utente di AWS IAM Identity Center.
AWS Identity and Access Management (Non consigliato)	Utilizza credenziali a lungo termine per accedere ai servizi. AWS	Segui le istruzioni contenute in Create an for emergency access nella IAM User Guide.	Configura l'accesso programmatico tramite Gestisci le chiavi di accesso per gli utenti nella Guida per l'utente IAM.

Fasi successive

Dopo aver creato le credenziali Account AWS e IAM, per utilizzarle AWS CLI puoi effettuare una delle seguenti operazioni:

- Installa l'ultima versione della AWS CLI versione 2 sul tuo computer.
- Installa una AWS CLI versione precedente della versione 2 sul tuo computer.
- Accedi alla AWS CLI versione 2 dal tuo computer utilizzando un'immagine Docker.
- Accedi alla AWS CLI versione 2 nella AWS console dal tuo browser utilizzando AWS CloudShell. Per ulteriori informazioni, consulta la Guida per l'utente AWS CloudShell.

Installazione o aggiornamento dell'ultima versione di AWS CLI

Questo argomento descrive come installare o aggiornare l'ultima versione di AWS Command Line Interface (AWS CLI) sui sistemi operativi supportati. Per informazioni sulle ultime versioni della AWS CLI, consulta AWS CLI version 2 Changelog su GitHub.

Per installare una versione precedente di AWS CLI, consulta the section called "Rilasci precedenti". Per le istruzioni di disinstallazione, consulta Disinstallazione.

> **Important**
>
> L'AWS CLI versioni 1 e 2 usa lo stesso nome del comando `aws`. Se in precedenza è stato installato AWS CLI versione 1, consulta Guida alla migrazione per la AWS CLI versione 2.

Argomenti

- Istruzioni per l'installazione e l'aggiornamento di AWS CLI
- Risoluzione degli errori di installazione e disinstallazione della AWS CLI
- Passaggi successivi

Istruzioni per l'installazione e l'aggiornamento di AWS CLI

Per le istruzioni di installazione, espandere la sezione relativa al sistema operativo in uso.

Linux

Requisiti di installazione e aggiornamento

- È necessario estrarre o "decomprimere" il pacchetto scaricato. Se il sistema operativo non dispone del comando `unzip` integrato, utilizzare un comando equivalente.
- AWS CLI utilizza `glibc`, `groff` e `less`. Questi sono inclusi per impostazione predefinita nella maggior parte delle principali distribuzioni di Linux.
- AWS CLI è supportato sulle versioni a 64 bit delle distribuzioni recenti di CentOS, Fedora, Ubuntu, Amazon Linux 1, Amazon Linux 2, Amazon Linux 2023 e Linux ARM.
- Dal momento che AWS non gestisce repository di terze parti oltre a `snap`, non è possibile garantire che contengano l'ultima versione di AWS CLI.

Installazione o aggiornamento di AWS CLI

> **Warning**
>
> Se è la prima volta che si esegue un aggiornamento su Amazon Linux, per installare la versione più recente di AWS CLI, è necessario disinstallare la versione `yum` preinstallata utilizzando il seguente comando:
>
> ```
> $ sudo yum remove awscli
> ```
>
> Dopo aver rimosso l'installazione `yum` di AWS CLI, seguire le istruzioni di installazione di Linux riportate di seguito.

È possibile installare AWS CLI utilizzando uno dei metodi seguenti:

- Il programma di installazione da riga di comando è una buona opzione per il controllo delle versioni, in quanto consente di specificare la versione da installare. Questa opzione non si aggiorna automaticamente e ogni volta che si esegue l'aggiornamento è necessario scaricare un nuovo programma di installazione per sostituire la versione precedente.
- Il pacchetto **`snap`** ufficialmente supportato è una buona opzione per avere sempre la versione più recente di AWS CLI dal momento che i pacchetti snap si aggiornano automaticamente. Non include un supporto integrato per la selezione delle versioni secondarie di AWS CLI e pertanto non è un metodo di installazione ottimale se il team deve aggiungere versioni.

Command line installer - Linux x86 (64-bit)

Per aggiornare l'installazione corrente della AWS CLI, è necessario scaricare un nuovo programma di installazione ogni volta che si esegue l'aggiornamento per sostituire le versioni precedenti. Segui queste fasi dalla riga di comando per installare AWS CLI su Linux.

Di seguito sono riportati i passaggi di installazione rapida, in un singolo gruppo da copiare e incollare, i quali forniscono un'installazione di base. Per istruzioni dettagliate, consulta i passaggi a seguire.

> **Note**
>
> (Facoltativo) Il seguente blocco di comandi scarica e installa la AWS CLI senza prima verificare l'integrità del download. Per verificare l'integrità del download, utilizza le istruzioni dettagliate riportate di seguito.

Come installare AWS CLI, eseguire i comandi riportati di seguito.

```
$ curl "https://awscli.amazonaws.com/awscli-exe-linux-x86_64.zip" -o "awscliv2.zip"
unzip awscliv2.zip
sudo ./aws/install
```

Per aggiornare l'installazione corrente della AWS CLI, aggiungere il collegamento simbolico esistente e le informazioni sul programma di installazione per costruire il comando `install` utilizzando i parametri `--bin-dir`, `--install-dir` e `--update`. Il seguente blocco di comandi utilizza il collegamento simbolico di esempio */usr/local/bin* e la posizione del programma di installazione di esempio */usr/local/aws-cli* per installare AWS CLI localmente per l'utente corrente.

```
$ curl "https://awscli.amazonaws.com/awscli-exe-linux-x86_64.zip" -o "awscliv2.zip"
unzip awscliv2.zip
sudo ./aws/install --bin-dir /usr/local/bin --install-dir /usr/local/aws-cli --
update
```

Istruzioni dettagliate per l'installazione

1. Scaricare il file di installazione in uno dei seguenti modi:

   - Utilizzare il comando **`curl`**: l'opzione `-o` specifica il nome del file in cui viene scritto il pacchetto scaricato. Le opzioni sul seguente comando di esempio determinano la scrittura del file scaricato nella directory corrente con il nome locale `awscliv2.zip`.

     ```
     $ curl "https://awscli.amazonaws.com/awscli-exe-linux-x86_64.zip" -o
      "awscliv2.zip"
     ```

   - Scaricare dall'URL: per scaricare il programma di installazione con il browser, utilizzare il seguente URL: https://awscli.amazonaws.com/awscli-exe-linux-x86_64.zip
2. (Facoltativo) Verificare l'integrità del file zip scaricato

Se hai scelto di scaricare manualmente il pacchetto di installazione `.zip` della AWS CLI nei passaggi precedenti, puoi utilizzare i seguenti passaggi per verificare le firme tramite lo strumento GnuPG.

I file `.zip` del pacchetto di installazione di AWS CLI sono firmati crittograficamente mediante le firme PGP. Se i file sono in qualche modo danneggiati o alterati, questa verifica non va a buon fine e l'installazione deve essere arrestata.

a. Scaricare e installare il comando gpg utilizzando il programma di gestione dei pacchetti. Per ulteriori informazioni su GnuPG, consulta il sito Web GnuPG.

b. Per creare il file della chiave pubblica, creare un file di testo e incollare il testo seguente.

```
-----BEGIN PGP PUBLIC KEY BLOCK-----

mQINBF2Cr7UBEADJZHcgusOJl7ENSyumXh85z0TRV0xJorM2B/JL0kHOyigQluUG
ZMLhENaG0bYatdrKP+3H91lvK050pXwnO/R7fB/FSTouki4ciIx5OuLlnJZIxSzx
PqGl0mkxImLNbGWoi6Lto0LYxqHN2iQtzlwTVmq9733zd3XfcXrZ3+LblHAgEt5G
TfNxEKJ8soPLyWmwDH6HWCnjZ/aIQRBTIQ05uVeEoYxSh6wOai7ss/KveoSNBbYz
gbdzoqI2Y8cgH2nbfgp3DSasaLZEdCSsIsK1u05CinE7k2qZ7KgKAUIcT/cR/grk
C6VwsnDU0OUCideXcQ8WeHutqvgZH1JgKDbznoIzeQHJD238GEu+eKhRHcz8/jeG
94zkcgJOz3KbZGYMiTh277Fvj9zzvZsbMBCedV1BTg3TqgvdX4bdkhf5cH+7NtWO
lrFj6UwAsGukBTAOxC0l/dnSmZhJ7Z1KmEWilro/gOrjtOxqRQutlIqG22TaqoPG
fYVN+en3Zwbt97kcgZDwqbuykNt64oZWc4XKCa3mprEGC3IbJTBFqglXmZ7l9ywG
EEUJYOlb2XrSuPWml39beWdKM8kzr1OjnlOm6+lpTRCBfo0wa9F8YZRhHPAkwKkX
XDeOGpWRj4ohOx0d2GWkyV5xyN14p2tQOCdOODmz80yUTgRpPVQUtOEhXQARAQAB
tCFBV1MgQ0xJIFRlYW0gPGF3cy1jbGlAYW1hem9uLmNvbT6JAlQEEwEIAD4CGwMF
CwkIBwIGFQoJCAsCBBYCAwECHgECF4AWIQT7Xbd/1cEYuAURraimMQrMRnJHXAUC
aGveYQUJDMpiLAAKCRCmMQrMRnJHXKBYD/9Ab0qQdGiO5hObchG8xh8Rpb4Mjyf6
0JrVo6m8GNjNj6BHkSc8fuTQJ/FaEhaQxj3pjZ3GXPrXjIIVChmICLlFuRXYzrXc
Pw0lniybypsZEVai5kO0tCNBCCFuMN9RsmmRG8mf7lC4FSTbUDmxG/QlYK+0IV/l
uJkzxWa+rySkdpm0JdqumjegNRgObdXHAQDWlubWQHWyZyIQ2B4U7AxqSpcdJp6I
S4Zds4wVLd1WE5pquYQ8vS2cNlDm4QNg8wTj58e3lKN47hXHMIb6CHxRnb947oJa
pg189LLPR5koh+EorNkA1wu5mAJtJvy5YMsppy2y/kIjp3lyY6AmPT1posgGk70Z
CmToEZ5rbd7AREXtlh76A0cabMDFlEHDIK8RNUOSRr7L64+KxOUegKBfQHb9dADY
qqiKqpCbKgvtWlds909Ms74JBgr2KwZCSY1HaOxnIr4CY43QRqAq5YHOay/mU+6w
hhmdF18vpyK0vfkvvGresWtSXbag7Hkt3XjaEw76BzxQH21EBDqU8WJVjHgU6ru+
DJTs+SxgJbaT3hb/vyjlw0lK+hFfhWKRwgOXH8vqducF95NRSUxtS4fpqxWVaw3Q
V2OWSjbne99A5EPEySzryFTKbMGwaTlAwMCwYevt4YT6eb7NmFhTx0Fis4TalUs+
j+c7Kg92pDx2uQ==
=OBAt
-----END PGP PUBLIC KEY BLOCK-----
```

Per riferimento, di seguito sono riportati i dettagli della chiave pubblica.

```
Key ID:          A6310ACC4672475C
Type:            RSA
Size:            4096/4096
Created:         2019-09-18
Expires:         2026-07-07
User ID:         AWS CLI Team <aws-cli@amazon.com>
Key fingerprint:  FB5D B77F D5C1 18B8 0511  ADA8 A631 0ACC 4672 475C
```

c. Importare la chiave pubblica AWS CLI con il comando seguente, sostituendo *public-key-file-name* con il nome file della chiave pubblica creata.

```
$ gpg --import public-key-file-name
gpg: /home/username/.gnupg/trustdb.gpg: trustdb created
gpg: key A6310ACC4672475C: public key "AWS CLI Team <aws-cli@amazon.com>"
 imported
gpg: Total number processed: 1
gpg:               imported: 1
```

d. Scarica il file della firma AWS CLI per il pacchetto scaricato. Il percorso e il nome sono identici a quelli del file `.zip` a cui corrisponde, ma con estensione `.sig`. Negli esempi seguenti, viene salvato nella cartella corrente come un file denominato `awscliv2.sig`.

   Per la versione più recente della AWS CLI, utilizza il seguente blocco di comandi:

```
$ curl -o awscliv2.sig https://awscli.amazonaws.com/awscli-exe-linux-
x86_64.zip.sig
```

e. Verifica la firma, passando entrambi i nomi file `.sig` e `.zip` scaricati come parametri del comando gpg.

```
$ gpg --verify awscliv2.sig awscliv2.zip
```

   L'output visualizzato dovrebbe essere simile al seguente:

```
gpg: Signature made Mon Nov  4 19:00:01 2019 PST
gpg:                using RSA key FB5D B77F D5C1 18B8 0511 ADA8 A631 0ACC
 4672 475C
gpg: Good signature from "AWS CLI Team <aws-cli@amazon.com>" [unknown]
```

```
gpg: WARNING: This key is not certified with a trusted signature!
gpg:          There is no indication that the signature belongs to the
 owner.
Primary key fingerprint: FB5D B77F D5C1 18B8 0511  ADA8 A631 0ACC 4672 475C
```

> **Important**
>
> L'avviso nell'output è previsto e non indica un problema. Si verifica perché non esiste una catena di trust tra la chiave PGP personale (se una è disponibile) e la chiave PGP AWS CLI. Per ulteriori informazioni, consulta Web of trust.

3. Decomprimi il programma di installazione. Se la distribuzione Linux non dispone di un comando `unzip` integrato, utilizzare un comando equivalente per la decompressione. Il comando di esempio seguente decomprime il pacchetto e crea una directory denominata `aws` sotto la directory corrente.

```
$ unzip awscliv2.zip
```

> **Note**
>
> Quando si esegue l'aggiornamento da una versione precedente, il comando `unzip` richiede di sostituire i file esistenti. Per ignorare questi prompt, ad esempio con l'automazione degli script, utilizzare il flag di aggiornamento `-u` per `unzip`. Questo flag aggiorna automaticamente i file esistenti e ne crea di nuovi in base alle necessità.
>
> ```
> $ unzip -u awscliv2.zip
> ```

4. Esegui il programma di installazione. Il comando di installazione utilizza un file denominato `install` nella `aws` directory appena decompressa. Per impostazione predefinita, i file vengono tutti installati in `/usr/local/aws-cli` e viene creato un collegamento simbolico in `/usr/local/bin`. Il comando include `sudo` per concedere autorizzazioni in scrittura a tali directory.

```
$ sudo ./aws/install
```

Puoi eseguire l'installazione senza `sudo` se specifichi le directory per le quali disponi già di autorizzazioni in scrittura. Utilizzare le seguenti istruzioni per il comando `install` per specificare il percorso di installazione:

- Assicurarsi che i percorsi forniti ai parametri `-i` e `-b` non contengano nomi di volume o directory contenenti caratteri di spazio o altri caratteri di spazio vuoto. Se c'è uno spazio, l'installazione non riesce.
- `--install-dir` o `-i`: questa opzione specifica la directory in cui vengono copiati tutti i file.

  Il valore predefinito è `/usr/local/aws-cli`.
- `--bin-dir` o `-b`: questa opzione specifica che il programma `aws` principale nella directory di installazione è collegato simbolicamente al file `aws` nel percorso specificato. È necessario disporre delle autorizzazioni di scrittura nella directory specificata. La creazione di un collegamento simbolico a una directory già presente nel percorso elimina la necessità di aggiungere la directory di installazione alla variabile `$PATH` dell'utente.

  Il valore predefinito è `/usr/local/bin`.

```
$ ./aws/install -i /usr/local/aws-cli -b /usr/local/bin
```

> **Note**
>
> Per aggiornare l'installazione corrente della AWS CLI, aggiungere il collegamento simbolico esistente e le informazioni sul programma di installazione per costruire il comando `install` con il parametro `--update`.
>
> ```
> $ sudo ./aws/install --bin-dir /usr/local/bin --install-dir /usr/local/
> aws-cli --update
> ```
>
> Per individuare il collegamento simbolico e la directory di installazione esistenti, utilizza i seguenti passaggi:
>
> 1. Usare il comando `which` per individuare il collegamento simbolico. Questo ti fornisce il percorso da utilizzare con il parametro `--bin-dir`.

```
$ which aws
/usr/local/bin/aws
```

2. Utilizzare il comando `ls` per individuare la directory a cui punta il collegamento simbolico. Questo ti fornisce il percorso da utilizzare con il parametro `--install-dir`.

```
$ ls -l /usr/local/bin/aws
lrwxrwxrwx 1 ec2-user ec2-user 49 Oct 22 09:49 /usr/local/bin/aws -> /usr/local/aws-cli/v2/current/bin/aws
```

5. Conferma l'installazione con il seguente comando.

```
$ aws --version
aws-cli/2.27.41 Python/3.11.6 Linux/5.10.205-195.807.amzn2.x86_64
```

Se il comando `aws` non viene trovato, potrebbe essere necessario riavviare il terminale o seguire la procedura di risoluzione dei problemi riportata in the section called "Risoluzione degli errori".

Command line - Linux ARM

Per aggiornare l'installazione corrente della AWS CLI, è necessario scaricare un nuovo programma di installazione ogni volta che si esegue l'aggiornamento per sostituire le versioni precedenti. Segui queste fasi dalla riga di comando per installare AWS CLI su Linux.

Di seguito sono riportati i passaggi di installazione rapida, in un singolo gruppo da copiare e incollare, i quali forniscono un'installazione di base. Per istruzioni dettagliate, consulta i passaggi a seguire.

> **Note**
>
> (Facoltativo) Il seguente blocco di comandi scarica e installa la AWS CLI senza prima verificare l'integrità del download. Per verificare l'integrità del download, utilizza le istruzioni dettagliate riportate di seguito.

Come installare AWS CLI, eseguire i comandi riportati di seguito.

```
$ curl "https://awscli.amazonaws.com/awscli-exe-linux-aarch64.zip" -o "awscliv2.zip"
unzip awscliv2.zip
sudo ./aws/install
```

Per aggiornare l'installazione corrente della AWS CLI, aggiungere il collegamento simbolico esistente e le informazioni sul programma di installazione per costruire il comando `install` utilizzando i parametri `--bin-dir`, `--install-dir` e `--update`. Il seguente blocco di comandi utilizza il collegamento simbolico di esempio */usr/local/bin* e la posizione del programma di installazione di esempio */usr/local/aws-cli*.

```
$ curl "https://awscli.amazonaws.com/awscli-exe-linux-aarch64.zip" -o "awscliv2.zip"
unzip awscliv2.zip
sudo ./aws/install --bin-dir /usr/local/bin --install-dir /usr/local/aws-cli --
update
```

Istruzioni dettagliate per l'installazione

1. Scaricare il file di installazione in uno dei seguenti modi:

   - Utilizzare il comando **`curl`**: l'opzione `-o` specifica il nome del file in cui viene scritto il pacchetto scaricato. Le opzioni sul seguente comando di esempio determinano la scrittura del file scaricato nella directory corrente con il nome locale `awscliv2.zip`.

     ```
     $ curl "https://awscli.amazonaws.com/awscli-exe-linux-aarch64.zip" -o
      "awscliv2.zip"
     ```

   - Scaricare dall'URL: per scaricare il programma di installazione con il browser, utilizzare il seguente URL: https://awscli.amazonaws.com/awscli-exe-linux-aarch64.zip

2. (Facoltativo) Verificare l'integrità del file zip scaricato

   Se hai scelto di scaricare manualmente il pacchetto di installazione `.zip` della AWS CLI nei passaggi precedenti, puoi utilizzare i seguenti passaggi per verificare le firme tramite lo strumento `GnuPG`.

   I file `.zip` del pacchetto di installazione di AWS CLI sono firmati crittograficamente mediante le firme PGP. Se i file sono in qualche modo danneggiati o alterati, questa verifica non va a buon fine e l'installazione deve essere arrestata.

a. Scaricare e installare il comando gpg utilizzando il programma di gestione dei pacchetti. Per ulteriori informazioni su GnuPG, consulta il sito Web GnuPG.

b. Per creare il file della chiave pubblica, creare un file di testo e incollare il testo seguente.

```
-----BEGIN PGP PUBLIC KEY BLOCK-----

mQINBF2Cr7UBEADJZHcgusOJl7ENSyumXh85z0TRV0xJorM2B/JL0kHOyigQluUG
ZMLhENaG0bYatdrKP+3H91lvK050pXwnO/R7fB/FSTouki4ciIx5OuLlnJZIxSzx
PqGl0mkxImLNbGWoi6Lto0LYxqHN2iQtzlwTVmq9733zd3XfcXrZ3+LblHAgEt5G
TfNxEKJ8soPLyWmwDH6HWCnjZ/aIQRBTIQ05uVeEoYxSh6wOai7ss/KveoSNBbYz
gbdzoqI2Y8cgH2nbfgp3DSasaLZEdCSsIsK1u05CinE7k2qZ7KgKAUIcT/cR/grk
C6VwsnDU0OUCideXcQ8WeHutqvgZH1JgKDbznoIzeQHJD238GEu+eKhRHcz8/jeG
94zkcgJOz3KbZGYMiTh277Fvj9zzvZsbMBCedV1BTg3TqgvdX4bdkhf5cH+7NtWO
lrFj6UwAsGukBTAOxC0l/dnSmZhJ7Z1KmEWilro/gOrjtOxqRQutlIqG22TaqoPG
fYVN+en3Zwbt97kcgZDwqbuykNt64oZWc4XKCa3mprEGC3IbJTBFqglXmZ7l9ywG
EEUJYOlb2XrSuPWml39beWdKM8kzr1OjnlOm6+lpTRCBfo0wa9F8YZRhHPAkwKkX
XDeOGpWRj4ohOx0d2GWkyV5xyN14p2tQOCdOODmz80yUTgRpPVQUtOEhXQARAQAB
tCFBV1MgQ0xJIFRlYW0gPGF3cy1jbGlAYW1hem9uLmNvbT6JAlQEEwEIAD4CGwMF
CwkIBwIGFQoJCAsCBBYCAwECHgECF4AWIQT7Xbd/1cEYuAURraimMQrMRnJHXAUC
aGveYQUJDMpiLAAKCRCmMQrMRnJHXKBYD/9Ab0qQdGiO5hObchG8xh8Rpb4Mjyf6
0JrVo6m8GNjNj6BHkSc8fuTQJ/FaEhaQxj3pjZ3GXPrXjIIVChmICLlFuRXYzrXc
Pw0lniybypsZEVai5kO0tCNBCCFuMN9RsmmRG8mf7lC4FSTbUDmxG/QlYK+0IV/l
uJkzxWa+rySkdpm0JdqumjegNRgObdXHAQDWlubWQHWyZyIQ2B4U7AxqSpcdJp6I
S4Zds4wVLd1WE5pquYQ8vS2cNlDm4QNg8wTj58e3lKN47hXHMIb6CHxRnb947oJa
pg189LLPR5koh+EorNkA1wu5mAJtJvy5YMsppy2y/kIjp3lyY6AmPT1posgGk70Z
CmToEZ5rbd7ARExtlh76A0cabMDFlEHDIK8RNUOSRr7L64+KxOUegKBfQHb9dADY
qqiKqpCbKgvtWlds909Ms74JBgr2KwZCSY1HaOxnIr4CY43QRqAq5YHOay/mU+6w
hhmdF18vpyK0vfkvvGresWtSXbag7Hkt3XjaEw76BzxQH21EBDqU8WJVjHgU6ru+
DJTs+SxgJbaT3hb/vyjlw0lK+hFfhWKRwgOXH8vqducF95NRSUxtS4fpqxWVaw3Q
V2OWSjbne99A5EPEySzryFTKbMGwaTlAwMCwYevt4YT6eb7NmFhTx0Fis4TalUs+
j+c7Kg92pDx2uQ==
=OBAt
-----END PGP PUBLIC KEY BLOCK-----
```

Per riferimento, di seguito sono riportati i dettagli della chiave pubblica.

```
Key ID:          A6310ACC4672475C
Type:            RSA
Size:            4096/4096
Created:         2019-09-18
Expires:         2026-07-07
```

```
User ID:          AWS CLI Team <aws-cli@amazon.com>
Key fingerprint:  FB5D B77F D5C1 18B8 0511  ADA8 A631 0ACC 4672 475C
```

c. Importare la chiave pubblica AWS CLI con il comando seguente, sostituendo *public-key-file-name* con il nome file della chiave pubblica creata.

```
$ gpg --import public-key-file-name
gpg: /home/username/.gnupg/trustdb.gpg: trustdb created
gpg: key A6310ACC4672475C: public key "AWS CLI Team <aws-cli@amazon.com>"
 imported
gpg: Total number processed: 1
gpg:               imported: 1
```

d. Scarica il file della firma AWS CLI per il pacchetto scaricato. Il percorso e il nome sono identici a quelli del file `.zip` a cui corrisponde, ma con estensione `.sig`. Negli esempi seguenti, viene salvato nella cartella corrente come un file denominato `awscliv2.sig`.

   Per la versione più recente della AWS CLI, utilizza il seguente blocco di comandi:

```
$ curl -o awscliv2.sig https://awscli.amazonaws.com/awscli-exe-linux-
aarch64.zip.sig
```

e. Verifica la firma, passando entrambi i nomi file `.sig` e `.zip` scaricati come parametri del comando gpg.

```
$ gpg --verify awscliv2.sig awscliv2.zip
```

   L'output visualizzato dovrebbe essere simile al seguente:

```
gpg: Signature made Mon Nov  4 19:00:01 2019 PST
gpg:                using RSA key FB5D B77F D5C1 18B8 0511 ADA8 A631 0ACC
 4672 475C
gpg: Good signature from "AWS CLI Team <aws-cli@amazon.com>" [unknown]
gpg: WARNING: This key is not certified with a trusted signature!
gpg:          There is no indication that the signature belongs to the
 owner.
Primary key fingerprint: FB5D B77F D5C1 18B8 0511  ADA8 A631 0ACC 4672 475C
```

> **Important**
>
> L'avviso nell'output è previsto e non indica un problema. Si verifica perché non esiste una catena di trust tra la chiave PGP personale (se una è disponibile) e la chiave PGP AWS CLI. Per ulteriori informazioni, consulta Web of trust.

3. Decomprimi il programma di installazione. Se la distribuzione Linux non dispone di un comando `unzip` integrato, utilizzare un comando equivalente per la decompressione. Il comando di esempio seguente decomprime il pacchetto e crea una directory denominata `aws` sotto la directory corrente.

```
$ unzip awscliv2.zip
```

> **Note**
>
> Quando si esegue l'aggiornamento da una versione precedente, il comando `unzip` richiede di sostituire i file esistenti. Per ignorare questi prompt, ad esempio con l'automazione degli script, utilizzare il flag di aggiornamento `-u` per `unzip`. Questo flag aggiorna automaticamente i file esistenti e ne crea di nuovi in base alle necessità.
>
> ```
> $ unzip -u awscliv2.zip
> ```

4. Esegui il programma di installazione. Il comando di installazione utilizza un file denominato `install` nella `aws` directory appena decompressa. Per impostazione predefinita, i file vengono tutti installati in `/usr/local/aws-cli` e viene creato un collegamento simbolico in `/usr/local/bin`. Il comando include `sudo` per concedere autorizzazioni in scrittura a tali directory.

```
$ sudo ./aws/install
```

Puoi eseguire l'installazione senza `sudo` se specifichi le directory per le quali disponi già di autorizzazioni in scrittura. Utilizzare le seguenti istruzioni per il comando `install` per specificare il percorso di installazione:

- Assicurarsi che i percorsi forniti ai parametri `-i` e `-b` non contengano nomi di volume o directory contenenti caratteri di spazio o altri caratteri di spazio vuoto. Se c'è uno spazio, l'installazione non riesce.
- `--install-dir` o `-i`: questa opzione specifica la directory in cui vengono copiati tutti i file.

  Il valore predefinito è `/usr/local/aws-cli`.
- `--bin-dir` o `-b`: questa opzione specifica che il programma `aws` principale nella directory di installazione è collegato simbolicamente al file `aws` nel percorso specificato. È necessario disporre delle autorizzazioni di scrittura nella directory specificata. La creazione di un collegamento simbolico a una directory già presente nel percorso elimina la necessità di aggiungere la directory di installazione alla variabile `$PATH` dell'utente.

  Il valore predefinito è `/usr/local/bin`.

```
$ ./aws/install -i /usr/local/aws-cli -b /usr/local/bin
```

> **Note**
>
> Per aggiornare l'installazione corrente della AWS CLI, aggiungere il collegamento simbolico esistente e le informazioni sul programma di installazione per costruire il comando `install` con il parametro `--update`.
>
> ```
> $ sudo ./aws/install --bin-dir /usr/local/bin --install-dir /usr/local/
> aws-cli --update
> ```
>
> Per individuare il collegamento simbolico e la directory di installazione esistenti, utilizza i seguenti passaggi:
>
> 1. Usare il comando `which` per individuare il collegamento simbolico. Questo ti fornisce il percorso da utilizzare con il parametro `--bin-dir`.
>
>    ```
>    $ which aws
>    /usr/local/bin/aws
>    ```

2. Utilizzare il comando `ls` per individuare la directory a cui punta il collegamento simbolico. Questo ti fornisce il percorso da utilizzare con il parametro `--install-dir`.

```
$ ls -l /usr/local/bin/aws
lrwxrwxrwx 1 ec2-user ec2-user 49 Oct 22 09:49 /usr/local/bin/aws -> /
usr/local/aws-cli/v2/current/bin/aws
```

5. Conferma l'installazione con il seguente comando.

```
$ aws --version
aws-cli/2.27.41 Python/3.11.6 Linux/5.10.205-195.807.amzn2.x86_64
```

Se il comando `aws` non viene trovato, potrebbe essere necessario riavviare il terminale o seguire la procedura di risoluzione dei problemi riportata in the section called "Risoluzione degli errori".

Snap package

Viene fornita una versione ufficiale AWS supportata di AWS CLI su `snap`. Per avere sempre l'ultima versione di AWS CLI installata sul sistema, un pacchetto snap fornisce questa opzione, in quanto si aggiorna automaticamente. Non include un supporto integrato per la selezione delle versioni secondarie di AWS CLI e pertanto non è un metodo di installazione ottimale se il team deve aggiungere versioni. Per installare una versione secondaria specifica di AWS CLI, è consigliabile utilizzare il programma di installazione da riga di comando.

1. Se la piattaforma Linux non include l'installazione di `snap`, installare `snap` sulla piattaforma.

   a. Per informazioni sull'installazione di `snap`, consulta Installing the daemon nella Documentazione di Snap.

   b. Potrebbe essere necessario riavviare il sistema in modo che le variabili `PATH` vengano aggiornate correttamente. Se si riscontrano problemi di installazione, seguire i passaggi riportati in Fix common issues nella Documentazione di Snap.

   c. Per verificare che `snap` sia installato correttamente, eseguire il comando seguente.

```
$ snap version
```

2. Eseguire il comando `snap install` seguente per AWS CLI.

```
$ snap install aws-cli --classic
```

A seconda delle autorizzazioni, potrebbe essere necessario aggiungere `sudo` al comando.

```
$ sudo snap install aws-cli --classic
```

> **Note**
>
> Per visualizzare il repository di snap per AWS CLI, comprese le istruzioni aggiuntive per `snap`, consulta la pagina `aws-cli` nel sito web Canonical Snapcraft.

3. Verifica che l'AWS CLI sia installata correttamente.

```
$ aws --version
aws-cli/2.27.41 Python/3.11.6 Linux/5.10.205-195.807.amzn2.x86_64
```

Se si verifica un errore, consulta Risoluzione degli errori per AWS CLI.

macOS

Requisiti di installazione e aggiornamento

- AWS CLI è supportato in macOS 11 e versioni successive. Per ulteriori informazioni, consulta macOS support policy updates for the AWS CLI v2 sul Blog sugli strumenti per sviluppatori AWS.
- Poiché AWS non gestisce repository di terza parte, non possiamo garantire che contenga l'ultima versione dell'AWS CLI.

Matrice di supporto della versione macOS

Versione AWS CLI	Versione macOS supportata
2.21.0 (attuale)	11+
2.17.0 - 2.20.0	10.15+

Versione AWS CLI	Versione macOS supportata
2.0.0 - 2.16.12	10.14 e versioni precedenti

Installazione o aggiornamento di AWS CLI

Se si esegue l'aggiornamento alla versione più recente, utilizzare lo stesso metodo di installazione utilizzato nella versione corrente. È possibile installare AWS CLI su macOS in uno dei modi riportati di seguito.

GUI installer

I passaggi seguenti mostrano come installare la versione più recente di AWS CLI utilizzando l'interfaccia utente standard di macOS e il browser.

1. Nel browser, scaricare il file `pkg` macOS: https://awscli.amazonaws.com/AWSCLIV2.pkg
2. Esegui il file scaricato e segui le istruzioni visualizzate. È possibile scegliere di installare AWS CLI nei seguenti modi:
   - Per tutti gli utenti sul computer (richiede **`sudo`**)
     - È possibile installare in qualsiasi cartella o scegliere la cartella predefinita consigliata `/usr/local/aws-cli`.
     - Il programma di installazione crea automaticamente un collegamento simbolico `/usr/local/bin/aws` che collega il programma principale nella cartella di installazione scelta.
   - Solo per l'utente corrente (non richiede **`sudo`**)
     - È possibile installare in qualsiasi cartella per cui si dispone dell'autorizzazione in scrittura.
     - A causa delle autorizzazioni utente standard, al termine del programma di installazione, è necessario creare manualmente un file di collegamento simbolico nel proprio `$PATH` che punta ai programma `aws` e `aws_completer` utilizzando i seguenti comandi al prompt dei comandi. La posizione predefinita di un collegamento simbolico è `/usr/local/bin/`:

```
$ ln -s /folder/installed/aws-cli/aws /usr/local/bin/aws
$ ln -s /folder/installed/aws-cli/aws_completer /usr/local/bin/aws_completer
```

Se non sono disponibili le autorizzazioni di scrittura per la cartella, potrebbe essere necessario utilizzare `sudo` nel comando. L'esempio seguente utilizza `sudo` con la posizione predefinita per un collegamento simbolico in `/usr/local/bin/`:

```
$ sudo ln -s /folder/installed/aws-cli/aws /usr/local/bin/aws
$ sudo ln -s /folder/installed/aws-cli/aws_completer /usr/local/bin/
aws_completer
```

> **Note**
>
> È possibile visualizzare i log di debug dell'installazione premendo CMD+L in qualsiasi momento nel programma di installazione. Verrà aperto un riquadro di log che consente di filtrare e salvare il log. Il file di log viene salvato automaticamente in `/var/log/install.log`.

3. Per verificare che la shell possa trovare ed eseguire il comando `aws` in `$PATH`, utilizzare i seguenti comandi.

```
$ which aws
/usr/local/bin/aws
$ aws --version
aws-cli/2.27.41 Python/3.11.6 Darwin/23.3.0
```

Se il comando `aws` non viene trovato, potrebbe essere necessario riavviare il terminale o seguire la procedura di risoluzione dei problemi riportata in the section called "Risoluzione degli errori".

Command line installer - All users

Se disponi delle autorizzazioni `sudo`, è possibile effettuare l'installazione di AWS CLI per tutti gli utenti sul computer. Forniamo le fasi in un gruppo facile da copiare e incollare. Consulta le descrizioni di ogni riga nelle fasi seguenti.

```
$ curl "https://awscli.amazonaws.com/AWSCLIV2.pkg" -o "AWSCLIV2.pkg"
$ sudo installer -pkg AWSCLIV2.pkg -target /
```

Istruzioni dettagliate per l'installazione

1. Scaricare il file utilizzando il comando `curl`. L'opzione `-o` specifica il nome del file su cui viene scritto il pacchetto scaricato. In questo esempio, il file viene scritto in `AWSCLIV2.pkg` nella cartella corrente.

   ```
   $ curl "https://awscli.amazonaws.com/AWSCLIV2.pkg" -o "AWSCLIV2.pkg"
   ```

2. Eseguire il programma `installer` macOS standard, specificando come origine il file `.pkg` scaricato. Utilizzare il parametro `-pkg` per specificare il nome del pacchetto da installare e il parametro `-target /` per indicare l'unità in cui installare il pacchetto. I file vengono installati in `/usr/local/aws-cli` e viene creato automaticamente un collegamento simbolico in `/usr/local/bin`. È necessario includere `sudo` nel comando per concedere autorizzazioni di scrittura a queste cartelle.

   ```
   $ sudo installer -pkg ./AWSCLIV2.pkg -target /
   ```

   Al termine dell'installazione, i log di debug vengono scritti in `/var/log/install.log`.

3. Per verificare che la shell possa trovare ed eseguire il comando `aws` in `$PATH`, utilizzare i seguenti comandi.

   ```
   $ which aws
   /usr/local/bin/aws
   $ aws --version
   aws-cli/2.27.41 Python/3.11.6 Darwin/23.3.0
   ```

   Se il comando `aws` non viene trovato, potrebbe essere necessario riavviare il terminale o seguire la procedura di risoluzione dei problemi riportata in the section called "Risoluzione degli errori".

Command line - Current user

1. Per specificare la cartella in cui è installato AWS CLI, è necessario creare un file XML e assegnare un nome al file. Questo file è in formato XML, simile al seguente esempio. Lasciare il valore come indicato, sostituendo il percorso */Users/myusername* nella riga 9 con il percorso della cartella in cui si desidera installare AWS CLI. La cartella deve già esistere o il comando non riesce. L'esempio XML seguente, denominato *choices.xml*,

specifica il programma di installazione per installare AWS CLI nella cartella `/Users/myusername`, in cui crea una cartella denominata `aws-cli`.

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE plist PUBLIC "-//Apple//DTD PLIST 1.0//EN" "http://www.apple.com/DTDs/
PropertyList-1.0.dtd">
<plist version="1.0">
  <array>
    <dict>
      <key>choiceAttribute</key>
      <string>customLocation</string>
      <key>attributeSetting</key>
      <string>/Users/myusername</string>
      <key>choiceIdentifier</key>
      <string>default</string>
    </dict>
  </array>
</plist>
```

2. Scaricare il programma di installazione `pkg` utilizzando il comando `curl`. L'opzione `-o` specifica il nome del file su cui viene scritto il pacchetto scaricato. In questo esempio, il file viene scritto in `AWSCLIV2.pkg` nella cartella corrente.

```
$ curl "https://awscli.amazonaws.com/AWSCLIV2.pkg" -o "AWSCLIV2.pkg"
```

3. Eseguire il programma `installer` macOS standard con le seguenti opzioni:

   - Specificare il nome del pacchetto da installare utilizzando il parametro `-pkg`.
   - Specifica un'installazione solo per l'utente corrente impostando il parametro `-target` su `CurrentUserHomeDirectory`.
   - Specificare il percorso (relativo della cartella corrente) e il nome del file XML creato nel parametro `-applyChoiceChangesXML`.

   Nell'esempio seguente l'AWS CLI viene installata nella cartella `/Users/myusername/aws-cli`.

```
$ installer -pkg AWSCLIV2.pkg \
            -target CurrentUserHomeDirectory \
            -applyChoiceChangesXML choices.xml
```

4. Poiché le autorizzazioni utente standard in genere non consentono la scrittura nelle cartelle in $PATH, il programma di installazione in questa modalità non tenta di aggiungere i collegamenti simbolici ai programmi `aws` e `aws_completer`. Affinché l'AWS CLI funzioni correttamente, è necessario creare manualmente i collegamenti simbolici al termine del programma di installazione. Se l'$PATH include una cartella in cui è possibile scrivere, e si specifica questa cartella come percorso di destinazione, è possibile eseguire il comando seguente senza `sudo`. Se non si dispone di una cartella scrivibile in $PATH, è necessario utilizzare `sudo` per le autorizzazioni per scrivere nella cartella di destinazione specificata. La posizione predefinita di un collegamento simbolico è `/usr/local/bin/`. Sostituire `folder/installed` con il percorso dell'installazione di AWS CLI.

   ```
   $ sudo ln -s /folder/installed/aws-cli/aws /usr/local/bin/aws
   $ sudo ln -s /folder/installed/aws-cli/aws_completer /usr/local/bin/
   aws_completer
   ```

   Al termine dell'installazione, i log di debug vengono scritti in `/var/log/install.log`.

5. Per verificare che la shell possa trovare ed eseguire il comando `aws` in $PATH, utilizzare i seguenti comandi.

   ```
   $ which aws
   /usr/local/bin/aws
   $ aws --version
   aws-cli/2.27.41 Python/3.11.6 Darwin/23.3.0
   ```

   Se il comando `aws` non viene trovato, potrebbe essere necessario riavviare il terminale o seguire la procedura di risoluzione dei problemi riportata in the section called "Risoluzione degli errori".

Windows

Requisiti di installazione e aggiornamento

- La AWS CLI è supportata sulle versioni di Windows a 64 bit supportate da Microsoft.
- Diritti di amministratore per installare il software

Installazione o aggiornamento di AWS CLI

Per aggiornare l'installazione corrente della AWS CLI su Windows, scaricare un nuovo programma di installazione ogni volta che si esegue l'aggiornamento per sostituire le versioni precedenti. AWS CLI viene aggiornato regolarmente. Per vedere quando è stata rilasciata l'ultima versione, consulta il changelog della AWS CLI versione 2 su GitHub.

1. Scaricare ed eseguire il programma di installazione MSI di AWS CLI per Windows (a 64 bit):

   https://awscli.amazonaws.com/AWSCLIV2.msi

   In alternativa, è possibile eseguire il comando `msiexec` per eseguire il programma di installazione MSI.

   ```
   C:\> msiexec.exe /i https://awscli.amazonaws.com/AWSCLIV2.msi
   ```

   Per i vari parametri utilizzabili con `msiexec`, consulta msiexec nel sito web Microsoft Docs. Ad esempio, è possibile utilizzare il flag `/qn` per eseguire un'installazione senza l'intervento dell'utente.

   ```
   C:\> msiexec.exe /i https://awscli.amazonaws.com/AWSCLIV2.msi /qn
   ```

2. Per confermare l'installazione, aprire il menu Start, ricercare `cmd` per aprire una finestra dei prompt dei comandi e al prompt dei comandi utilizzare il comando `aws --version`.

   ```
   C:\> aws --version
   aws-cli/2.27.41 Python/3.11.6 Windows/10 exe/AMD64 prompt/off
   ```

   Se Windows non riesce a trovare il programma, potrebbe essere necessario chiudere e riaprire la finestra del prompt dei comandi per aggiornare il percorso oppure seguire la procedura di risoluzione dei problemi riportata in the section called "Risoluzione degli errori".

Risoluzione degli errori di installazione e disinstallazione della AWS CLI

Se riscontri problemi dopo l'installazione o la disinstallazione della AWS CLI, consulta la procedura the section called "Risoluzione degli errori" per la risoluzione dei problemi. Per le procedure di risoluzione dei problemi più importanti, consulta the section called "Errori di comando non trovato", the section called "Il comando `aws --version` restituisce una versione diversa da quella installata"

e the section called "Il comando "`aws --version`" restituisce una versione dopo la disinstallazione di AWS CLI".

Passaggi successivi

Dopo aver installato correttamente AWS CLI, è possibile eliminare in modo sicuro i file del programma di installazione scaricati. Dopo aver completato i passaggi descritti in the section called "Prerequisiti" e aver installato la AWS CLI, è necessario eseguire la configurazione come indicato in the section called "Configurazione".

Installazione dei rilasci precedenti della AWS CLI versione 2

Questo argomento illustra come installare i rilasci precedenti della AWS Command Line Interface versione 2 (AWS CLI) sui sistemi operativi supportati. Per informazioni sulle release della AWS CLI versione 2, consulta il changelog della AWS CLI versione 2 su GitHub.

Istruzioni per l'installazione della AWS CLI versione 2:

Linux

Requisiti di installazione

- Sai quale release della AWS CLI versione 2 desideri installare. Per un elenco della versioni, consulta il changelog della AWS CLI versione 2 su GitHub.
- È necessario estrarre o decomprimere il pacchetto scaricato. Se il sistema operativo non dispone del comando `unzip` integrato, utilizza un comando equivalente.
- La AWS CLI versione 2 utilizza `glibc`, `groff` e `less`. Questi sono inclusi per impostazione predefinita nella maggior parte delle principali distribuzioni di Linux.
- La AWS CLI versione 2 è supportata sulle versioni a 64 bit delle distribuzioni recenti di CentOS, Fedora, Ubuntu, Amazon Linux 1, Amazon Linux 2 e Linux ARM.
- Poiché AWS non gestisce repository di terze parti, non possiamo garantire che contenga l'ultima versione della AWS CLI.

Istruzioni di installazione

Segui queste fasi dalla riga di comando per installare la AWS CLI su Linux.

Le fasi vengono fornite in un unico gruppo facile da copiare e incollare a seconda se si utilizza Linux a 64 bit o Linux ARM. Consulta le descrizioni di ogni riga nelle fasi seguenti.

Linux x86 (64-bit)

> **Note**
>
> (Facoltativo) Il seguente blocco di comandi scarica e installa la AWS CLI senza prima verificare l'integrità del download. Per verificare l'integrità del download, utilizza le istruzioni dettagliate riportate di seguito.

Per un elenco della versioni, consulta il changelog della AWS CLI versione 2 su GitHub.

Come installare AWS CLI, eseguire i comandi riportati di seguito.

Per specificare una versione, aggiungi un trattino e il numero di versione al nome del file. In questo esempio, il nome del file per la versione *2.0.30* sarebbe `awscli-exe-linux-x86_64-2.0.30.zip`, generando il comando seguente:

```
$ curl "https://awscli.amazonaws.com/awscli-exe-linux-x86_64-2.0.30.zip" -o
 "awscliv2.zip"
unzip awscliv2.zip
sudo ./aws/install
```

Per aggiornare l'installazione corrente della AWS CLI, aggiungere il collegamento simbolico esistente e le informazioni sul programma di installazione per costruire il comando `install` utilizzando i parametri `--bin-dir`, `--install-dir` e `--update`. Il seguente blocco di comandi utilizza il collegamento simbolico di esempio */usr/local/bin* e la posizione del programma di installazione di esempio */usr/local/aws-cli*.

```
$ curl "https://awscli.amazonaws.com/awscli-exe-linux-x86_64-2.0.30.zip" -o
 "awscliv2.zip"
unzip awscliv2.zip
sudo ./aws/install --bin-dir /usr/local/bin --install-dir /usr/local/aws-cli --
update
```

Linux ARM

> **Note**
>
> (Facoltativo) Il seguente blocco di comandi scarica e installa la AWS CLI senza prima verificare l'integrità del download. Per verificare l'integrità del download, utilizza le istruzioni dettagliate riportate di seguito.

Per un elenco della versioni, consulta il changelog della AWS CLI versione 2 su GitHub.

Come installare AWS CLI, eseguire i comandi riportati di seguito.

Per specificare una versione, aggiungi un trattino e il numero di versione al nome del file. In questo esempio, il nome del file per la versione *2.0.30* sarebbe `awscli-exe-linux-aarch64-2.0.30.zip`, generando il comando seguente:

```
$ curl "https://awscli.amazonaws.com/awscli-exe-linux-aarch64-2.0.30.zip" -o
 "awscliv2.zip"
unzip awscliv2.zip
sudo ./aws/install
```

Per aggiornare l'installazione corrente della AWS CLI, aggiungere il collegamento simbolico esistente e le informazioni sul programma di installazione per costruire il comando `install` utilizzando i parametri `--bin-dir`, `--install-dir` e `--update`. Il seguente blocco di comandi utilizza il collegamento simbolico di esempio */usr/local/bin* e la posizione del programma di installazione di esempio */usr/local/aws-cli*.

```
$ curl "https://awscli.amazonaws.com/awscli-exe-linux-aarch64-2.0.30.zip" -o
 "awscliv2.zip"
unzip awscliv2.zip
sudo ./aws/install --bin-dir /usr/local/bin --install-dir /usr/local/aws-cli --
update
```

1. Scaricare il file di installazione in uno dei seguenti modi:

Linux x86 (64-bit)

- Utilizzare il comando **`curl`**: l'opzione `-o` specifica il nome del file in cui viene scritto il pacchetto scaricato. Le opzioni sul seguente comando di esempio determinano la scrittura del file scaricato nella directory corrente con il nome locale `awscliv2.zip`.

  Per specificare una versione, aggiungi un trattino e il numero di versione al nome del file. In questo esempio, il nome del file per la versione *`2.0.30`* sarebbe `awscli-exe-linux-x86_64-2.0.30.zip`, generando il comando seguente:

  ```
  $ curl "https://awscli.amazonaws.com/awscli-exe-linux-x86_64-2.0.30.zip" -o
   "awscliv2.zip"
  ```

  Per un elenco della versioni, consulta il changelog della AWS CLI versione 2 su GitHub.

- Download dall'URL –

  Nel browser, scarica la versione specifica della AWS CLI aggiungendo un trattino e il numero di versione al nome del file.

  ```
  https://awscli.amazonaws.com/awscli-exe-linux-x86_64-version.number.zip
  ```

  In questo esempio, il nome del file per la versione *`2.0.30`* sarebbe awscli-exe-linux-x86_64-2.0.30.zip, generando il seguente link: https://awscli.amazonaws.com/awscli-exe-linux-x86_64-2.0.30.zip

Linux ARM

- Utilizzare il comando **`curl`**: l'opzione `-o` specifica il nome del file in cui viene scritto il pacchetto scaricato. Le opzioni sul seguente comando di esempio determinano la scrittura del file scaricato nella directory corrente con il nome locale `awscliv2.zip`.

  Per specificare una versione, aggiungi un trattino e il numero di versione al nome del file. In questo esempio, il nome del file per la versione *`2.0.30`* sarebbe `awscli-exe-linux-aarch64-2.0.30.zip`, generando il comando seguente:

  ```
  $ curl "https://awscli.amazonaws.com/awscli-exe-linux-aarch64-2.0.30.zip" -o
   "awscliv2.zip"
  unzip awscliv2.zip
  ```

```
sudo ./aws/install
```

- Download dall'URL –

  Nel browser, scarica la versione specifica della AWS CLI aggiungendo un trattino e il numero di versione al nome del file.

  ```
  https://awscli.amazonaws.com/awscli-exe-linux-aarch64-version.number.zip
  ```

  In questo esempio, il nome del file per la versione *2.0.30* sarebbe `awscli-exe-linux-aarch64-2.0.30.zip`, generando il seguente link: https://awscli.amazonaws.com/awscli-exe-linux-aarch64-2.0.30.zip

2. (Facoltativo) Verificare l'integrità del file zip scaricato

   Se hai scelto di scaricare manualmente il pacchetto di installazione `.zip` della AWS CLI nei passaggi precedenti, puoi utilizzare i seguenti passaggi per verificare le firme tramite lo strumento GnuPG.

   I file `.zip` del pacchetto di installazione della AWS CLI sono firmati crittograficamente mediante le firme PGP. Se i file sono in qualche modo danneggiati o alterati, questa verifica non va a buon fine e l'installazione deve essere arrestata.

   a. Scarica e installa il comando gpg utilizzando il programma di gestione dei pacchetti. Per ulteriori informazioni su GnuPG, consulta il sito Web GnuPG.

   b. Per creare il file della chiave pubblica, crea un file di testo e incolla il testo seguente.

   ```
   -----BEGIN PGP PUBLIC KEY BLOCK-----

   mQINBF2Cr7UBEADJZHcgusOJl7ENSyumXh85z0TRV0xJorM2B/JL0kHOyigQluUG
   ZMLhENaG0bYatdrKP+3H91lvK050pXwnO/R7fB/FSTouki4ciIx5OuLlnJZIxSzx
   PqGl0mkxImLNbGWoi6Lto0LYxqHN2iQtzlwTVmq9733zd3XfcXrZ3+LblHAgEt5G
   TfNxEKJ8soPLyWmwDH6HWCnjZ/aIQRBTIQ05uVeEoYxSh6wOai7ss/KveoSNBbYz
   gbdzoqI2Y8cgH2nbfgp3DSasaLZEdCSsIsK1u05CinE7k2qZ7KgKAUIcT/cR/grk
   C6VwsnDU0OUCideXcQ8WeHutqvgZH1JgKDbznoIzeQHJD238GEu+eKhRHcz8/jeG
   94zkcgJOz3KbZGYMiTh277Fvj9zzvZsbMBCedV1BTg3TqgvdX4bdkhf5cH+7NtWO
   lrFj6UwAsGukBTAOxC0l/dnSmZhJ7Z1KmEWilro/gOrjtOxqRQutlIqG22TaqoPG
   fYVN+en3Zwbt97kcgZDwqbuykNt64oZWc4XKCa3mprEGC3IbJTBFqglXmZ7l9ywG
   EEUJYOlb2XrSuPWml39beWdKM8kzr1OjnlOm6+lpTRCBfo0wa9F8YZRhHPAkwKkX
   XDeOGpWRj4ohOx0d2GWkyV5xyN14p2tQOCdOODmz80yUTgRpPVQUtOEhXQARAQAB
   tCFBV1MgQ0xJIFRlYW0gPGF3cy1jbGlAYW1hem9uLmNvbT6JAlQEEwEIAD4CGwMF
   ```

```
CwkIBwIGFQoJCAsCBBYCAwECHgECF4AWIQT7Xbd/1cEYuAURraimMQrMRnJHXAUC
aGveYQUJDMpiLAAKCRCmMQrMRnJHXKBYD/9Ab0qQdGiO5hObchG8xh8Rpb4Mjyf6
0JrVo6m8GNjNj6BHkSc8fuTQJ/FaEhaQxj3pjZ3GXPrXjIIVChmICLlFuRXYzrXc
Pw0lniybypsZEVai5kO0tCNBCCFuMN9RsmmRG8mf71C4FSTbUDmxG/QlYK+0IV/l
uJkzxWa+rySkdpm0JdqumjegNRgObdXHAQDWlubWQHWyZyIQ2B4U7AxqSpcdJp6I
S4Zds4wVLd1WE5pquYQ8vS2cNlDm4QNg8wTj58e3lKN47hXHMIb6CHxRnb947oJa
pg189LLPR5koh+EorNkA1wu5mAJtJvy5YMsppy2y/kIjp3lyY6AmPT1posgGk70Z
CmToEZ5rbd7ARExtlh76A0cabMDFlEHDIK8RNUOSRr7L64+KxOUegKBfQHb9dADY
qqiKqpCbKgvtWlds909Ms74JBgr2KwZCSY1HaOxnIr4CY43QRqAq5YHOay/mU+6w
hhmdF18vpyK0vfkvvGresWtSXbag7Hkt3XjaEw76BzxQH21EBDqU8WJVjHgU6ru+
DJTs+SxgJbaT3hb/vyjlw0lK+hFfhWKRwgOXH8vqducF95NRSUxtS4fpqxWVaw3Q
V2OWSjbne99A5EPEySzryFTKbMGwaTlAwMCwYevt4YT6eb7NmFhTx0Fis4TalUs+
j+c7Kg92pDx2uQ==
=OBAt
-----END PGP PUBLIC KEY BLOCK-----
```

Per riferimento, di seguito sono riportati i dettagli della chiave pubblica.

```
Key ID:          A6310ACC4672
Type:            RSA
Size:            4096/4096
Created:         2019-09-18
Expires:         2026-07-07
User ID:         AWS CLI Team <aws-cli@amazon.com>
Key fingerprint:  FB5D B77F D5C1 18B8 0511  ADA8 A631 0ACC 4672 475C
```

c. Importa la chiave pubblica AWS CLI con il comando seguente, sostituendo *public-key-file-name* con il nome file della chiave pubblica creata.

```
$ gpg --import public-key-file-name
gpg: /home/username/.gnupg/trustdb.gpg: trustdb created
gpg: key A6310ACC4672475C: public key "AWS CLI Team <aws-cli@amazon.com>"
 imported
gpg: Total number processed: 1
gpg:               imported: 1
```

d. Scarica il file della firma AWS CLI per il pacchetto scaricato. Il percorso e il nome sono identici a quelli del file `.zip` a cui corrisponde, ma con estensione `.sig`. Negli esempi seguenti, viene salvato nella cartella corrente come un file denominato `awscliv2.sig`.

Linux x86 (64-bit)

Per la versione più recente della AWS CLI, utilizza il seguente blocco di comandi:

```
$ curl -o awscliv2.sig https://awscli.amazonaws.com/awscli-exe-linux-x86_64.zip.sig
```

Per una versione specifica della AWS CLI, aggiungi un trattino e il numero di versione al nome del file. In questo esempio, il nome del file per la versione *2.0.30* sarebbe `awscli-exe-linux-x86_64-2.0.30.zip.sig`, generando il comando seguente:

```
$ curl -o awscliv2.sig https://awscli.amazonaws.com/awscli-exe-linux-x86_64-2.0.30.zip.sig
```

Per un elenco della versioni, consulta il changelog della AWS CLI versione 2 su GitHub.

Linux ARM

Per la versione più recente della AWS CLI, utilizza il seguente blocco di comandi:

```
$ curl -o awscliv2.sig https://awscli.amazonaws.com/awscli-exe-linux-aarch64.zip.sig
```

Per una versione specifica della AWS CLI, aggiungi un trattino e il numero di versione al nome del file. In questo esempio, il nome del file per la versione *2.0.30* sarebbe `awscli-exe-linux-aarch64-2.0.30.zip.sig`, generando il comando seguente:

```
$ curl -o awscliv2.sig https://awscli.amazonaws.com/awscli-exe-linux-aarch64-2.0.30.zip.sig
```

Per un elenco della versioni, consulta il changelog della AWS CLI versione 2 su GitHub.

e. Verifica la firma, passando entrambi i nomi file `.sig` e `.zip` scaricati come parametri del comando gpg.

```
$ gpg --verify awscliv2.sig awscliv2.zip
```

L'output visualizzato dovrebbe essere simile al seguente:

```
gpg: Signature made Mon Nov  4 19:00:01 2019 PST
gpg:                using RSA key FB5D B77F D5C1 18B8 0511 ADA8 A631 0ACC 4672
 475C
gpg: Good signature from "AWS CLI Team <aws-cli@amazon.com>" [unknown]
gpg: WARNING: This key is not certified with a trusted signature!
gpg:          There is no indication that the signature belongs to the owner.
Primary key fingerprint: FB5D B77F D5C1 18B8 0511  ADA8 A631 0ACC 4672 475C
```

> **Important**
>
> L'avviso nell'output è previsto e non indica un problema. Si verifica perché non esiste una catena di trust tra la chiave PGP personale (se una è disponibile) e la chiave PGP AWS CLI. Per ulteriori informazioni, consulta Web of trust.

3. Decomprimi il programma di installazione. Se la distribuzione Linux non dispone di un comando `unzip` integrato, usa un comando equivalente per la decompressione. Il comando di esempio seguente decomprime il pacchetto e crea una directory denominata `aws` sotto la directory corrente.

```
$ unzip awscliv2.zip
```

4. Esegui il programma di installazione. Il comando di installazione utilizza un file denominato `install` nella `aws` directory appena decompressa. Per impostazione predefinita, i file vengono tutti installati in `/usr/local/aws-cli` e viene creato un collegamento simbolico in `/usr/local/bin`. Il comando include `sudo` per concedere autorizzazioni in scrittura a tali directory.

```
$ sudo ./aws/install
```

   Puoi eseguire l'installazione senza `sudo` se specifichi le directory per le quali disponi già di autorizzazioni in scrittura. Usa le seguenti istruzioni per il comando `install` per specificare il percorso di installazione:

   - Assicurati che i percorsi forniti ai parametri `-i` e `-b` non contengano nomi di volume o directory contenenti caratteri di spazio o altri caratteri di spazio vuoto. Se c'è uno spazio, l'installazione non riesce.
   - `--install-dir` o `-i`: questa opzione specifica la directory in cui vengono copiati tutti i file.

     Il valore predefinito è `/usr/local/aws-cli`.

- `--bin-dir` o `-b`: questa opzione specifica che il programma `aws` principale nella directory di installazione è collegato simbolicamente al file `aws` nel percorso specificato. È necessario disporre delle autorizzazioni di scrittura nella directory specificata. La creazione di un collegamento simbolico a una directory già presente nel percorso elimina la necessità di aggiungere la directory di installazione alla variabile `$PATH` dell'utente.

  Il valore predefinito è `/usr/local/bin`.

```
$ ./aws/install -i /usr/local/aws-cli -b /usr/local/bin
```

> **Note**
>
> Per aggiornare l'installazione corrente della AWS CLI versione 2 a una versione più recente, aggiungi il collegamento simbolico esistente e le informazioni sul programma di installazione per creare il comando `install` con il parametro `--update`.
>
> ```
> $ sudo ./aws/install --bin-dir /usr/local/bin --install-dir /usr/local/aws-cli --update
> ```
>
> Per individuare il collegamento simbolico e la directory di installazione esistenti, utilizza i seguenti passaggi:
>
> 1. Usa il comando `which` per individuare il collegamento simbolico. Questo ti fornisce il percorso da utilizzare con il parametro `--bin-dir`.
>
>    ```
>    $ which aws
>    /usr/local/bin/aws
>    ```
>
> 2. Usa il comando `ls` per individuare la directory a cui punta il collegamento simbolico. Questo ti fornisce il percorso da utilizzare con il parametro `--install-dir`.
>
>    ```
>    $ ls -l /usr/local/bin/aws
>    lrwxrwxrwx 1 ec2-user ec2-user 49 Oct 22 09:49 /usr/local/bin/aws -> /usr/local/aws-cli/v2/current/bin/aws
>    ```

5. Conferma l'installazione con il seguente comando.

```
$ aws --version
aws-cli/2.27.41 Python/3.11.6 Linux/5.10.205-195.807.amzn2.x86_64
```

Se il comando `aws` non viene trovato, potrebbe essere necessario riavviare il terminale o seguire la procedura di risoluzione dei problemi riportata in the section called "Risoluzione degli errori".

macOS

Requisiti di installazione

- Sai quale release della AWS CLI versione 2 desideri installare. Per un elenco della versioni, consulta il changelog della AWS CLI versione 2 su GitHub.
- La AWS CLI versione 2 è supportata sulle versioni di macOS a 64 bit supportate da Apple.
- Poiché AWS non gestisce repository di terze parti, non possiamo garantire che contenga l'ultima versione della AWS CLI.

Istruzioni di installazione

È possibile installare la AWS CLI versione 2 su macOS nei seguenti modi.

GUI installer

Le fasi seguenti mostrano come installare o aggiornare la versione più recente della AWS CLI versione 2 utilizzando l'interfaccia utente standard di macOS e il browser. Se si esegue l'aggiornamento alla versione più recente, utilizza lo stesso metodo di installazione impiegato per la versione corrente.

1. Nel browser, scarica la versione specifica della AWS CLI aggiungendo un trattino e il numero di versione al nome del file.

   ```
   https://awscli.amazonaws.com/AWSCLIV2-version.number.pkg
   ```

   In questo esempio, il nome del file per la versione *2.0.30* sarebbe `AWSCLIV2-2.0.30.pkg`, generando il seguente link: https://awscli.amazonaws.com/AWSCLIV2-2.0.30.pkg.

2. Esegui il file scaricato e segui le istruzioni visualizzate. È possibile scegliere di installare la AWS CLI versione 2 nei seguenti modi:

   - Per tutti gli utenti sul computer (richiede **sudo**)
     - È possibile installare in qualsiasi cartella o scegliere la cartella predefinita consigliata `/usr/local/aws-cli`.
     - Il programma di installazione crea automaticamente un collegamento simbolico `/usr/local/bin/aws` che collega il programma principale nella cartella di installazione scelta.
   - Solo per l'utente corrente (non richiede **sudo**)
     - È possibile installare in qualsiasi cartella per cui si dispone dell'autorizzazione in scrittura.
     - A causa delle autorizzazioni utente standard, al termine del programma di installazione, è necessario creare manualmente un file di collegamento simbolico nel proprio `$PATH` che punta ai programmi `aws` e `aws_completer` utilizzando i seguenti comandi al prompt dei comandi. Se `$PATH` include una cartella in cui è possibile scrivere, eseguire il comando seguente senza `sudo` se si specifica tale cartella come percorso di destinazione. Se non si dispone di una cartella scrivibile nel `$PATH`, è necessario utilizzare `sudo` nei comandi per ottenere le autorizzazioni per scrivere nella cartella di destinazione specificata. La posizione predefinita di un collegamento simbolico è `/usr/local/bin/`.

     ```
     $ sudo ln -s /folder/installed/aws-cli/aws /usr/local/bin/aws
     $ sudo ln -s /folder/installed/aws-cli/aws_completer /usr/local/bin/
     aws_completer
     ```

   > **Note**
   >
   > È possibile visualizzare i log di debug dell'installazione premendo CMD+L in qualsiasi momento nel programma di installazione. Si aprirà un riquadro di log che consente di filtrare e salvare il log. Il file di log viene salvato automaticamente in `/var/log/install.log`.

3. Per verificare che la shell possa trovare ed eseguire il comando `aws` in `$PATH`, utilizza i seguenti comandi.

   ```
   $ which aws
   ```

```
/usr/local/bin/aws
$ aws --version
aws-cli/2.27.41 Python/3.11.6 Darwin/23.3.0
```

Se il comando `aws` non viene trovato, potrebbe essere necessario riavviare il terminale o seguire la procedura di risoluzione dei problemi riportata in the section called "Risoluzione degli errori".

Command line installer - All users

Se disponi delle autorizzazioni `sudo`, è possibile installare la AWS CLI versione 2 per tutti gli utenti sul computer. Forniamo le fasi in un gruppo facile da copiare e incollare. Consulta le descrizioni di ogni riga nelle fasi seguenti.

Per una versione specifica della AWS CLI, aggiungi un trattino e il numero di versione al nome del file. In questo esempio, il nome del file per la versione *`2.0.30`* sarebbe `AWSCLIV2-2.0.30.pkg`, generando il comando seguente:

```
$ curl "https://awscli.amazonaws.com/AWSCLIV2-2.0.30.pkg" -o "AWSCLIV2.pkg"
$ sudo installer -pkg AWSCLIV2.pkg -target /
```

1. Scarica il file utilizzando il comando `curl`. L'opzione `-o` specifica il nome del file su cui viene scritto il pacchetto scaricato. In questo esempio, il file viene scritto in `AWSCLIV2.pkg` nella cartella corrente.

   Per una versione specifica della AWS CLI, aggiungi un trattino e il numero di versione al nome del file. In questo esempio, il nome del file per la versione *`2.0.30`* sarebbe `AWSCLIV2-2.0.30.pkg`, generando il comando seguente:

   ```
   $ curl "https://awscli.amazonaws.com/AWSCLIV2-2.0.30.pkg" -o "AWSCLIV2.pkg"
   ```

   Per un elenco della versioni, consulta il changelog della AWS CLI versione 2 su GitHub.
2. Esegui il programma `installer` macOS standard, specificando come origine il file `.pkg` scaricato. Utilizza il parametro `-pkg` per specificare il nome del pacchetto da installare e il parametro `-target /` per indicare l'unità in cui installare il pacchetto. I file vengono installati in `/usr/local/aws-cli` e viene creato automaticamente un collegamento simbolico in `/usr/local/bin`. È necessario includere `sudo` nel comando per concedere autorizzazioni di scrittura a queste cartelle.

```
$ sudo installer -pkg ./AWSCLIV2.pkg -target /
```

Al termine dell'installazione, i log di debug vengono scritti in `/var/log/install.log`.

3. Per verificare che la shell possa trovare ed eseguire il comando `aws` in `$PATH`, utilizza i seguenti comandi.

```
$ which aws
/usr/local/bin/aws
$ aws --version
aws-cli/2.27.41 Python/3.11.6 Darwin/23.3.0
```

Se il comando `aws` non viene trovato, potrebbe essere necessario riavviare il terminale o seguire la procedura di risoluzione dei problemi riportata in the section called "Risoluzione degli errori".

Command line - Current user

1. Per specificare la cartella in cui è installato AWS CLI, è necessario creare un file XML. Questo file è in formato XML, simile al seguente esempio. Lascia tutti i valori come indicato, sostituendo il percorso */Users/myusername* nella riga 9 con il percorso della cartella in cui desideri installare la AWS CLI versione 2. La cartella deve già esistere o il comando non riesce. Questo esempio di XML specifica che il programma di installazione sta installando la AWS CLI nella cartella `/Users/myusername`, dove crea una cartella denominata `aws-cli`.

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE plist PUBLIC "-//Apple//DTD PLIST 1.0//EN" "http://www.apple.com/DTDs/
PropertyList-1.0.dtd">
<plist version="1.0">
  <array>
    <dict>
      <key>choiceAttribute</key>
      <string>customLocation</string>
      <key>attributeSetting</key>
      <string>/Users/myusername</string>
      <key>choiceIdentifier</key>
      <string>default</string>
    </dict>
  </array>
```

```
</plist>
```

2. Scarica il programma di installazione pkg utilizzando il comando `curl`. L'opzione `-o` specifica il nome del file su cui viene scritto il pacchetto scaricato. In questo esempio, il file viene scritto in `AWSCLIV2.pkg` nella cartella corrente.

   Per la versione specifica della AWS CLI, aggiungi un trattino e il numero di versione al nome del file. In questo esempio, il nome del file per la versione *`2.0.30`* sarebbe `AWSCLIV2-2.0.30.pkg`, generando il comando seguente:

   ```
   $ curl "https://awscli.amazonaws.com/AWSCLIV2-2.0.30.pkg" -o "AWSCLIV2.pkg"
   ```

   Per un elenco della versioni, consulta il changelog della AWS CLI versione 2 su GitHub.

3. Esegui il programma `installer` macOS standard con le seguenti opzioni:

   - Specifica il nome del pacchetto da installare utilizzando il parametro `-pkg`.
   - Specifica un'installazione solo per l'utente corrente impostando il parametro `-target` su `CurrentUserHomeDirectory`.
   - Specifica il percorso (relativo della cartella corrente) e il nome del file XML creato nel parametro `-applyChoiceChangesXML`.

   Nell'esempio seguente, la AWS CLI viene installata nella cartella `/Users/myusername/aws-cli`.

   ```
   $ installer -pkg AWSCLIV2.pkg \
               -target CurrentUserHomeDirectory \
               -applyChoiceChangesXML choices.xml
   ```

4. Poiché le autorizzazioni utente standard in genere non consentono la scrittura nelle cartelle in `$PATH`, il programma di installazione in questa modalità non tenta di aggiungere i collegamenti simbolici ai programmi `aws` e `aws_completer`. Affinché la AWS CLI funzioni correttamente, è necessario creare manualmente i collegamenti simbolici al termine del programma di installazione. Se la `$PATH` include una cartella in cui è possibile scrivere, e si specifica questa cartella come percorso di destinazione, è possibile eseguire il comando seguente senza `sudo`. Se non si dispone di una cartella scrivibile in `$PATH`, è necessario utilizzare `sudo` per le autorizzazioni per scrivere nella cartella di destinazione specificata. La posizione predefinita di un collegamento simbolico è `/usr/local/bin/`.

```
$ sudo ln -s /folder/installed/aws-cli/aws /usr/local/bin/aws
$ sudo ln -s /folder/installed/aws-cli/aws_completer /usr/local/bin/
aws_completer
```

Al termine dell'installazione, i log di debug vengono scritti in `/var/log/install.log`.

5. Per verificare che la shell possa trovare ed eseguire il comando `aws` in `$PATH`, utilizza i seguenti comandi.

```
$ which aws
/usr/local/bin/aws
$ aws --version
aws-cli/2.27.41 Python/3.11.6 Darwin/23.3.0
```

Se il comando `aws` non viene trovato, potrebbe essere necessario riavviare il terminale o seguire la procedura di risoluzione dei problemi riportata in the section called "Risoluzione degli errori".

Windows

Requisiti di installazione

- Sai quale release della AWS CLI versione 2 desideri installare. Per un elenco della versioni, consulta il changelog della AWS CLI versione 2 su GitHub.
- La AWS CLI è supportata sulle versioni di Windows a 64 bit supportate da Microsoft.
- Diritti di amministratore per installare il software

Istruzioni di installazione

Per aggiornare l'installazione corrente della AWS CLI versione 2 su Windows, scarica un nuovo programma di installazione ogni volta che esegui l'aggiornamento per sovrascrivere le versioni precedenti. La AWS CLI viene aggiornata regolarmente. Per vedere quando è stata rilasciata l'ultima versione, consulta il changelog della AWS CLI versione 2 su GitHub.

1. Scarica ed esegui il programma di installazione MSI della AWS CLI per Windows (64 bit) in uno dei seguenti modi:

- Download ed esecuzione del programma di installazione MSI: per creare il link di download di una versione specifica della AWS CLI, aggiungi un trattino e il numero di versione al nome del file.

  ```
  https://awscli.amazonaws.com/AWSCLIV2-version.number.msi
  ```

  In questo esempio, il nome del file per la versione *2.0.30* sarebbe `AWSCLIV2-2.0.30.msi`, generando il seguente link: https://awscli.amazonaws.com/AWSCLIV2-2.0.30.msi.

- Uso del comando msiexec: in alternativa, è possibile utilizzare il programma di installazione MSI aggiungendo il link al comando `msiexec`. Per una versione specifica della AWS CLI, aggiungi un trattino e il numero di versione al nome del file.

  ```
  C:\> msiexec.exe /i https://awscli.amazonaws.com/AWSCLIV2-version.number.msi
  ```

  In questo esempio, il nome del file per la versione *2.0.30* sarebbe `AWSCLIV2-2.0.30.msi`, generando il seguente link: https://awscli.amazonaws.com/AWSCLIV2-2.0.30.msi.

  ```
  C:\> msiexec.exe /i https://awscli.amazonaws.com/AWSCLIV2-2.0.30.msi
  ```

  Per i vari parametri utilizzabili con `msiexec`, consulta msiexec nel sito web Microsoft Docs.

Per un elenco della versioni, consulta il changelog della AWS CLI versione 2 su GitHub.

2. Per confermare l'installazione, apri il menu Start, ricerca `cmd` per aprire una finestra dei prompt dei comandi e, al prompt dei comandi, utilizza il comando `aws --version`.

   ```
   C:\> aws --version
   aws-cli/2.27.41 Python/3.11.6 Windows/10 exe/AMD64 prompt/off
   ```

   Se Windows non riesce a trovare il programma, potrebbe essere necessario chiudere e riaprire la finestra del prompt dei comandi per aggiornare il percorso oppure seguire la procedura di risoluzione dei problemi riportata in the section called "Risoluzione degli errori".

Risoluzione degli errori di installazione e disinstallazione della AWS CLI

Se riscontri problemi dopo l'installazione o la disinstallazione della AWS CLI, consulta la procedura the section called "Risoluzione degli errori" per la risoluzione dei problemi. Per le procedure di risoluzione dei problemi più importanti, consulta the section called "Errori di comando non trovato", the section called "Il comando `aws --version` restituisce una versione diversa da quella installata" e the section called "Il comando "`aws --version`" restituisce una versione dopo la disinstallazione di AWS CLI".

Passaggi successivi

Dopo aver completato i passaggi descritti in the section called "Prerequisiti" e aver installato la AWS CLI, è necessario eseguire la configurazione come indicato in the section called "Configurazione".

Compilazione e installazione di AWS CLI dall'origine

Questo argomento descrive come installare o aggiornare dall'origine l'ultima versione di AWS Command Line Interface (AWS CLI) sui sistemi operativi supportati.

Per informazioni sulle ultime versioni della AWS CLI, consulta AWS CLI version 2 Changelog su GitHub.

> **Important**
>
> L'AWS CLI versioni 1 e 2 usa lo stesso nome del comando `aws`. Se in precedenza è stato installato AWS CLI versione 1, consulta Guida alla migrazione per la AWS CLI versione 2.

Argomenti

- Perché compilare dall'origine?
- Passaggi rapidi
- Passaggio 1: configurare tutti i requisiti
- Passaggio 2: configurare l'installazione dell'origine di AWS CLI
- Passaggio 3: compilare AWS CLI
- Passaggio 4: installare AWS CLI

- Passaggio 5: verificare l'installazione di AWS CLI
- Esempi di flussi di lavoro
- Risoluzione degli errori di installazione e disinstallazione della AWS CLI
- Passaggi successivi

Perché compilare dall'origine?

AWS CLI è disponibile con programmi di installazione predefiniti per la maggior parte delle piattaforme e degli ambienti, nonché come immagine Docker.

In genere, questi programmi di installazione forniscono la copertura per la maggior parte dei casi d'uso. Le istruzioni per l'installazione dall'origine servono a risolvere i casi d'uso non coperti dai programmi di installazione. Alcuni di questi casi d'uso sono i seguenti:

- I programmi di installazione predefiniti non supportano l'ambiente in uso. Ad esempio, ARM a 32 bit non è supportato dai programmi di installazione predefiniti.
- I programmi di installazione predefiniti hanno dipendenze che non sono presenti nell'ambiente. Ad esempio, Alpine Linux utilizza `musl`, ma gli attuali programmi di installazione richiedono `glibc`, il che impedisce il funzionamento immediato dei programmi di installazione predefiniti.
- I programmi di installazione predefiniti richiedono risorse a cui l'ambiente limita l'accesso. Ad esempio, i sistemi con hardening della sicurezza potrebbero non fornire le autorizzazioni per la memoria condivisa. Questo comportamento è necessario per il programma di installazione `aws` bloccato.
- I programmi di installazione predefiniti spesso rappresentano un ostacolo per i manutentori dei gestori di pacchetti, in quanto è preferibile avere il pieno controllo sul processo di compilazione del codice e dei pacchetti. La compilazione dall'origine consente ai manutentori della distribuzione di utilizzare un processo più semplice per mantenere AWS CLI aggiornato. Il supporto per i manutentori fornisce ai clienti le versioni aggiornate di AWS CLI quando installano da un gestore di pacchetti di terze parti come `brew`, `yum` e `apt`.
- I clienti che applicano la patch della funzionalità AWS CLI richiedono la compilazione e l'installazione di AWS CLI dall'origine. Questo approccio è particolarmente importante per i membri della community che desiderano testare le modifiche apportate all'origine prima di fornirle al repository AWS CLI in GitHub.

Passaggi rapidi

> **Note**
>
> Si presume che tutti gli esempi di codice vengano eseguiti dalla root della directory di origine.

Per compilare e installare AWS CLI dall'origine, seguire i passaggi indicati in questa sezione. AWS CLI utilizza GNU Autotools per l'installazione dall'origine. Nel caso più semplice, AWS CLI può essere installato dall'origine eseguendo i comandi di esempio predefiniti dalla root del repository AWS CLI in GitHub.

1. Impostare tutti i requisiti per l'ambiente. Ciò include la possibilità di eseguire file generati da GNU Autotools e installare Python 3.8 o versioni successive.
2. Nel terminale, passare al livello superiore della cartella di origine AWS CLI ed eseguire il comando `./configure`. Questo comando verifica la presenza di tutte le dipendenze richieste nel sistema e genera un `Makefile` per la compilazione e l'installazione di AWS CLI in base alle configurazioni rilevate e specificate.

   Linux and macOS

   Il seguente comando `./configure` di esempio imposta la configurazione della compilazione per AWS CLI utilizzando le impostazioni predefinite.

   ```
   $ ./configure
   ```

   Windows PowerShell

   Prima di eseguire qualsiasi comando che chiami MSYS2, è necessario proteggere la directory di lavoro corrente:

   ```
   PS C:\> $env:CHERE_INVOKING = 'yes'
   ```

   Quindi utilizzare il seguente comando `./configure` di esempio per impostare la configurazione della compilazione per AWS CLIutilizzando il percorso locale dell'eseguibile Python, installando in C:\Program Files\AWSCLI e scaricando tutte le dipendenze.

```
PS C:\> C:\msys64\usr\bin\bash  -lc " PYTHON='C:\path\to\python.exe' ./configure
 --prefix='C:\Program Files\AWSCLI' --with-download-deps "
```

Per i dettagli, le opzioni di configurazione disponibili e le informazioni sulle impostazioni predefinite, consulta la sezione the section called "Passaggio 2: configurare l'installazione dell'origine di AWS CLI".

3. Esegui il comando `make`. Questo comando compila AWS CLI in base alle impostazioni di configurazione.

   Il seguente comando `make` di esempio compila con le opzioni predefinite utilizzando le impostazioni `./configure` esistenti.

   Linux and macOS

   ```
   $ make
   ```

   Windows PowerShell

   ```
   PS C:\> C:\msys64\usr\bin\bash -lc "make"
   ```

   Per i dettagli e le opzioni di compilazione disponibili, consulta la sezione the section called "Passaggio 3: compilare AWS CLI".

4. Esegui il comando `make install`. Questo comando installa la compilazione di AWS CLI nella posizione configurata del sistema.

   Il seguente comando `make install` di esempio installa la compilazione di AWS CLI e crea collegamenti simbolici nelle posizioni configurate utilizzando le impostazioni di comando predefinite.

   Linux and macOS

   ```
   $ make install
   ```

Windows PowerShell

```
PS C:\> C:\msys64\usr\bin\bash -lc "make install"
```

Dopo l'installazione, aggiungere il percorso ad AWS CLI utilizzando il seguente comando:

```
PS C:\> $Env:PATH +=";C:\Program Files\AWSCLI\bin\"
```

Per i dettagli e le opzioni di installazione disponibili, consulta la sezione the section called "Passaggio 4: installare AWS CLI".

5. Confermare di aver installato AWS CLI correttamente utilizzando il seguente comando:

```
$ aws --version
aws-cli/2.27.41 Python/3.11.6 Windows/10 exe/AMD64 prompt/off
```

Per il passaggi della risoluzione dei problemi di installazione, consulta la sezione the section called "Risoluzione degli errori di installazione e disinstallazione della AWS CLI".

Passaggio 1: configurare tutti i requisiti

Per compilare AWS CLI dall'origine è necessario completare prima le seguenti operazioni:

> **Note**
>
> Si presume che tutti gli esempi di codice vengano eseguiti dalla root della directory di origine.

1. Scaricare l'origine AWS CLI eseguendo il fork del repository AWS CLI in GitHub o scaricando il tarball di origine. Seguire una delle seguenti istruzioni:

   - Eseguire il fork e clonare il repository AWS CLI da GitHub. Per ulteriori informazioni, consulta Fork a repo nella Documentazione di GitHub.
   - Scaricare il tarball di origine più recente da https://awscli.amazonaws.com/awscli.tar.gz ed estrarne il contenuto utilizzando i seguenti comandi:

     ```
     $ curl "https://awscli.amazonaws.com/awscli.tar.gz" -o "awscli.tar.gz"
     ```

```
$ tar -xzf awscli.tar.gz
```

> **Note**
>
> Per scaricare una versione specifica, utilizzare il seguente formato di collegamento: https://awscli.amazonaws.com/awscli-*versionnumber*.tar.gz
> Ad esempio, per la versione 2.10.0 il collegamento è: https://awscli.amazonaws.com/awscli-*2.10.0*.tar.gz
> Le versioni di origine sono disponibili a partire dalla versione 2.10.0 di AWS CLI.

(Facoltativo) Verificare l'integrità del file zip scaricato completando i seguenti passaggi:

1. Puoi utilizzare le fasi seguenti per verificare le firme tramite lo strumento GnuPG.

   I file `.zip` del pacchetto di installazione di AWS CLI sono firmati crittograficamente mediante le firme PGP. Se i file sono in qualche modo danneggiati o alterati, questa verifica non va a buon fine e l'installazione deve essere arrestata.

2. Scaricare e installare il comando gpg utilizzando il programma di gestione dei pacchetti. Per ulteriori informazioni su GnuPG, consulta il sito Web GnuPG.

3. Per creare il file della chiave pubblica, creare un file di testo e incollare il testo seguente.

```
-----BEGIN PGP PUBLIC KEY BLOCK-----

mQINBF2Cr7UBEADJZHcgusOJl7ENSyumXh85z0TRV0xJorM2B/JL0kHOyigQluUG
ZMLhENaG0bYatdrKP+3H91lvK050pXwnO/R7fB/FSTouki4ciIx5OuLlnJZIxSzx
PqGl0mkxImLNbGWoi6Lto0LYxqHN2iQtzlwTVmq9733zd3XfcXrZ3+LblHAgEt5G
TfNxEKJ8soPLyWmwDH6HWCnjZ/aIQRBTIQ05uVeEoYxSh6wOai7ss/KveoSNBbYz
gbdzoqI2Y8cgH2nbfgp3DSasaLZEdCSsIsK1u05CinE7k2qZ7KgKAUIcT/cR/grk
C6VwsnDU0OUCideXcQ8WeHutqvgZH1JgKDbznoIzeQHJD238GEu+eKhRHcz8/jeG
94zkcgJOz3KbZGYMiTh277Fvj9zzvZsbMBCedV1BTg3TqgvdX4bdkhf5cH+7NtWO
lrFj6UwAsGukBTAOxC0l/dnSmZhJ7Z1KmEWilro/gOrjtOxqRQutlIqG22TaqoPG
fYVN+en3Zwbt97kcgZDwqbuykNt64oZWc4XKCa3mprEGC3IbJTBFqglXmZ7l9ywG
EEUJYOlb2XrSuPWml39beWdKM8kzr1OjnlOm6+lpTRCBfo0wa9F8YZRhHPAkwKkX
XDeOGpWRj4ohOx0d2GWkyV5xyN14p2tQOCdOODmz80yUTgRpPVQUtOEhXQARAQAB
tCFBV1MgQ0xJIFRlYW0gPGF3cy1jbGlAYW1hem9uLmNvbT6JAlQEEwEIAD4CGwMF
CwkIBwIGFQoJCAsCBBYCAwECHgECF4AWIQT7Xbd/1cEYuAURraimMQrMRnJHXAUC
aGveYQUJDMpiLAAKCRCmMQrMRnJHXKBYD/9Ab0qQdGiO5hObchG8xh8Rpb4Mjyf6
0JrVo6m8GNjNj6BHkSc8fuTQJ/FaEhaQxj3pjZ3GXPrXjIIVChmICLlFuRXYzrXc
Pw0lniybypsZEVai5kO0tCNBCCFuMN9RsmmRG8mf7lC4FSTbUDmxG/QlYK+0IV/l
```

```
uJkzxWa+rySkdpm0JdqumjegNRgObdXHAQDWlubWQHWyZyIQ2B4U7AxqSpcdJp6I
S4Zds4wVLd1WE5pquYQ8vS2cNlDm4QNg8wTj58e3lKN47hXHMIb6CHxRnb947oJa
pg189LLPR5koh+EorNkA1wu5mAJtJvy5YMsppy2y/kIjp3lyY6AmPT1posgGk70Z
CmToEZ5rbd7ARExtlh76A0cabMDFlEHDIK8RNUOSRr7L64+KxOUegKBfQHb9dADY
qqiKqpCbKgvtWlds909Ms74JBgr2KwZCSY1HaOxnIr4CY43QRqAq5YHOay/mU+6w
hhmdF18vpyK0vfkvvGresWtSXbag7Hkt3XjaEw76BzxQH21EBDqU8WJVjHgU6ru+
DJTs+SxgJbaT3hb/vyjlw0lK+hFfhWKRwgOXH8vqducF95NRSUxtS4fpqxWVaw3Q
V2OWSjbne99A5EPEySzryFTKbMGwaTlAwMCwYevt4YT6eb7NmFhTx0Fis4TalUs+
j+c7Kg92pDx2uQ==
=OBAt
-----END PGP PUBLIC KEY BLOCK-----
```

Per riferimento, di seguito sono riportati i dettagli della chiave pubblica.

```
Key ID:          A6310ACC4672
Type:            RSA
Size:            4096/4096
Created:         2019-09-18
Expires:         2026-07-07
User ID:         AWS CLI Team <aws-cli@amazon.com>
Key fingerprint:  FB5D B77F D5C1 18B8 0511  ADA8 A631 0ACC 4672 475C
```

4. Importare la chiave pubblica AWS CLI con il comando seguente, sostituendo *public-key-file-name* con il nome file della chiave pubblica creata.

```
$ gpg --import public-key-file-name
gpg: /home/username/.gnupg/trustdb.gpg: trustdb created
gpg: key A6310ACC4672475C: public key "AWS CLI Team <aws-cli@amazon.com>"
 imported
gpg: Total number processed: 1
gpg:               imported: 1
```

5. Scaricare il file della firma AWS CLI per il pacchetto scaricato da https://awscli.amazonaws.com/awscli.tar.gz.sig. Il percorso e il nome sono identici a quelli del file tarball corrispondente, ma con estensione `.sig`. Salvare il file nello stesso percorso del file tarball. In alternativa, utilizzare il seguente blocco di comandi:

```
$ curl awscliv2.sig https://awscli.amazonaws.com/ -o awscli.tar.gz.sig
```

6. Verifica la firma, passando entrambi i nomi file `.sig` e `.zip` scaricati come parametri del comando gpg.

```
$ gpg --verify awscliv2.sig awscli.tar.gz
```

L'output visualizzato dovrebbe essere simile al seguente:

```
gpg: Signature made Mon Nov  4 19:00:01 2019 PST
gpg:                using RSA key FB5D B77F D5C1 18B8 0511 ADA8 A631 0ACC 4672
 475C
gpg: Good signature from "AWS CLI Team <aws-cli@amazon.com>" [unknown]
gpg: WARNING: This key is not certified with a trusted signature!
gpg:          There is no indication that the signature belongs to the owner.
Primary key fingerprint: FB5D B77F D5C1 18B8 0511  ADA8 A631 0ACC 4672 475C
```

> **Important**
>
> L'avviso nell'output è previsto e non indica un problema. Si verifica perché non esiste una catena di trust tra la chiave PGP personale (se una è disponibile) e la chiave PGP AWS CLI. Per ulteriori informazioni, consulta Web of trust.

2. È disponibile un ambiente in grado di eseguire file generati da GNU Autotools come `configure` e `Makefile`. Questi file sono ampiamente portabili nelle piattaforme POSIX.

   Linux and macOS

   Se Autotools non è installato nell'ambiente o è necessario aggiornarlo, seguire le istruzioni di installazione disponibili in How do I install the Autotools (as user)? o Basic Installation nella Documentazione di GNU.

   Windows PowerShell

   > **Warning**
   >
   > Se si utilizza un ambiente Windows, è consigliabile utilizzare i programmi di installazione predefiniti. Per le istruzioni di installazione dei programmi di installazione predefiniti, consulta the section called "Installazione o aggiornamento".

   Poiché Windows non dispone di una shell compatibile con POSIX, è necessario installare un software aggiuntivo per installare AWS CLI dall'origine. MSYS2 fornisce una raccolta di

strumenti e librerie per aiutare a compilare e installare software Windows, in particolare per lo scripting basato su POSIX utilizzato da Autotools.

1. Installare MSYS2. Per informazioni sull'installazione e sull'utilizzo di MSYS2, consulta le istruzioni di installazione e utilizzo nella Documentazione di MSYS2.
2. Aprire il terminale MSYS2 e installare Autotools utilizzando il seguente comando.

```
$ pacman -S autotools
```

> **Note**
>
> Quando si utilizzano gli esempi di codice di configurazione, compilazione e installazione riportati in questa guida per Windows, si presuppone che il percorso di installazione predefinito di MSYS2 sia `C:\msys64\usr\bin\bash`. Quando si chiama MSYS2 all'interno di PowerShell, si utilizza il seguente formato, con il comando bash tra virgolette:
>
> ```
> PS C:\> C:\msys64\usr\bin\bash  -lc "command example"
> ```
>
> Il seguente comando di esempio chiama il comando `./configure`.
>
> ```
> PS C:\> C:\msys64\usr\bin\bash  -lc "./configure"
> ```

3. È installato un interprete Python 3.8 o versioni successive. La versione minima di Python richiesta segue le tempistiche ufficiali indicate in Python support policy for AWS SDKs and Tools. Un interprete è supportato solo per 6 mesi dopo la data di fine supporto.
4. (Facoltativo) Installare tutte le dipendenze della libreria Python di compilazione e runtime di AWS CLI. Il comando `./configure` informa l'utente se mancano delle dipendenze e come installarle.

   È possibile installare e utilizzare automaticamente queste dipendenze tramite la configurazione, consulta the section called "Download delle dipendenze" per ulteriori informazioni.

Passaggio 2: configurare l'installazione dell'origine di AWS CLI

La configurazione per la compilazione e l'installazione di AWS CLI viene specificata utilizzando lo script `configure`. Per la documentazione di tutte le opzioni di configurazione, eseguire lo script `configure` con l'opzione `--help`:

Linux and macOS

```
$ ./configure --help
```

Windows PowerShell

```
PS C:\> C:\msys64\usr\bin\bash -lc "./configure --help"
```

Le opzioni più importanti sono:

- Posizione di installazione
- Interprete Python
- Download delle dipendenze
- Tipo di installazione

Posizione di installazione

L'installazione dell'origine di AWS CLI utilizza due directory configurabili per installare AWS CLI.

- `libdir`: directory principale in cui verrà installato AWS CLI. Il percorso di installazione di AWS CLI è *<libdir-value>*`/aws-cli`. Il valore `libdir` predefinito per Linux e macOS è `/usr/local/lib`, pertanto la directory di installazione predefinita è */usr/local/lib*`/aws-cli`.
- `bindir`: directory in cui sono installati gli eseguibili AWS CLI. Il percorso predefinito è `/usr/local/bin`.

Le seguenti opzioni `configure` controllano le directory utilizzate:

- `--prefix`: imposta il prefisso di directory da utilizzare per l'installazione. Il valore predefinito per Linux e macOS è `/usr/local`.

- `--libdir`: imposta la `libdir` da utilizzare per l'installazione di AWS CLI. Il valore predefinito è `<prefix-value>/lib`. Se entrambi `--libdir` e `--prefix` non sono specificati, l'impostazione predefinita per Linux e macOS è `/usr/local/lib/`.
- `--bindir`: imposta la `bindir` da utilizzare per l'installazione degli eseguibili AWS CLI `aws` e `aws_completer`. Il valore predefinito è `<prefix-value>/bin`. Se entrambi `bindir` e `--prefix` non sono specificati, l'impostazione predefinita per Linux e macOS è `/usr/local/bin/`.

Linux and macOS

Il seguente comando di esempio utilizza l'opzione `--prefix` per eseguire un'installazione utente locale di AWS CLI. Questo comando installa AWS CLI in `$HOME/.local/lib/aws-cli` e gli eseguibili in `$HOME/.local/bin`:

```
$ ./configure --prefix=$HOME/.local
```

Il seguente comando di esempio utilizza l'opzione `--libdir` per installare AWS CLI come applicazione aggiuntiva nella directory `/opt`. Questo comando installa AWS CLI in `/opt/aws-cli` e gli eseguibili nella posizione predefinita `/usr/local/bin`.

```
$ ./configure --libdir=/opt
```

Windows PowerShell

Il seguente comando di esempio utilizza l'opzione `--prefix` per eseguire un'installazione utente locale di AWS CLI. Questo comando installa AWS CLI in `$HOME/.local/lib/aws-cli` e gli eseguibili in `$HOME/.local/bin`:

```
$ C:\msys64\usr\bin\bash -lc "./configure --prefix='C:\Program Files\AWSCLI'"
```

Il seguente comando di esempio utilizza l'opzione `--libdir` per installare AWS CLI come applicazione aggiuntiva nella directory `/opt`. Questo comando installa AWS CLI in `C:\Program Files\AWSCLI\opt\aws-cli`.

Interprete Python

> **Note**
>
> Si consiglia vivamente di specificare l'interprete Python durante l'installazione per Windows.

Lo script `./configure` seleziona automaticamente un interprete Python 3.8 o versioni successive installato da utilizzare per la compilazione e l'esecuzione di AWS CLI utilizzando la macro Autoconf AM_PATH_PYTHON.

L'interprete Python da utilizzare può essere impostato in modo esplicito utilizzando la variabile di ambiente PYTHON durante l'esecuzione dello script `configure`:

Linux and macOS

```
$ PYTHON=/path/to/python ./configure
```

Windows PowerShell

```
PS C:\> C:\msys64\usr\bin\bash -lc "PYTHON='C:\path\to\python' ./configure"
```

Download delle dipendenze

Per impostazione predefinita, è necessario che tutte le dipendenze di compilazione e runtime di AWS CLI siano già installate nel sistema. Sono incluse le dipendenze della libreria Python. Tutte le dipendenze vengono controllate durante l'esecuzione dello script `configure` e se nel sistema mancano delle dipendenze Python, lo script `configure` genera errori.

Il seguente codice di esempio genera errori quando nel sistema mancano le dipendenze:

Linux and macOS

```
$ ./configure
checking for a Python interpreter with version >= 3.8... python
checking for python... /Users/username/.envs/env3.11/bin/python
checking for python version... 3.11
checking for python platform... darwin
checking for GNU default python prefix... ${prefix}
```

```
checking for GNU default python exec_prefix... ${exec_prefix}
checking for python script directory (pythondir)... ${PYTHON_PREFIX}/lib/python3.11/
site-packages
checking for python extension module directory (pyexecdir)... ${PYTHON_EXEC_PREFIX}/
lib/python3.11/site-packages
checking for --with-install-type... system-sandbox
checking for --with-download-deps... Traceback (most recent call last):
  File "<frozen runpy>", line 198, in _run_module_as_main
  File "<frozen runpy>", line 88, in _run_code
  File "/Users/username/aws-code/aws-cli/./backends/build_system/__main__.py", line
 125, in <module>
    main()
  File "/Users/username/aws-code/aws-cli/./backends/build_system/__main__.py", line
 121, in main
    parsed_args.func(parsed_args)
  File "/Users/username/aws-code/aws-cli/./backends/build_system/__main__.py", line
 49, in validate
    validate_env(parsed_args.artifact)
  File "/Users/username/aws-code/aws-cli/./backends/build_system/validate_env.py",
 line 68, in validate_env
    raise UnmetDependenciesException(unmet_deps, in_venv)
validate_env.UnmetDependenciesException: Environment requires following Python
 dependencies:

awscrt (required: ('>=0.12.4', '<0.17.0')) (version installed: None)

We recommend using --with-download-deps flag to automatically create a virtualenv
 and download the dependencies.

If you want to manage the dependencies yourself instead, run the following pip
 command:
/Users/username/.envs/env3.11/bin/python -m pip install --prefer-binary
 'awscrt>=0.12.4,<0.17.0'

configure: error: "Python dependencies not met."
```

Windows PowerShell

```
PS C:\> C:\msys64\usr\bin\bash -lc "./configure"
checking for a Python interpreter with version >= 3.8... python
checking for python... /Users/username/.envs/env3.11/bin/python
checking for python version... 3.11
checking for python platform... darwin
```

```
checking for GNU default python prefix... ${prefix}
checking for GNU default python exec_prefix... ${exec_prefix}
checking for python script directory (pythondir)... ${PYTHON_PREFIX}/lib/python3.11/
site-packages
checking for python extension module directory (pyexecdir)... ${PYTHON_EXEC_PREFIX}/
lib/python3.11/site-packages
checking for --with-install-type... system-sandbox
checking for --with-download-deps... Traceback (most recent call last):
  File "<frozen runpy>", line 198, in _run_module_as_main
  File "<frozen runpy>", line 88, in _run_code
  File "/Users/username/aws-code/aws-cli/./backends/build_system/__main__.py", line
 125, in <module>
    main()
  File "/Users/username/aws-code/aws-cli/./backends/build_system/__main__.py", line
 121, in main
    parsed_args.func(parsed_args)
  File "/Users/username/aws-code/aws-cli/./backends/build_system/__main__.py", line
 49, in validate
    validate_env(parsed_args.artifact)
  File "/Users/username/aws-code/aws-cli/./backends/build_system/validate_env.py",
 line 68, in validate_env
    raise UnmetDependenciesException(unmet_deps, in_venv)
validate_env.UnmetDependenciesException: Environment requires following Python
 dependencies:

awscrt (required: ('>=0.12.4', '<0.17.0')) (version installed: None)

We recommend using --with-download-deps flag to automatically create a virtualenv
 and download the dependencies.

If you want to manage the dependencies yourself instead, run the following pip
 command:
/Users/username/.envs/env3.11/bin/python -m pip install --prefer-binary
 'awscrt>=0.12.4,<0.17.0'

configure: error: "Python dependencies not met."
```

Per installare automaticamente le dipendenze Python richieste, utilizzare l'opzione `--with-download-deps`. Quando si utilizza questo flag, il processo di compilazione esegue le seguenti operazioni:

- Salta il controllo delle dipendenze della libreria Python.

- Configura le impostazioni per scaricare tutte le dipendenze Python richieste e utilizzare solo le dipendenze scaricate per compilare AWS CLI durante la compilazione di `make`.

Il seguente comando configure di esempio utilizza l'opzione `--with-download-deps` per scaricare e utilizzare le dipendenze Python:

Linux and macOS

```
$ ./configure --with-download-deps
```

Windows PowerShell

```
PS C:\> C:\msys64\usr\bin\bash -lc "./configure --with-download-deps"
```

Tipo di installazione

Il processo di installazione dall'origine supporta i seguenti tipi di installazione:

- `system-sandbox`: (impostazione predefinita) crea un ambiente virtuale Python isolato, installa AWS CLI nell'ambiente virtuale ed effettua il collegamento simbolico agli eseguibili `aws` e `aws_completer` nell'ambiente virtuale. Questa installazione di AWS CLI dipende direttamente dall'interprete Python selezionato per il runtime.

  Questo è un meccanismo di installazione leggero per installare AWS CLI su un sistema e segue le best practice di Python eseguendo il sandbox dell'installazione in un ambiente virtuale. Questa installazione è destinata ai clienti che desiderano installare AWS CLI dall'origine nel modo più semplice possibile, con l'installazione abbinata all'installazione di Python.
- `portable-exe`: blocca AWS CLI in un eseguibile standalone che può essere distribuito in ambienti con architetture simili. Questo è lo stesso processo utilizzato per generare gli eseguibili ufficiali predefiniti di AWS CLI. `portable-exe` blocca in una copia dell'interprete Python scelto nel passaggio `configure` da utilizzare per il runtime di AWS CLI. Questo consente di spostarlo su altri computer che potrebbero non avere un interprete Python.

  Questo tipo di compilazione è utile perché è possibile assicurarsi che l'installazione di AWS CLI non sia abbinata alla versione di Python installata nell'ambiente e distribuire una compilazione su altri sistemi in cui potrebbe non essere installato Python. Questo approccio consente di controllare le dipendenze e la sicurezza degli eseguibili AWS CLI utilizzati.

Per configurare il tipo di installazione, utilizzare l'opzione `--with-install-type` e specificare il valore `portable-exe` o `system-sandbox`.

Il seguente comando `./configure` di esempio specifica il valore `portable-exe`:

Linux and macOS

```
$ ./configure --with-install-type=portable-exe
```

Windows PowerShell

```
PS C:\> C:\msys64\usr\bin\bash -lc "./configure --with-install-type=portable-exe"
```

Passaggio 3: compilare AWS CLI

Utilizzare il comando `make` per compilare AWS CLI utilizzando le impostazioni di configurazione:

Linux and macOS

```
$ make
```

Windows PowerShell

```
PS C:\> C:\msys64\usr\bin\bash -lc "make"
```

> **Note**
>
> Quando si utilizza il comando **`make`**, i seguenti passaggi vengono completati in background:
>
> 1. Un ambiente virtuale viene creato nella directory di compilazione utilizzando il modulo Python `venv`. L'ambiente virtuale viene avviato con una versione di pip fornita nella libreria standard Python.
> 2. Le dipendenze della libreria Python vengono copiate. A seconda se il flag `--with-download-deps` è stato specificato nel comando `configure`, questo passaggio esegue una delle seguenti operazioni:

- `--with-download-deps` è specificato. Le dipendenze Python eseguono l'installazione pip. Sono incluse `wheel`, `setuptools` e tutte le dipendenze di runtime AWS CLI. Se si sta compilando `portable-exe`, viene installato `pyinstaller`. Questi requisiti sono tutti specificati nei file di blocco generati da `pip-compile`.
- `--with-download-deps` non è specificato. Le librerie Python del pacchetto del sito dell'interprete Python più eventuali script (ad esempio `pyinstaller`) vengono copiati nell'ambiente virtuale utilizzato per la compilazione.

3. `pip install` viene eseguito direttamente sulla codebase AWS CLI per eseguire una compilazione offline ad albero e AWS CLI viene installato nell'ambiente virtuale di compilazione. Questa installazione utilizza i flag pip --no-build-isolation , --use-feature=in-tree-build , --no-cache-dir e `--no-index`.
4. (Facoltativo) Se `--install-type` è impostato su `portable-exe` nel comando `configure`, viene compilato un eseguibile standalone utilizzando `pyinstaller`.

Passaggio 4: installare AWS CLI

Il comando `make install` installa la compilazione di AWS CLI nella posizione configurata del sistema.

Linux and macOS

Il seguente comando di esempio installa AWS CLI utilizzando le impostazioni di configurazione e compilazione:

```
$ make install
```

Windows PowerShell

Il seguente comando di esempio installa AWS CLI utilizzando le impostazioni di configurazione e compilazione, quindi aggiunge una variabile di ambiente con il percorso per AWS CLI:

```
PS C:\> C:\msys64\usr\bin\bash -lc " make install "
PS C:\> $Env:PATH +=";C:\Program Files\AWSCLI\bin\"
```

La regola `make install` supporta la variabile `DESTDIR`. Quando è specificata, questa variabile imposta per il percorso specificato il prefisso del percorso di installazione già configurato durante

l'installazione di AWS CLI. Per impostazione predefinita, non è impostato alcun valore per questa variabile.

Linux and macOS

Il seguente codice di esempio utilizza un flag `--prefix=/usr/local` per configurare una posizione di installazione, quindi modifica tale destinazione utilizzando `DESTDIR=/tmp/stage` per il comando `make install`. Questi comandi eseguono l'installazione di AWS CLI in */tmp/stage/*`usr/local/lib/aws-cli` e posizionano i relativi eseguibili in */tmp/stage/*`usr/local/bin`.

```
$ ./configure --prefix=/usr/local
$ make
$ make DESTDIR=/tmp/stage install
```

Windows PowerShell

Il seguente codice di esempio utilizza un flag `--prefix=\awscli` per configurare una posizione di installazione, quindi modifica tale destinazione utilizzando `DESTDIR=C:\Program Files` per il comando `make install`. Questi comandi eseguono l'installazione di AWS CLI in *C:\Program Files*`\awscli`.

```
$ ./configure --prefix=\awscli
$ make
$ make DESTDIR='C:\Program Files' install
```

> **Note**
>
> Durante l'esecuzione di **`make install`**, i seguenti passaggi vengono completati in background
>
> 1. Sposta uno dei seguenti elementi nella directory di installazione configurata:
>   - Se il tipo di installazione è `system-sandbox`, sposta l'ambiente virtuale compilato.
>   - Se il tipo di installazione è `portable-exe`, sposta l'eseguibile standalone compilato.
> 2. Crea collegamenti simbolici per gli eseguibili `aws` e `aws_completer` nella directory bin configurata.

Passaggio 5: verificare l'installazione di AWS CLI

Confermare di aver installato AWS CLI correttamente utilizzando il seguente comando:

```
$ aws --version
aws-cli/2.27.41 Python/3.11.6 Windows/10 exe/AMD64 prompt/off
```

Se il comando `aws` non viene riconosciuto, potrebbe essere necessario riavviare il terminale per aggiornare i nuovi collegamenti simbolici. Se riscontri altri problemi dopo l'installazione o la disinstallazione di AWS CLI, consulta le procedure comuni per la risoluzione dei problemi in the section called "Risoluzione degli errori"

Esempi di flussi di lavoro

Questa sezione fornisce alcuni esempi di flussi di lavoro di base per l'installazione dall'origine.

Installazione di base per Linux e macOS

L'esempio seguente è un flusso di lavoro di installazione di base in cui AWS CLI viene installato nella posizione predefinita `/usr/local/lib/aws-cli`.

```
$ cd path/to/cli/respository/
$ ./configure
$ make
$ make install
```

Installazione automatizzata per Windows

> **Note**
>
> È necessario eseguire PowerShell come amministratore per utilizzare questo flusso di lavoro.

MSYS2 può essere utilizzato in modalità automatizzata in un'impostazione CI, consulta Using MSYS2 in CI nella Documentazione di MSYS2 per ulteriori informazioni.

Downloaded Tarball

Scaricare il file `awscli.tar.gz`, estrarre e installare AWS CLI. Quando si utilizzano i comandi seguenti, sostituire i percorsi seguenti:

- `C:\msys64\usr\bin\bash` con la posizione del percorso MSYS2.
- `.\awscli-2.x.x\` con il nome della cartella `awscli.tar.gz` estratta.
- `PYTHON='C:\path\to\python.exe'` con il percorso Python locale.

Il seguente codice di esempio automatizza la compilazione e l'installazione di AWS CLI da PowerShell utilizzando MSYS2 e specifica quale installazione locale di Python utilizzare:

```
PS C:\> curl "https://awscli.amazonaws.com/awscli.tar.gz" -o "awscliv2.zip"  #
 Download the awscli.tar.gz file in the current working directory
PS C:\> tar -xvzf .\awscli.tar.gz #  Extract awscli.tar.gz file
PS C:\> cd .\awscli-2.x.x\ #  Navigate to the root of the extracted files
PS C:\> $env:CHERE_INVOKING = 'yes' #  Preserve the current working directory
PS C:\> C:\msys64\usr\bin\bash  -lc " PYTHON='C:\path\to\python.exe' ./configure --
prefix='C:\Program Files\AWSCLI' --with-download-deps "
PS C:\> C:\msys64\usr\bin\bash  -lc "make"
PS C:\> C:\msys64\usr\bin\bash  -lc "make install"
PS C:\> $Env:PATH +=";C:\Program Files\AWSCLI\bin\"
PS C:\> aws --version
aws-cli/2.27.41 Python/3.11.6 Windows/10 source-sandbox/AMD64
```

GitHub Repository

Scaricare il file `awscli.tar.gz`, estrarre e installare AWS CLI. Quando si utilizzano i comandi seguenti, sostituire i percorsi seguenti:

- `C:\msys64\usr\bin\bash` con la posizione del percorso MSYS2.
- `C:path\to\cli\repository\` con il percorso del repository AWS CLI clonato in GitHub. Per ulteriori informazioni, consulta Fork a repo nella Documentazione di GitHub
- `PYTHON='C:\path\to\python.exe'` con il percorso Python locale.

Il seguente codice di esempio automatizza la compilazione e l'installazione di AWS CLI da PowerShell utilizzando MSYS2 e specifica quale installazione locale di Python utilizzare:

```
PS C:\> cd C:path\to\cli\repository\
PS C:\> $env:CHERE_INVOKING = 'yes' #  Preserve the current working directory
PS C:\> C:\msys64\usr\bin\bash  -lc " PYTHON='C:\path\to\python.exe' ./configure --
prefix='C:\Program Files\AWSCLI' --with-download-deps "
PS C:\> C:\msys64\usr\bin\bash  -lc "make"
PS C:\> C:\msys64\usr\bin\bash  -lc "make install"
```

```
PS C:\> $Env:PATH +=";C:\Program Files\AWSCLI\bin\"
PS C:\> aws --version
```

Container Alpine Linux

Di seguito è riportato un esempio di Dockerfile che può essere utilizzato per ottenere un'installazione funzionante di AWS CLI in un container Alpine Linux come alternativa ai file binari predefiniti per Alpine. Quando si utilizza questo esempio, sostituire *AWSCLI_VERSION* con il numero di versione AWS CLI desiderato:

```
FROM python:3.8-alpine AS builder

ENV AWSCLI_VERSION=2.10.1

RUN apk add --no-cache \
    curl \
    make \
    cmake \
    gcc \
    g++ \
    libc-dev \
    libffi-dev \
    openssl-dev \
    && curl https://awscli.amazonaws.com/awscli-${AWSCLI_VERSION}.tar.gz | tar -xz \
    && cd awscli-${AWSCLI_VERSION} \
    && ./configure --prefix=/opt/aws-cli/ --with-download-deps \
    && make \
    && make install

FROM python:3.8-alpine

RUN apk --no-cache add groff

COPY --from=builder /opt/aws-cli/ /opt/aws-cli/

ENTRYPOINT ["/opt/aws-cli/bin/aws"]
```

Questa immagine viene compilata e AWS CLI viene invocato da un container simile a quello compilato su Amazon Linux 2:

```
$ docker build --tag awscli-alpine .
```

```
$ docker run --rm -it awscli-alpine --version
aws-cli/2.2.1 Python/3.8.11 Linux/5.10.25-linuxkit source-sandbox/x86_64.alpine.3
 prompt/off
```

La dimensione finale di questa immagine è inferiore alla dimensione dell'immagine Docker AWS CLI ufficiale. Per informazioni sull'immagine Docker ufficiale, consulta the section called "Amazon ECR Public/Docker".

Risoluzione degli errori di installazione e disinstallazione della AWS CLI

Per i passaggi della risoluzione dei problemi di installazione, consulta the section called "Risoluzione degli errori" per le procedure più comuni. Per le procedure di risoluzione dei problemi più importanti, consulta the section called "Errori di comando non trovato", the section called "Il comando `aws --version` restituisce una versione diversa da quella installata" e the section called "Il comando "`aws --version`" restituisce una versione dopo la disinstallazione di AWS CLI".

Per eventuali problemi non trattati nelle guide alla risoluzione dei problemi, cercare i problemi con l'etichetta `source-distribution` nel repository AWS CLI in GitHub. Se nessun problema esistente è relativo agli errori riscontrati, creare un nuovo problema per ricevere assistenza dai manutentori AWS CLI.

Passaggi successivi

Dopo aver installato AWS CLI, è necessario eseguire la configurazione come indicato in the section called "Configurazione".

Esecuzione delle immagini ufficiali di Amazon ECR Public o Docker per la AWS CLI

Questo argomento descrive come eseguire, controllare la versione e configurare la AWS CLI versione 2 su Docker utilizzando l'immagine ufficiale di Amazon Elastic Container Registry Public (Amazon ECR Public) o Docker Hub. Per ulteriori informazioni su come utilizzare Docker, consulta la documentazione di Docker.

Le immagini ufficiali forniscono isolamento, portabilità e sicurezza supporti e gestiti direttamente da AWS. Ciò consente di utilizzare la AWS CLI versione 2 in un ambiente basato su container senza dover gestire autonomamente l'installazione.

Argomenti

- Prerequisiti
- Decidere tra Amazon ECR Public e Docker Hub
- Esegui le immagini ufficiali della AWS CLI versione 2
- Note sulle interfacce e sulla retrocompatibilità delle immagini ufficiali
- Utilizzo di versioni e tag specifici
- Aggiornamento all'ultima immagine ufficiale
- Condivisione di file host, credenziali, variabili d'ambiente e configurazione
- Abbreviazione del comando di esecuzione Docker

Prerequisiti

È necessario che Docker sia installato. Per istruzioni sull'installazione, consulta il sito Web Docker.

Per verificare l'installazione di Docker, eseguire il comando seguente e confermare l'esistenza di output.

```
$ docker --version
Docker version 19.03.1
```

Decidere tra Amazon ECR Public e Docker Hub

per le immagini della AWS CLI, noi consigliamo di utilizzare Amazon ECR Public rispetto a Docker Hub. Docker Hub prevede limiti di velocità più rigorosi per i consumatori pubblici, causando eventualmente problemi di limitazione (della larghezza di banda della rete). Inoltre, Amazon ECR Public replica le immagini in più di una Regione per fornire una solida disponibilità e gestire i problemi di interruzione della Regione.

Per ulteriori informazioni sui limiti di frequenza di Docker Hub, consulta Informazioni sulla limitazione della frequenza di Docker Hub nel sito web di Docker.

Esegui le immagini ufficiali della AWS CLI versione 2

La prima volta che si utilizza il comando `docker run`, l'immagine più recente viene scaricata sul computer. Ogni successivo utilizzo del comando `docker run` viene eseguito dalla copia locale.

Per eseguire le immagini Docker della AWS CLI versione 2, utilizzare il comando `docker run`.

Amazon ECR Public

L'immagine ufficiale della AWS CLI versione 2 Amazon ECR Public è ospitata su Amazon ECR Public nel repository `aws-cli/aws-cli`.

```
$ docker run --rm -it public.ecr.aws/aws-cli/aws-cli command
```

Docker Hub

L'immagine Docker ufficiale della AWS CLI versione 2 è ospitata su Docker Hub nel repository `amazon/aws-cli`.

```
$ docker run --rm -it amazon/aws-cli command
```

Ecco come funziona il comando:

- `docker run --rm -it` *`repository/name`* - L'equivalente dell'eseguibile `aws`. Ogni volta che esegui questo comando, Docker rende operativo un container dell'immagine scaricata ed esegue il comando `aws`. Per impostazione predefinita, l'immagine utilizza l'ultima versione della AWS CLI versione 2.

  Ad esempio, per chiamare il comando `aws --version` nel Docker, eseguire quanto segue.

  Amazon ECR Public

  ```
  $ docker run --rm -it public.ecr.aws/aws-cli/aws-cli --version
  aws-cli/2.27.41 Python/3.7.3 Linux/4.9.184-linuxkit botocore/2.4.5dev10
  ```

  Docker Hub

  ```
  $ docker run --rm -it amazon/aws-cli --version
  aws-cli/2.27.41 Python/3.7.3 Linux/4.9.184-linuxkit botocore/2.4.5dev10
  ```

- `--rm` - Specifica di eseguire la pulizia del container dopo l'uscita del comando.
- `-it` - Specifica di aprire uno pseudo-TTY con `stdin`. Ciò consente di fornire input alla AWS CLI versione 2 mentre è in esecuzione in un container, ad esempio, utilizzando i comandi `aws configure` e `aws help`. Se scegli se omettere `-it`, considera quanto segue:

- Se stai eseguendo degli script, `-it` non è necessario.
- Se riscontri degli errori con gli script, l'omissione della `-it` dalla chiamata Docker potrebbe risolvere il problema.
- Se stai cercando di reindirizzare l'output, la `-it` potrebbe causare degli errori e l'omissione della `-it` dalla chiamata Docker potrebbe risolvere il problema. Se desideri mantenere il flag `-it`, ma desideri comunque reindirizzare l'output, la disattivazione del pager lato client utilizzato dalla AWS CLI per impostazione predefinita dovrebbe risolvere il problema.

Per ulteriori informazioni sul comando `docker run`, consulta la Guida di riferimento di Docker.

Note sulle interfacce e sulla retrocompatibilità delle immagini ufficiali

- L'unico strumento supportato sull'immagine è la AWS CLI. Solo l'eseguibile `aws` non dovrebbe mai essere eseguito direttamente. Ad esempio, anche se `less` e `groff` sono installati in modo esplicito sull'immagine, non devono essere eseguiti direttamente all'esterno di un comando della AWS CLI.
- La directory di lavoro di `/aws` è controllata dall'utente. L'immagine non verrà scritta in questa directory, a meno che l'utente non richieda istruzioni sull'esecuzione di un comando della AWS CLI.
- Non ci sono garanzie di compatibilità con le versioni precedenti quando si fa affidamento sul tag più recente. Per garantire la compatibilità con le versioni precedenti, è necessario aggiungere un tag specifico `<major.minor.patch>` in quanto tali tag sono immutabili e verranno inseriti una sola volta.

Utilizzo di versioni e tag specifici

L'immagine ufficiale della AWS CLI versione 2 ha più versioni che è possibile utilizzare, a partire dalla versione `2.0.6`. Per eseguire una versione specifica della AWS CLI versione 2, accodare il tag appropriato al comando `docker run`. La prima volta che utilizzi il comando `docker run` con un tag, l'ultima immagine Docker per quel tag viene scaricata nel computer. Ogni successivo utilizzo del comando `docker run` con quel tag viene eseguito dalla copia locale.

È possibile utilizzare due tipi di tag:

- `latest` - Definisce l'ultima versione della AWS CLI versione 2 per l'immagine Docker. Suggeriamo di utilizzare il tag `latest` se desideri l'ultima versione della AWS CLI versione 2. Tuttavia, non ci sono garanzie di compatibilità con le versioni precedenti quando si fa affidamento su questo tag. Il

tag `latest` viene utilizzato per impostazione predefinita nel comando `docker run`. Per utilizzare esplicitamente il tag `latest`, aggiungere il tag al nome dell'immagine del container.

Amazon ECR Public

```
$ docker run --rm -it public.ecr.aws/aws-cli/aws-cli:latest command
```

Docker Hub

```
$ docker run --rm -it amazon/aws-cli:latest command
```

- `<major.minor.patch>` - Definisce una versione specifica della AWS CLI versione 2 per l'immagine. Se prevedi di utilizzare un'immagine ufficiale in produzione, suggeriamo di utilizzare una versione specifica della AWS CLI versione 2 per garantire la compatibilità con le versioni precedenti. Ad esempio, per eseguire la versione `2.0.6`, aggiungere la versione al nome dell'immagine del container.

  Amazon ECR Public

  ```
  $ docker run --rm -it public.ecr.aws/aws-cli/aws-cli:2.0.6 command
  ```

  Docker Hub

  ```
  $ docker run --rm -it amazon/aws-cli:2.0.6 command
  ```

Aggiornamento all'ultima immagine ufficiale

Poiché l'immagine più recente viene scaricata nel computer solo la prima volta che si utilizza il comando `docker run`, è necessario estrarre manualmente un'immagine aggiornata. Per eseguire manualmente l'aggiornamento alla versione più recente, ti consigliamo di estrarre l'immagine `latest` con tag. Estraendo l'immagine viene scaricata la versione più recente sul computer.

Amazon ECR Public

```
$ docker pull public.ecr.aws/aws-cli/aws-cli:latest
```

Docker Hub

```
$ docker pull amazon/aws-cli:latest
```

Condivisione di file host, credenziali, variabili d'ambiente e configurazione

Poiché la AWS CLI versione 2 viene eseguita in un container, per impostazione predefinita l'interfaccia a riga di comando non può accedere al file system host, che include la configurazione e le credenziali. Per condividere il file system host, le credenziali e la configurazione nel container, montare la directory `~/.aws` del sistema host sul container `/root/.aws` con il flag `-v` per il comando `docker run`. Ciò consente alla AWS CLI versione 2 in esecuzione nel container di individuare le informazioni sul file host.

Amazon ECR Public

Linux e macOS

```
$ docker run --rm -it -v ~/.aws:/root/.aws public.ecr.aws/aws-cli/aws-cli command
```

Prompt dei comandi di Windows

```
$ docker run --rm -it -v %userprofile%\.aws:/root/.aws public.ecr.aws/aws-cli/aws-
cli command
```

Windows PowerShell

```
C:\> docker run --rm -it -v $env:userprofile\.aws:/root/.aws  public.ecr.aws/aws-
cli/aws-cli command
```

Docker Hub

Linux e macOS

```
$ docker run --rm -it -v ~/.aws:/root/.aws amazon/aws-cli command
```

Prompt dei comandi di Windows

```
$ docker run --rm -it -v %userprofile%\.aws:/root/.aws amazon/aws-cli command
```

Windows PowerShell

```
C:\> docker run --rm -it -v $env:userprofile\.aws:/root/.aws  amazon/aws-cli command
```

Per ulteriori informazioni sul flag `-v` e sul montaggio, consulta la Guida di riferimento Docker.

> **Note**
>
> Per informazioni sui file `config` e `credentials`, consulta the section called "Impostazioni di configurazione".

Esempio 1: fornire credenziali e configurazione

In questo esempio, vengono fornite le credenziali host e la configurazione durante l'esecuzione del comando `s3 ls` per elencare i bucket in Amazon Simple Storage Service (Amazon S3). Gli esempi seguenti utilizzano la posizione predefinita per le credenziali e i file di configurazione della AWS CLI; per utilizzare una posizione diversa, modificare il percorso del file.

Amazon ECR Public

Linux e macOS

```
$ docker run --rm -it -v ~/.aws:/root/.aws public.ecr.aws/aws-cli/aws-cli s3 ls
2020-03-25 00:30:48 aws-cli-docker-demo
```

Prompt dei comandi di Windows

```
$ docker run --rm -it -v %userprofile%\.aws:/root/.aws public.ecr.aws/aws-cli/aws-
cli s3 ls
2020-03-25 00:30:48 aws-cli-docker-demo
```

Windows PowerShell

```
C:\> docker run --rm -it -v $env:userprofile\.aws:/root/.aws public.ecr.aws/aws-cli/
aws-cli s3 ls
```

Docker Hub

Linux e macOS

```
$ docker run --rm -it -v ~/.aws:/root/.aws amazon/aws-cli s3 ls
2020-03-25 00:30:48 aws-cli-docker-demo
```

Prompt dei comandi di Windows

```
$ docker run --rm -it -v %userprofile%\.aws:/root/.aws amazon/aws-cli s3 ls
2020-03-25 00:30:48 aws-cli-docker-demo
```

Windows PowerShell

```
C:\> docker run --rm -it -v $env:userprofile\.aws:/root/.aws amazon/aws-cli s3 ls
```

È possibile chiamare variabili di ambiente di sistema specifiche utilizzando il flag `-e`. Per utilizzare una variabile di ambiente, chiamala per nome.

Amazon ECR Public

Linux e macOS

```
$ docker run --rm -it -v ~/.aws:/root/.aws -e ENVVAR_NAME public.ecr.aws/aws-cli/
aws-cli s3 ls
2020-03-25 00:30:48 aws-cli-docker-demo
```

Prompt dei comandi di Windows

```
$ docker run --rm -it -v %userprofile%\.aws:/root/.aws -e ENVVAR_NAME
 public.ecr.aws/aws-cli/aws-cli s3 ls
2020-03-25 00:30:48 aws-cli-docker-demo
```

Windows PowerShell

```
C:\> docker run --rm -it -v $env:userprofile\.aws:/root/.aws -e ENVVAR_NAME
 public.ecr.aws/aws-cli/aws-cli s3 ls
```

Docker Hub

Linux e macOS

```
$ docker run --rm -it -v ~/.aws:/root/.aws -e ENVVAR_NAME amazon/aws-cli s3 ls
2020-03-25 00:30:48 aws-cli-docker-demo
```

Prompt dei comandi di Windows

```
$ docker run --rm -it -v %userprofile%\.aws:/root/.aws -e ENVVAR_NAME amazon/aws-cli
 s3 ls
```

```
2020-03-25 00:30:48 aws-cli-docker-demo
```

Windows PowerShell

```
C:\> docker run --rm -it -v $env:userprofile\.aws:/root/.aws -e ENVVAR_NAME amazon/
aws-cli s3 ls
```

Esempio 2: download di un file Amazon S3 nel sistema host

Per alcuni comandi AWS CLI versione 2, è possibile leggere i file dal sistema host nel container o scrivere file dal container al sistema host.

Questo esempio illustra come scaricare l'oggetto S3 `s3://aws-cli-docker-demo/hello` nel file system locale montando la directory di lavoro corrente nella directory `/aws` del container. Scaricando l'oggetto `hello` nella directory `/aws` del container, il file viene salvato anche nella directory di lavoro corrente del sistema host.

Amazon ECR Public

Linux e macOS

```
$ docker run --rm -it -v ~/.aws:/root/.aws -v $(pwd):/aws public.ecr.aws/aws-cli/
aws-cli s3 cp s3://aws-cli-docker-demo/hello .
download: s3://aws-cli-docker-demo/hello to ./hello
```

Prompt dei comandi di Windows

```
$ docker run --rm -it -v %userprofile%\.aws:/root/.aws -v %cd%:/aws public.ecr.aws/
aws-cli/aws-cli s3 cp s3://aws-cli-docker-demo/hello .
download: s3://aws-cli-docker-demo/hello to ./hello
```

Windows PowerShell

```
C:\> docker run --rm -it -v $env:userprofile\.aws:/root/.aws -v $pwd\aws:/aws
 public.ecr.aws/aws-cli/aws-cli s3 cp s3://aws-cli-docker-demo/hello .
```

Docker Hub

Linux e macOS

```
$ docker run --rm -it -v ~/.aws:/root/.aws -v $(pwd):/aws amazon/aws-cli s3 cp s3://
aws-cli-docker-demo/hello .
download: s3://aws-cli-docker-demo/hello to ./hello
```

Prompt dei comandi di Windows

```
$ docker run --rm -it -v %userprofile%\.aws:/root/.aws -v %cd%:/aws amazon/aws-cli
 s3 cp s3://aws-cli-docker-demo/hello .
download: s3://aws-cli-docker-demo/hello to ./hello
```

Windows PowerShell

```
C:\> docker run --rm -it -v $env:userprofile\.aws:/root/.aws -v $pwd\aws:/aws
 amazon/aws-cli s3 cp s3://aws-cli-docker-demo/hello .
```

Per verificare che il file scaricato esista nel file system locale, eseguire quanto segue.

Linux e macOS

```
$ cat hello
Hello from Docker!
```

Windows PowerShell

```
$ type hello
Hello from Docker!
```

Esempio 3: utilizzo della variabile di ambiente AWS_PROFILE

È possibile chiamare variabili di ambiente di sistema specifiche utilizzando il flag `-e`. Chiama ogni variabile di ambiente che desideri utilizzare. In questo esempio, vengono fornite le credenziali host, la configurazione e la variabile d'ambiente *`AWS_PROFILE`* durante l'esecuzione del comando `s3 ls` per elencare i bucket in Amazon Simple Storage Service (Amazon S3).

Amazon ECR Public

Linux e macOS

```
$ docker run --rm -it -v ~/.aws:/root/.aws -e AWS_PROFILE public.ecr.aws/aws-cli/
aws-cli s3 ls
```

```
2020-03-25 00:30:48 aws-cli-docker-demo
```

Prompt dei comandi di Windows

```
$ docker run --rm -it -v %userprofile%\.aws:/root/.aws -e AWS_PROFILE
 public.ecr.aws/aws-cli/aws-cli s3 ls
2020-03-25 00:30:48 aws-cli-docker-demo
```

Windows PowerShell

```
C:\> docker run --rm -it -v $env:userprofile\.aws:/root/.aws -e AWS_PROFILE
 public.ecr.aws/aws-cli/aws-cli s3 ls
```

Docker Hub

Linux e macOS

```
$ docker run --rm -it -v ~/.aws:/root/.aws -e AWS_PROFILE amazon/aws-cli s3 ls
2020-03-25 00:30:48 aws-cli-docker-demo
```

Prompt dei comandi di Windows

```
$ docker run --rm -it -v %userprofile%\.aws:/root/.aws -e AWS_PROFILE amazon/aws-cli
 s3 ls
2020-03-25 00:30:48 aws-cli-docker-demo
```

Windows PowerShell

```
C:\> docker run --rm -it -v $env:userprofile\.aws:/root/.aws -e AWS_PROFILE amazon/
aws-cli s3 ls
```

Abbreviazione del comando di esecuzione Docker

Per abbreviare il comando `docker run`, suggeriamo di utilizzare la capacità del sistema operativo per creare un `symbolic link` (collegamento simbolico) o un `alias` in Linux e macOS o `doskey` in Windows. Per impostare l'alias `aws`, è possibile eseguire uno dei seguenti comandi.

- Per l'accesso di base ai comandi `aws`, eseguire le seguenti operazioni.

Amazon ECR Public

Linux e macOS

```
$ alias aws='docker run --rm -it public.ecr.aws/aws-cli/aws-cli'
```

Prompt dei comandi di Windows

```
C:\> doskey aws=docker run --rm -it public.ecr.aws/aws-cli/aws-cli $*
```

Windows PowerShell

```
C:\> Function AWSCLI {docker run --rm -it public.ecr.aws/aws-cli/aws-cli $args}
Set-Alias -Name aws -Value AWSCLI
```

Docker Hub

Linux e macOS

```
$ alias aws='docker run --rm -it amazon/aws-cli'
```

Prompt dei comandi di Windows

```
C:\> doskey aws=docker run --rm -it amazon/aws-cli $*
```

Windows PowerShell

```
C:\> Function AWSCLI {docker run --rm -it amazon/aws-cli $args}
Set-Alias -Name aws -Value AWSCLI
```

- Per accedere al file system host e alle impostazioni di configurazione quando si utilizzano i comandi `aws` eseguire le seguenti operazioni.

  Amazon ECR Public

  Linux e macOS

```
$ alias aws='docker run --rm -it -v ~/.aws:/root/.aws -v $(pwd):/aws
 public.ecr.aws/aws-cli/aws-cli'
```

Prompt dei comandi di Windows

```
C:\> doskey aws=docker run --rm -it -v %userprofile%\.aws:/root/.aws -v %cd%:/aws
 public.ecr.aws/aws-cli/aws-cli $*
```

Windows PowerShell

```
C:\> Function AWSCLI {docker run --rm -it -v $env:userprofile\.aws:/root/.aws -v
 $pwd\aws:/aws public.ecr.aws/aws-cli/aws-cli $args}
Set-Alias -Name aws -Value AWSCLI
```

Docker Hub

Linux e macOS

```
$ alias aws='docker run --rm -it -v ~/.aws:/root/.aws -v $(pwd):/aws amazon/aws-
cli'
```

Prompt dei comandi di Windows

```
C:\> doskey aws=docker run --rm -it -v %userprofile%\.aws:/root/.aws -v %cd%:/aws
 amazon/aws-cli $*
```

Windows PowerShell

```
C:\> Function AWSCLI {docker run --rm -it -v $env:userprofile\.aws:/root/.aws -v
 $pwd\aws:/aws amazon/aws-cli $args}
Set-Alias -Name aws -Value AWSCLI
```

- Per assegnare una versione specifica da utilizzare nell'alias `aws`, aggiungere il tag di versione.

  Amazon ECR Public

  Linux e macOS

  ```
  $ alias aws='docker run --rm -it -v ~/.aws:/root/.aws -v $(pwd):/aws
   public.ecr.aws/aws-cli/aws-cli:2.0.6'
  ```

  Prompt dei comandi di Windows

```
C:\> doskey aws=docker run --rm -it -v %userprofile%\.aws:/root/.aws -v %cd%:/aws
 public.ecr.aws/aws-cli/aws-cli:2.0.6 $*
```

Windows PowerShell

```
C:\> Function AWSCLI {docker run --rm -it -v $env:userprofile\.aws:/root/.aws -v
 $pwd\aws:/aws public.ecr.aws/aws-cli/aws-cli:2.0.6 $args}
Set-Alias -Name aws -Value AWSCLI
```

Docker Hub

Linux e macOS

```
$ alias aws='docker run --rm -it -v ~/.aws:/root/.aws -v $(pwd):/aws amazon/aws-
cli:2.0.6'
```

Prompt dei comandi di Windows

```
C:\> doskey aws=docker run --rm -it -v %userprofile%\.aws:/root/.aws -v %cd%:/aws
 amazon/aws-cli:2.0.6 $*
```

Windows PowerShell

```
C:\> Function AWSCLI {docker run --rm -it -v $env:userprofile\.aws:/root/.aws -v
 $pwd\aws:/aws amazon/aws-cli:2.0.6 $args}
Set-Alias -Name aws -Value AWSCLI
```

Dopo aver impostato l'alias, è possibile eseguire la AWS CLI versione 2 dall'interno di un container come se fosse installata sul sistema host.

```
$ aws --version
aws-cli/2.27.41 Python/3.7.3 Linux/4.9.184-linuxkit botocore/2.4.5dev10
```

Configurazione del AWS CLI

Questo argomento spiega come configurare rapidamente le impostazioni di base utilizzate da AWS Command Line Interface (AWS CLI) per interagire AWS. Queste includono le credenziali di sicurezza, il formato di output predefinito e la AWS regione predefinita.

Argomenti

- Raccogliere le informazioni sulle credenziali per l'accesso programmatico
- Configurazione di nuove configurazioni e credenziali
- Utilizzo di file di configurazione e credenziali esistenti

Raccogliere le informazioni sulle credenziali per l'accesso programmatico

Avrai bisogno dell'accesso programmatico se desideri interagire con l' AWS esterno di. Console di gestione AWS Per istruzioni di autenticazione e credenziali, scegli una delle seguenti opzioni:

Tipo di autenticazione	Scopo	Istruzioni
AWS Credenziali della console di gestione	(Consigliato) Utilizza credenziali a breve termine accedendo alla AWS CLI con le credenziali della console. Consigliato se utilizzi root, utenti IAM o federazione con IAM per l'accesso all'account AWS	the section called "Credenziali della console"
Credenziali a breve termine per gli utenti della forza lavoro del Centro identità IAM	Utilizza credenziali a breve termine per un utente della forza lavoro di IAM Identity Center. Le best practice di sicurezza sono da utilizzare AWS Organizations con IAM Identity Center. Combina credenzia li a breve termine con una	the section called "Autentic azione in Centro identità IAM"

Tipo di autenticazione	Scopo	Istruzioni
	directory utente, ad esempio la directory integrata del Centro identità IAM o di Active Directory.	
Credenziali a breve termine per utenti IAM	Utilizza le credenziali a breve termine degli utenti IAM, che sono più sicure delle credenziali a lungo termine. Se le tue credenziali sono compromesse, è previsto un periodo di tempo limitato in cui possono essere utilizzate prima della scadenza.	the section called "Credenziali a breve termine"
Utenti IAM o IAM Identity Center su un' EC2 istanza Amazon.	Utilizza i metadati delle EC2 istanze Amazon per richieder e credenziali temporanee utilizzando il ruolo assegnato all'istanza Amazon EC2 .	the section called "Metadati Amazon EC2"
Assumere ruoli per le autorizzazioni	Associa un altro metodo di credenziali e assumi un ruolo per l'accesso temporaneo a cui l'utente dei Servizi AWS potrebbe non avere accesso.	the section called "Ruoli IAM"
Credenziali a lungo termine per utenti IAM	(Non raccomandato) Utilizzar e credenziali a lungo termine, che non hanno scadenza.	the section called "Utenti IAM"

Tipo di autenticazione	Scopo	Istruzioni
Archiviazione esterna degli utenti della forza lavoro IAM o del Centro identità IAM	(Non raccomandato) Associare un altro metodo di credenziali ma archiviar e i valori delle credenziali in una posizione esterna alla AWS CLI. Questo metodo è sicuro solo quanto la posizione esterna in cui sono archiviate le credenziali.	the section called "Credenziali esterne"

Configurazione di nuove configurazioni e credenziali

AWS CLI Memorizza le informazioni di configurazione e credenziali in un profilo (una raccolta di impostazioni) nei file and. `credentials` `config`

Esistono principalmente due metodi per eseguire rapidamente la configurazione:

- Configurazione tramite i comandi della AWS CLI
- Modifica manuale delle credenziali e dei file di configurazione

Gli esempi seguenti utilizzano valori di esempio per ciascuno dei metodi di autenticazione. Sostituire i valori di esempio con i propri.

Configurazione tramite i comandi della AWS CLI

Per uso generale, i comandi `aws configure sso` o `aws configure` nel terminale preferito sono il modo più veloce per configurare l'installazione della AWS CLI . In base al metodo di credenziale che preferisci, ti AWS CLI vengono richieste le informazioni pertinenti. Per impostazione predefinita, le informazioni in questo profilo vengono utilizzate quando si esegue un AWS CLI comando che non specifica esplicitamente un profilo da utilizzare.

Per ulteriori informazioni sui file `credentials` e `config`, consulta Impostazioni dei file di configurazione e delle credenziali nella AWS CLI.

Console sessions as short-term credentials

Questo esempio consente di utilizzare le credenziali della console esistenti con il `aws login` comando. Questo metodo di accesso può essere utilizzato con credenziali root create durante la configurazione iniziale dell'account, con un utente IAM o con un'identità federata del provider di identità. Per ulteriori informazioni, consulta Accesso per lo sviluppo AWS locale utilizzando le credenziali della console.

```
$ aws login
No AWS region has been configured. The AWS region is the geographic location of your
 AWS resources.

If you've used AWS before and already have resources in your account, tell us which
 region they were
created in. If you haven't created resources in your account before, you can pick
 the region closest to you:
https://docs.aws.amazon.com/global-infrastructure/latest/regions/aws-regions.html.
 You are able to change
the region in the CLI at any time with the command `aws configure set region
 NEW_REGION`.
AWS Region [us-east-1]: us-east-1
Attempting to open the login page for `us-east-1` in your default browser. If the
 browser does not open,
use the following URL to complete your login:

https://us-east-1.signin.aws.amazon.com/v1/authorize<abbreviated>

If you cannot connect to this URL, make sure that you have specified a valid region.

Waiting for login...

Logged in with role `arn:aws:sts::012345678910:user/iam-user`, and configured
 profile
`default` to use `us-east-1`. This session will expire on October 14, 2025 at 2:04
 PST. After this time, you
can renew your session with `aws login`.
```

IAM Identity Center (SSO)

Questo esempio riguarda l' AWS IAM Identity Center utilizzo della `aws configure sso` procedura guidata. Per ulteriori informazioni, consulta the section called "Autenticazione in Centro identità IAM".

```
$ aws configure sso
SSO session name (Recommended): my-sso
SSO start URL [None]: https://my-sso-portal.awsapps.com/start
SSO region [None]:us-east-1

Attempting to automatically open the SSO authorization page in your default browser.

There are 2 AWS accounts available to you.
> DeveloperAccount, developer-account-admin@example.com (111122223333)
  ProductionAccount, production-account-admin@example.com (444455556666)

Using the account ID 111122223333

There are 2 roles available to you.
> ReadOnly
  FullAccess

Using the role name "ReadOnly"

CLI default client Region [None]: us-west-2
CLI default output format [None]: json
CLI profile name [123456789011_ReadOnly]: user1
```

IAM Identity Center (Legacy SSO)

Questo esempio riguarda il metodo precedente di AWS IAM Identity Center utilizzo della `aws configure sso` procedura guidata. Per utilizzare l'SSO precedente, lascia vuoto il nome della sessione. Per ulteriori informazioni, consulta the section called "Autenticazione in Centro identità IAM".

```
$ aws configure sso
SSO session name (Recommended):
SSO start URL [None]: https://my-sso-portal.awsapps.com/start
SSO region [None]:us-east-1

SSO authorization page has automatically been opened in your default browser.
Follow the instructions in the browser to complete this authorization request.

There are 2 AWS accounts available to you.
> DeveloperAccount, developer-account-admin@example.com (111122223333)
  ProductionAccount, production-account-admin@example.com (444455556666)
```

```
Using the account ID 111122223333

There are 2 roles available to you.
> ReadOnly
  FullAccess

Using the role name "ReadOnly"

CLI default client Region [None]: us-west-2
CLI default output format [None]: json
CLI profile name [123456789011_ReadOnly]: user1
```

Short-term credentials

Questo esempio riguarda le credenziali a breve termine di AWS Identity and Access Management. Dopo aver utilizzato la procedura guidata aws configure per impostare i valori iniziali, il comando `aws configure set` assegna l'ultimo valore necessario. Per ulteriori informazioni, consulta the section called "Credenziali a breve termine".

```
$ aws configure
AWS Access Key ID [None]: AKIAIOSFODNN7EXAMPLE
AWS Secret Access Key [None]: wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY
Default region name [None]: us-west-2
Default output format [None]: json
$ aws configure set
 aws_session_token fcZib3JpZ2luX2IQoJb3JpZ2luX2IQoJb3JpZ2luX2IQoJb3JpZ2luX2IQoJb3JpZVERYLONG
```

IAM role

Questo esempio serve per assumere un ruolo IAM. I profili che utilizzano i ruoli IAM estraggono le credenziali da un altro profilo e quindi applicano le autorizzazioni dei ruoli IAM. Negli esempi seguenti, `default` è il profilo di origine per le credenziali e `user1` prende in prestito le stesse credenziali per poi assumere un nuovo ruolo. Non esiste una procedura guidata per questo processo, pertanto ogni valore viene impostato utilizzando il comando `aws configure set`. Per ulteriori informazioni, consulta the section called "Ruoli IAM".

```
$ aws configure set role_arn arn:aws:iam::123456789012:role/defaultrole
$ aws configure set source_profile default
$ aws configure set role_session_name session_user1
$ aws configure set region us-west-2
$ aws configure set output json
```

Amazon EC2 instance metadata credentials

Questo esempio riguarda le credenziali ottenute dai metadati dell' EC2 istanza Amazon di hosting. Non esiste una procedura guidata per questo processo, pertanto ogni valore viene impostato utilizzando il comando `aws configure set`. Per ulteriori informazioni, consulta the section called “Metadati Amazon EC2”.

```
$ aws configure set role_arn arn:aws:iam::123456789012:role/defaultrole
$ aws configure set credential_source Ec2InstanceMetadata
$ aws configure set region us-west-2
$ aws configure set output json
```

Long-term credentials

> **Warning**
>
> Per evitare rischi per la sicurezza, non utilizzare gli utenti IAM per l'autenticazione quando sviluppi software creato ad hoc o lavori con dati reali. Utilizza invece la federazione con un provider di identità come AWS IAM Identity Center.

Questo esempio riguarda le credenziali a lungo termine di AWS Identity and Access Management. Per ulteriori informazioni, consulta the section called “Utenti IAM”.

```
$ aws configure
AWS Access Key ID [None]: AKIAIOSFODNN7EXAMPLE
AWS Secret Access Key [None]: wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY
Default region name [None]: us-west-2
Default output format [None]: json
```

Per informazioni più dettagliate sui metodi di autenticazione e credenziali, consulta Credenziali di autenticazione e accesso.

Modifica manuale delle credenziali e dei file di configurazione

Quando copi e incolli le informazioni, ti suggeriamo di modificare manualmente il file `config` e `credentials`. In base al metodo di credenziali preferito, i file vengono configurati in modo diverso.

I file vengono memorizzati nella home directory all'interno della cartella `.aws`. La posizione della home directory varia in base al sistema operativo, ma viene definita utilizzando le variabili di ambiente

`%UserProfile%` in Windows e `$HOME` o ~ (tilde) nei sistemi basati su Unix. Per ulteriori informazioni su dove sono memorizzate queste impostazioni, consulta the section called "Dove vengono archiviate le impostazioni di configurazione?".

Gli esempi seguenti mostrano un profilo `default` e un profilo denominato `user1` e utilizzano valori di esempio. Sostituire i valori di esempio con i propri. Per ulteriori informazioni sui file `credentials` e `config`, consulta Impostazioni dei file di configurazione e delle credenziali nella AWS CLI.

IAM Identity Center (SSO)

Questo esempio è per. AWS IAM Identity Center Per ulteriori informazioni, consulta the section called "Autenticazione in Centro identità IAM".

File di credenziali

Il file `credentials` non viene utilizzato per questo metodo di autenticazione.

File di configurazione

```
[default]
sso_session = my-sso
sso_account_id = 111122223333
sso_role_name = readOnly
region = us-west-2
output = text

[profile user1]
sso_session = my-sso
sso_account_id = 444455556666
sso_role_name = readOnly
region = us-east-1
output = json

[sso-session my-sso]
sso_region = us-east-1
sso_start_url = https://my-sso-portal.awsapps.com/start
sso_registration_scopes = sso:account:access
```

IAM Identity Center (Legacy SSO)

Questo esempio riguarda il metodo legacy di AWS IAM Identity Center. Per ulteriori informazioni, consulta the section called "Autenticazione in Centro identità IAM".

File di credenziali

Il file `credentials` non viene utilizzato per questo metodo di autenticazione.

File di configurazione

```
[default]
sso_start_url = https://my-sso-portal.awsapps.com/start
sso_region = us-east-1
sso_account_id = 111122223333
sso_role_name = readOnly
region = us-west-2
output = text

[profile user1]
sso_start_url = https://my-sso-portal.awsapps.com/start
sso_region = us-east-1
sso_account_id = 444455556666
sso_role_name = readOnly
region = us-east-1
output = json
```

Short-term credentials

Questo esempio riguarda le credenziali a breve termine di AWS Identity and Access Management. Per ulteriori informazioni, consulta the section called "Credenziali a breve termine".

File di credenziali

```
[default]
aws_access_key_id=ASIAIOSFODNN7EXAMPLE
aws_secret_access_key=wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY
aws_session_token
 = IQoJb3JpZ2luX2IQoJb3JpZ2luX2IQoJb3JpZ2luX2IQoJb3JpZ2luX2IQoJb3JpZVERYLONGSTRINGEXAMPLE

[user1]
aws_access_key_id=ASIAI44QH8DHBEXAMPLE
aws_secret_access_key=je7MtGbClwBF/2Zp9Utk/h3yCo8nvbEXAMPLEKEY
aws_session_token
 = fcZib3JpZ2luX2IQoJb3JpZ2luX2IQoJb3JpZ2luX2IQoJb3JpZ2luX2IQoJb3JpZVERYLONGSTRINGEXAMPLE
```

File di configurazione

```
[default]
region=us-west-2
output=json

[profile user1]
region=us-east-1
output=text
```

IAM role

Questo esempio serve per assumere un ruolo IAM. I profili che utilizzano i ruoli IAM estraggono le credenziali da un altro profilo e quindi applicano le autorizzazioni dei ruoli IAM. Negli esempi seguenti, `default` è il profilo di origine per le credenziali e `user1` prende in prestito le stesse credenziali per poi assumere un nuovo ruolo. Per ulteriori informazioni, consulta the section called "Ruoli IAM".

File di credenziali

Il file `credentials` dipende dall'autenticazione utilizzata dal profilo di origine. Nell'esempio seguente, il profilo di origine utilizza credenziali a breve termine.

```
[default]
aws_access_key_id=ASIAIOSFODNN7EXAMPLE
aws_secret_access_key=wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY
aws_session_token
 = IQoJb3JpZ2luX2IQoJb3JpZ2luX2IQoJb3JpZ2luX2IQoJb3JpZ2luX2IQoJb3JpZVERYLONGSTRINGEXAMPLE
```

File di configurazione

```
[default]
region=us-west-2
output=json

[profile user1]
role_arn=arn:aws:iam::777788889999:role/user1role
source_profile=default
role_session_name=session_user1
region=us-east-1
output=text
```

Amazon EC2 instance metadata credentials

Questo esempio riguarda le credenziali ottenute dai metadati dell' EC2 istanza Amazon di hosting. Per ulteriori informazioni, consulta the section called "Metadati Amazon EC2".

File di credenziali

Il file `credentials` non viene utilizzato per questo metodo di autenticazione.

File di configurazione

```
[default]
role_arn=arn:aws:iam::123456789012:role/defaultrole
credential_source=Ec2InstanceMetadata
region=us-west-2
output=json

[profile user1]
role_arn=arn:aws:iam::777788889999:role/user1role
credential_source=Ec2InstanceMetadata
region=us-east-1
output=text
```

Long-term credentials

> **Warning**
>
> Per evitare rischi per la sicurezza, non utilizzare gli utenti IAM per l'autenticazione quando sviluppi software creato ad hoc o lavori con dati reali. Utilizza invece la federazione con un provider di identità come AWS IAM Identity Center.

Questo esempio riguarda le credenziali a lungo termine di AWS Identity and Access Management. Per ulteriori informazioni, consulta the section called "Utenti IAM".

File di credenziali

```
[default]
aws_access_key_id=AKIAIOSFODNN7EXAMPLE
aws_secret_access_key=wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY

[user1]
```

```
aws_access_key_id=AKIAI44QH8DHBEXAMPLE
aws_secret_access_key=je7MtGbClwBF/2Zp9Utk/h3yCo8nvbEXAMPLEKEY
```

File di configurazione

```
[default]
region=us-west-2
output=json

[profile user1]
region=us-east-1
output=text
```

Per informazioni più dettagliate sui metodi di autenticazione e credenziali, consulta Credenziali di autenticazione e accesso.

Utilizzo di file di configurazione e credenziali esistenti

Se disponi di file di configurazione e credenziali esistenti, questi possono essere utilizzati per AWS CLI.

Per utilizzare i file `credentials` e `config`, spostali nella cartella denominata `.aws` nella tua home directory. La posizione della home directory varia in base al sistema operativo, ma viene definita utilizzando le variabili di ambiente `%UserProfile%` in Windows e `$HOME` o ~ (tilde) nei sistemi basati su Unix.

Puoi specificare una posizione non predefinita per i file `config` e `credentials` impostando le variabili di ambiente `AWS_CONFIG_FILE` e `AWS_SHARED_CREDENTIALS_FILE` su un altro percorso locale. Per informazioni dettagliate, vedi Configurazione delle variabili d'ambiente per AWS CLI.

Per informazioni più dettagliate sui file di configurazione e credenziali, consulta the section called "Impostazioni di configurazione".

Configurazione delle impostazioni della AWS CLI.

In questa sezione viene illustrato come configurare le impostazioni che l'AWS Command Line Interface (AWS CLI) utilizza per interagire con AWS. Questi sono i seguenti:

- Le credenziali identificano chi sta chiamando l'API. Le credenziali di accesso vengono utilizzate per crittografare la richiesta ai server AWS per confermare l'identità e recuperare le policy di autorizzazione associate. Queste autorizzazioni determinano le azioni che è possibile eseguire. Per informazioni relative alla configurazione delle credenziali, consulta Credenziali di autenticazione e accesso.
- Altri dettagli di configurazione per indicare alla AWS CLI il modo in cui elaborare le richieste, come il formato di output predefinito e la Regione AWS predefiniti.

> **Note**
>
> AWS richiede che tutte le richieste in entrata siano firmate crittograficamente. AWS CLI lo fa per te La "firma" include un timestamp/data. Pertanto, è necessario assicurarsi che la data e l'ora del computer siano impostate correttamente. In caso contrario, e se la data/ora nella firma è troppo distante dalla data/ora riconosciuta dal servizio AWS, AWS rifiuta la richiesta.

Configurazione e precedenza delle credenziali

Le credenziali e le impostazioni di configurazione sono situate in più posizioni, ad esempio le variabili di ambiente del sistema o dell'utente, in file di configurazione AWS locali oppure dichiarati esplicitamente come parametri dalla linea di comando. Alcune posizioni hanno la precedenza su altre. Le credenziali e le impostazioni di configurazione AWS CLI hanno la precedenza nell'ordine seguente:

1. Opzioni della riga di comando: sovrascrive le impostazioni in qualsiasi altra posizione, come i parametri `--region`, `--output` e `--profile`.
2. Variabili di ambiente: puoi archiviare i valori nelle variabili di ambiente del sistema.
3. Assumere un ruolo: assumi le autorizzazioni di un ruolo IAM tramite la configurazione o il comando `assume-role`.

4. Assumere un ruolo con l'identità web: assumi le autorizzazioni di un ruolo IAM utilizzando l'identità web tramite la configurazione o il comando `assume-role-with-web-identity`.
5. AWS IAM Identity Center: le impostazioni di configurazione del centro identità IAM memorizzate nel file `config` vengono aggiornate quando si esegue il comando `aws configure sso`. Le credenziali vengono quindi autenticate quando si esegue il comando `aws sso login`. Il file `config` si trova in `~/.aws/config` su Linux o macOS o in `C:\Users\`*`USERNAME`*`\.aws\config` su Windows.
6. File delle credenziali della CLI: i file `credentials` e `config` vengono aggiornati quando esegui il comando `aws configure`. Il file `credentials` si trova in `~/.aws/credentials` su Linux o macOS o in `C:\Users\`*`USERNAME`*`\.aws\credentials` su Windows.
7. Processo personalizzato: ottieni le tue credenziali da una fonte esterna.
8. File di configurazione: i file `credentials` e `config` vengono aggiornati quando esegui il comando `aws configure`. Il file `config` si trova in `~/.aws/config` su Linux o macOS o in `C:\Users\`*`USERNAME`*`\.aws\config` su Windows.
9. Credenziali del container: puoi associare un ruolo IAM a ciascuna delle definizioni di attività di Amazon Elastic Container Service (Amazon ECS). Le credenziali temporanee per tale ruolo sono quindi disponibili per i contenitori dell'attività. Per ulteriori informazioni, consulta Ruoli IAM per le attività nella Guida per sviluppatori di Amazon Elastic Container Service.
10. Credenziali del profilo dell'istanza Amazon EC2: è possibile associare un ruolo IAM a ciascuna istanza Amazon Elastic Compute Cloud (Amazon EC2). Le credenziali temporanee per tale ruolo sono quindi disponibili per il codice in esecuzione nell'istanza. Le credenziali vengono distribuite tramite il servizio di metadati di Amazon EC2. Per ulteriori informazioni, consulta Ruoli IAM per Amazon EC2 nella Guida per l'utente di Amazon EC2 e in Utilizzo dei profili delle istanze nella Guida per l'utente di IAM.

Argomenti aggiuntivi in questa sezione

- the section called "Impostazioni di configurazione"
- the section called "Variabili di ambiente"
- the section called "Opzioni della riga di comando"
- the section called " Completamento dei comandi"
- the section called "Tentativi"
- the section called "Proxy HTTP"

Impostazioni dei file di configurazione e delle credenziali nella AWS CLI

Puoi salvare le impostazioni di configurazione e le credenziali utilizzate di frequente nei file gestiti da AWS CLI.

I file sono suddivisi in `profiles`. Per impostazione predefinita, AWS CLI utilizza le impostazioni presenti nel profilo denominato`default`. Per usare impostazioni alternative, puoi creare profili aggiuntivi e farvi riferimento.

Puoi sostituire una singola impostazione configurando una delle variabili di ambiente supportate oppure utilizzando un parametro della riga di comando. Per ulteriori informazioni sulla precedenza delle impostazioni di configurazione, consulta Configurazione delle impostazioni della AWS CLI..

> **Note**
>
> Per informazioni relative alla configurazione delle credenziali, consulta Credenziali di autenticazione e accesso.

Argomenti

- Formato dei file di configurazione e delle credenziali.
- Dove vengono archiviate le impostazioni di configurazione?
- Utilizzo di profili denominati
- Definizione e visualizzazione delle impostazioni di configurazione utilizzando i comandi
- Esempi di comandi per impostare nuove configurazioni e credenziali
- Impostazioni supportate del file config

Formato dei file di configurazione e delle credenziali.

I file `config` e `credentials` sono organizzati in sezioni. Le sezioni includono profiles, sso-sessions e services. Una sezione è una raccolta denominata di impostazioni e prosegue fino a quando non si incontra un'altra riga di definizione della sezione. Nei file `config` e `credentials` è possibile archiviare più profili e sezioni.

Si tratta di file con testo normale che utilizzano il seguente formato:

- I nomi delle sezioni sono racchiusi tra parentesi [] come `[default]`, `[profile user1]` e `[sso-session]`.
- Tutte le voci di una sezione assumono il formato generale `setting_name=value`.
- È possibile commentare le righe iniziandole con un carattere cancelletto (#).

I file config e credentials contengono i seguenti tipi di sezione:

- Tipo di sezione: profile
- Tipo di sezione: sso-session
- Tipo di sezione: services

Tipo di sezione: **`profile`**

A seconda del file, i nomi delle sezioni del profilo utilizzano il seguente formato:

- File di configurazione: `[default]` `[profile user1]`
- file delle credenziali: `[default]` `[user1]`

  Non utilizzare la parola `profile` durante la creazione di una voce nel file `credentials`.

Ogni profilo può specificare non solo credenziali diverse, ma anche Regioni AWS e formati di output differenti. Quando assegni un nome al profilo in un file `config`, includi la parola di prefisso "`profile`", ma non includerla nel file `credentials`.

Negli esempi seguenti viene mostrato un file `config` e `credentials` con due profili, una Regione e un output specificati. Il primo [default] viene utilizzato quando esegui un comando AWS CLI senza alcun profilo specificato. Il secondo viene utilizzato quando si esegue un AWS CLI comando con il `--profile user1` parametro.

IAM Identity Center (SSO)

Questo esempio è per AWS IAM Identity Center. Per ulteriori informazioni, consulta the section called "Autenticazione in Centro identità IAM".

File di credenziali

Il file `credentials` non viene utilizzato per questo metodo di autenticazione.

File di configurazione

```
[default]
sso_session = my-sso
sso_account_id = 111122223333
sso_role_name = readOnly
region = us-west-2
output = text

[profile user1]
sso_session = my-sso
sso_account_id = 444455556666
sso_role_name = readOnly
region = us-east-1
output = json

[sso-session my-sso]
sso_region = us-east-1
sso_start_url = https://my-sso-portal.awsapps.com/start
sso_registration_scopes = sso:account:access
```

IAM Identity Center (Legacy SSO)

Questo esempio riguarda il metodo legacy di AWS IAM Identity Center. Per ulteriori informazioni, consulta the section called "Autenticazione in Centro identità IAM".

File di credenziali

Il file `credentials` non viene utilizzato per questo metodo di autenticazione.

File di configurazione

```
[default]
sso_start_url = https://my-sso-portal.awsapps.com/start
sso_region = us-east-1
sso_account_id = 111122223333
sso_role_name = readOnly
region = us-west-2
output = text

[profile user1]
sso_start_url = https://my-sso-portal.awsapps.com/start
sso_region = us-east-1
```

```
sso_account_id = 444455556666
sso_role_name = readOnly
region = us-east-1
output = json
```

Short-term credentials

Questo esempio riguarda le credenziali a breve termine di AWS Identity and Access Management. Per ulteriori informazioni, consulta the section called "Credenziali a breve termine".

File di credenziali

```
[default]
aws_access_key_id=ASIAIOSFODNN7EXAMPLE
aws_secret_access_key=wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY
aws_session_token
 = IQoJb3JpZ2luX2IQoJb3JpZ2luX2IQoJb3JpZ2luX2IQoJb3JpZ2luX2IQoJb3JpZVERYLONGSTRINGEXAMPLE

[user1]
aws_access_key_id=ASIAI44QH8DHBEXAMPLE
aws_secret_access_key=je7MtGbClwBF/2Zp9Utk/h3yCo8nvbEXAMPLEKEY
aws_session_token
 = fcZib3JpZ2luX2IQoJb3JpZ2luX2IQoJb3JpZ2luX2IQoJb3JpZ2luX2IQoJb3JpZVERYLONGSTRINGEXAMPLE
```

File di configurazione

```
[default]
region=us-west-2
output=json

[profile user1]
region=us-east-1
output=text
```

IAM role

Questo esempio serve per assumere un ruolo IAM. I profili che utilizzano i ruoli IAM estraggono le credenziali da un altro profilo e quindi applicano le autorizzazioni dei ruoli IAM. Negli esempi seguenti, `default` è il profilo di origine per le credenziali e `user1` prende in prestito le stesse credenziali per poi assumere un nuovo ruolo. Per ulteriori informazioni, consulta the section called "Ruoli IAM".

File di credenziali

Il file `credentials` dipende dall'autenticazione utilizzata dal profilo di origine. Nell'esempio seguente, il profilo di origine utilizza credenziali a breve termine.

```
[default]
aws_access_key_id=ASIAIOSFODNN7EXAMPLE
aws_secret_access_key=wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY
aws_session_token
 = IQoJb3JpZ2luX2IQoJb3JpZ2luX2IQoJb3JpZ2luX2IQoJb3JpZ2luX2IQoJb3JpZVERYLONGSTRINGEXAMPLE
```

File di configurazione

```
[default]
region=us-west-2
output=json

[profile user1]
role_arn=arn:aws:iam::777788889999:role/user1role
source_profile=default
role_session_name=session_user1
region=us-east-1
output=text
```

Amazon EC2 instance metadata credentials

Questo esempio riguarda le credenziali ottenute dai metadati dell' EC2 istanza Amazon di hosting. Per ulteriori informazioni, consulta the section called "Metadati Amazon EC2".

File di credenziali

Il file `credentials` non viene utilizzato per questo metodo di autenticazione.

File di configurazione

```
[default]
role_arn=arn:aws:iam::123456789012:role/defaultrole
credential_source=Ec2InstanceMetadata
region=us-west-2
output=json
```

```
[profile user1]
role_arn=arn:aws:iam::777788889999:role/user1role
credential_source=Ec2InstanceMetadata
region=us-east-1
output=text
```

Long-term credentials

> **Warning**
>
> Per evitare rischi per la sicurezza, non utilizzare gli utenti IAM per l'autenticazione quando sviluppi software creato ad hoc o lavori con dati reali. Utilizza invece la federazione con un provider di identità come AWS IAM Identity Center.

Questo esempio riguarda le credenziali a lungo termine di AWS Identity and Access Management. Per ulteriori informazioni, consulta the section called "Utenti IAM".

File di credenziali

```
[default]
aws_access_key_id=AKIAIOSFODNN7EXAMPLE
aws_secret_access_key=wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY

[user1]
aws_access_key_id=AKIAI44QH8DHBEXAMPLE
aws_secret_access_key=je7MtGbClwBF/2Zp9Utk/h3yCo8nvbEXAMPLEKEY
```

File di configurazione

```
[default]
region=us-west-2
output=json

[profile user1]
region=us-east-1
output=text
```

Per ulteriori informazioni e metodi di autorizzazione e credenziali aggiuntivi, consulta the section called "Utenti IAM".

Tipo di sezione: **sso-session**

La `sso-session` sezione del `config` file viene utilizzata per raggruppare le variabili di configurazione per l'acquisizione di token di accesso SSO, che possono quindi essere utilizzati per acquisire le credenziali. AWS Vengono utilizzate le impostazioni seguenti:

- (Obbligatorio) `sso_start_url`
- (Richiesto) `sso_region`
- `sso_account_id`
- `sso_role_name`
- `sso_registration_scopes`

È necessario definire una sezione `sso-session` e associarla a un profilo. `sso_region` e `sso_start_url` devono essere impostati all'interno della sezione `sso-session`. In genere, le impostazioni `sso_account_id` e `sso_role_name` devono essere configurate nella sezione `profile` in modo che l'SDK possa richiedere le credenziali SSO.

L'esempio seguente configura l'SDK in modo da richiedere le credenziali SSO e supporta l'aggiornamento automatico dei token:

```
[profile dev]
sso_session = my-sso
sso_account_id = 111122223333
sso_role_name = SampleRole

[sso-session my-sso]
sso_region = us-east-1
sso_start_url = https://my-sso-portal.awsapps.com/start
```

In questo modo, è possibile riutilizzare le configurazioni `sso-session` su più profili:

```
[profile dev]
sso_session = my-sso
sso_account_id = 111122223333
sso_role_name = SampleRole

[profile prod]
sso_session = my-sso
sso_account_id = 111122223333
```

```
sso_role_name = SampleRole2

[sso-session my-sso]
sso_region = us-east-1
sso_start_url = https://my-sso-portal.awsapps.com/start
```

Tuttavia, le impostazioni `sso_account_id` e `sso_role_name` non sono necessarie per tutti gli scenari di configurazione dei token SSO. Se l'applicazione utilizza solo servizi AWS che supportano l'autenticazione tramite token di connessione, le credenziali AWS tradizionali non sono necessarie. Questo tipo di autenticazione è uno schema di autenticazione HTTP che utilizza token di sicurezza noti come token di connessione. In questo scenario le impostazioni `sso_account_id` e `sso_role_name` non sono obbligatorie. Consulta la guida individuale del tuo AWS servizio per determinare se supporta l'autorizzazione del token al portatore.

Inoltre, gli ambiti di registrazione possono essere configurati all'interno di un parametro `sso-session`. Scope è un meccanismo della OAuth versione 2.0 per limitare l'accesso di un'applicazione all'account di un utente. Un'applicazione può richiedere uno o più ambiti e il token di accesso emesso all'applicazione sarà limitato agli ambiti consentiti. Questi ambiti definiscono le autorizzazioni richieste per l'autorizzazione per il client OIDC registrato e i token di accesso recuperati dal client. L'esempio seguente imposta `sso_registration_scopes` in modo da fornire l'accesso necessario per elencare account/ruoli:

```
[sso-session my-sso]
sso_region = us-east-1
sso_start_url = https://my-sso-portal.awsapps.com/start
sso_registration_scopes = sso:account:access
```

Il token di autenticazione viene memorizzato nella cache su disco all'interno della directory `~/.aws/sso/cache` con un nome di file basato sul nome della sessione.

Per ulteriori informazioni su questo tipo di configurazione, consulta the section called "Autenticazione in Centro identità IAM".

Tipo di sezione: **`services`**

La `services` sezione è un gruppo di impostazioni che configura gli endpoint personalizzati per Servizio AWS le richieste. Il profilo viene quindi collegato a una sezione `services`.

```
[profile dev]
```

```
services = my-services
```

La sezione `services` è suddivisa in sottosezioni da righe `<SERVICE> = `, dove `<SERVICE>` è la chiave identificativa Servizio AWS . L' Servizio AWS identificatore si basa sul modello API e sostituisce tutti gli spazi con caratteri `serviceId` di sottolineatura e tutte le lettere minuscole. Per un elenco di tutte le chiavi identificative del servizio da utilizzare nella sezione `services`, consulta Utilizzo degli endpoint in AWS CLI. La chiave identificativa del servizio è seguita da impostazioni annidate, ciascuna sulla propria riga e con un rientro di due spazi.

L'esempio seguente configura l'endpoint da utilizzare per le richieste effettuate al Amazon DynamoDB servizio nella *my-services* sezione utilizzata nel profilo. *dev* Tutte le righe con rientro immediatamente successive vengono incluse in tale sottosezione e si applicano a quel servizio.

```
[profile dev]
services = my-services

[services my-services]
dynamodb =
  endpoint_url = http://localhost:8000
```

Per ulteriori informazioni sugli endpoint specifici del servizio, consulta Utilizzo degli endpoint in AWS CLI.

Se il tuo profilo dispone di credenziali basate sui ruoli configurate tramite un parametro `source_profile` per consentire a IAM di assumere la funzionalità del ruolo, l'SDK utilizza solo le configurazioni di servizio relative al profilo specificato. Non utilizza profili a cui sono associati ruoli concatenati. Ad esempio, può utilizzare il seguente file `config` condiviso:

```
[profile A]
credential_source = Ec2InstanceMetadata
endpoint_url = https://profile-a-endpoint.aws/

[profile B]
source_profile = A
role_arn = arn:aws:iam::123456789012:role/roleB
services = profileB

[services profileB]
ec2 =
  endpoint_url = https://profile-b-ec2-endpoint.aws
```

Se usi il profilo B ed effettui una chiamata nel tuo codice verso Amazon EC2, l'endpoint si risolve come. `https://profile-b-ec2-endpoint.aws` Se il codice invia una richiesta a qualsiasi altro servizio, la risoluzione dell'endpoint non seguirà alcuna logica personalizzata. L'endpoint non viene risolto come l'endpoint globale definito nel profilo A. Affinché un endpoint globale abbia effetto sul profilo B, è necessario impostare `endpoint_url` direttamente all'interno del profilo B.

Dove vengono archiviate le impostazioni di configurazione?

AWS CLI Memorizza le informazioni riservate sulle credenziali specificate `aws configure` in un file locale denominato`credentials`, in una cartella denominata `.aws` nella home directory. Le altre opzioni di configurazione meno riservate specificate con `aws configure` vengono archiviate in un file locale denominato `config` che è archiviato nella cartella `.aws` della home directory.

> **Archiviazione delle credenziali nel file di configurazione**
>
> Puoi conservare tutte le impostazioni del tuo profilo in un unico file in quanto AWS CLI possono leggere le credenziali dal `config` file. Se in entrambi i file sono presenti credenziali per un profilo che condivide lo stesso nome, le chiavi nel file delle credenziali hanno la precedenza. Suggeriamo di conservare le credenziali nei file `credentials`. Questi file vengono utilizzati anche dai kit di sviluppo software in varie lingue ()SDKs. Se utilizzi uno di questi SDKs in aggiunta a AWS CLI, conferma se le credenziali devono essere archiviate nel loro file.

La posizione della home directory varia in base al sistema operativo, ma viene definita utilizzando le variabili di ambiente `%UserProfile%` in Windows e `$HOME` o ~ (tilde) nei sistemi basati su Unix. Puoi specificare una posizione non predefinita per i file impostando le variabili di ambiente `AWS_CONFIG_FILE` e `AWS_SHARED_CREDENTIALS_FILE` su un altro percorso locale. Per informazioni dettagliate, vedi Configurazione delle variabili d'ambiente per AWS CLI.

Quando utilizzi un profilo condiviso che specifica un ruolo AWS Identity and Access Management (IAM), AWS CLI chiama l' AWS STS `AssumeRole`operazione per recuperare le credenziali temporanee. Queste credenziali vengono quindi archiviate (in `~/.aws/cli/cache`). AWS CLI I comandi successivi utilizzano le credenziali temporanee memorizzate nella cache fino alla loro scadenza e, a quel punto, le aggiornano automaticamente. AWS CLI

Utilizzo di profili denominati

Se nessun profilo è definito in modo esplicito, viene utilizzato il profilo `default`.

Per usare un profilo denominato, aggiungi l'opzione `--profile` *`profile-name`* al comando. L'esempio seguente elenca tutte le tue EC2 istanze Amazon utilizzando le credenziali e le impostazioni definite nel `user1` profilo.

```
$ aws ec2 describe-instances --profile user1
```

Per utilizzare un profilo denominato per più comandi, puoi evitare di specificare il profilo in ogni comando impostando la variabile di ambiente `AWS_PROFILE` come profilo predefinito. È possibile sovrascrivere questa impostazione utilizzando il parametro `--profile`.

Linux or macOS

```
$ export AWS_PROFILE=user1
```

Windows

```
C:\> setx AWS_PROFILE user1
```

Se si utilizza `set` per impostare una variabile di ambiente, il valore utilizzato viene modificato fino al termine della sessione del prompt dei comandi corrente o finché non imposti la variabile su un valore diverso.

L'uso di `setx` per impostare una variabile di ambiente comporta la modifica del valore in tutte le shell di comando create dopo l'esecuzione del comando. Non influisce su alcuna shell di comando già in esecuzione nel momento in cui si esegue il comando. Chiudere e riavviare la shell dei comandi per visualizzare gli effetti della modifica.

L'impostazione della variabile di ambiente modifica il profilo di default fino al termine della sessione della shell o finché non si imposta la variabile su un valore diverso. Puoi rendere le variabili di ambiente persistenti per le sessioni future inserendole nello script di avvio della shell. Per ulteriori informazioni, consulta Configurazione delle variabili d'ambiente per AWS CLI.

Definizione e visualizzazione delle impostazioni di configurazione utilizzando i comandi

Esistono diversi modi per visualizzare e definire le impostazioni di configurazione utilizzando i comandi.

aws configure

Esegui questo comando per impostare e visualizzare rapidamente le credenziali, la Regione e il formato di output. L'esempio seguente illustra i valori di esempio.

```
$ aws configure
AWS Access Key ID [None]: AKIAIOSFODNN7EXAMPLE
AWS Secret Access Key [None]: wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY
Default region name [None]: us-west-2
Default output format [None]: json
```

aws configure set

Puoi impostare le credenziali o le impostazioni di configurazione utilizzando `aws configure set`. Specifica il profilo che intendi visualizzare o modificare con l'impostazione `--profile`.

Ad esempio, il comando seguente configura l'impostazione `region` nel profilo denominato `integ`.

```
$ aws configure set region us-west-2 --profile integ
```

Per rimuovere un'impostazione, eliminala manualmente nei file `config` e `credentials` in un editor di testo.

aws configure get

Puoi recuperare le credenziali o le impostazioni di configurazione impostate utilizzando `aws configure get`. Specifica il profilo che intendi visualizzare o modificare con l'impostazione `--profile`.

Ad esempio, il comando seguente recupera l'impostazione `region` nel profilo denominato `integ`.

```
$ aws configure get region --profile integ
us-west-2
```

Se l'output è vuoto, l'impostazione non è configurata esplicitamente e viene utilizzato il valore predefinito.

aws configure import

Importa le credenziali CSV generate dalla console web IAM. Non si tratta delle credenziali generate dal Centro identità IAM. I clienti che utilizzano il Centro identità IAM devono utilizzare

aws configure sso. Viene importato un file CSV con il nome del profilo corrispondente al nome utente. Il file .csv deve contenere le seguenti intestazioni.

- Nome utente
- ID chiave di accesso
- Chiave di accesso segreta

> **Note**
>
> Durante la creazione iniziale della coppia di chiavi, una volta chiusa la finestra di dialogo Scarica il file.csv, non è possibile accedere alla chiave di accesso segreta. Se hai bisogno di un file `.csv`, dovrai crearne uno personalmente con le intestazioni richieste e le informazioni sulle chiavi di accesso archiviate. Se non hai accesso alle informazioni sulla coppia di chiavi, devi crearne una nuova.

```
$ aws configure import --csv file://credentials.csv
```

aws configure list

Per elencare tutti i dati di configurazione, utilizza il comando `aws configure list`. Questo comando elenca le informazioni di configurazione del profilo, della chiave di accesso, della chiave segreta e della Regione utilizzate per il profilo specificato. Per ogni elemento di configurazione, mostra il valore, dove è stato recuperato il valore di configurazione e il nome della variabile di configurazione.

Ad esempio, se fornisci la variabile Regione AWS in un ambiente, questo comando ti mostra il nome della regione che hai configurato, che questo valore proviene da una variabile di ambiente e il nome della variabile di ambiente.

Per quanto riguarda i metodi per le credenziali temporanee, come ruoli e Centro identità IAM, il comando visualizza la chiave di accesso e la chiave di accesso segreta temporaneamente memorizzate nella cache.

```
$ aws configure list
NAME       : VALUE                : TYPE                    : LOCATION
profile    : <not set>            : None                    : None
access_key : ****************ABCD : shared-credentials-file :
secret_key : ****************ABCD : shared-credentials-file :
```

```
region     : us-west-2           : env                    : AWS_DEFAULT_REGION
```

aws configure list-profiles

Per elencare tutti i nomi del profilo, utilizza il comando `aws configure list-profiles`.

```
$ aws configure list-profiles
default
test
```

aws configure mfa-login

Esegui questo comando per configurare un nuovo profilo da utilizzare con l'autenticazione a più fattori (MFA) e le tue credenziali utente IAM nel profilo specificato. Se non viene specificato alcun profilo, l'MFA si basa sul `default` profilo. Se non è configurato alcun profilo predefinito, il `mfa-login` comando richiede AWS le credenziali prima di richiedere le informazioni sull'MFA. Il seguente esempio di comando utilizza la configurazione predefinita e crea un profilo MFA.

```
$ aws configure mfa-login
MFA serial number or ARN: arn:aws:iam::123456789012:mfa/MFADeviceName
MFA token code: 123456
Profile to update [session-MFADeviceName]:
Temporary credentials written to profile 'session-MFADeviceName'
Credentials will expire at 2023-05-19 18:06:10 UTC
To use these credentials, specify --profile session-MFADeviceName when running AWS
 CLI commands
```

Per aggiornare un profilo esistente, utilizzare il `--update-profile` parametro.

```
$ aws configure mfa-login --profile myprofile --update-profile mfaprofile
MFA token code: 123456
Temporary credentials written to profile 'mfaprofile'
Credentials will expire at 2023-05-19 18:06:10 UTC
To use these credentials, specify --profile mfaprofile when running AWS CLI commands
```

Attualmente questo comando supporta solo autenticatori OTP (One-Time Password) basati su hardware o software. Le password e i dispositivi U2F non sono attualmente supportati con questo comando.

Per ulteriori informazioni sull'utilizzo della MFA con IAM, consulta Autenticazione a AWS più fattori in IAM nella Guida per l'AWS Identity and Access Management utente.

aws configure sso

Esegui questo comando per impostare e visualizzare rapidamente AWS IAM Identity Center le credenziali, la regione e il formato di output. L'esempio seguente illustra i valori di esempio.

```
$ aws configure sso
SSO session name (Recommended): my-sso
SSO start URL [None]: https://my-sso-portal.awsapps.com/start
SSO region [None]: us-east-1
SSO registration scopes [None]: sso:account:access
```

aws configure sso-session

Esegui questo comando per impostare e visualizzare rapidamente le AWS IAM Identity Center credenziali, la regione e il formato di output nella `sso-session` sezione dei file `credentials` and`config`. L'esempio seguente illustra i valori di esempio.

```
$ aws configure sso-session
SSO session name: my-sso
SSO start URL [None]: https://my-sso-portal.awsapps.com/start
SSO region [None]: us-east-1
SSO registration scopes [None]: sso:account:access
```

aws configure export-credentials

Esegui questo comando per esportare le credenziali attualmente impostate nel formato specificato. Per impostazione predefinita, il comando esporta le credenziali predefinite nel `process` formato, che è un formato JSON supportato dal formato di credenziali AWS SDKs and Tools.

```
$ aws configure export-credentials
{
  "Version": 1,
  "AccessKeyId": "AKIAIOSFODNN7EXAMPLE",
  "SecretAccessKey": "wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY"
}
```

Per esportare un profilo e un formato specifici, utilizza le opzioni `--profile` e `--format`. Le opzioni di formato sono:

- (impostazione predefinita) **`process`**- Il formato JSON supportato dalla configurazione and Tools. AWS SDKs `credential_process`

- **`env`**: variabili di ambiente in formato shell esportato.
- **`env-no-export`**: variabili di ambiente non esportate in formato shell.
- **`powershell`**- Variabili d'ambiente in PowerShell formato.
- **`windows-cmd`**: variabili di ambiente in formato riga di comando di Windows.

L'esempio seguente esporta il profilo `user1` in un formato shell esportato.

```
$ aws configure export-credentials --profile user1 --format env
export AWS_ACCESS_KEY_ID=AKIAIOSFODNN7EXAMPLE
export AWS_SECRET_ACCESS_KEY=wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY
```

Esempi di comandi per impostare nuove configurazioni e credenziali

Negli esempi seguenti viene mostrata la configurazione di un profilo predefinito con credenziali, Regione e output specificati per diversi metodi di autenticazione.

Console sessions as short-term credentials

Questo esempio consente di utilizzare le credenziali di console esistenti con il `aws login` comando. Questo metodo di accesso può essere utilizzato con credenziali root create durante la configurazione iniziale dell'account, con un utente IAM o con un'identità federata del provider di identità. Per ulteriori informazioni, consulta Accesso per lo sviluppo AWS locale utilizzando le credenziali della console.

```
$ aws login
No AWS region has been configured. The AWS region is the geographic location of your
 AWS resources.

If you've used AWS before and already have resources in your account, tell us which
 region they were
created in. If you haven't created resources in your account before, you can pick
 the region closest to you:
https://docs.aws.amazon.com/global-infrastructure/latest/regions/aws-regions.html.
 You are able to change
the region in the CLI at any time with the command `aws configure set region
 NEW_REGION`.
AWS Region [us-east-1]: us-east-1
Attempting to open the login page for `us-east-1` in your default browser. If the
 browser does not open,
```

```
use the following URL to complete your login:

https://us-east-1.signin.aws.amazon.com/v1/authorize<abbreviated>

If you cannot connect to this URL, make sure that you have specified a valid region.

Waiting for login...

Logged in with role `arn:aws:sts::012345678910:user/iam-user`, and configured
 profile
`default` to use `us-east-1`. This session will expire on October 14, 2025 at 2:04
 PST. After this time, you
can renew your session with `aws login`.
```

IAM Identity Center (SSO)

Questo esempio riguarda l' AWS IAM Identity Center utilizzo della `aws configure sso` procedura guidata. Per ulteriori informazioni, consulta the section called "Autenticazione in Centro identità IAM".

```
$ aws configure sso
SSO session name (Recommended): my-sso
SSO start URL [None]: https://my-sso-portal.awsapps.com/start
SSO region [None]:us-east-1

Attempting to automatically open the SSO authorization page in your default browser.

There are 2 AWS accounts available to you.
> DeveloperAccount, developer-account-admin@example.com (111122223333)
  ProductionAccount, production-account-admin@example.com (444455556666)

Using the account ID 111122223333

There are 2 roles available to you.
> ReadOnly
  FullAccess

Using the role name "ReadOnly"

CLI default client Region [None]: us-west-2
CLI default output format [None]: json
CLI profile name [123456789011_ReadOnly]: user1
```

IAM Identity Center (Legacy SSO)

Questo esempio riguarda il metodo precedente di AWS IAM Identity Center utilizzo della `aws configure sso` procedura guidata. Per utilizzare l'SSO precedente, lascia vuoto il nome della sessione. Per ulteriori informazioni, consulta the section called "Autenticazione in Centro identità IAM".

```
$ aws configure sso
SSO session name (Recommended):
SSO start URL [None]: https://my-sso-portal.awsapps.com/start
SSO region [None]:us-east-1

SSO authorization page has automatically been opened in your default browser.
Follow the instructions in the browser to complete this authorization request.

There are 2 AWS accounts available to you.
> DeveloperAccount, developer-account-admin@example.com (111122223333)
  ProductionAccount, production-account-admin@example.com (444455556666)

Using the account ID 111122223333

There are 2 roles available to you.
> ReadOnly
  FullAccess

Using the role name "ReadOnly"

CLI default client Region [None]: us-west-2
CLI default output format [None]: json
CLI profile name [123456789011_ReadOnly]: user1
```

Short-term credentials

Questo esempio riguarda le credenziali a breve termine di AWS Identity and Access Management. Dopo aver utilizzato la procedura guidata aws configure per impostare i valori iniziali, il comando `aws configure set` assegna l'ultimo valore necessario. Per ulteriori informazioni, consulta the section called "Credenziali a breve termine".

```
$ aws configure
AWS Access Key ID [None]: AKIAIOSFODNN7EXAMPLE
AWS Secret Access Key [None]: wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY
Default region name [None]: us-west-2
```

```
Default output format [None]: json
$ aws configure set
 aws_session_token fcZib3JpZ2luX2IQoJb3JpZ2luX2IQoJb3JpZ2luX2IQoJb3JpZ2luX2IQoJb3JpZVERYLONG
```

IAM role

Questo esempio serve per assumere un ruolo IAM. I profili che utilizzano i ruoli IAM estraggono le credenziali da un altro profilo e quindi applicano le autorizzazioni dei ruoli IAM. Negli esempi seguenti, `default` è il profilo di origine per le credenziali e `user1` prende in prestito le stesse credenziali per poi assumere un nuovo ruolo. Non esiste una procedura guidata per questo processo, pertanto ogni valore viene impostato utilizzando il comando `aws configure set`. Per ulteriori informazioni, consulta the section called "Ruoli IAM".

```
$ aws configure set role_arn arn:aws:iam::123456789012:role/defaultrole
$ aws configure set source_profile default
$ aws configure set role_session_name session_user1
$ aws configure set region us-west-2
$ aws configure set output json
```

Amazon EC2 instance metadata credentials

Questo esempio riguarda le credenziali ottenute dai metadati dell' EC2 istanza Amazon di hosting. Non esiste una procedura guidata per questo processo, pertanto ogni valore viene impostato utilizzando il comando `aws configure set`. Per ulteriori informazioni, consulta the section called "Metadati Amazon EC2".

```
$ aws configure set role_arn arn:aws:iam::123456789012:role/defaultrole
$ aws configure set credential_source Ec2InstanceMetadata
$ aws configure set region us-west-2
$ aws configure set output json
```

Long-term credentials

> **Warning**
>
> Per evitare rischi per la sicurezza, non utilizzare gli utenti IAM per l'autenticazione quando sviluppi software creato ad hoc o lavori con dati reali. Utilizza invece la federazione con un provider di identità come AWS IAM Identity Center.

Questo esempio riguarda le credenziali a lungo termine di AWS Identity and Access Management. Per ulteriori informazioni, consulta the section called "Utenti IAM".

```
$ aws configure
AWS Access Key ID [None]: AKIAIOSFODNN7EXAMPLE
AWS Secret Access Key [None]: wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY
Default region name [None]: us-west-2
Default output format [None]: json
```

Impostazioni supportate del file `config`

Argomenti

- Impostazioni globali
- Impostazioni dei comandi S3 personalizzati

Le seguenti impostazioni sono supportate nel file `config`. Vengono utilizzati i valori elencati nel profilo specificato (o nel profilo predefinito), a meno che non vengano sostituiti grazie alla presenza di una variabile di ambiente con lo stesso nome o di un'opzione della riga di comando con lo stesso nome. Per ulteriori informazioni sulle impostazioni di ordine che hanno la precedenza, consulta Configurazione delle impostazioni della AWS CLI.

Impostazioni globali

`account_id_endpoint_mode`

Speciifica se utilizzare un endpoint AWS IDs basato sull'account per le chiamate ai servizi supportati. Servizi AWS Per ulteriori informazioni sugli endpoint basati sull'account, consulta the section called "Endpoint basati sull'account".

Questa impostazione può essere configurata come segue:

- (Impostazione predefinita) **`preferred`**: l'endpoint deve includere l'ID dell'account, se disponibile.
- **`disabled`**: un endpoint risolto non include l'ID dell'account.
- **`required`**: l'endpoint deve includere l'ID dell'account. Se l'ID dell'account non è disponibile, l'SDK genera un errore.

Può essere sostituita dalla variabile di ambiente AWS_ACCOUNT_ID_ENDPOINT_MODE. Per utilizzare gli endpoint basati sull'account, l'ID deve essere impostato nella variabile di ambiente AWS_ACCOUNT_ID o nell'impostazione aws_account_id.

```
account_id_endpoint_mode = preferred
```

> **Priorità degli endpoint**
>
> Le impostazioni di configurazione degli endpoint si trovano in più posizioni, ad esempio le variabili di sistema o di ambiente utente, i file di AWS configurazione locali o sono dichiarate esplicitamente nella riga di comando come parametro. AWS CLI Controlla queste impostazioni dell'endpoint in un ordine particolare e utilizza l'impostazione dell'endpoint con la massima precedenza. Per l'elenco delle priorità degli endpoint, consulta the section called "Configurazione dell'endpoint e precedenza delle impostazioni".

aws_access_key_id

Speciifica la chiave di AWS accesso utilizzata come parte delle credenziali per autenticare la richiesta di comando. Sebbene tale impostazione possa essere archiviata nel file `config`, ti consigliamo di archiviarla nel file `credentials`.

Può essere sostituita dalla variabile di ambiente `AWS_ACCESS_KEY_ID`. Non è possibile specificare l'ID chiave di accesso come un'opzione della riga di comando.

```
aws_access_key_id = AKIAIOSFODNN7EXAMPLE
```

aws_account_id

Speciifica l'ID dell'endpoint AWS basato sull'account da utilizzare per le chiamate ai servizi supportati. Servizi AWS Per ulteriori informazioni sugli endpoint basati sull'account, consulta the section called "Endpoint basati sull'account".

Può essere sostituita dalla variabile di ambiente AWS_ACCOUNT_ID. La variabile di ambiente AWS_ACCOUNT_ID_ENDPOINT_MODE o l'impostazione account_id_endpoint_mode deve essere impostata su `preferred` o `required` per utilizzare questa impostazione.

```
aws_account_id = 123456789EXAMPLE
```

> **Priorità degli endpoint**
>
> Le impostazioni di configurazione degli endpoint si trovano in più posizioni, ad esempio le variabili di sistema o di ambiente utente, i file di AWS configurazione locali o sono dichiarate esplicitamente nella riga di comando come parametro. AWS CLI Controlla queste impostazioni dell'endpoint in un ordine particolare e utilizza l'impostazione dell'endpoint con la massima precedenza. Per l'elenco delle priorità degli endpoint, consulta the section called "Configurazione dell'endpoint e precedenza delle impostazioni".

`aws_secret_access_key`

Speciifica la chiave AWS segreta utilizzata come parte delle credenziali per autenticare la richiesta di comando. Sebbene tale impostazione possa essere archiviata nel file `config`, ti consigliamo di archiviarla nel file `credentials`.

Può essere sostituita dalla variabile di ambiente `AWS_SECRET_ACCESS_KEY`. Non è possibile specificare la chiave di accesso segreta come un'opzione della riga di comando.

```
aws_secret_access_key = wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY
```

`aws_session_token`

Specifica un token di sessione. AWS È richiesto un token di sessione solo se si specificano manualmente credenziali di sicurezza temporanee. Sebbene tale impostazione possa essere archiviata nel file `config`, ti consigliamo di archiviarla nel file `credentials`.

Può essere sostituita dalla variabile di ambiente `AWS_SESSION_TOKEN`. Non è possibile specificare il token di sessione come un'opzione della riga di comando.

```
aws_session_token = AQoEXAMPLEH4aoAH0gNCAPyJxz4BlCFFxWNE1OPTgk5TthT
+FvwqnKwRcOIfrRh3c/LTo6UDdyJwOOvEVPvLXCrrrUtdnniCEXAMPLE/
IvU1dYUg2RVAJBanLiHb4IgRmpRV3zrkuWJOgQs8IZZaIv2BXIa2R4Olgk
```

`ca_bundle`

Specifica un bundle di certificati CA (file con estensione `.pem`) utilizzato per verificare i certificati SSL.

Può essere sostituita dalla variabile di ambiente AWS_CA_BUNDLE o dall'opzione della riga di comando --ca-bundle.

```
ca_bundle = dev/apps/ca-certs/cabundle-2019mar05.pem
```

cli_auto_prompt

Abilita il prompt automatico per la AWS CLI versione 2. È possibile utilizzare due impostazioni:

- **on** utilizza la modalità di prompt automatico completa ogni volta che si tenta di eseguire un comando `aws`. Include il dover premere INVIO dopo un comando completo o incompleto.

  ```
  cli_auto_prompt = on
  ```

- **on-partial** utilizza la modalità di prompt automatico parziale. Se un comando è incompleto o non può essere eseguito a causa di errori di convalida sul lato client, viene utilizzato il prompt automatico. Questa modalità è particolarmente utile se sono disponibili script e runbook preesistenti o per ricevere un prompt automatico solo per i comandi che non si conoscono, piuttosto che per ogni comando.

  ```
  cli_auto_prompt = on-partial
  ```

Per sovrascrivere questa impostazione, utilizza la variabile di ambiente aws_cli_auto_prompt o i parametri della riga di comando --cli-auto-prompt e --no-cli-auto-prompt.

Per informazioni sulla funzionalità di richiesta automatica della AWS CLI versione 2, vedere. Attivazione e utilizzo dei prompt dei comandi in AWS CLI

cli_binary_format

Specifica in che modo la AWS CLI versione 2 interpreta i parametri di input binari. Può essere uno dei seguenti valori:

- Base64: si tratta del valore predefinito. Un parametro di input digitato come oggetto binario di grandi dimensioni (BLOB) accetta una stringa con codifica base64. Per passare il contenuto binario vero, inserire il contenuto in un file e fornire il percorso e il nome del file con il prefisso `fileb://` come valore del parametro. Per passare il testo codificato in base 64 contenuto in un file, specificare il percorso e il nome del file con il prefisso `file://` come valore del parametro.

- raw-in-base64-out: impostazione predefinita per la AWS CLI versione 1. Se il valore dell'impostazione è `raw-in-base64-out`, i file a cui si fa riferimento utilizzando il prefisso `file://` vengono letti come testo e quindi la AWS CLI tenta di codificarli in codice binario.

Questa voce non dispone di una variabile di ambiente equivalente. È possibile specificare il valore su un singolo comando utilizzando il parametro `--cli-binary-format raw-in-base64-out`.

```
cli_binary_format = raw-in-base64-out
```

Se si fa riferimento a un valore binario in un file utilizzando la notazione del `fileb://` prefisso, si aspetta AWS CLI sempre che il file contenga contenuto binario non elaborato e non tenta di convertire il valore.

Se si fa riferimento a un valore binario in un file utilizzando la notazione del `file://` prefisso, AWS CLI gestisce il file in base all'impostazione corrente. `cli_binary_format` Se il valore di tale impostazione è `base64` (l'impostazione predefinita quando non è impostata in modo esplicito), si AWS CLI aspetta che il file contenga testo con codifica base64. Se il valore di tale impostazione è`raw-in-base64-out`, si AWS CLI aspetta che il file contenga contenuto binario non elaborato.

`cli_help_output`

A partire dalla versione `2.31.0` La visualizzazione del comando `help` è configurata dall'impostazione `cli_help_output` e presenta i seguenti valori:

- (impostazione predefinita) `terminal` - Apre la pagina man nel terminale.
- `browser` - Apre la pagina man come file HTML locale nel browser predefinito. Viene stampato un avviso sul terminale quando viene aperto il browser predefinito e un messaggio di errore se AWS CLI non è possibile aprire il browser.
- `url`- Stampa l'URL della Guida AWS CLI di riferimento online per la versione di AWS CLI cui hai installato. Le impostazioni per la paginazione lato client, come la variabile di `AWS_PAGER` ambiente, vengono rispettate.

```
cli_help_output = browser
```

Per ulteriori informazioni sul comando `help`, consulta the section called "Chiedere aiuto".

`cli_history`

Disabilitato per impostazione predefinita. Questa impostazione abilita la cronologia dei comandi per la AWS CLI. Dopo aver abilitato questa impostazione, AWS CLI registra la cronologia dei comandi. `aws`

```
cli_history = enabled
```

Puoi generare l'elenco della cronologia con il comando `aws history list` e utilizzare gli `command_ids` risultanti nel comando `aws history show` per i dettagli. Per ulteriori informazioni, consulta `aws history` nella Guida di riferimento della AWS CLI .

`cli_pager`

Specifica il programma pager utilizzato per l'output. Per impostazione predefinita, AWS CLI la versione 2 restituisce tutto l'output tramite il programma pager predefinito del sistema operativo.

Può essere sostituita dalla variabile di ambiente AWS_PAGER.

```
cli_pager=less
```

`cli_timestamp_format`

Specifica il formato dei valori di timestamp inclusi nell'output. Puoi specificare uno dei seguenti valori:

- iso8601 — Il valore predefinito per la AWS CLI versione 2. Se specificato, AWS CLI riformatta tutti i timestamp nell'output in base alla norma ISO 8601. I timestamp in formato ISO 8601 assomigliano ai seguenti esempi. L'esempio seguente dimostra come si formatta l'ora separando la data e l'ora con una `T` e inserendo una `Z` dopo l'ora.

  ```
  YYYY-MM-DDThh:mm:ssZ
  ```

  Negli esempi seguenti viene mostrato un timestamp che utilizza la formattazione precedente.

  ```
  2024-05-08T15:16:43Z
  ```

- wire — Il valore predefinito per la versione 1. AWS CLI Se specificato, AWS CLI visualizza tutti i valori del timestamp esattamente come ricevuti nella risposta alla query HTTP.

Questa impostazione non dispone di una variabile di ambiente o di un'opzione della riga di comando equivalente. Questa impostazione non modifica gli input relativi al timestamp, ma solo la formattazione dell'output.

```
cli_timestamp_format = iso8601
```

credential_process

Speciifica un comando esterno da eseguire per generare o recuperare le AWS CLI credenziali di autenticazione da utilizzare per questo comando. Il comando deve restituire le credenziali in un formato specifico. Per ulteriori informazioni sull'utilizzo di questa impostazione, consulta Acquisizione di credenziali con un processo esterno nella CLI.

Questa voce non dispone di una variabile di ambiente o di un'opzione della riga di comando equivalente.

```
credential_process = /opt/bin/awscreds-retriever --username susan
```

credential_source

Utilizzato all'interno di EC2 istanze o contenitori Amazon per specificare dove AWS CLI trovare le credenziali da utilizzare per assumere il ruolo specificato con il `role_arn` parametro. Non è possibile specificare sia `source_profile` sia `credential_source` nello stesso profilo.

Questo parametro può avere uno dei seguenti tre valori:

- Ambiente: specifica che deve recuperare le AWS CLI credenziali di origine dalle variabili di ambiente.
- Ec2 InstanceMetadata: specifica che deve essere utilizzato il ruolo IAM AWS CLI collegato al profilo dell'EC2 istanza per ottenere le credenziali di origine.
- EcsContainer— Speciifica che deve utilizzare AWS CLI il ruolo IAM collegato al contenitore ECS come credenziali di origine.

```
credential_source = Ec2InstanceMetadata
```

duration_seconds

Specifica la durata massima della sessione del ruolo, in secondi. Questo valore può variare da 900 secondi (15 minuti) fino alla durata massima della sessione per il ruolo (massimo 43200

minuti). Si tratta di un parametro facoltativo e, per impostazione predefinita, il valore è impostato su 3600 secondi.

`endpoint_url`

Specifica l'endpoint utilizzato per tutte le richieste del servizio. Se questa impostazione viene utilizzata nella sezione `services` del file `config`, l'endpoint viene utilizzato solo per il servizio specificato. Per ulteriori informazioni, consulta the section called "Imposta un endpoint globale per tutti i Servizi AWS".

L'esempio seguente utilizza l'endpoint globale `http://localhost:1234` e un endpoint specifico del servizio di `http://localhost:4567` per Amazon S3.

```
[profile dev]
endpoint_url = http://localhost:1234
services = s3-specific

[services s3-specific]
s3 =
  endpoint_url = http://localhost:4567
```

> **Priorità degli endpoint**
>
> Le impostazioni di configurazione degli endpoint si trovano in più posizioni, ad esempio le variabili di sistema o di ambiente utente, i file di AWS configurazione locali o sono dichiarate esplicitamente nella riga di comando come parametro. AWS CLI Controlla queste impostazioni dell'endpoint in un ordine particolare e utilizza l'impostazione dell'endpoint con la massima precedenza. Per l'elenco delle priorità degli endpoint, consulta the section called "Configurazione dell'endpoint e precedenza delle impostazioni".

`ignore_configure_endpoint_urls`

Se abilitato, AWS CLI ignora tutte le configurazioni personalizzate degli endpoint specificate nel file. `config` I valori validi sono **`true`** e **`false`**.

```
ignore_configure_endpoint_urls = true
```

> **Priorità degli endpoint**
>
> Le impostazioni di configurazione degli endpoint si trovano in più posizioni, ad esempio le variabili di sistema o di ambiente utente, i file di AWS configurazione locali o sono dichiarate esplicitamente nella riga di comando come parametro. AWS CLI Controlla queste impostazioni dell'endpoint in un ordine particolare e utilizza l'impostazione dell'endpoint con la massima precedenza. Per l'elenco delle priorità degli endpoint, consulta the section called "Configurazione dell'endpoint e precedenza delle impostazioni".

`external_id`

Specifica un identificatore univoco che viene utilizzato da terze parti per assumere un ruolo negli account dei relativi clienti. Questo è mappato al parametro `ExternalId` nell'operazione `AssumeRole`. Questo parametro è necessario solo se la policy di attendibilità per il ruolo specifica un valore per `ExternalId`. Per ulteriori informazioni, consulta Come utilizzare un ID esterno per concedere l'accesso alle tue AWS risorse a una terza parte nella Guida per l'utente IAM.

`max_attempts`

Speciifica il valore massimo di tentativi utilizzato dal gestore dei tentativi, in AWS CLI base al quale la chiamata iniziale viene conteggiata ai fini del `max_attempts` valore fornito.

È possibile sovrascrivere questo valore utilizzando la variabile di ambiente `AWS_MAX_ATTEMPTS`.

```
max_attempts = 3
```

`mfa_serial`

Il numero di identificazione del dispositivo MFA da utilizzare durante l'assunzione di un ruolo. È obbligatorio specificare questo valore se la policy di attendibilità del ruolo assunto include una condizione che richiede l'autenticazione MFA. Il valore può essere un numero di serie per un dispositivo hardware (ad esempio `GAHT12345678`) o un Amazon Resource Name (ARN) per un dispositivo MFA virtuale (ad esempio `arn:aws:iam::123456789012:mfa/user`).

`output`

Specifica il formato di output predefinito per i comandi richiesti utilizzando questo profilo. Puoi specificare uno qualsiasi dei valori seguenti:

- **`json`**: l'output è formattato come una stringa JSON.

- **`yaml`**: l'output è formattato come una stringa YAML.
- **`yaml-stream`**: l'output viene inviato in streaming e formattato come una stringa YAML. Lo streaming consente una gestione più rapida di tipi di dati di grandi dimensioni.
- **`text`**: l'output è formattato come più righe di valori di stringa separati da tabulazioni. Questa formattazione può essere utile per passare l'output a un elaboratore di testi, ad esempio `grep`, `sed` o `awk`.
- **`table`**: l'output è formattato come una tabella in cui si utilizzano i caratteri +|- per formare i bordi delle celle. In genere presenta le informazioni in un formato comprensibile molto più semplice da leggere rispetto ad altri, ma non altrettanto utile a livello programmatico.

Può essere sostituita dalla variabile di ambiente `AWS_DEFAULT_OUTPUT` o dall'opzione della riga di comando `--output`.

```
output = table
```

`parameter_validation`

Speciifica se il AWS CLI client tenta di convalidare i parametri prima di inviarli all'endpoint del servizio. AWS

- true: si tratta del valore predefinito. Se specificato, AWS CLI esegue la convalida locale dei parametri della riga di comando.
- false — Se specificato, AWS CLI non convalida i parametri della riga di comando prima di inviarli all'endpoint del AWS servizio.

Questa voce non dispone di una variabile di ambiente o di un'opzione della riga di comando equivalente.

```
parameter_validation = false
```

`region`

Speciifica a chi Regione AWS inviare le richieste per i comandi richiesti utilizzando questo profilo.

- Puoi specificare i codici di Regione disponibili per il servizio scelto, come indicato in Regioni ed endpoint AWS in Riferimenti generali di Amazon Web Services.
- `aws_global`consente di specificare l'endpoint globale per i servizi che supportano un endpoint globale oltre agli endpoint regionali, come AWS Security Token Service ()AWS STS e Amazon Simple Storage Service (Amazon S3).

È possibile sovrascrivere questo valore utilizzando la variabile di ambiente `AWS_REGION`, la variabile di ambiente `AWS_DEFAULT_REGION` o l'opzione della riga di comando `--region`.

```
region = us-west-2
```

`request_checksum_calculation`

Specifica quando viene calcolato un checksum per i payload di richiesta e dispone delle seguenti opzioni:

- `when_supported`: (impostazione predefinita) il checksum del payload di richiesta viene calcolato quando un'operazione specifica un algoritmo di checksum nel modello di servizio o richiede i checksum di richiesta.
- `when_required`: il checksum del payload di richiesta viene calcolato quando un'operazione richiede i checksum di richiesta o quando un utente fornisce un `requestAlgorithmMember` modellato dal Servizio AWS.

```
request_checksum_calculation = when_supported
```

La variabile di ambiente AWS_REQUEST_CHECKSUM_CALCULATION sovrascrive questa impostazione.

`response_checksum_validation`

Specifica quando viene eseguita la convalida del checksum per i payload di risposta e dispone delle seguenti opzioni:

- `when_supported`— (Impostazione predefinita) La convalida del checksum del payload di risposta viene eseguita quando un'operazione specifica un algoritmo di risposta nel modello di servizio supportato. AWS CLI
- `when_required`— La convalida del checksum del payload di risposta viene eseguita quando un'operazione specifica un algoritmo di risposta nel suo modello di servizio AWS CLI supportato e l'utente imposta il modello su nell'input dell'operazione. `requestValidationModeMember` ENABLED

```
response_checksum_validation = when_supported
```

La variabile di ambiente AWS_RESPONSE_CHECKSUM_VALIDATION sovrascrive questa impostazione.

retry_mode

Speciifica la modalità di ripetizione utilizzata. AWS CLI Sono disponibili tre modalità di ripetizione dei tentativi: `standard` (impostazione predefinita), `legacy` (impostazione predefinita) e `adaptive`. Per ulteriori informazioni sui tentativi, consulta Tentativi della AWS CLI nella AWS CLI.

È possibile sovrascrivere questo valore utilizzando la variabile di ambiente `AWS_RETRY_MODE`.

```
retry_mode = standard
```

role_arn

Speciifica l'Amazon Resource Name (ARN) di un ruolo IAM che desideri utilizzare per eseguire AWS CLI i comandi. Puoi anche specificare uno dei seguenti parametri per identificare le credenziali che dispongono dell'autorizzazione per assumere questo ruolo:

- source_profile
- credential_source

```
role_arn = arn:aws:iam::123456789012:role/role-name
```

La variabile di ambiente `AWS_ROLE_ARN` sovrascrive questa impostazione.

Per ulteriori informazioni sull'uso di identità web, consulta the section called "Assunzione di un ruolo con un'identità Web".

role_session_name

Specifica il nome da associare alla sessione del ruolo. Questo valore viene fornito al parametro `RoleSessionName` quando ' AWS CLI chiama l'operazione `AssumeRole` e diventa parte dell'ARN utente del ruolo assunto: `arn:aws:sts::123456789012:assumed-role/role_name/role_session_name`. Si tratta di un parametro facoltativo. Se non fornisci questo valore, viene automaticamente generato un nome della sessione. Questo nome viene visualizzato nei log di AWS CloudTrail per le voci associate a questa sessione.

```
role_session_name = maria_garcia_role
```

La variabile di ambiente `AWS_ROLE_SESSION_NAME` sovrascrive questa impostazione.

Per ulteriori informazioni sull'uso di identità web, consulta the section called "Assunzione di un ruolo con un'identità Web".

`services`

Specifica la configurazione del servizio da utilizzare per il profilo.

```
[profile dev-s3-specific-and-global]
endpoint_url = http://localhost:1234
services = s3-specific

[services s3-specific]
s3 =
  endpoint_url = http://localhost:4567
```

Per ulteriori informazioni sulla sezione `services`, consulta the section called "`services`".

La variabile di ambiente `AWS_ROLE_SESSION_NAME` sovrascrive questa impostazione.

Per ulteriori informazioni sull'uso di identità web, consulta the section called "Assunzione di un ruolo con un'identità Web".

`sdk_ua_app_id`

Un singolo Account AWS può essere utilizzato da più applicazioni del cliente a cui effettuare chiamate. Servizi AWS L'ID dell'applicazione identifica quale applicazione di origine ha effettuato una serie di chiamate utilizzando un Servizio AWS. AWS SDKs e i servizi non utilizzano o interpretano questo valore se non per riemergere nelle comunicazioni con i clienti. Ad esempio, questo valore può essere incluso nelle e-mail operative per identificare in modo univoco quale delle applicazioni è associata alla notifica.

L'ID dell'applicazione è una stringa composta al massimo da 50 caratteri. Sono consentite lettere, numeri e i seguenti caratteri speciali: `! $ % & * + - . , ^ _ ` | ~` Per impostazione predefinita, non viene assegnato alcun valore.

```
sdk_ua_app_id = prod1
```

Questa impostazione può essere sovrascritta utilizzando la variabile di ambiente `AWS_SDK_UA_APP_ID`. Non è possibile impostare questo valore come parametro della riga di comando.

`sigv4a_signing_region_set`

Specifica le Regioni da utilizzare quando si firma con SigV4a utilizzando un elenco delimitato da virgole. Se questa variabile non è impostata, AWS CLI utilizza l'impostazione predefinita utilizzata da Servizio AWS. Se non Servizio AWS ha un valore predefinito, la firma della richiesta diventa valida in tutte le regioni utilizzando un valore di*.

```
sigv4a_signing_region_set = us-west-2, us-east-1
```

Per ulteriori informazioni su SigV4a, consulta AWS Signature Version 4 per le richieste API nella Guida per l'utente IAM

Questa impostazione può essere sovrascritta utilizzando la variabile di ambiente AWS_SIGV4A_SIGNING_REGION_SET. Non è possibile impostare questo valore come parametro della riga di comando.

`source_profile`

Specifica un profilo denominato con credenziali a lungo termine che l' AWS CLI può utilizzare per assumere un ruolo specificato con il parametro `role_arn`. Non è possibile specificare sia `source_profile` sia `credential_source` nello stesso profilo.

```
source_profile = production-profile
```

`sso_account_id`

Speciifica l'ID dell' AWS account che contiene il ruolo IAM con l'autorizzazione che desideri concedere all'utente IAM Identity Center associato.

Questa impostazione non dispone di una variabile di ambiente o opzione della riga di comando.

```
sso_account_id = 123456789012
```

`sso_region`

Speciifica la AWS regione che contiene l'host del portale di AWS accesso. Questa è separata e può essere una regione diversa rispetto al parametro `region` predefinito della CLI.

Questa impostazione non dispone di una variabile di ambiente o opzione della riga di comando.

```
sso_region = us_west-2
```

sso_registration_scopes

Un elenco di ambiti delimitati da virgola autorizzati per la `sso-session`. Gli ambiti autorizzano l'accesso agli endpoint autorizzati con token di connessione del Centro identità IAM. Un ambito valido è una stringa, ad esempio `sso:account:access`. Questa impostazione non è applicabile alla configurazione precedente non aggiornabile.

```
sso_registration_scopes = sso:account:access
```

sso_role_name

Specifica il nome descrittivo del ruolo IAM che definisce le autorizzazioni dell'utente quando si utilizza questo profilo.

Questa impostazione non dispone di una variabile di ambiente o opzione della riga di comando.

```
sso_role_name = ReadAccess
```

sso_start_url

Speciifica l'URL che rimanda al portale di AWS accesso dell'organizzazione. AWS CLI Utilizza questo URL per stabilire una sessione con il servizio IAM Identity Center per autenticare i suoi utenti. Per trovare l'URL del portale di AWS accesso, utilizza uno dei seguenti:

- Apri l'e-mail di invito, l'URL del portale di AWS accesso è elencato.
- Apri la AWS IAM Identity Center console all'indirizzo https://console.aws.amazon.com/singlesignon/. L'URL del portale di AWS accesso è elencato nelle impostazioni.

Questa impostazione non dispone di una variabile di ambiente o opzione della riga di comando.

```
sso_start_url = https://my-sso-portal.awsapps.com/start
```

use_dualstack_endpoint

Abilita l'uso di endpoint dual-stack per inviare richieste. AWS Per ulteriori informazioni sugli endpoint dual-stack, che supportano sia il traffico che il IPv6 traffico, consulta Using IPv4 Using Amazon S3 dual-stack endpoint nella Amazon Simple Storage Service User Guide. Gli endpoint

dual-stack sono disponibili per determinati servizi in alcune Regioni. Se non esiste un endpoint dual-stack per il servizio o se la richiesta ha esito negativo. Regione AWS Le impostazioni valide sono `true` e `false`. Questa opzione è disabilitata per impostazione predefinita. Per ulteriori informazioni, consulta the section called "Impostato per utilizzare gli endpoint dual-stack per tutti i Servizi AWS".

Questa impostazione e l'impostazione `use_accelerate_endpoint` si escludono a vicenda.

> **Priorità degli endpoint**
>
> Le impostazioni di configurazione degli endpoint si trovano in più posizioni, ad esempio nelle variabili di sistema o di ambiente utente, nei file di AWS configurazione locali o sono dichiarate esplicitamente nella riga di comando come parametro. AWS CLI Controlla queste impostazioni dell'endpoint in un ordine particolare e utilizza l'impostazione dell'endpoint con la massima precedenza. Per l'elenco delle priorità degli endpoint, consulta the section called "Configurazione dell'endpoint e precedenza delle impostazioni".

`use_fips_endpoint`

Alcuni AWS servizi offrono endpoint che supportano il Federal Information Processing Standard (FIPS) 140-2, in altri. Regioni AWS Quando il AWS servizio supporta FIPS, questa impostazione specifica quale endpoint FIPS deve utilizzare. AWS CLI A differenza degli AWS endpoint standard, gli endpoint FIPS utilizzano una libreria software TLS conforme a FIPS 140-2. Questi endpoint potrebbero essere necessari ad aziende che interagiscono con il governo degli Stati Uniti. Per ulteriori informazioni, consultare the section called "Imposta per utilizzare gli endpoint FIP per tutti i Servizi AWS".

Se questa impostazione è abilitata, ma non esiste un endpoint FIPS per il servizio in uso, il comando potrebbe non riuscire. Regione AWS AWS In questo caso, specifica manualmente l'endpoint da utilizzare nel comando utilizzando l'opzione `--endpoint-url` oppure utilizza gli endpoint specifici del servizio.

> **Priorità degli endpoint**
>
> Le impostazioni di configurazione degli endpoint si trovano in più posizioni, ad esempio nelle variabili di sistema o di ambiente utente, nei file di AWS configurazione

locali o sono dichiarate esplicitamente nella riga di comando come parametro. AWS CLI Controlla queste impostazioni dell'endpoint in un ordine particolare e utilizza l'impostazione dell'endpoint con la massima precedenza. Per l'elenco delle priorità degli endpoint, consulta the section called "Configurazione dell'endpoint e precedenza delle impostazioni".

`web_identity_token_file`

Speciifica il percorso di un file che contiene un token di accesso OAuth 2.0 o un token ID OpenID Connect fornito da un provider di identità. L' AWS CLI carica il contenuto di questo file e lo passa come argomento `WebIdentityToken` all'operazione `AssumeRoleWithWebIdentity`.

La variabile di ambiente `AWS_WEB_IDENTITY_TOKEN_FILE` sovrascrive questa impostazione.

Per ulteriori informazioni sull'uso di identità web, consulta the section called "Assunzione di un ruolo con un'identità Web".

`tcp_keepalive`

Speciifica se il AWS CLI client utilizza pacchetti keep-alive TCP.

Questa voce non dispone di una variabile di ambiente o di un'opzione della riga di comando equivalente.

```
tcp_keepalive = false
```

Impostazioni dei comandi S3 personalizzati

Amazon S3 supporta diverse impostazioni che configurano il modo in cui AWS CLI esegue le operazioni di Amazon S3. Alcune si applicano a tutti i comandi S3 in entrambi gli spazi dei nomi `s3api` e `s3`. Altre sono specifiche per i comandi «personalizzati» di S3 che astraggono le operazioni comuni e fanno più di una one-to-one mappatura a un'operazione API. I comandi di trasferimento `aws s3` `cp`, `sync`, `mv` e `rm` dispongono di impostazioni aggiuntive che puoi utilizzare per controllare i trasferimenti S3.

Tutte queste opzioni possono essere configurate specificando l'impostazione nidificata `s3` nel file `config`. A ogni impostazione viene quindi applicato il rientro nella riga.

> **Note**
>
> Queste impostazioni sono completamente facoltative. Dovresti essere in grado di utilizzare i comandi di trasferimento `aws s3` senza configurare tali impostazioni, le quali vengono rese disponibili per consentirti di ottenere prestazioni ottimizzate o prendere in considerazione l'ambiente specifico in cui esegui questi comandi `aws s3`.

Queste impostazioni vengono tutte configurate in una chiave `s3` di livello superiore nel file `config`, come mostrato nell'esempio seguente per il profilo `development`:

```
[profile development]
s3 =
  max_concurrent_requests = 20
  max_queue_size = 10000
  multipart_threshold = 64MB
  multipart_chunksize = 16MB
  max_bandwidth = 50MB/s
  use_accelerate_endpoint = true
  addressing_style = path
```

Le impostazioni seguenti si applicano a qualsiasi comando S3 negli spazi dei nomi `s3` o `s3api`.

`addressing_style`

Specifica quale stile di indirizzamento utilizzare. Controlla se il nome del bucket è nell'hostname o in una parte dell'URL. I valori validi sono`path`, `virtual` e `auto`. Il valore predefinito è `auto`.

Sono disponibili due modi per creare un endpoint Amazon S3. Il primo è denominato `virtual` e include il nome del bucket come parte del nome host. Ad esempio: `https://`*`bucketname`*`.s3.amazonaws.com`. In alternativa, lo stile `path` consente di considerare il nome del bucket come un percorso nell'URI, ad esempio, `https://s3.amazonaws.com/`*`bucketname`*. Per impostazione predefinita, nell'interfaccia a riga di comando viene utilizzato `auto`, che tenta di utilizzare lo stile `virtual` se possibile, applicando di nuovo lo stile `path` quando necessario. Se non è compatibile con il DNS, il nome del bucket non può ad esempio essere parte del nome host e deve trovarsi nel percorso. Con `auto`, l'interfaccia a riga di comando rileverà la condizione e passerà automaticamente allo stile `path`. Se imposti lo stile di indirizzamento su`path`, devi quindi assicurarti che la AWS regione che hai configurato AWS CLI corrisponda alla regione del tuo bucket.

payload_signing_enabled

Specifica se SHA256 firmare i payload sigv4. Per impostazione predefinita, è disabilitata per i caricamenti in streaming (`UploadPart` e `PutObject`) quando si usa HTTPS. Per impostazione predefinita, è configurata su `false` per i caricamenti in streaming (`UploadPart` e `PutObject`), ma solo se è presente `ContentMD5` (generato per impostazione predefinita) e l'endpoint utilizza HTTPS.

Se impostato su true, le richieste S3 ricevono un'ulteriore convalida del contenuto sotto forma di un SHA256 checksum calcolato automaticamente e incluso nella firma della richiesta. Se impostata su false, il checksum non viene calcolato. Può essere utile disabilitare questa impostazione per ridurre il sovraccarico in termini di prestazioni dovuto al calcolo del checksum.

use_accelerate_endpoint

Utilizza l'endpoint Amazon S3 Accelerate per tutti i comandi `s3` e `s3api`. Il valore predefinito è false. Questa impostazione e l'impostazione `use_dualstack_endpoint` si escludono a vicenda.

Se impostato su true, AWS CLI indirizza tutte le richieste Amazon S3 `S3 Accelerate` all'endpoint a. `s3-accelerate.amazonaws.com` Per usare questo endpoint, dovrai abilitare l'utilizzo di `S3 Accelerate` per il bucket. Tutte le richieste vengono inviate utilizzando lo stile virtuale di indirizzamento al bucket: *`my-bucket`*`.s3-accelerate.amazonaws.com`. Le richieste `ListBuckets`, `CreateBucket` e `DeleteBucket` non verranno inviate all'endpoint di accelerazione S3 in quanto tale endpoint non supporta queste operazioni. Questo comportamento può essere configurato anche se il parametro `--endpoint-url` è impostato su `https://s3-accelerate.amazonaws.com` o `http://s3-accelerate.amazonaws.com` per qualsiasi comando `s3` o `s3api`.

use_dualstack_endpoint

Consente l'uso di endpoint dual-stack per inviare richieste `s3` e `s3api`. Per ulteriori informazioni sugli endpoint dual-stack, che supportano sia il traffico che il IPv6 traffico, consulta Using IPv4 Using Amazon S3 dual-stack endpoint nella Amazon Simple Storage Service User Guide. Gli endpoint dual-stack sono disponibili per determinati servizi in alcune Regioni. Se non esiste un endpoint dual-stack per il servizio o se la richiesta ha esito negativo. Regione AWS Le impostazioni valide sono `true` e `false`. Questa opzione è disabilitata per impostazione predefinita. Per ulteriori informazioni, consulta the section called "Impostato per utilizzare gli endpoint dual-stack per tutti i Servizi AWS".

Questa impostazione e l'impostazione `use_accelerate_endpoint` si escludono a vicenda.

Le impostazioni seguenti si applicano solo ai comandi nel set di comandi dello spazio dei nomi `s3`:

`max_bandwidth`

Specifica la larghezza di banda massima che può essere utilizzata per il caricamento e il download dei dati da e verso Amazon S3. Il valore predefinito è senza limiti.

Questa impostazione limita la larghezza di banda massima che i comandi S3 possono utilizzare per trasferire i dati da e verso Amazon S3. Tale valore viene applicato solo ai caricamenti e ai download, non alle operazioni di copia o eliminazione. Il valore viene espresso in byte al secondo. Il valore può essere specificato come segue:

- Come un intero, `1048576` imposta ad esempio l'utilizzo massimo della larghezza di banda su 1 MB al secondo.
- Come un intero seguito da un suffisso di velocità, che può essere specificato utilizzando `KB/s`, `MB/s` o `GB/s`. Ad esempio, `300KB/s`, `10MB/s`.

In generale, ti consigliamo di provare in primo luogo a ridurre il consumo di larghezza di banda riducendo `max_concurrent_requests`. Se ciò non è sufficiente per limitare il consumo di larghezza di banda in modo appropriato alle tue esigenze, puoi utilizzare l'impostazione `max_bandwidth` per limitarlo ulteriormente. Ciò accade perché `max_concurrent_requests` controlla il numero di thread attualmente in esecuzione. Se invece prima si abbassa `max_bandwidth` ma si lascia un'impostazione `max_concurrent_requests` alta, ciò può comportare che i thread debbano attendere inutilmente. Ciò può portare a un consumo eccessivo di risorse e a timeout di connessione.

`max_concurrent_requests`

Specifica il numero massimo di richieste simultanee. Il valore predefinito è 10.

I comandi di trasferimento `aws s3` sono a più thread. In qualsiasi momento, possono essere eseguite più richieste Amazon S3. Ad esempio, quando usi il comando `aws s3 cp localdir s3://bucket/ --recursive` per caricare file in un bucket S3, AWS CLI possono caricare i file `localdir/file1` e in `localdir/file3` parallelo. `localdir/file2` L'impostazione `max_concurrent_requests` specifica il numero massimo di operazioni di trasferimento che possono essere eseguite nello stesso momento.

Potrebbe essere necessario modificare questo valore per vari motivi:

- Riduzione del valore. In alcuni ambienti, il valore predefinito di 10 richieste simultanee può sovraccaricare il sistema. Questo può causare un timeout di connessione o una riduzione

della velocità di risposta del sistema. Se si riduce questo valore, i comandi di trasferimento S3 utilizzano un numero minore di risorse. Il completamento dei trasferimenti S3 può tuttavia richiedere più tempo. La riduzione di questo valore potrebbe essere necessaria se utilizzi uno strumento per limitare la larghezza di banda.

- Aumento del valore. In alcuni scenari potrebbe essere necessario che i trasferimenti Amazon S3 vengano completati il più rapidamente possibile, utilizzando la quantità di larghezza di banda della rete necessaria. In questo scenario, il numero predefinito di richieste simultanee potrebbe non essere sufficiente per utilizzare tutta la larghezza di banda di rete disponibile. L'aumento di questo valore può migliorare il tempo necessario per completare un trasferimento Amazon S3.

`max_queue_size`

Specifica il numero massimo di attività nella coda. Il valore predefinito è 1000.

AWS CLI Internamente utilizza un modello in cui mette in coda le attività di Amazon S3 che vengono poi eseguite da consumatori il cui numero è limitato a. `max_concurrent_requests` Un'attività viene in genere mappata a una singola operazione Amazon S3. Un'attività potrebbe ad esempio essere `PutObjectTask`, `GetObjectTask` o `UploadPartTask`. La velocità con cui le attività vengono aggiunte alla coda può essere molto maggiore rispetto a quella con cui i consumatori terminano le attività. Per evitare la crescita illimitata, esiste un limite specifico per la dimensione della coda delle attività. Questa impostazione consente di modificare il valore del numero massimo.

In generale, non è necessario modificare questa impostazione, Questa impostazione corrisponde anche al numero di attività di cui AWS CLI è a conoscenza e che devono essere eseguite. Ciò significa che, per impostazione predefinita, AWS CLI possono visualizzare solo 1000 attività future. L'aumento di questo valore significa che AWS CLI può conoscere più rapidamente il numero totale di attività necessarie, supponendo che la velocità di attesa sia superiore alla velocità di completamento delle attività. Il compromesso è che un valore max_queue_size maggiore richiede più memoria.

`multipart_chunksize`

Speciifica la dimensione del blocco AWS CLI utilizzata per i trasferimenti multiparte di singoli file. Il valore predefinito è 8 MB, con un minimo di 5 MB.

Quando un trasferimento file supera il valore `multipart_threshold`, la AWS CLI divide il file in blocchi di queste dimensioni. Questo valore può essere specificato utilizzando la stessa sintassi di `multipart_threshold`, come numero di byte sotto forma di intero o utilizzando una dimensione e un suffisso.

`multipart_threshold`

Speciifica la soglia di dimensione AWS CLI utilizzata per i trasferimenti multiparte di singoli file. Il valore predefinito è 8 MB.

Se durante il caricamento, il download o la copia, un file supera tali dimensioni, i comandi Amazon S3 avviano operazioni in più parti. Puoi specificare questo valore in uno dei due modi seguenti:

- Dimensione dei file in byte. Ad esempio, `1048576`.
- Dimensione dei file con un suffisso per la dimensione. Puoi utilizzare `KB`, `MB`, `GB` o `TB`. Ad esempio, `10MB`, `1GB`.

> **Note**
>
> S3 può imporre vincoli per i valori validi che possono essere utilizzati per operazioni in più parti. Per ulteriori informazioni, consulta S3 Multipart Upload documentation in Amazon Simple Storage Service User Guide.

Configurazione delle variabili d'ambiente per AWS CLI

Le variabili di ambiente offrono un altro modo per specificare le opzioni di configurazione e le credenziali e possono essere utili per la creazione di script.

Precedenza delle opzioni

- Se specifichi un'opzione utilizzando una delle variabili di ambiente descritte in questo argomento, questa sostituisce qualsiasi valore caricato da un profilo nel file di configurazione.
- Se si specifica un'opzione utilizzando un parametro sulla riga di AWS CLI comando, questo sovrascrive qualsiasi valore della variabile di ambiente corrispondente o di un profilo nel file di configurazione.

Per ulteriori informazioni sulla precedenza e su come AWS CLI determina le credenziali da utilizzare, vedere. Configurazione delle impostazioni della AWS CLI.

Argomenti

- Come impostare le variabili di ambiente
- AWS CLI variabili di ambiente supportate

Come impostare le variabili di ambiente

L'esempio seguente mostra come configurare le variabili di ambiente per l'utente predefinito.

Linux or macOS

```
$ export AWS_ACCESS_KEY_ID=AKIAIOSFODNN7EXAMPLE
$ export AWS_SECRET_ACCESS_KEY=wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY
$ export AWS_DEFAULT_REGION=us-west-2
```

L'impostazione della variabile di ambiente modifica il valore utilizzato fino al termine della sessione della shell o finché non imposti la variabile su un valore diverso. Puoi rendere le variabili persistenti per le sessioni future impostandole nello script di avvio della shell.

Windows Command Prompt

Impostazione per tutte le sessioni

```
C:\> setx AWS_ACCESS_KEY_ID AKIAIOSFODNN7EXAMPLE
C:\> setx AWS_SECRET_ACCESS_KEY wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY
C:\> setx AWS_DEFAULT_REGION us-west-2
```

Se si utilizza `setx` per impostare una variabile di ambiente, il valore utilizzato viene modificato nella sessione del prompt dei comandi corrente e in tutte le sessioni del prompt dei comandi create dopo l'esecuzione del comando. Ciò non ha alcun impatto su altre shell di comando già in esecuzione quando esegui il comando. Potrebbe essere necessario riavviare il terminale per caricare le impostazioni.

Impostazione solo per la sessione corrente

Se si utilizza `set` per impostare una variabile di ambiente, il valore utilizzato viene modificato fino al termine della sessione del prompt dei comandi corrente o finché non imposti la variabile su un valore diverso.

```
C:\> set AWS_ACCESS_KEY_ID=AKIAIOSFODNN7EXAMPLE
C:\> set AWS_SECRET_ACCESS_KEY=wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY
C:\> set AWS_DEFAULT_REGION=us-west-2
```

PowerShell

```
PS C:\> $Env:AWS_ACCESS_KEY_ID="AKIAIOSFODNN7EXAMPLE"
```

```
PS C:\> $Env:AWS_SECRET_ACCESS_KEY="wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY"
PS C:\> $Env:AWS_DEFAULT_REGION="us-west-2"
```

Se impostate una variabile di ambiente al PowerShell prompt, come mostrato negli esempi precedenti, il valore viene salvato solo per la durata della sessione corrente. Per rendere persistente l'impostazione della variabile di ambiente in tutte PowerShell le sessioni del prompt dei comandi, memorizzatela utilizzando l'applicazione System nel Pannello di controllo. In alternativa, puoi impostare la variabile per tutte le PowerShell sessioni future aggiungendola al tuo PowerShell profilo. Consulta la PowerShell documentazione per ulteriori informazioni sulla memorizzazione delle variabili di ambiente o sulla loro persistenza tra le sessioni.

AWS CLI variabili di ambiente supportate

AWS CLI supporta le seguenti variabili di ambiente.

`AWS_ACCESS_KEY_ID`

Specifica una chiave di AWS accesso associata a un account IAM.

Se definita, questa variabile di ambiente sostituisce il valore per l'impostazione del profilo `aws_access_key_id`. Non è possibile specificare l'ID chiave di accesso utilizzando un'opzione della riga di comando.

`AWS_ACCOUNT_ID`

Speciifica l'ID dell'endpoint AWS basato sull'account da utilizzare per le chiamate ai servizi di assistenza. Servizi AWS Per ulteriori informazioni sugli endpoint basati sull'account, consulta the section called "Endpoint basati sull'account".

Questa impostazione sostituisce l'impostazione `aws_account_id`. La variabile di ambiente `AWS_ACCOUNT_ID_ENDPOINT_MODE` o l'impostazione `account_id_endpoint_mode` deve essere impostata su `preferred` o `required` per utilizzare questa impostazione.

> **Priorità degli endpoint**
>
> Le impostazioni di configurazione degli endpoint si trovano in più posizioni, ad esempio le variabili di sistema o di ambiente utente, i file di AWS configurazione locali o sono dichiarate esplicitamente nella riga di comando come parametro. AWS CLI Controlla queste impostazioni dell'endpoint in un ordine particolare e utilizza l'impostazione dell'endpoint con la massima precedenza. Per l'elenco delle priorità degli

endpoint, consulta the section called "Configurazione dell'endpoint e precedenza delle impostazioni".

AWS_ACCOUNT_ID_ENDPOINT_MODE

Speciifica se utilizzare un endpoint AWS basato sull'account per le chiamate ai servizi supportati. IDs Servizi AWS Per ulteriori informazioni sugli endpoint basati sull'account, consulta the section called "Endpoint basati sull'account".

Questa impostazione può essere configurata come segue:

- (Impostazione predefinita) **`preferred`**: l'endpoint deve includere l'ID dell'account, se disponibile.
- **`disabled`**: un endpoint risolto non include l'ID dell'account.
- **`required`**: l'endpoint deve includere l'ID dell'account. Se l'ID dell'account non è disponibile, l'SDK genera un errore.

Questa impostazione sostituisce l'impostazione `account_id_endpoint_mode`. Per utilizzare gli endpoint basati sull'account, l'ID deve essere impostato nella variabile di ambiente `AWS_ACCOUNT_ID` o nell'impostazione `aws_account_id`.

> **Priorità degli endpoint**
>
> Le impostazioni di configurazione degli endpoint si trovano in più posizioni, ad esempio le variabili di sistema o di ambiente utente, i file di AWS configurazione locali o sono dichiarate esplicitamente nella riga di comando come parametro. AWS CLI Controlla queste impostazioni dell'endpoint in un ordine particolare e utilizza l'impostazione dell'endpoint con la massima precedenza. Per l'elenco delle priorità degli endpoint, consulta the section called "Configurazione dell'endpoint e precedenza delle impostazioni".

AWS_CA_BUNDLE

Specifica il percorso di un bundle di certificati da utilizzare per la convalida dei certificati HTTPS.

Se definita, questa variabile di ambiente sostituisce il valore per l'impostazione del profilo `ca_bundle`. Puoi sovrascrivere questa variabile di ambiente utilizzando il parametro della riga di comando `--ca-bundle`.

AWS_CLI_AUTO_PROMPT

Abilita il prompt automatico per la AWS CLI versione 2. È possibile utilizzare due impostazioni:

- **on** utilizza la modalità di prompt automatico completa ogni volta che si tenta di eseguire un comando `aws`. Include il dover premere INVIO dopo un comando completo o incompleto.
- **on-partial** utilizza la modalità di prompt automatico parziale. Se un comando è incompleto o non può essere eseguito a causa di errori di convalida sul lato client, viene utilizzato il prompt automatico. Questa modalità è utile se sono disponibili script e runbook preesistenti o per ricevere un prompt automatico solo per i comandi che non si conoscono piuttosto che per ogni comando.

Se definita, questa variabile di ambiente sostituisce il valore per l'impostazione del profilo `cli_auto_prompt`. È possibile sostituire questa variabile di ambiente utilizzando i parametri della riga di comando `--cli-auto-prompt` e `--no-cli-auto-prompt`.

Per informazioni sulla funzionalità di richiesta automatica della AWS CLI versione 2, vedere. Attivazione e utilizzo dei prompt dei comandi in AWS CLI

AWS_CLI_FILE_ENCODING

Specifica la codifica utilizzata per i file di testo. Per impostazione predefinita, la codifica corrisponde alle impostazioni locali. Per impostare una codifica diversa dalle impostazioni locali, utilizzare la variabile di ambiente `aws_cli_file_encoding`. Ad esempio, se si utilizza Windows con la codifica predefinita `CP1252`, l'impostazione `aws_cli_file_encoding=UTF-8` imposta la CLI per aprire i file di testo utilizzando `UTF-8`.

AWS_CLI_S3_MV_VALIDATE_SAME_S3_PATHS

Se i bucket di origine e di destinazione sono gli stessi quando si utilizza il comando `s3 mv`, il file o l'oggetto di origine può essere spostato su se stesso, il che può comportare l'eliminazione accidentale del file o dell'oggetto di origine. La variabile e l'`--validate-same-s3-paths`opzione di `AWS_CLI_S3_MV_VALIDATE_SAME_S3_PATHS` ambiente specificano se convalidare il punto di accesso ARNs o gli alias del punto di accesso nell'origine o nella destinazione di Amazon S3. URIs

> **Note**
>
> La convalida del percorso per `s3 mv` richiede chiamate API aggiuntive.

`AWS_CONFIG_FILE`

Speciifica la posizione del file utilizzato per archiviare i profili di configurazione. AWS CLI Il percorso predefinito è `~/.aws/config`.

Non è possibile specificare questo valore in un'impostazione profilo denominato o utilizzando un parametro della riga di comando.

`AWS_DATA_PATH`

Un elenco di directory aggiuntive da controllare al di fuori del percorso di ricerca integrato durante il caricamento AWS CLI dei `~/.aws/models` dati. L'impostazione di questa variabile di ambiente indica le directory aggiuntive da controllare prima di eseguire il fallback sul percorso di ricerca integrato. Voci diverse devono essere separate con il carattere `os.pathsep` che è `:` su Linux o macOS e `;` su Windows.

`AWS_DEFAULT_OUTPUT`

Specifica il formato di output da utilizzare.

Se definita, questa variabile di ambiente sostituisce il valore per l'impostazione del profilo `output`. Puoi sovrascrivere questa variabile di ambiente utilizzando il parametro della riga di comando `--output`.

`AWS_DEFAULT_REGION`

Per impostazione predefinita, `Default region name` identifica la AWS regione ai cui server desideri inviare le richieste. Questa è tipicamente la regione più vicina per l'utente, ma può essere qualsiasi regione. Ad esempio, è possibile digitare `us-west-2` per utilizzare Stati Uniti occidentali (Oregon). Questa è la regione a cui verranno inviate tutte le richieste successive, se non specificato altrimenti in un singolo comando.

> **Note**
>
> È necessario specificare una AWS regione quando si utilizza la AWS CLI, in modo esplicito o impostando una regione predefinita. Per un elenco delle regioni disponibili, consulta Regioni ed endpoint. I designatori di regione utilizzati da AWS CLI sono gli stessi nomi che vedi negli Console di gestione AWS URLs endpoint del servizio.

Se definita, questa variabile di ambiente sostituisce il valore per l'impostazione del profilo `region`. È possibile sovrascrivere questa variabile di ambiente utilizzando il parametro della `--region` riga di comando e la variabile di ambiente compatibile con AWS SDK. `AWS_REGION`

`AWS_EC2_METADATA_DISABLED`

Disattiva l'uso del servizio di metadati delle EC2 istanze Amazon (IMDS).

Se impostato su true, le credenziali o la configurazione dell'utente (come la Regione) non vengono richieste al servizio IMDS.

`AWS_ENDPOINT_URL`

Specifica l'endpoint utilizzato per tutte le richieste del servizio. Per ulteriori informazioni, consulta the section called "Imposta un endpoint globale per tutti i Servizi AWS".

> **Priorità degli endpoint**
>
> Le impostazioni di configurazione degli endpoint si trovano in più posizioni, ad esempio le variabili di sistema o di ambiente utente, i file di AWS configurazione locali o sono dichiarate esplicitamente nella riga di comando come parametro. AWS CLI Controlla queste impostazioni dell'endpoint in un ordine particolare e utilizza l'impostazione dell'endpoint con la massima precedenza. Per l'elenco delle priorità degli endpoint, consulta the section called "Configurazione dell'endpoint e precedenza delle impostazioni".

`AWS_ENDPOINT_URL_<SERVICE>`

Speciifica un endpoint personalizzato utilizzato per un servizio specifico, dove `<SERVICE>` viene sostituito con l'identificatore. Servizio AWS Ad esempio, Amazon DynamoDB ha un. `serviceId` `DynamoDB` Per questo servizio, la variabile di ambiente dell'URL dell'endpoint è `AWS_ENDPOINT_URL_DYNAMODB`.

Per l'elenco di tutte le variabili di ambiente specifiche del servizio, consulta Elenco di identificatori specifici del servizio.

> **Priorità degli endpoint**
>
> Le impostazioni di configurazione degli endpoint si trovano in più posizioni, ad esempio nelle variabili di sistema o di ambiente utente, nei file di AWS configurazione locali,

oppure sono dichiarate esplicitamente nella riga di comando come parametro. AWS CLI Controlla queste impostazioni dell'endpoint in un ordine particolare e utilizza l'impostazione dell'endpoint con la massima precedenza. Per l'elenco delle priorità degli endpoint, consulta the section called "Configurazione dell'endpoint e precedenza delle impostazioni".

`AWS_IGNORE_CONFIGURED_ENDPOINT_URLS`

Se abilitato, AWS CLI ignora tutte le configurazioni personalizzate degli endpoint. I valori validi sono **`true`** e **`false`**. Per ulteriori informazioni, consulta the section called "Imposta un endpoint globale per tutti i Servizi AWS".

Le impostazioni di configurazione degli endpoint si trovano in più posizioni, ad esempio nelle variabili di sistema o di ambiente utente, nei file di AWS configurazione locali, oppure sono dichiarate esplicitamente nella riga di comando come parametro. Per la priorità degli endpoint, consulta the section called "Configurazione dell'endpoint e precedenza delle impostazioni".

Priorità degli endpoint

Le impostazioni di configurazione degli endpoint si trovano in più posizioni, ad esempio le variabili di ambiente di sistema o utente, i file di AWS configurazione locali o sono dichiarate esplicitamente nella riga di comando come parametro. AWS CLI Controlla queste impostazioni dell'endpoint in un ordine particolare e utilizza l'impostazione dell'endpoint con la massima precedenza. Per l'elenco delle priorità degli endpoint, consulta the section called "Configurazione dell'endpoint e precedenza delle impostazioni".

`AWS_MAX_ATTEMPTS`

Specifica il numero massimo di tentativi utilizzato dal gestore dei tentativi, in AWS CLI base al quale la chiamata iniziale viene conteggiata ai fini del valore fornito. Per ulteriori informazioni sui tentativi, consulta Tentativi della AWS CLI nella AWS CLI.

Se definita, questa variabile di ambiente sostituisce il valore per l'impostazione del profilo `max_attempts`.

AWS_METADATA_SERVICE_NUM_ATTEMPTS

Quando tenta di recuperare le credenziali su un' EC2 istanza Amazon configurata con un ruolo IAM, AWS CLI tenta di recuperare le credenziali una volta dal servizio di metadati dell'istanza prima di interrompersi. Se sai che i tuoi comandi verranno eseguiti su un' EC2 istanza Amazon, puoi aumentare questo valore per AWS CLI riprovare più volte prima di rinunciare.

AWS_METADATA_SERVICE_TIMEOUT

Il numero di secondi prima che una connessione al servizio di metadati di istanza scada. Per impostazione predefinita, quando si tenta di recuperare le credenziali su un' EC2istanza Amazon configurata con un ruolo IAM, la connessione al servizio di metadati dell'istanza scade dopo 1 secondo. Se sai di essere in esecuzione su un' EC2 istanza Amazon con un ruolo IAM configurato, puoi aumentare questo valore se necessario.

AWS_PAGER

Specifica il programma pager utilizzato per l'output. Per impostazione predefinita, AWS CLI la versione 2 restituisce tutto l'output tramite il programma pager predefinito del sistema operativo.

Per disabilitare l'uso di un programma di paging esterno, imposta la variabile su una stringa vuota.

Se definita, questa variabile di ambiente sostituisce il valore per l'impostazione del profilo `cli_pager`.

AWS_PROFILE

Specificate il nome del AWS CLI profilo con le credenziali e le opzioni da utilizzare. Può essere il nome di un profilo archiviato in un file `credentials` o `config`, oppure il valore `default` per utilizzare il profilo di default.

Se definita, questa variabile di ambiente sostituisce il comportamento di utilizzo del profilo denominato `[default]` nel file di configurazione. Puoi sovrascrivere questa variabile di ambiente utilizzando il parametro della riga di comando `--profile`.

AWS_REGION

La variabile di ambiente compatibile con AWS SDK che specifica la AWS regione a cui inviare la richiesta.

Se definita, questa variabile di ambiente sostituisce il valore della variabile di ambiente `AWS_DEFAULT_REGION` e l'impostazione del profilo `region`. Puoi sovrascrivere questa variabile di ambiente utilizzando il parametro della riga di comando `--region`.

AWS_REQUEST_CHECKSUM_CALCULATION

Specifica quando viene calcolato un checksum per i payload di richiesta e dispone delle seguenti opzioni:

- `when_supported`: (impostazione predefinita) il checksum del payload di richiesta viene calcolato quando un'operazione specifica un algoritmo di checksum nel modello di servizio o richiede i checksum di richiesta.
- `when_required`: il checksum del payload di richiesta viene calcolato quando un'operazione richiede i checksum di richiesta o quando un utente fornisce un `requestAlgorithmMember` modellato dal Servizio AWS.

Se definita, questa variabile di ambiente sostituisce il valore dell'impostazione del profilo `request_checksum_calculation`.

AWS_RESPONSE_CHECKSUM_VALIDATION

Specifica quando viene eseguita la convalida del checksum per i payload di risposta e dispone delle seguenti opzioni:

- `when_supported`— (Impostazione predefinita) La convalida del checksum del payload di risposta viene eseguita quando un'operazione specifica un algoritmo di risposta nel modello di servizio supportato. AWS CLI
- `when_required`— La convalida del checksum del payload di risposta viene eseguita quando un'operazione specifica un algoritmo di risposta nel suo modello di servizio AWS CLI supportato e l'utente imposta il modello su nell'input dell'operazione. `requestValidationModeMember` `ENABLED`

Se definita, questa variabile di ambiente sostituisce il valore per l'impostazione del profilo `response_checksum_validation`.

AWS_RETRY_MODE

Speciifica la modalità di ripetizione utilizzata. AWS CLI Sono disponibili tre modalità di riprova: legacy (predefinita), standard e adattiva. Per ulteriori informazioni sui tentativi, consulta Tentativi della AWS CLI nella AWS CLI.

Se definita, questa variabile di ambiente sostituisce il valore per l'impostazione del profilo `retry_mode`.

`AWS_ROLE_ARN`

Specifica l'Amazon Resource Name (ARN) di un ruolo IAM con un provider di identità web che desideri utilizzare per eseguire AWS CLI i comandi.

Si utilizza con le variabili di ambiente `AWS_WEB_IDENTITY_TOKEN_FILE` e `AWS_ROLE_SESSION_NAME`.

Se definita, questa variabile di ambiente sostituisce il valore per l'impostazione del profilo `role_arn`. Non è possibile specificare un nome di sessione del ruolo come un parametro della riga di comando.

> **Note**
>
> Questa variabile di ambiente si applica solo a un ruolo assunto con un provider di identità Web e non alla configurazione generale del provider di ruoli assunti.

Per ulteriori informazioni sull'uso di identità Web, consulta the section called "Assunzione di un ruolo con un'identità Web".

`AWS_ROLE_SESSION_NAME`

Specifica il nome da associare alla sessione del ruolo. Questo valore viene fornito al `RoleSessionName` parametro quando AWS CLI chiama l'`AssumeRole`operazione e diventa parte del ruolo assunto dall'utente ARN:. `arn:aws:sts::123456789012:assumed-role/role_name/role_session_name` Si tratta di un parametro facoltativo. Se non fornisci questo valore, viene automaticamente generato un nome della sessione. Questo nome appare nei AWS CloudTrail registri delle voci associate a questa sessione.

Se definita, questa variabile di ambiente sostituisce il valore per l'impostazione del profilo `role_session_name`.

Si utilizza con le variabili di ambiente `AWS_ROLE_ARN` e `AWS_WEB_IDENTITY_TOKEN_FILE`.

Per ulteriori informazioni sull'uso di identità Web, consulta the section called "Assunzione di un ruolo con un'identità Web".

> **Note**
>
> Questa variabile di ambiente si applica solo a un ruolo assunto con un provider di identità Web e non alla configurazione generale del provider di ruoli assunti.

AWS_SDK_UA_APP_ID

Una singola Account AWS può essere utilizzata da più applicazioni del cliente a cui effettuare Servizi AWS chiamate. L'ID dell'applicazione identifica quale applicazione di origine ha effettuato una serie di chiamate utilizzando un Servizio AWS. AWS SDKs e i servizi non utilizzano o interpretano questo valore se non per riemergere nelle comunicazioni con i clienti. Ad esempio, questo valore può essere incluso nelle e-mail operative per identificare in modo univoco quale delle applicazioni è associata alla notifica.

Per impostazione predefinita, non è previsto un valore.

L'ID dell'applicazione è una stringa composta al massimo da 50 caratteri. Sono consentiti lettere, numeri, spazi e i seguenti caratteri speciali:

```
! $ % & * + - . , ^ _ ` | ~
```

Se definita, questa variabile di ambiente sostituisce il valore per l'impostazione del profilo `sdk_ua_app_id`. Non è possibile specificare l'ID dell'applicazione come un'opzione della riga di comando.

AWS_SECRET_ACCESS_KEY

Specifica la chiave segreta associata alla chiave di accesso. Si tratta essenzialmente della "password" per la chiave di accesso.

Se definita, questa variabile di ambiente sostituisce il valore per l'impostazione del profilo `aws_secret_access_key`. Non è possibile specificare l'ID della chiave di accesso segreta come un'opzione della riga di comando.

AWS_SESSION_TOKEN

Specifica il valore del token di sessione richiesto se si utilizzano credenziali di sicurezza temporanee recuperate direttamente dalle operazioni AWS STS . Per ulteriori informazioni, consulta Output del comando assume-role in AWS CLI Command Reference.

Se definita, questa variabile di ambiente sostituisce il valore per l'impostazione del profilo `aws_session_token`.

`AWS_SHARED_CREDENTIALS_FILE`

Speciifica la posizione del file AWS CLI utilizzato per archiviare le chiavi di accesso. Il percorso predefinito è `~/.aws/credentials`.

Non è possibile specificare questo valore in un'impostazione profilo denominato o utilizzando un parametro della riga di comando.

`AWS_SIGV4A_SIGNING_REGION_SET`

Specifica le Regioni da utilizzare quando si firma con SigV4a utilizzando un elenco delimitato da virgole. Se questa variabile non è impostata, AWS CLI utilizza l'impostazione predefinita utilizzata da. Servizio AWS Se non Servizio AWS ha un valore predefinito, la firma della richiesta diventa valida in tutte le regioni utilizzando un valore di`*`.

Per ulteriori informazioni su SigV4a, consulta AWS Signature Version 4 per le richieste API nella Guida per l'utente IAM

Se definita, questa variabile di ambiente sostituisce il valore per l'impostazione del profilo `sigv4a_signing_region_set`.

`AWS_USE_DUALSTACK_ENDPOINT`

Abilita l'uso di endpoint dual-stack per inviare richieste. AWS Per ulteriori informazioni sugli endpoint dual-stack, che supportano sia il traffico che il IPv6 traffico, consulta Using IPv4 Using Amazon S3 dual-stack endpoint nella Amazon Simple Storage Service User Guide. Gli endpoint dual-stack sono disponibili per determinati servizi in alcune Regioni. Se non esiste un endpoint dual-stack per il servizio o se la richiesta ha esito negativo. Regione AWS Questa opzione è disabilitata per impostazione predefinita. Per ulteriori informazioni, consulta the section called "Impostato per utilizzare gli endpoint dual-stack per tutti i Servizi AWS".

> **Priorità degli endpoint**
>
> Le impostazioni di configurazione degli endpoint si trovano in più posizioni, ad esempio nelle variabili di sistema o di ambiente utente, nei file di AWS configurazione locali o sono dichiarate esplicitamente nella riga di comando come parametro. AWS CLI Controlla queste impostazioni dell'endpoint in un ordine particolare e utilizza l'impostazione dell'endpoint con la massima precedenza. Per l'elenco delle priorità degli

endpoint, consulta the section called "Configurazione dell'endpoint e precedenza delle impostazioni".

AWS_USE_FIPS_ENDPOINT

Alcuni AWS servizi offrono endpoint che supportano il Federal Information Processing Standard (FIPS) 140-2, in altri. Regioni AWS Quando il servizio AWS supporta lo standard FIPS, questa impostazione specifica quale endpoint FIPS la AWS CLI deve utilizzare. A differenza degli AWS endpoint standard, gli endpoint FIPS utilizzano una libreria software TLS conforme a FIPS 140-2. Questi endpoint potrebbero essere necessari ad aziende che interagiscono con il governo degli Stati Uniti. Per ulteriori informazioni, consultare the section called "Imposta per utilizzare gli endpoint FIP per tutti i Servizi AWS".

Se questa impostazione è abilitata, ma non esiste un endpoint FIPS per il servizio in uso, il comando potrebbe non riuscire. Regione AWS AWS In questo caso, specifica manualmente l'endpoint da utilizzare nel comando utilizzando l'opzione `--endpoint-url` oppure utilizza gli endpoint specifici del servizio.

> **Priorità degli endpoint**
>
> Le impostazioni di configurazione degli endpoint si trovano in più posizioni, ad esempio nelle variabili di sistema o di ambiente utente, nei file di AWS configurazione locali o sono dichiarate esplicitamente nella riga di comando come parametro. AWS CLI Controlla queste impostazioni dell'endpoint in un ordine particolare e utilizza l'impostazione dell'endpoint con la massima precedenza. Per l'elenco delle priorità degli endpoint, consulta the section called "Configurazione dell'endpoint e precedenza delle impostazioni".

AWS_WEB_IDENTITY_TOKEN_FILE

Speciifica il percorso di un file che contiene un token di accesso OAuth 2.0 o un token ID OpenID Connect fornito da un provider di identità. L' AWS CLI carica il contenuto di questo file e lo passa come argomento `WebIdentityToken` all'operazione `AssumeRoleWithWebIdentity`.

Si utilizza con le variabili di ambiente `AWS_ROLE_ARN` e `AWS_ROLE_SESSION_NAME`.

Se definita, questa variabile di ambiente sostituisce il valore per l'impostazione del profilo `web_identity_token_file`.

Per ulteriori informazioni sull'uso di identità Web, consulta the section called "Assunzione di un ruolo con un'identità Web".

> **Note**
>
> Questa variabile di ambiente si applica solo a un ruolo assunto con un provider di identità Web e non alla configurazione generale del provider di ruoli assunti.

Opzioni della riga di comando in AWS CLI

In AWS CLI, le opzioni della riga di comando sono parametri globali che è possibile utilizzare per sovrascrivere le impostazioni di configurazione predefinite, qualsiasi impostazione del profilo corrispondente o l'impostazione della variabile di ambiente per quel singolo comando. Non puoi utilizzare le opzioni della riga di comando per specificare direttamente le credenziali, anche se puoi specificare il profilo da utilizzare.

Argomenti

- Come utilizzare le opzioni della riga di comando
- AWS CLI opzioni della riga di comando globali supportate
- Utilizzi comuni delle opzioni della riga di comando

Come utilizzare le opzioni della riga di comando

La maggior parte delle opzioni della riga di comando sono stringhe semplici, come il nome del profilo `profile1` nell'esempio seguente:

```
$ aws s3 ls --profile profile1
amzn-s3-demo-bucket1
amzn-s3-demo-bucket2
...
```

Ogni opzione che accetta un argomento richiede uno spazio o un simbolo di uguale ("=") che separa l'argomento dal nome dell'opzione. Se il valore dell'argomento è una stringa che contiene uno spazio,

occorre racchiudere l'argomento tra virgolette. Per informazioni dettagliate sui tipi di argomenti e sulla formattazione dei parametri, consulta Specificare dei valori di parametro nella AWS CLI.

AWS CLI opzioni della riga di comando globali supportate

In AWS CLI è possibile utilizzare le seguenti opzioni della riga di comando per sovrascrivere le impostazioni di configurazione predefinite, qualsiasi impostazione del profilo corrispondente o l'impostazione della variabile di ambiente per quel singolo comando.

--ca-bundle ***<string>***

Specificare il bundle di certificati CA (Certificate Authority, autorità di certificazione) da utilizzare per la verifica dei certificati SSL.

Se definita, questa opzione sostituisce il valore per l'impostazione del profilo `ca_bundle` e la variabile di ambiente `AWS_CA_BUNDLE`.

--cli-auto-prompt

Abilita la modalità di prompt automatico per un singolo comando. Come illustrato negli esempi seguenti, è possibile specificarlo in qualsiasi momento.

```
$ aws --cli-auto-prompt
$ aws dynamodb --cli-auto-prompt
$ aws dynamodb describe-table --cli-auto-prompt
```

Questa impostazione sostituisce la variabile di ambiente `aws_cli_auto_prompt` e la configurazione del profilo `cli_auto_prompt`.

Per informazioni sulla funzionalità di richiesta automatica della AWS CLI versione 2, vedere. Attivazione e utilizzo dei prompt dei comandi in AWS CLI

--cli-binary-format

Specifica in che modo la AWS CLI versione 2 interpreta i parametri di input binari. Può essere uno dei seguenti valori:

- Base64: si tratta del valore predefinito. Un parametro di input digitato come oggetto binario di grandi dimensioni (BLOB) accetta una stringa con codifica base64. Per passare il contenuto binario vero, inserire il contenuto in un file e fornire il percorso e il nome del file con il prefisso `fileb://` come valore del parametro. Per passare il testo codificato in base 64 contenuto

in un file, specificare il percorso e il nome del file con il prefisso `file://` come valore del parametro.

- raw-in-base64-out: impostazione predefinita per la AWS CLI versione 1. Se il valore dell'impostazione è`raw-in-base64-out`, i file a cui si fa riferimento utilizzando il `file://` prefisso vengono letti come testo e quindi i AWS CLI tentativi di codificarli in formato binario.

Questo sovrascrive l'impostazione di configurazione del file `cli_binary_format`.

```
$ aws lambda invoke \
    --cli-binary-format raw-in-base64-out \
    --function-name my-function \
    --invocation-type Event \
    --payload '{ "name": "Bob" }' \
    response.json
```

Se si fa riferimento a un valore binario in un file utilizzando la notazione del `fileb://` prefisso, si aspetta AWS CLI sempre che il file contenga contenuto binario non elaborato e non tenta di convertire il valore.

Se si fa riferimento a un valore binario in un file utilizzando la notazione del `file://` prefisso, AWS CLI gestisce il file in base all'impostazione corrente. `cli_binary_format` Se il valore di tale impostazione è `base64` (l'impostazione predefinita quando non è impostata in modo esplicito), si AWS CLI aspetta che il file contenga testo con codifica base64. Se il valore di tale impostazione è `raw-in-base64-out`, la AWS CLI prevede che nel file sia presente contenuto binario non elaborato.

--cli-connect-timeout ***<integer>***

Specifica il tempo di connessione socket massimo in secondi. Se il valore è impostato su zero (0), la connessione socket attende per un periodo di tempo indefinito (si blocca) e non si verifica alcun timeout. Il valore predefinito è 60 secondi.

--cli-read-timeout ***<integer>***

Specifica il tempo di lettura del socket massimo in secondi. Se il valore è impostato su zero (0), la lettura del socket attende per un periodo di tempo indefinito (si blocca) e non si verifica alcun timeout. Il valore predefinito è 60 secondi.

--colore ***<string>***

Specifica il supporto per l'output di colori. I valori validi sono on, `off` e `auto`. Il valore predefinito è `auto`.

--debug

Un interruttore booleano che abilita la registrazione di debug. Per AWS CLI impostazione predefinita, fornisce informazioni dettagliate su eventuali successi o fallimenti relativi ai risultati dei comandi nell'output del comando. L'opzione `--debug` fornisce i log completi di Python. Sono incluse informazioni di diagnostica `stderr` aggiuntive sull'operazione del comando che possono essere utili per capire i motivi per cui un comando restituisce risultati imprevisti. Per visualizzare facilmente i log di debug, suggeriamo di inviare i log in un file per cercare più facilmente le informazioni. Puoi farlo usando i seguenti metodi:

Per inviare solo le informazioni di diagnostica `stderr`, aggiungi `2> debug.txt` dove `debug.txt` è il nome che desideri utilizzare per il file di debug:

```
$ aws servicename commandname options --debug 2> debug.txt
```

Per inviare sia le informazioni di output che quelle di diagnostica `stderr`, aggiungi `&> debug.txt` dove `debug.txt` è il nome che desideri utilizzare per il file di debug:

```
$ aws servicename commandname options --debug &> debug.txt
```

--endpoint-url *<string>*

Specifica l'URL a cui inviare la richiesta. Per la maggior parte dei comandi, determina AWS CLI automaticamente l'URL in base al servizio selezionato e alla regione specificata. AWS Tuttavia, alcuni comandi ti richiedono di specificare un URL specifico dell'account. Puoi anche configurare alcuni AWS servizi per ospitare un endpoint direttamente all'interno del tuo VPC privato, che potrebbe quindi dover essere specificato.

Il comando di esempio seguente utilizza un URL personalizzato per l'endpoint Amazon S3.

```
$ aws s3 ls --endpoint-url http://localhost:4567
```

> **Priorità degli endpoint**
>
> Le impostazioni di configurazione degli endpoint si trovano in più posizioni, ad esempio le variabili di sistema o di ambiente utente, i file di AWS configurazione locali o sono dichiarate esplicitamente nella riga di comando come parametro. AWS CLI Controlla queste impostazioni dell'endpoint in un ordine particolare e utilizza l'impostazione dell'endpoint con la massima precedenza. Per l'elenco delle priorità degli

endpoint, consulta the section called "Configurazione dell'endpoint e precedenza delle impostazioni".

--no-cli-auto-prompt

Disabilita la modalità di prompt automatico per un singolo comando.

```
$ aws dynamodb describe-table --table-name Table1 --no-cli-auto-prompt
```

Questa impostazione sostituisce la variabile di ambiente `aws_cli_auto_prompt` e la configurazione del profilo `cli_auto_prompt`.

Per informazioni sulla funzionalità di richiesta automatica della AWS CLI versione 2, vedere. Attivazione e utilizzo dei prompt dei comandi in AWS CLI

--no-cli-pager

Un interruttore booleano che disabilita l'uso di un pager per l'output del comando.

--no-paginate

Uno switch booleano che disattiva le chiamate multiple AWS CLI esegue automaticamente la ricezione di tutti i risultati dei comandi che creano l'impaginazione dell'output. Ciò significa che viene visualizzata solo la prima pagina dell'output.

--no-sign-request

Un interruttore booleano che disabilita la firma delle richieste HTTP all'endpoint del servizio. AWS In questo modo si evita il caricamento delle credenziali.

--no-verify-ssl

Per impostazione predefinita, AWS CLI utilizza SSL per comunicare con i servizi. AWS Per ogni connessione e chiamata SSL, AWS CLI verifica i certificati SSL. L'utilizzo di questa opzione ha la precedenza sul comportamento predefinito di verifica dei certificati SSL.

> **Warning**
>
> Questa non è una best practice. Se lo utilizzi`--no-verify-ssl`, il traffico tra il cliente e i AWS servizi non è più sicuro. Ciò significa che il traffico rappresenta un rischio per la sicurezza ed è vulnerabile agli man-in-the-middle exploit. Se riscontri problemi con i

certificati, è preferibile risolverli. Per la procedura di risoluzione dei problemi relativi ai certificati, consulta the section called "Errori di certificato SSL".

--uscita ***<string>***

Specifica il formato di output da utilizzare per questo comando. Puoi specificare uno qualsiasi dei valori seguenti:

- **`json`**: l'output è formattato come una stringa JSON.
- **`yaml`**: l'output è formattato come una stringa YAML.
- **`yaml-stream`**: l'output viene inviato in streaming e formattato come una stringa YAML. Lo streaming consente una gestione più rapida di tipi di dati di grandi dimensioni.
- **`text`**: l'output è formattato come più righe di valori di stringa separati da tabulazioni. Questa formattazione può essere utile per passare l'output a un elaboratore di testi, ad esempio `grep`, `sed` o `awk`.
- **`table`**: l'output è formattato come una tabella in cui si utilizzano i caratteri +|- per formare i bordi delle celle. In genere presenta le informazioni in un formato comprensibile molto più semplice da leggere rispetto ad altri, ma non altrettanto utile a livello programmatico.

--profilo ***<string>***

Specifica il profilo denominato da utilizzare per questo comando. Per impostare profili denominati aggiuntivi, puoi utilizzare il comando `aws configure` con l'opzione `--profile`.

```
$ aws configure --profile <profilename>
```

--interrogazione ***<string>***

Specifica una JMESPath query da utilizzare per filtrare i dati di risposta. Per ulteriori informazioni, consulta Uscita di filtraggio nel AWS CLI.

--region ***<string>***

Speciifica a quale AWS regione inviare la AWS richiesta di questo comando. Per un elenco di tutte le regioni che puoi specificare, consulta Regioni ed endpoint AWS nella Riferimenti generali di Amazon Web Services.

--version

Un interruttore booleano che visualizza la versione corrente del programma in esecuzione. AWS CLI

Utilizzi comuni delle opzioni della riga di comando

Usi comuni per le opzioni della riga di comando includono il controllo delle risorse in più regioni AWS e la modifica del formato di output per maggiore leggibilità o facilità d'uso al momento della creazione di script. Negli esempi seguenti, eseguiamo il comando describe-instances su ciascuna Regione finché non individuiamo in quale Regione si trova la nostra istanza.

```
$ aws ec2 describe-instances --output table --region us-west-1
------------------
|DescribeInstances|
+-----------------+
$ aws ec2 describe-instances --output table --region us-west-2
-------------------------------------------------------------------------------
|                            DescribeInstances                                |
+-----------------------------------------------------------------------------+
||                              Reservations                                 ||
|+-----------------------------------+-----------------------------------+|
||  OwnerId                          |  012345678901                     ||
||  ReservationId                    |  r-abcdefgh                       ||
|+-----------------------------------+-----------------------------------+|
|||                               Instances                               |||
||+-------------------------+-----------------------------------------------+||
|||  AmiLaunchIndex         |  0                                            |||
|||  Architecture           |  x86_64                                       |||
...
```

Configurazione del completamento dei comandi nella AWS CLI.

La AWS CLI (AWS Command Line Interface) include una funzionalità di completamento dei comandi compatibile con Bash che consente di completare un comando digitato parzialmente utilizzando il tasto Tab. Sulla maggior parte dei sistemi è necessario configurarlo manualmente.

Per informazioni sulla funzionalità di prompt automatico della AWS CLI versione 2, consulta Attivazione e utilizzo dei prompt dei comandi in AWS CLI.

Argomenti

- Come funziona
- Configurazione del completamento dei comandi su Linux o macOS
- Configurazione del completamento dei comandi su Windows

Come funziona

Quando si immette parzialmente un comando, un parametro o un'opzione, la caratteristica di completamento dei comandi completa automaticamente il comando o visualizza un elenco suggerito di comandi. Per richiedere il completamento del comando, inserire parzialmente un comando e premere il tasto di completamento, tipicamente *Tab* nella maggior parte delle shell.

Negli esempi seguenti vengono illustrati diversi modi in cui è possibile utilizzare il completamento dei comandi:

- Immettere parzialmente un comando e premere *Tab* per visualizzare un elenco di comandi suggeriti.

```
$ aws dynamodb dTAB
delete-backup                       describe-global-table
delete-item                         describe-global-table-settings
delete-table                        describe-limits
describe-backup                     describe-table
describe-continuous-backups         describe-table-replica-auto-scaling
describe-contributor-insights       describe-time-to-live
describe-endpoints
```

- Immettere parzialmente un parametro e premere *Tab* per visualizzare un elenco di parametri suggeriti.

```
$ aws dynamodb delete-table --TAB
--ca-bundle             --endpoint-url          --profile
--cli-connect-timeout   --generate-cli-skeleton --query
--cli-input-json        --no-paginate           --region
--cli-read-timeout      --no-sign-request       --table-name
--color                 --no-verify-ssl         --version
--debug                 --output
```

- Immettere un parametro e premere *Tab* per visualizzare un elenco di valori delle risorse suggeriti. Questa funzionalità è disponibile solo nella versione 2 della AWS CLI.

```
$ aws dynamodb delete-table --table-name TAB
Table 1                 Table 2                 Table 3
```

Configurazione del completamento dei comandi su Linux o macOS

Per configurare il completamento dei comandi, su Linux o macOS occorrono due informazioni: il nome della shell che si sta utilizzando e la posizione dello script `aws_completer`.

> **Note**
>
> Il completamento dei comandi viene configurato e abilitato automaticamente per impostazione predefinita sulle istanze Amazon EC2 che eseguono Amazon Linux.

Argomenti

- Conferma che la cartella del programma di completamento si trova nel tuo percorso
- Abilitazione del completamento dei comandi
- Verifica del completamento dei comandi

Conferma che la cartella del programma di completamento si trova nel tuo percorso

AWSAffinché il programma di completamento funzioni correttamente, `aws_completer` deve trovarsi nel percorso della shell. Il comando `which` può verificare se il programma di completamento si trova nel percorso dell'utente.

```
$ which aws_completer
/usr/local/bin/aws_completer
```

Se il comando which non riesce a trovare il programma di completamento, utilizza le seguenti fasi per aggiungere relativa la cartella al tuo percorso.

Fase 1: individuare il programma di completamento AWS

La posizione del programma di completamento AWS può variare in base al metodo di installazione utilizzato.

- Programma di gestione dei pacchetti - Programmi come `pip`, `yum`, `brew` e `apt-get` generalmente installano il programma di completamento AWS (o un link simbolico a esso) in un percorso standard.
  - Se `pip` è stato utilizzato senza il parametro `--user`, il percorso predefinito è `/usr/local/bin/aws_completer`.

  - Se `pip` è stato utilizzato con il parametro `--user`, il percorso predefinito è `/home/username/.local/bin/aws_completer`.
- Programma di installazione in bundle - Se hai utilizzato il programma di installazione in bundle, il percorso predefinito è `/usr/local/bin/aws_completer`.

Se non riesci a individuarlo, puoi utilizzare il comando `find` per cercare il file system per il programma di completamento AWS.

```
$ find / -name aws_completer
/usr/local/bin/aws_completer
```

Fase 2: identificazione della shell

Per identificare la shell in uso, è possibile utilizzare uno dei seguenti comandi.

- echo $SHELL - Visualizza il nome del file di programma della shell. Generalmente corrisponde al nome della shell in uso, a meno che non venga avviata un'altra shell dopo avere effettuato l'accesso.

  ```
  $ echo $SHELL
  /bin/bash
  ```
- ps - Mostra i processi in esecuzione per l'utente corrente. Uno di questi è la shell.

  ```
  $ ps
    PID TTY          TIME CMD
   2148 pts/1    00:00:00 bash
   8756 pts/1    00:00:00 ps
  ```

Fase 3: aggiunta della cartella del programma di completamento al percorso

1. Trova lo script del profilo della tua shell nella tua cartella utente.

   ```
   $ ls -a ~/
   .  ..  .bash_logout  .bash_profile  .bashrc  Desktop  Documents  Downloads
   ```

   - Bash - `.bash_profile`, `.profile` o `.bash_login`
   - Zsh - `.zshrc`

- Tcsh - `.tcshrc`, `.cshrc`, o `.login`

2. Aggiungi un comando di esportazione al termine dello script del profilo simile a quello dell'esempio seguente. Sostituisci */usr/local/bin/* con la cartella individuata nella sezione precedente.

```
export PATH=/usr/local/bin/:$PATH
```

3. Ricarica il profilo nella sessione corrente per rendere effettive tali modifiche. Sostituisci *.bash_profile* con il nome dello script della shell individuato nella prima sezione.

```
$ source ~/.bash_profile
```

Abilitazione del completamento dei comandi

Dopo aver verificato che il programma di completamento si trovi nel tuo percorso, abilita il completamento del comando eseguendo il comando appropriato per la shell che stai utilizzando. Puoi aggiungere il comando al profilo della shell per eseguirlo ogni volta che apri una nuova shell. In ogni comando, sostituisci il percorso */usr/local/bin/* con quello trovato sul tuo sistema in Conferma che la cartella del programma di completamento si trova nel tuo percorso.

- **`bash`** - Utilizza il comando integrato `complete`.

```
$ complete -C '/usr/local/bin/aws_completer' aws
```

  Aggiungi il comando precedente a `~/.bashrc` per eseguirlo ogni volta che apri una nuova shell. Per accertarti che il comando venga eseguito anche nelle shell di accesso, l'origine di `~/.bash_profile` deve essere `~/.bashrc`.

- **`zsh`** - Per eseguire il completamento del comando, è necessario eseguire `bashcompinit` aggiungendo la seguente riga di caricamento automatico alla fine dello script del profilo `~/.zshrc`.

```
$ autoload bashcompinit && bashcompinit
$ autoload -Uz compinit && compinit
```

  Per abilitare il completamento dei comandi, utilizzare il comando integrato `complete`.

```
$ complete -C '/usr/local/bin/aws_completer' aws
```

  Aggiungi i comandi precedenti a `~/.zshrc` per eseguirlo ogni volta che apri una nuova shell.

- **`tcsh`** - Il completamento di `tcsh` richiede un tipo di parola e un modello per definire il comportamento di completamento.

```
> complete aws 'p/*/`aws_completer`/'
```

  Aggiungi il comando precedente a `~/.tcshrc` per eseguirlo ogni volta che apri una nuova shell.

Dopo aver abilitato il completamento dei comandi, Verifica del completamento dei comandi è in funzione.

Verifica del completamento dei comandi

Dopo avere abilitato il completamento dei comandi, ricarica la shell, immetti un comando parziale e premi il tasto Tab per visualizzare i comandi disponibili.

```
$ aws sTAB
s3              ses             sqs             sts             swf
s3api           sns             storagegateway  support
```

Configurazione del completamento dei comandi su Windows

> **Note**
>
> Per informazioni su come PowerShell gestisce il completamento, incluse le varie chiavi di completamento, consulta about_Tab_Expansion in Microsoft PowerShell Docs.

Per abilitare il completamento dei comandi per PowerShell su Windows, completa le fasi seguenti in PowerShell.

1. Apri il file `$PROFILE` tramite il comando seguente:

```
PS C:\> Notepad $PROFILE
```

   Se non disponi di un account `$PROFILE`, crea un profilo utente utilizzando il comando seguente.

```
PS C:\> if (!(Test-Path -Path $PROFILE ))
{ New-Item -Type File -Path $PROFILE -Force }
```

Per ulteriori informazioni sui profili PowerShell, consulta How to Use Profiles in Windows PowerShell ISE sul sito web Microsoft Docs.

2. Per abilitare il completamento dei comandi, aggiungi il seguente blocco di codice al tuo profilo, salva e chiudi il file.

```
Register-ArgumentCompleter -Native -CommandName aws -ScriptBlock {
    param($commandName, $wordToComplete, $cursorPosition)
        $env:COMP_LINE=$wordToComplete
        if ($env:COMP_LINE.Length -lt $cursorPosition){
            $env:COMP_LINE=$env:COMP_LINE + " "
        }
        $env:COMP_POINT=$cursorPosition
        aws_completer.exe | ForEach-Object {
            [System.Management.Automation.CompletionResult]::new($_, $_,
 'ParameterValue', $_)
        }
        Remove-Item Env:\COMP_LINE
        Remove-Item Env:\COMP_POINT
}
```

3. Dopo avere abilitato il completamento dei comandi, ricarica la shell, immetti un comando parziale e premi il tasto Tab per passare tra i comandi disponibili.

```
$ aws sTab
```

```
$ aws s3
```

Per visualizzare tutti i comandi disponibili per il completamento, immetti un comando parziale e premi Ctrl + Spazio.

```
$ aws sCtrl + Space
s3              ses             sqs             sts             swf
s3api           sns             storagegateway  support
```

Tentativi della AWS CLI nella AWS CLI

Questo argomento descrive come la AWS CLI potrebbe vedere le chiamate a servizi AWS non riusciti a causa di problemi imprevisti. Questi problemi possono verificarsi sul lato server o potrebbero

non riuscire a causa della limitazione della velocità imposta dal servizio AWS che stai tentando di chiamare. Questi tipi di errori di solito non richiedono una gestione speciale e la chiamata viene rieffettuata automaticamente, spesso dopo un breve periodo di attesa. La AWS CLI fornisce molte funzionalità che aiutano a ritentare le chiamate dei client ai servizi AWS quando si verificano questi tipi di errori o eccezioni.

Argomenti

- Modalità di ripetizione disponibili
- Configurazione di una modalità di riprova
- Visualizzazione dei log dei tentativi di riprova

Modalità di ripetizione disponibili

La AWS CLI presenta diverse modalità tra cui scegliere a seconda della versione:

- Modalità di riprova standard (impostazione predefinita)
-
- Modalità di riprova adattiva

Modalità di riprova standard (impostazione predefinita)

La modalità `Standard` è un set standard di regole di riprova tra gli AWS SDK con più funzionalità rispetto alla versione precedente ed è la modalità predefinita della AWS CLI. La modalità standard è stata creata per la AWS CLI versione 2 ed è stata sottoposta al backport nella AWS CLI versione 1. Le funzionalità della modalità standard includono:

- Il valore predefinito è 2 per il numero massimo di tentativi di riprova, per un totale di 3 tentativi di chiamata. Questo valore può essere sovrascritto tramite il parametro di configurazione `max_attempts`.
- Riprova i tentativi per il seguente elenco esteso di errori/eccezioni:
  - Errori ed eccezioni transitori
    - `RequestTimeout`
    - `RequestTimeoutException`
    - `PriorRequestNotComplete`
    - `ConnectionError`

    - `HTTPClientError`
  - Errori e eccezioni di limitazione/limitazione (della larghezza di banda della rete) lato servizio:
    - `Throttling`
    - `ThrottlingException`
    - `ThrottledException`
    - `RequestThrottledException`
    - `TooManyRequestsException`
    - `ProvisionedThroughputExceededException`
    - `TransactionInProgressException`
    - `RequestLimitExceeded`
    - `BandwidthLimitExceeded`
    - `LimitExceededException`
    - `RequestThrottled`
    - `SlowDown`
    - `EC2ThrottledException`

- Riprova con codici di errore transitori e non descrittivi. In particolare, questi codici di stato HTTP: 500, 502, 503, 504.
- Qualsiasi nuovo tentativo includerà un backoff esponenziale di un fattore base pari a 2 per un tempo di backoff massimo di 20 secondi.

La modalità `Legacy` utilizza un gestore di tentativi precedente con funzionalità limitate che includono:

- Il valore predefinito è 4 per il numero massimo di tentativi di riprova, per un totale di 5 tentativi di chiamata. Questo valore può essere sovrascritto tramite il parametro di configurazione `max_attempts`.
- DynamoDB ha un valore predefinito di 9 per il numero massimo di tentativi di riprova, per un totale di 10 tentativi di chiamata. Questo valore può essere sovrascritto tramite il parametro di configurazione `max_attempts`.
- Riprova i tentativi per il seguente numero limitato di errori/eccezioni:
  - Errori generali di socket/connessione:

    - `ConnectionError`
    - `ConnectionClosedError`
    - `ReadTimeoutError`
    - `EndpointConnectionError`
  - Errori e eccezioni di limitazione/limitazione (della larghezza di banda della rete) lato servizio:
    - `Throttling`
    - `ThrottlingException`
    - `ThrottledException`
    - `RequestThrottledException`
    - `ProvisionedThroughputExceededException`
- Tentativi su diversi codici di stato HTTP, tra cui 429, 500, 502, 503, 504 e 509.
- Qualsiasi nuovo tentativo includerà un backoff esponenziale di un fattore base pari a 2.

Modalità di riprova adattiva

> **Warning**
>
> La modalità adattiva è una modalità sperimentale ed è soggetta a modifiche, sia nelle funzionalità che nel comportamento.

La modalità di riprova `Adaptive` comporta la ripetizione sperimentale che include tutte le funzionalità della modalità standard. Oltre alle funzionalità della modalità standard, la modalità adattiva introduce anche la limitazione della velocità lato client mediante l'uso di un token bucket e di variabili con limite di velocità che vengono aggiornate dinamicamente a ogni nuovo tentativo. Questa modalità offre flessibilità nei tentativi lato client e si adatta alla risposta allo stato di errore/eccezione di un servizio AWS.

Ad ogni nuovo tentativo, la modalità adattiva modifica le variabili del limite di velocità in base all'errore, all'eccezione o al codice di stato HTTP presentato nella risposta del servizio AWS. Queste variabili relative al limite di velocità vengono quindi utilizzate per calcolare una nuova velocità di chiamata per il client. Ogni eccezione/errore o risposta HTTP non riuscita (fornita nell'elenco precedente) di un servizio AWS aggiorna le variabili del limite di frequenza man mano che si verificano nuovi tentativi fino al raggiungimento del successo, all'esaurimento del bucket token o al raggiungimento del valore massimo di tentativi configurato.

Configurazione di una modalità di riprova

La AWS CLI include una varietà di configurazioni di nuovi tentativi e metodi di configurazione da considerare durante la creazione dell'oggetto client.

Metodi di connessione disponibili

Nella AWS CLI, gli utenti possono configurare nuovi tentativi nei modi seguenti:

- Variabili di ambiente
- File di configurazione della AWS CLI

Gli utenti possono personalizzare le seguenti opzioni per la riprova:

- Modalità Riprova: specifica la modalità di riprova utilizzata dalla AWS CLI. Come descritto in precedenza, sono disponibili tre modalità di riprova: legacy, standard e adattiva. Il valore predefinito per la . La AWS CLI versione 2 è standard.
- Numero massimo di tentativi: specifica il valore massimo di tentativi utilizzato dal gestore dei tentativi della AWS CLI, in cui la chiamata iniziale viene conteggiata ai fini del valore fornito. Il valore predefinito si basa sulla modalità di riprova.

Definizione di una configurazione di una riprova nelle variabili di ambiente

Per definire la configurazione dei nuovi tentativi per la AWS CLI, aggiorna le variabili di ambiente del sistema operativo.

Le variabili di ambiente della riprova sono:

- `AWS_RETRY_MODE`
- `AWS_MAX_ATTEMPTS`

Per ulteriori informazioni sulle variabili di ambiente, consulta Configurazione delle variabili d'ambiente per AWS CLI.

Definizione di una nuova configurazione nel file di configurazione AWS

Per modificare la configurazione del nuovo tentativo, aggiorna il file di configurazione globale AWS. La posizione predefinita per il file di configurazione AWS è ~/.aws/config.

Di seguito è riportato un esempio di un file AWS:

```
[default]
retry_mode = standard
max_attempts = 6
```

Per ulteriori informazioni sui file di configurazione, consulta Impostazioni dei file di configurazione e delle credenziali nella AWS CLI.

Visualizzazione dei log dei tentativi di riprova

La AWS CLI utilizza la metodologia di riprova e la registrazione di log di Boto3. È possibile utilizzare l'opzione `--debug` su qualsiasi comando per ricevere i log di debug. Per informazioni su come utilizzare l'opzione `--debug`, consulta Opzioni della riga di comando in AWS CLI.

Se cerchi "retry" nei log di debug, troverai le informazioni sui nuovi tentativi di cui hai bisogno. Le voci di log del client relative ai tentativi dipendono dalla modalità di riprova abilitata.

Modalità legacy:

I messaggi di riprova vengono generati da botocore.retryhandler. Vedrai uno dei tre messaggi seguenti:

- `No retry needed`
- `Retry needed, action of: <action_name>`
- `Reached the maximum number of retry attempts: <attempt_number>`

Modalità standard o adattiva:

I messaggi di riprova vengono generati da botocore.retries.standard. Vedrai uno dei tre messaggi seguenti:

- `No retrying request`
- `Retry needed, retrying request after delay of: <delay_value>`
- `Retry needed but retry quota reached, not retrying request`

Per il file di definizione completo dei tentativi di botocore, consulta _retry.json nel repository GitHub di botocore.

Utilizzo di un proxy HTTP per la AWS CLI

Per accedere ad AWS tramite i server proxy, puoi configurare le variabili di ambiente `HTTP_PROXY` e `HTTPS_PROXY` con i nomi di dominio DNS o gli indirizzi IP e i numeri di porta utilizzati dai server proxy.

Argomenti

- Utilizzo degli esempi
- Autenticazione in un proxy
- Utilizzo di un proxy sulle istanze Amazon EC2
- Risoluzione dei problemi

Utilizzo degli esempi

> **Note**
>
> Gli esempi seguenti mostrano il nome della variabile di ambiente in lettere maiuscole. Tuttavia, se si specifica una variabile due volte utilizzando lettere maiuscole e minuscole, le lettere minuscole hanno la precedenza. Ti consigliamo di definire ogni variabile solo una volta per evitare confusione nel sistema e un comportamento imprevisto.

Nell'esempio seguente viene illustrato come utilizzare l'indirizzo IP esplicito del proxy o un nome DNS che viene risolto nell'indirizzo IP del proxy. Entrambi possono essere seguiti da due punti e dal numero di porta a cui devono essere inviate le query.

Linux or macOS

```
$ export HTTP_PROXY=http://10.15.20.25:1234
$ export HTTP_PROXY=http://proxy.example.com:1234
$ export HTTPS_PROXY=http://10.15.20.25:5678
$ export HTTPS_PROXY=http://proxy.example.com:5678
```

Windows Command Prompt

Impostazione per tutte le sessioni

```
C:\> setx HTTP_PROXY http://10.15.20.25:1234
```

```
C:\> setx HTTP_PROXY http://proxy.example.com:1234
C:\> setx HTTPS_PROXY http://10.15.20.25:5678
C:\> setx HTTPS_PROXY http://proxy.example.com:5678
```

Se si utilizza `setx` per impostare una variabile di ambiente, il valore utilizzato viene modificato nella sessione del prompt dei comandi corrente e in tutte le sessioni del prompt dei comandi create dopo l'esecuzione del comando. Ciò non ha alcun impatto su altre shell di comando già in esecuzione quando esegui il comando.

Impostazione solo per la sessione corrente

Se si utilizza `set` per impostare una variabile di ambiente, il valore utilizzato viene modificato fino al termine della sessione del prompt dei comandi corrente o finché non imposti la variabile su un valore diverso.

```
C:\> set HTTP_PROXY=http://10.15.20.25:1234
C:\> set HTTP_PROXY=http://proxy.example.com:1234
C:\> set HTTPS_PROXY=http://10.15.20.25:5678
C:\> set HTTPS_PROXY=http://proxy.example.com:5678
```

Autenticazione in un proxy

> **Note**
>
> AWS CLI non supporta i proxy NTLM. Se utilizzi un proxy protocollo NTLM o Kerberos, puoi connetterti tramite un proxy di autenticazione quale Cntlm.

L'AWS CLI supporta l'autenticazione di base HTTP. Specifica il nome utente e la password nell'URL del proxy come riportato di seguito.

Linux or macOS

```
$ export HTTP_PROXY=http://username:password@proxy.example.com:1234
$ export HTTPS_PROXY=http://username:password@proxy.example.com:5678
```

Windows Command Prompt

Impostazione per tutte le sessioni

```
C:\> setx HTTP_PROXY http://username:password@proxy.example.com:1234
C:\> setx HTTPS_PROXY http://username:password@proxy.example.com:5678
```

Impostazione solo per la sessione corrente

```
C:\> set HTTP_PROXY=http://username:password@proxy.example.com:1234
C:\> set HTTPS_PROXY=http://username:password@proxy.example.com:5678
```

Utilizzo di un proxy sulle istanze Amazon EC2

Se configuri un proxy su un'istanza Amazon EC2 avviata con un ruolo IAM collegato, assicurati di esentare l'indirizzo utilizzato per accedere ai metadati dell'istanza. A questo scopo, imposta la variabile di ambiente NO_PROXY sull'indirizzo IP del servizio di metadati di istanza, 169.254.169.254. Questo indirizzo non varia.

Linux or macOS

```
$ export NO_PROXY=169.254.169.254
```

Windows Command Prompt

Impostazione per tutte le sessioni

```
C:\> setx NO_PROXY 169.254.169.254
```

Impostazione solo per la sessione corrente

```
C:\> set NO_PROXY=169.254.169.254
```

Risoluzione dei problemi

Se riscontri problemi con la AWS CLI, consulta la procedura the section called "Risoluzione degli errori" per la risoluzione dei problemi. Per le procedure di risoluzione dei problemi più importanti, consulta the section called "Errori di certificato SSL".

Utilizzo degli endpoint in AWS CLI

Per connettersi a livello di programmazione a un Servizio AWS, è possibile utilizzare un endpoint. Un endpoint è l'URL del punto di ingresso di un servizio Web AWS. La AWS CLI (AWS Command Line Interface) utilizza automaticamente l'endpoint predefinito per ogni servizio in una Regione AWS, ma puoi specificare un endpoint alternativo per le tue richieste API.

Argomenti sugli endpoint

- Impostare l'endpoint per un unico comando
- Imposta un endpoint globale per tutti i Servizi AWS
- Imposta per utilizzare gli endpoint FIP per tutti i Servizi AWS
- Impostato per utilizzare gli endpoint dual-stack per tutti i Servizi AWS
- Imposta endpoint specifici del servizio
  - Endpoint specifici del servizio: variabili di ambiente
  - Endpoint specifici del servizio: file config condiviso
  - Endpoint specifici del servizio: elenco di identificatori specifici del servizio
- Endpoint basati sull'account
- Configurazione dell'endpoint e precedenza delle impostazioni

Impostare l'endpoint per un unico comando

Per sovrascrivere le impostazioni degli endpoint o le variabili di ambiente per un singolo comando, utilizza l'opzione `--endpoint-url` della riga di comando. Il comando di esempio seguente utilizza un URL personalizzato per l'endpoint Amazon S3.

```
$ aws s3 ls --endpoint-url http://localhost:4567
```

Imposta un endpoint globale per tutti i Servizi AWS

Per indirizzare le richieste di tutti i servizi a un URL endpoint personalizzato, utilizza una delle seguenti impostazioni:

- Variabili di ambiente:
  - `AWS_IGNORE_CONFIGURED_ENDPOINT_URLS` - Ignora gli URL degli endpoint configurati.

Linux or macOS

```
$ export AWS_IGNORE_CONFIGURED_ENDPOINT_URLS=true
```

Windows Command Prompt

Impostazione per tutte le sessioni

```
C:\> setx AWS_IGNORE_CONFIGURED_ENDPOINT_URLS true
```

Impostazione solo per la sessione corrente

```
C:\> set AWS_IGNORE_CONFIGURED_ENDPOINT_URLS=true
```

PowerShell

```
PS C:\> $Env:AWS_IGNORE_CONFIGURED_ENDPOINT_URLS="true"
```

- AWS_ENDPOINT_URL - Imposta l'URL globale dell'endpoint.

  Linux or macOS

  ```
  $ export AWS_ENDPOINT_URL=http://localhost:4567
  ```

  Windows Command Prompt

  Impostazione per tutte le sessioni

  ```
  C:\> setx AWS_ENDPOINT_URL http://localhost:4567
  ```

  Impostazione solo per la sessione corrente

  ```
  C:\> set AWS_ENDPOINT_URL=http://localhost:4567
  ```

  PowerShell

  ```
  PS C:\> $Env:AWS_ENDPOINT_URL="http://localhost:4567"
  ```

- Il file `config`

- ignore_configure_endpoint_urls - Ignora gli URL degli endpoint configurati.

```
ignore_configure_endpoint_urls = true
```

- endpoint_url - Imposta l'URL globale dell'endpoint.

```
endpoint_url = http://localhost:4567
```

Gli endpoint specifici del servizio e l'opzione della riga di comando `--endpoint-url` sostituiscono qualsiasi endpoint globale.

Imposta per utilizzare gli endpoint FIP per tutti i Servizi AWS

Per indirizzare le richieste di tutti i servizi all'utilizzo degli endpoint FIPS, utilizza uno dei seguenti:

- Variabile di ambiente AWS_USE_FIPS_ENDPOINT.

  Linux or macOS

  ```
  $ export AWS_USE_FIPS_ENDPOINT=true
  ```

  Windows Command Prompt

  Impostazione per tutte le sessioni

  ```
  C:\> setx AWS_USE_FIPS_ENDPOINT true
  ```

  Impostazione solo per la sessione corrente

  ```
  C:\> set AWS_USE_FIPS_ENDPOINT=true
  ```

  PowerShell

  ```
  PS C:\> $Env:AWS_USE_FIPS_ENDPOINT="true"
  ```

- Impostazione del file use_fips_endpoint.

```
use_fips_endpoint = true
```

Alcuni servizi AWS offrono endpoint che supportano lo standard FIPS (Federal Information Processing Standard) 140-2 in determinate Regioni AWS. Quando il servizio AWS supporta lo standard FIPS, questa impostazione specifica quale endpoint FIPS la AWS CLI deve utilizzare. A differenza degli endpoint AWS standard, gli endpoint FIPS utilizzano una libreria software TLS conforme agli standard FIPS 140-2. Questi endpoint potrebbero essere necessari ad aziende che interagiscono con il governo degli Stati Uniti.

Se questa impostazione è abilitata, ma non esiste un endpoint FIPS per il servizio nella Regione AWS, il comando AWS potrebbe non riuscire. In questo caso, specifica manualmente l'endpoint da utilizzare nel comando utilizzando l'opzione `--endpoint-url` oppure utilizza gli endpoint specifici del servizio.

Per ulteriori informazioni sulla specificazione degli endpoint FIPS tramite Regione AWS, consulta Endpoint FIPS per servizio.

Impostato per utilizzare gli endpoint dual-stack per tutti i Servizi AWS

Per indirizzare le richieste relative a tutti i servizi in modo che utilizzino gli endpoint dual-stack quando disponibili, utilizza una delle seguenti impostazioni:

- Variabile di ambiente `AWS_USE_DUALSTACK_ENDPOINT`.

  Linux or macOS

  ```
  $ export AWS_USE_DUALSTACK_ENDPOINT=true
  ```

  Windows Command Prompt

  Impostazione per tutte le sessioni

  ```
  C:\> setx AWS_USE_DUALSTACK_ENDPOINT true
  ```

  Impostazione solo per la sessione corrente

  ```
  C:\> set AWS_USE_DUALSTACK_ENDPOINT=true
  ```

  PowerShell

  ```
  PS C:\> $Env:AWS_USE_DUALSTACK_ENDPOINT="true"
  ```

- Impostazione del file `use_dualstack_endpoint`.

```
use_dualstack_endpoint = true
```

Consente l'uso di endpoint dual-stack per inviare richieste AWS. Per ulteriori informazioni sugli endpoint dual-stack, che supportano sia il traffico IPv4 che IPv6, consulta Using Amazon S3 dual-stack endpoints nella Guida per l'utente di Amazon Simple Storage Service. Gli endpoint dual-stack sono disponibili per determinati servizi in alcune Regioni. Se non esiste un endpoint dual-stack per il servizio o la Regione AWS, la richiesta ha esito negativo. Questa opzione è disabilitata per impostazione predefinita.

Imposta endpoint specifici del servizio

La configurazione dell'endpoint specifica per il servizio offre la possibilità di utilizzare un endpoint persistente di tua scelta per le richieste della AWS CLI. Queste impostazioni offrono la flessibilità necessaria per supportare endpoint locali, endpoint VPC e ambienti di sviluppo locale di terze parti AWS. È possibile utilizzare endpoint diversi per ambienti di test e produzione. È possibile specificare un URL di endpoint per singoli utenti dei Servizi AWS.

Gli endpoint specifici del servizio possono essere specificati nei seguenti modi:

- L'opzione della riga di comando `--endpoint-url` per un singolo comando.
- Variabili di ambiente:
  - `AWS_IGNORE_CONFIGURED_ENDPOINT_URLS` - Ignora tutti gli URL degli endpoint configurati, a meno che non siano specificati nella riga di comando.
  - `AWS_ENDPOINT_URL_<SERVICE>` - Specifica un endpoint personalizzato utilizzato per un servizio specifico, dove `<SERVICE>` viene sostituito con l'identificatore Servizio AWS. Per tutte le variabili specifiche del servizio, consulta the section called "Elenco di identificatori specifici del servizio".
- `config` file
  - `ignore_configure_endpoint_urls` - Ignora tutti gli URL degli endpoint configurati, a meno che non venga specificato tramite variabili di ambiente o sulla riga di comando.
  - La sezione `services` del file `config` combinata con l'impostazione del file `endpoint_url`.

Argomenti relativi agli endpoint specifici del servizio:

- Endpoint specifici del servizio: variabili di ambiente
- Endpoint specifici del servizio: file config condiviso
- Endpoint specifici del servizio: elenco di identificatori specifici del servizio

Endpoint specifici del servizio: variabili di ambiente

Le variabili di ambiente sostituiscono le impostazioni nel file di configurazione, ma non sostituiscono le opzioni specificate nella riga di comando. Utilizza le variabili di ambiente se desideri che tutti i profili impieghino gli stessi endpoint sul tuo dispositivo.

Le seguenti sono variabili di ambiente specifiche del servizio:

- `AWS_IGNORE_CONFIGURED_ENDPOINT_URLS` - Ignora tutti gli URL degli endpoint configurati, a meno che non siano specificati nella riga di comando.

  Linux or macOS

  ```
  $ export AWS_IGNORE_CONFIGURED_ENDPOINT_URLS=true
  ```

  Windows Command Prompt

  Impostazione per tutte le sessioni

  ```
  C:\> setx AWS_IGNORE_CONFIGURED_ENDPOINT_URLS true
  ```

  Impostazione solo per la sessione corrente

  ```
  C:\> set AWS_IGNORE_CONFIGURED_ENDPOINT_URLS=true
  ```

  PowerShell

  ```
  PS C:\> $Env:AWS_IGNORE_CONFIGURED_ENDPOINT_URLS="true"
  ```

- `AWS_ENDPOINT_URL_<SERVICE>` - Specifica un endpoint personalizzato utilizzato per un servizio specifico, dove `<SERVICE>` viene sostituito con l'identificatore Servizio AWS. Per tutte le variabili specifiche del servizio, consulta the section called "Elenco di identificatori specifici del servizio".

  I seguenti esempi di variabili di ambiente impostano un endpoint per AWS Elastic Beanstalk.

Linux or macOS

```
$ export AWS_ENDPOINT_URL_ELASTIC_BEANSTALK=http://localhost:4567
```

Windows Command Prompt

Impostazione per tutte le sessioni

```
C:\> setx AWS_ENDPOINT_URL_ELASTIC_BEANSTALK http://localhost:4567
```

Impostazione solo per la sessione corrente

```
C:\> set AWS_ENDPOINT_URL_ELASTIC_BEANSTALK=http://localhost:4567
```

PowerShell

```
PS C:\> $Env:AWS_ENDPOINT_URL_ELASTIC_BEANSTALK="http://localhost:4567"
```

Per ulteriori informazioni sull'impostazione delle variabili di ambiente, consulta the section called "Variabili di ambiente".

Endpoint specifici del servizio: file **`config`** condiviso

Nel file `config` condiviso, `endpoint_url` viene utilizzato in più sezioni. Per impostare un endpoint specifico del servizio, utilizza l'impostazione `endpoint_url`annidata sotto una chiave identificativa del servizio all'interno di una sezione dei `services`. Per i dettagli sulla definizione di una sezione `services` nel file `config` condiviso, consulta the section called "`services`".

L'esempio seguente utilizza una sezione `services` per configurare un URL dell'endpoint specifico del servizio per Amazon S3 e un endpoint globale personalizzato utilizzato per tutti gli altri servizi:

```
[profile dev1]
endpoint_url = http://localhost:1234
services = s3-specific

[services testing-s3]
s3 =
  endpoint_url = http://localhost:4567
```

Un singolo profilo può configurare gli endpoint per più servizi. L'esempio seguente imposta gli URL degli endpoint specifici del servizio per Amazon S3 e nello stesso profilo AWS Elastic Beanstalk.

Per un elenco di tutte le chiavi identificative del servizio da utilizzare nella sezione `services`, consulta Elenco di identificatori specifici del servizio.

```
[profile dev1]
services = testing-s3-and-eb

[services testing-s3-and-eb]
s3 =
  endpoint_url = http://localhost:4567
elastic_beanstalk =
  endpoint_url = http://localhost:8000
```

La sezione di configurazione del servizio può essere utilizzata in più profili. L'esempio seguente presenta due profili che utilizzano la stessa definizione dei `services`:

```
[profile dev1]
output = json
services = testing-s3

[profile dev2]
output = text
services = testing-s3

[services testing-s3]
s3 =
  endpoint_url = https://localhost:4567
```

Endpoint specifici del servizio: elenco di identificatori specifici del servizio

L'identificatore Servizio AWS si basa sul `serviceId` del modello API, sostituendo tutti gli spazi con caratteri di sottolineatura e rendendo tutte le lettere minuscole.

La tabella seguente elenca tutti gli identificatori, le chiavi di file e le variabili di ambiente specifici del servizio `config`.

Endpoint basati sull'account

Gli endpoint basati su account possono essere specificati nei seguenti modi:

- Variabili di ambiente
  - AWS_ACCOUNT_ID - Specifica l'ID dell'endpoint basato sull'account AWS da utilizzare per le chiamate ai Servizi AWS supportati.

    Linux or macOS

    ```
    $ export AWS_ACCOUNT_ID=<account-id>
    ```

    Windows Command Prompt

    Impostazione per tutte le sessioni

    ```
    C:\> setx AWS_ACCOUNT_ID <account-id>
    ```

    Impostazione solo per la sessione corrente

    ```
    C:\> set AWS_ACCOUNT_ID=<account-id>
    ```

    PowerShell

    ```
    PS C:\> $Env:AWS_ACCOUNT_ID="<account-id>"
    ```

  - AWS_ACCOUNT_ID_ENDPOINT_MODE - Specifica se utilizzare gli ID dell'endpoint basato sull'account AWS per le chiamate ai Servizi AWS supportati. Può essere impostato su `preferred`, `disabled` o `required`. Il valore predefinito è `preferred`.

    Linux or macOS

    ```
    $ export AWS_ACCOUNT_ID_ENDPOINT_MODE=preferred
    ```

    Windows Command Prompt

    Impostazione per tutte le sessioni

    ```
    C:\> setx AWS_ACCOUNT_ID_ENDPOINT_MODE preferred
    ```

    Impostazione solo per la sessione corrente

    ```
    C:\> set AWS_ACCOUNT_ID_ENDPOINT_MODE=preferred
    ```

PowerShell

```
PS C:\> $Env:AWS_ACCOUNT_ID_ENDPOINT_MODE="preferred"
```

- Il file `config`
  - `aws_account_id` - Specifica l'ID dell'endpoint basato sull'account AWS da utilizzare per le chiamate ai Servizi AWS supportati.

    ```
    aws_account_id = <account-id>
    ```

  - `account_id_endpoint_mode` - Specifica se utilizzare gli ID dell'endpoint basato sull'account AWS per le chiamate ai Servizi AWS supportati. Può essere impostato su `preferred`, `disabled` o `required`. Il valore predefinito è preferred.

    ```
    account_id_endpoint_mode = preferred
    ```

Gli endpoint basati sull'account aiutano a garantire prestazioni e scalabilità elevate utilizzando l'ID AWS dell'account per semplificare il routing delle richieste degli Servizio AWS che supportano questa funzionalità. Quando si utilizza un provider di credenziali e un servizio che supporta gli endpoint basati su account, AWS CLI crea e utilizza automaticamente un endpoint basato sull'account anziché un endpoint regionale.

Gli endpoint basati sull'account utilizzano il seguente formato, in cui `<account-id>` viene sostituito dall'ID dell'account AWS e `<region>` viene sostituito dalla tua Regione AWS:

```
https://<account-id>.myservice.<region>.amazonaws.com
```

Per impostazione predefinita nella AWS CLI, la modalità endpoint basata sull'account è impostata su `preferred`.

Configurazione dell'endpoint e precedenza delle impostazioni

Le impostazioni di configurazione dell'endpoint sono situate in più posizioni, ad esempio le variabili di ambiente del sistema o dell'utente e i file di configurazione AWS locali, oppure possono essere dichiarate esplicitamente come parametri dalla linea di comando. Le impostazioni di configurazione dell'endpoint AWS CLI hanno la precedenza nell'ordine seguente:

1. Opzione della riga di comando `--endpoint-url`.

2. Se abilitata, la variabile di ambiente globale AWS_IGNORE_CONFIGURED_ENDPOINT_URLS dell'endpoint o l'impostazione del profilo ignore_configure_endpoint_urls per ignorare gli endpoint personalizzati.
3. Il valore fornito da una variabile di ambiente AWS_ENDPOINT_URL_<SERVICE> specifica del servizio, ad esempio AWS_ENDPOINT_URL_DYNAMODB.
4. I valori forniti dalle variabili di ambiente AWS_USE_DUALSTACK_ENDPOINT, AWS_USE_FIPS_ENDPOINT e AWS_ENDPOINT_URL.
5. La variabile di ambiente AWS_ACCOUNT_ID_ENDPOINT_MODE è impostata su `preferred` o `required` e utilizza l'ID account nella variabile aws_account_id o nell'impostazione AWS_ACCOUNT_ID di ambiente.
6. Il valore dell'endpoint specifico del servizio fornito dall'impostazione endpoint_url all'interno di una sezione `services` del file condiviso `config`.
7. Il valore fornito dall'impostazione endpoint_url all'interno di un `profile` dei file `config` condivisi.
8. Impostazioni di use_dualstack_endpoint, use_fips_endpoint e endpoint_url.
9. L'impostazione account_id_endpoint_mode è configurata su `preferred` o `required` che utilizza l'ID account nella variabile di ambiente AWS_ACCOUNT_ID o nell'impostazione aws_account_id.
10. Qualsiasi URL di endpoint predefinito per il rispettivo Servizio AWS viene utilizzato per ultimo. Per un elenco degli endpoint del servizio standard disponibili in ogni regione, consulta Regioni ed endpoint AWS nella Riferimenti generali di Amazon Web Services.

Credenziali di autenticazione e accesso per la AWS CLI

È necessario stabilire la modalità di AWS CLI autenticazione AWS quando si sviluppa con i servizi. AWS Per configurare le credenziali per l'accesso programmatico per il AWS CLI, scegliete una delle seguenti opzioni. Le opzioni sono elencate in ordine di raccomandazione.

Tipo di autenticazione	Scopo	Istruzioni
AWS Credenziali della console di gestione	(Consigliato) Utilizza credenziali a breve termine accedendo alla AWS CLI con le credenziali della console. Consigliato se utilizzi root, utenti IAM o federazione con IAM per l'accesso all'account AWS	the section called "Credenziali della console"
Credenziali a breve termine per gli utenti della forza lavoro del Centro identità IAM	Utilizza credenziali a breve termine per un utente della forza lavoro di IAM Identity Center. Le best practice di sicurezza sono da utilizzare AWS Organizations con IAM Identity Center. Combina credenzia li a breve termine con una directory utente, ad esempio la directory integrata del Centro identità IAM o di Active Directory.	the section called "Autentic azione in Centro identità IAM"
Credenziali a breve termine per utenti IAM	Utilizza le credenziali a breve termine degli utenti IAM, che sono più sicure delle credenziali a lungo termine. Se le tue credenziali sono	the section called "Credenziali a breve termine"

Tipo di autenticazione	Scopo	Istruzioni
	compromesse, è previsto un periodo di tempo limitato in cui possono essere utilizzate prima della scadenza.	
Utenti IAM o IAM Identity Center su un' EC2 istanza Amazon.	Utilizza i metadati delle EC2 istanze Amazon per richieder e credenziali temporanee utilizzando il ruolo assegnato all'istanza Amazon EC2 .	the section called "Metadati Amazon EC2"
Assumere ruoli per le autorizzazioni	Associa un altro metodo di credenziali e assumi un ruolo per l'accesso temporaneo a cui l'utente dei Servizi AWS potrebbe non avere accesso.	the section called "Ruoli IAM"
Credenziali a lungo termine per utenti IAM	(Non raccomandato) Utilizzar e credenziali a lungo termine, che non hanno scadenza.	the section called "Utenti IAM"
Archiviazione esterna degli utenti della forza lavoro IAM o del Centro identità IAM	(Non raccomandato) Associare un altro metodo di credenziali ma archiviar e i valori delle credenziali in una posizione esterna alla AWS CLI. Questo metodo è sicuro solo quanto la posizione esterna in cui sono archiviate le credenziali.	the section called "Credenziali esterne"

Configurazione e precedenza delle credenziali

Le credenziali e le impostazioni di configurazione si trovano in più posizioni, come le variabili di sistema o di ambiente utente, i file di AWS configurazione locali o sono dichiarate esplicitamente

nella riga di comando come parametro. Alcune autenticazioni hanno la precedenza su altre. Le impostazioni di autenticazione della AWS CLI hanno la precedenza nell'ordine seguente:

1. Opzioni della riga di comando: sovrascrive le impostazioni in qualsiasi altra posizione, come i parametri `--region`, `--output` e `--profile`.
2. Variabili di ambiente: puoi archiviare i valori nelle variabili di ambiente del sistema.
3. Assumere un ruolo: assumi le autorizzazioni di un ruolo IAM tramite la configurazione o il comando `assume-role`.
4. Assumere un ruolo con l'identità web: assumi le autorizzazioni di un ruolo IAM utilizzando l'identità web tramite la configurazione o il comando `assume-role-with-web-identity`.
5. AWS IAM Identity Center: le impostazioni di configurazione del centro identità IAM memorizzate nel file `config` vengono aggiornate quando si esegue il comando `aws configure sso`. Le credenziali vengono quindi autenticate quando si esegue il comando `aws sso login`. Il file `config` si trova in `~/.aws/config` su Linux o macOS o in `C:\Users\`*`USERNAME`*`\.aws\config` su Windows.
6. File delle credenziali della CLI: i file `credentials` e `config` vengono aggiornati quando esegui il comando `aws configure`. Il file `credentials` si trova in `~/.aws/credentials` su Linux o macOS o in `C:\Users\`*`USERNAME`*`\.aws\credentials` su Windows.
7. Processo personalizzato: ottieni le tue credenziali da una fonte esterna.
8. File di configurazione: i file `credentials` e `config` vengono aggiornati quando esegui il comando `aws configure`. Il file `config` si trova in `~/.aws/config` su Linux o macOS o in `C:\Users\`*`USERNAME`*`\.aws\config` su Windows.
9. Credenziali del container: puoi associare un ruolo IAM a ciascuna delle definizioni di attività di Amazon Elastic Container Service (Amazon ECS). Le credenziali temporanee per tale ruolo sono quindi disponibili per i contenitori dell'attività. Per ulteriori informazioni, consulta Ruoli IAM per le attività nella Guida per sviluppatori di Amazon Elastic Container Service.
10. Credenziali del profilo dell' EC2istanza Amazon: puoi associare un ruolo IAM a ciascuna delle tue istanze Amazon Elastic Compute Cloud EC2 (Amazon). Le credenziali temporanee per tale ruolo sono quindi disponibili per il codice in esecuzione nell'istanza. Le credenziali vengono fornite tramite il servizio di EC2 metadati Amazon. Per ulteriori informazioni, consulta IAM Roles for Amazon EC2 nella Amazon EC2 User Guide e Using Instance Profiles nella IAM User Guide.

Argomenti aggiuntivi in questa sezione

- the section called "Autenticazione in Centro identità IAM"
- the section called "Credenziali a breve termine"
- the section called "Ruoli IAM"
- the section called "Utenti IAM"
- the section called "Metadati Amazon EC2"
- the section called "Credenziali esterne"

Accedi per lo sviluppo AWS locale utilizzando le credenziali della console

È possibile utilizzare le credenziali di accesso esistenti AWS della Console di gestione per l'accesso programmatico ai servizi. AWS Dopo un flusso di autenticazione basato su browser, AWS genera credenziali temporanee che funzionano con strumenti di sviluppo locali come AWS CLI, Tools for and. AWS PowerShell AWS SDKs Questa funzionalità semplifica il processo di configurazione e gestione delle credenziali AWS CLI, soprattutto se si preferisce l'autenticazione interattiva alla gestione delle chiavi di accesso a lungo termine.

Con questo processo, puoi autenticarti utilizzando le credenziali root create durante la configurazione iniziale dell'account, un utente IAM o un'identità federata del tuo provider di identità e la AWS CLI gestisce automaticamente le credenziali temporanee per te. Questo approccio migliora la sicurezza eliminando la necessità di archiviare le credenziali a lungo termine a livello locale.

Quando si esegue il `aws login` comando, è possibile selezionare una delle sessioni attive della console oppure accedere tramite il flusso di autenticazione basato sul browser. In questo modo verranno generate automaticamente credenziali temporanee. La CLI aggiornerà automaticamente queste credenziali per un massimo di 12 ore.

Una volta configurata, la sessione può essere utilizzata nella AWS CLI e in altri strumenti AWS SDKs .

Argomenti

- Prerequisiti
- Accedi alla AWS CLI con il aws login comando.
- Esegui un comando con il tuo profilo

- Esci dalla sessione utilizzando il comando aws logout
- Risoluzione dei problemi
- Risorse correlate

Prerequisiti

- Installa la AWS CLI. Per ulteriori informazioni, consulta Installazione o aggiornamento dell'ultima versione di AWS CLI. Per utilizzare il comando è richiesta una versione minima di 2.32.0. `aws login`
- Accesso per accedere alla console di AWS gestione come utente root, utente IAM o tramite federazione con IAM. Se utilizzi IAM Identity Center, vai Configurazione dell'autenticazione IAM Identity Center con AWS CLI invece a.
- Assicurati che l'identità IAM disponga delle autorizzazioni appropriate. Allega la policy SignInLocalDevelopmentAccessgestita al tuo utente, ruolo o gruppo IAM. Se accedi come utente root, non sono richieste autorizzazioni aggiuntive.

Accedi alla AWS CLI con il **`aws login`** comando.

Esegui il `aws login` comando per autenticarti utilizzando le credenziali AWS della console di gestione esistenti. Se non hai configurato in precedenza un profilo, ti verranno richieste ulteriori informazioni. Per accedere o configurare un profilo, procedi nel seguente modo.

1. Nel terminale che preferisci esegui il comando `aws login`.

```
$ aws login
```

Per accedere a un profilo denominato o crearne uno nuovo, utilizza l'`--profile`opzione.

```
$ aws login --profile my-dev-profile
```

- Se si tratta di un nuovo profilo o non è stata specificata alcuna AWS regione, la AWS CLI richiede di fornire una regione.

```
No AWS region has been configured. The AWS region is the geographic location of
your AWS resources.
```

```
If you've used AWS before and already have resources in your account, tell us
which region they were created in. If you haven't created resources in your
 account
before, you can pick the region closest to you:
https://docs.aws.amazon.com/global-infrastructure/latest/regions/aws-regions.html.
You are able to change the region in the CLI at any time with the command
`aws configure set region NEW_REGION`.
AWS Region [us-east-1]:
```

- Se il dispositivo che utilizza la AWS CLI non dispone di un browser, puoi utilizzare l'`--remote`opzione per fornire un URL da aprire su un dispositivo abilitato al browser.

```
$ aws login --remote
```

> **Note**
>
> Puoi controllare l'accesso all'autenticazione sullo stesso dispositivo () e su più dispositivi (`aws login`). `aws login --remote` Utilizza la seguente risorsa ARNs in qualsiasi policy IAM pertinente.
>
> - `arn:aws:signin:`*`region`*`:`*`account-id`*`:oauth2/public-client/localhost`— Usa questo ARN per l'autenticazione sullo stesso dispositivo con. `aws login`
> - `arn:aws:signin:`*`region`*`:`*`account-id`*`:oauth2/public-client/remote`— Utilizza questo ARN per l'autenticazione tra dispositivi con. `aws login --remote`

2. La AWS CLI tenta di aprire il browser predefinito per la procedura di accesso del tuo AWS account.

```
Attempting to open the login page for `us-east-1` in your default browser.
If the browser does not open, use the following URL to complete your login:
https://signin.us-east-1.amazonaws.com/authorize?<abbreviated>

If you cannot connect to this URL, make sure that you have specified a valid region.
```

- Se hai utilizzato l'`--remote`opzione, vengono visualizzate le istruzioni per avviare manualmente la procedura di accesso in base al tipo di autorizzazione che stai utilizzando. L'URL visualizzato è un URL univoco che inizia con: https://us-east-1.signin.amazonaws.com /authorize. Una volta completato l'accesso al browser, dovrai copiare e incollare il codice di autorizzazione risultante nella CLI.

```
Browser will not be automatically opened.
Please visit the following URL:
https://region.signin.amazonaws.com/authorize?<abbreviated>

Please enter the authorization code displayed in the browser:
```

3. Nel browser, seleziona le tue credenziali da utilizzare dall'elenco visualizzato e poi torna al tuo terminale.
   - Se il profilo che stai configurando ha una sessione di accesso precedentemente configurata che non corrisponde alla nuova sessione, la AWS CLI richiede di confermare che stai cambiando la sessione corrispondente al profilo esistente.

```
Profile signin is already configured to use session
 arn:aws:iam::0123456789012:user/ReadOnly.
Do you want to overwrite it to use arn:aws:iam::0123456789012:user/Admin instead?
 (y/n):.
```

4. Un messaggio finale descrive la configurazione del profilo completata. Puoi ora utilizzare questo profilo per richiedere le credenziali. Utilizza il comando `aws login` per richiedere e recuperare le credenziali necessarie per eseguire i comandi.

   Il token di autenticazione viene memorizzato nella cache su disco `.aws/login/cache` nella directory con un nome di file basato sul profilo risolto.

File di configurazione generato

Questi passaggi comportano la creazione del `default` profilo nel `config` file simile al seguente:

```
[default]
login_session = arn:aws:iam::0123456789012:user/username
region = us-east-1
```

Esegui un comando con il tuo profilo

Una volta effettuato l'accesso, puoi utilizzare le tue credenziali per richiamare i comandi AWS CLI con il profilo associato. L'esempio seguente chiama il `get-caller-identity` comando utilizzando il profilo: `default`

```
$ aws sts get-caller-identity
```

Per accedere a una sessione specifica, utilizzate l'`--profile`opzione.

```
$ aws sts get-caller-identity --profile my-dev-profile
```

La AWS CLI e SDKs aggiorneranno automaticamente le credenziali memorizzate nella cache ogni 15 minuti, se necessario. La sessione complessiva sarà valida fino alla durata della sessione impostata per il principale IAM (massimo 12 ore), dopodiché dovrai eseguire nuovamente. `aws login`

Esci dalla sessione utilizzando il comando aws logout

Quando hai finito di usare la sessione, puoi lasciare scadere le credenziali o eseguire il `aws logout` comando per eliminare le credenziali memorizzate nella cache. Se non viene specificato alcun profilo nella riga di comando o nella variabile di `AWS_PROFILE` ambiente, il comando disconnette l'utente dal profilo predefinito. L'esempio seguente ti disconnette dal tuo profilo predefinito.

```
$ aws logout
```

Per uscire da una sessione specifica, usa l'`--profile`opzione.

```
$ aws logout --profile my-dev-profile
```

Per uscire da tutti i profili che utilizzano le credenziali di accesso, utilizza l'`--all`opzione.

```
$ aws logout --all
```

Credenziali memorizzate nella cache

Le credenziali temporanee memorizzate nella cache e i metadati necessari per aggiornarle vengono archiviati per impostazione predefinita in `~/.aws/login/cache` Linux e macOS o su Windows. `%USERPROFILE%\.aws\login\cache`

Per archiviare la cache delle credenziali a breve termine in una posizione alternativa, imposta la variabile di ambiente. `AWS_LOGIN_CACHE_DIRECTORY`

Condivisione delle credenziali di accesso come credenziali di processo

Le versioni precedenti AWS SDKs o di altri strumenti di sviluppo potrebbero non supportare ancora le credenziali della console. Come soluzione alternativa, puoi configurare la AWS CLI in modo che

funga da fornitore di credenziali di processo. La CLI continuerà ad aggiornare le credenziali secondo necessità, condividendole con strumenti configurati per utilizzare il profilo credential_process.

In questo esempio, usa la AWS CLI per accedere prima per l'accesso al profilo:

```
$ aws login --profile signin
```

Quindi, configura manualmente un profilo con l'opzione credential_process, che rimanda al profilo di accesso. Ora puoi configurare i nostri strumenti SDKs per utilizzare il profilo di processo, che richiamerà la CLI per condividere le credenziali dal profilo di accesso.

```
[profile signin]
login_session = arn:aws:iam::0123456789012:user/username
region = us-east-1

[profile process]
credential_process = aws configure export-credentials --profile signin --format process
region = us-east-1
```

Risoluzione dei problemi

Questa pagina contiene consigli per la risoluzione dei problemi relativi all'accesso per lo sviluppo AWS locale utilizzando le credenziali della console per la CLI. AWS

> **Note**
>
> Per risolvere altri problemi che potresti riscontrare utilizzando la AWS CLI, vedi. Risoluzione degli errori per AWS CLI

ExpiredToken o AccessDeniedException errori dopo aver usato «aws login»

Quando si esegue un comando AWS CLI dopo l'esecuzione `aws login` per un determinato profilo, è possibile che si verifichi un errore di credenziali scadute o non valide.

```
$ aws s3 ls

An error occurred (ExpiredToken) when calling the ListBuckets operation: The provided
 token has expired.
```

Possibile causa: in quel profilo è presente una combinazione di credenziali esistenti e nuove credenziali di accesso

Esegui `aws configure list` o stampa `aws configure list --profile <profile name>` da dove la CLI sta risolvendo le credenziali per il profilo predefinito o specificato.

Se la colonna TYPE è diversa dal login, significa che nel profilo di destinazione è ancora impostato un tipo diverso di credenziali.

In questo esempio, le credenziali vengono risolte dal file di credenziali condivise nella home directory, che ha la precedenza sulle credenziali di accesso.

```
$  aws configure list
NAME       : VALUE                    : TYPE             : LOCATION
profile    : <not set>                : None             : None
access_key : ****************MPLE     : shared-credentials-file :
secret_key : ****************EKEY     : shared-credentials-file :
region     : us-east-1                : config-file      : ~/.aws/config
```

Per risolvere questo problema, rimuovete manualmente tutte le credenziali esistenti dal file di configurazione e delle credenziali per il profilo di destinazione. Dopo averlo fatto, dovresti vedere le credenziali di accesso quando esegui di nuovo. `aws configure list`

```
$  aws configure list
NAME       : VALUE                    : TYPE             : LOCATION
profile    : <not set>                : None             : None
access_key : ****************MPLE     : login            :
secret_key : ****************EKEY     : login            :
region     : us-east-1                : config-file      : ~/.aws/config
```

In alternativa, l'utilizzo dell'`--debug`opzione mostrerà da dove la CLI sta risolvendo le credenziali.

Firewall che blocca l'accesso alla rete durante l'esecuzione di «aws login»

Durante l'esecuzione, `aws login` è possibile che venga visualizzato un popup o un messaggio dal software firewall che impedisce alla AWS CLI di accedere alla rete.

Possibile causa: il firewall o il software di sicurezza impediscono alla AWS CLI di aprire la porta utilizzata per gestire il OAuth callback.

Per evitare questo problema, utilizzate invece l'`--remote`opzione. Verrà richiesto di copiare e incollare il codice di autorizzazione invece di utilizzare il OAuth callback.

```
$ aws login --remote
```

Risorse correlate

Di seguito sono elencate alcune risorse aggiuntive.

- Installazione o aggiornamento dell'ultima versione di AWS CLI
- `aws login`nella versione AWS CLI 2 Reference
- `aws logout`nella versione AWS CLI 2 Reference

Configurazione dell'autenticazione IAM Identity Center con AWS CLI

Questo argomento fornisce istruzioni su come configurare AWS CLI with AWS IAM Identity Center (IAM Identity Center) per recuperare le credenziali per eseguire i comandi. AWS CLI Esistono principalmente due modi per autenticare gli utenti con IAM Identity Center e ottenere le credenziali per eseguire AWS CLI comandi tramite il file: `config`

- (Scelta consigliata) Configurazione del provider di token SSO.
- Configurazione legacy non aggiornabile.

Per informazioni sull'utilizzo dell'autenticazione del portatore, che non utilizza l'ID e il ruolo dell'account, consulta Configurazione per l'utilizzo di AWS CLI with CodeCatalyst nella Amazon CodeCatalyst User Guide.

> **Note**
>
> Per una procedura guidata di utilizzo di IAM Identity Center con AWS CLI i comandi, consulta. the section called "Tutorial: AWS IAM Identity Center e Amazon S3"

Argomenti

- the section called "Prerequisiti"
- the section called "Configura il tuo profilo con la procedura guidata `aws configure sso`"

- the section called "Configura solo la sezione `sso-session` con la procedura guidata `aws configure sso-session`"
- the section called "Configurazione manuale utilizzando il file `config`"
- the section called "Accedere a una sessione di Centro identità IAM"
- the section called "Eseguire un comando con il profilo Centro identità IAM"
- the section called "Disconnettersi dalle sessioni in Centro identità IAM"
- the section called "Risoluzione dei problemi"
- the section called "Risorse correlate"

Prerequisiti

- Installa il AWS CLI. Per ulteriori informazioni, consulta the section called "Installazione o aggiornamento".
- Devi prima avere accesso all'autenticazione SSO all'interno di Centro di identità IAM. Scegli uno dei seguenti metodi per accedere alle tue AWS credenziali.

Non ho stabilito l'accesso tramite Centro identità IAM

Segui le istruzioni riportate in Nozioni di base nella Guida per l'utente di AWS IAM Identity Center . Questo processo attiva Centro identità IAM, crea un utente amministratore e aggiunge un set di autorizzazioni con privilegi minimi appropriato.

> **Note**
>
> Crea un set di autorizzazioni per l'applicazione di autorizzazioni con privilegio minimo. Consigliamo di utilizzare il set di autorizzazioni `PowerUserAccess` predefinito, a meno che il datore di lavoro non ne abbia creato uno personalizzato a questo scopo.

Esci dal portale e accedi nuovamente per visualizzare i Account AWS dettagli di accesso programmatici e le opzioni per `Administrator` o. `PowerUserAccess` Seleziona `PowerUserAccess` quando utilizzi l'SDK.

Ho già accesso AWS tramite un provider di identità federato gestito dal mio datore di lavoro (come Azure AD o Okta)

Accedi AWS tramite il portale del tuo provider di identità. Se il tuo amministratore cloud ti ha concesso le autorizzazioni `PowerUserAccess` (sviluppatore), vedi quelle a Account AWS cui hai accesso e il tuo set di autorizzazioni. Accanto al nome del set di autorizzazioni, vengono visualizzate le opzioni per accedere agli account manualmente o a livello di programmazione utilizzando il set di autorizzazioni.

Le implementazioni personalizzate possono dare luogo a esperienze diverse, ad esempio nomi di set di autorizzazioni diversi. In caso di dubbi su quale set di autorizzazioni utilizzare, contatta il team IT per assistenza.

Ho già accesso AWS tramite il portale di AWS accesso gestito dal mio datore di lavoro

Accedi AWS tramite il tuo portale di AWS accesso. Se il tuo amministratore cloud ti ha concesso le autorizzazioni `PowerUserAccess` (sviluppatore), vedi quelle a Account AWS cui hai accesso e il set di autorizzazioni. Accanto al nome del set di autorizzazioni, vengono visualizzate le opzioni per accedere agli account manualmente o a livello di programmazione utilizzando il set di autorizzazioni.

Ho già accesso AWS tramite un provider di identità personalizzato federato gestito dal mio datore di lavoro

Contatta il team IT per assistenza.

Dopo aver ottenuto l'accesso a Centro identità IAM, raccogli le informazioni su Centro identità IAM effettuando le operazioni seguenti:

1. Recupera i valori `SSO Start URL` e `SSO Region` necessari per eseguire `aws configure sso`

    a. Nel portale di AWS accesso, seleziona il set di autorizzazioni che utilizzi per lo sviluppo e seleziona il link Access keys.
    b. Nella finestra di dialogo Ottieni le credenziali seleziona la scheda corrispondente al tuo sistema operativo.
    c. Scegli il metodo Credenziali Centro identità IAM per ottenere i valori `SSO Start URL` e `SSO Region`.
2. In alternativa, a partire dalla versione 2.22.0 puoi utilizzare l'URL dell'emittente anziché l'URL di avvio. L'URL dell'emittente si trova nella AWS IAM Identity Center console in una delle seguenti posizioni:

- Nella pagina Dashboard l'URL dell'emittente si trova nel riepilogo delle impostazioni.
- Nella pagina Impostazioni l'URL dell'emittente si trova nelle impostazioni in Origine identità.

3. Per informazioni sul valore degli ambiti da registrare, consulta OAuth 2.0 Access scopes nella Guida per l'utente di IAM Identity Center.

Configura il tuo profilo con la procedura guidata `aws configure sso`

Per configurare un profilo Centro identità IAM per AWS CLI:

1. Nel terminale che preferisci esegui il comando `aws configure sso`.

   (Recommended) IAM Identity Center

   Crea un nome di sessione, fornisci l'URL di avvio di IAM Identity Center o l'URL dell'emittente, Regione AWS che ospita la directory IAM Identity Center e l'ambito di registrazione.

   ```
   $ aws configure sso
   SSO session name (Recommended): my-sso
   SSO start URL [None]: https://my-sso-portal.awsapps.com/start
   SSO region [None]: us-east-1
   SSO registration scopes [None]: sso:account:access
   ```

   Per il supporto dual-stack, utilizza l'URL di avvio SSO dual-stack:

   ```
   $ aws configure sso
   SSO session name (Recommended): my-sso
   SSO start URL [None]: https://ssoins-1234567890abcdef.portal.us-east-1.app.aws
   SSO region [None]: us-east-1
   SSO registration scopes [None]: sso:account:access
   ```

   L'autorizzazione Proof Key for Code Exchange (PKCE) viene utilizzata per impostazione predefinita AWS CLI a partire dalla versione 2.22.0 e deve essere utilizzata su dispositivi dotati di browser. Per continuare a utilizzare l'autorizzazione del dispositivo, aggiungi l'opzione `--use-device-code`.

   ```
   $ aws configure sso --use-device-code
   ```

Legacy IAM Identity Center

Crea un nome di sessione e fornisci l'URL di avvio di Centro identità IAM e la Regione AWS che ospita la directory per Centro identità.

```
$ aws configure sso
SSO session name (Recommended):
SSO start URL [None]: https://my-sso-portal.awsapps.com/start
SSO region [None]:us-east-1
```

Per il supporto dual-stack:

```
$ aws configure sso
SSO session name (Recommended):
SSO start URL [None]: https://ssoins-1234567890abcdef.portal.us-east-1.app.aws
SSO region [None]:us-east-1
```

2. I AWS CLI tentativi di aprire il browser predefinito per la procedura di accesso del tuo account IAM Identity Center. Questo processo potrebbe richiederti di consentire l' AWS CLI accesso ai tuoi dati. Poiché AWS CLI è basato sull'SDK per Python, i messaggi di autorizzazione possono contenere variazioni del `botocore` nome.

   - Se AWS CLI non è possibile aprire il browser, vengono visualizzate le istruzioni per avviare manualmente la procedura di accesso in base al tipo di autorizzazione utilizzato.

     PKCE authorization

     L'autorizzazione Proof Key for Code Exchange (PKCE) viene utilizzata per impostazione predefinita AWS CLI a partire dalla versione 2.22.0. L'URL visualizzato è un URL univoco che inizia con:

     - IPv4: *https://oidc.us-east-1.amazonaws.com/authorize*
     - Doppio stack: *https://oidc.us-east-1.api.aws/authorize*

     L'autorizzazione PKCE URLs deve essere aperta sullo stesso dispositivo con cui si effettua l'accesso e deve essere utilizzata per un dispositivo dotato di browser.

```
Attempting to automatically open the SSO authorization
 page in your
```

```
default browser.
If the browser does not open or you wish to use a
 different device to
authorize the request, open the following URL:

https://oidc.us-east-1.amazonaws.com/authorize?
<abbreviated>
```

Device authorization

L'autorizzazione del dispositivo OAuth 2.0 viene utilizzata AWS CLI per le versioni precedenti alla 2.22.0. Puoi abilitare questo metodo sulle versioni più recenti utilizzando l'opzione `--use-device-code`.

L'autorizzazione del dispositivo URLs non deve essere aperta sullo stesso dispositivo a cui stai effettuando l'accesso e può essere utilizzata per un dispositivo con o senza browser. Il formato dell'endpoint dipende dalla tua configurazione:

- IPv4: *https://device.sso.us-west-2.amazonaws.com/*
- Doppio stack: *https://device.sso.us-west-2.api.aws/*

```
If the browser does not open or you wish to use a
 different device to
authorize this request, open the following URL:
https://device.sso.us-west-2.amazonaws.com/

Then enter the code:
QCFK-N451
```

3. Seleziona l' AWS account da utilizzare dall'elenco visualizzato. Se sei autorizzato a utilizzare un solo account, lo seleziona AWS CLI automaticamente e salta la richiesta.

```
There are 2 AWS accounts available to you.
> DeveloperAccount, developer-account-admin@example.com (123456789011)
  ProductionAccount, production-account-admin@example.com (123456789022)
```

4. Seleziona il ruolo IAM da utilizzare nell'elenco visualizzato. Se è disponibile un solo ruolo, lo seleziona AWS CLI automaticamente e salta la richiesta.

```
Using the account ID 123456789011
```

```
There are 2 roles available to you.
> ReadOnly
  FullAccess
```

5. Specifica il formato di output predefinito, la Regione AWS predefinita a cui inviare i comandi e un nome per il profilo. Se specifichi `default` come nome del profilo, questo diventa il profilo predefinito utilizzato. Nell'esempio seguente l'utente immette la Regione predefinita, il formato di output predefinito e il nome del profilo.

```
Default client Region [None]: us-west-2<ENTER>
CLI default output format (json if not specified) [None]: json<ENTER>
Profile name [123456789011_ReadOnly]: my-dev-profile<ENTER>
```

6. Un messaggio finale descrive la configurazione del profilo completata. Puoi ora utilizzare questo profilo per richiedere le credenziali. Utilizza il comando `aws sso login` per richiedere e recuperare le credenziali necessarie per eseguire i comandi. Per istruzioni, consulta Accedere a una sessione di Centro identità IAM.

File di configurazione generato

Questa procedura permette di creare la sezione `sso-session` e il profilo denominato nel file `config`, che avrà l'aspetto seguente:

IAM Identity Center

```
[profile my-dev-profile]
sso_session = my-sso
sso_account_id = 123456789011
sso_role_name = readOnly
region = us-west-2
output = json

[sso-session my-sso]
sso_region = us-east-1
sso_start_url = https://my-sso-portal.awsapps.com/start
sso_registration_scopes = sso:account:access
```

Per il supporto dual-stack:

```
[profile my-dev-profile]
sso_session = my-sso
```

```
sso_account_id = 123456789011
sso_role_name = readOnly
region = us-west-2
output = json

[sso-session my-sso]
sso_region = us-east-1
sso_start_url = https://ssoins-1234567890abcdef.portal.us-east-1.app.aws
sso_registration_scopes = sso:account:access
```

Legacy IAM Identity Center

```
[profile my-dev-profile]
sso_start_url = https://my-sso-portal.awsapps.com/start
sso_region = us-east-1
sso_account_id = 123456789011
sso_role_name = readOnly
region = us-west-2
output = json
```

Per il supporto dual-stack:

```
[profile my-dev-profile]
sso_start_url = https://ssoins-1234567890abcdef.portal.us-east-1.app.aws
sso_region = us-east-1
sso_account_id = 123456789011
sso_role_name = readOnly
region = us-west-2
output = json
```

Configura solo la sezione **`sso-session`** con la procedura guidata **`aws configure sso-session`**

> **Note**
>
> Questa configurazione non è compatibile con la configurazione legacy di Centro identità IAM.

Il comando `aws configure sso-session` aggiorna le sezioni `sso-session` nel file `~/.aws/config`. Esegui il `aws configure sso-session` comando e fornisci l'URL di avvio o l'URL

dell'emittente del tuo IAM Identity Center e la AWS regione che ospita la directory IAM Identity Center.

```
$ aws configure sso-session
SSO session name: my-sso
SSO start URL [None]: https://my-sso-portal.awsapps.com/start
SSO region [None]: us-east-1
SSO registration scopes [None]: sso:account:access
```

Per il supporto dual-stack, utilizza l'URL di avvio SSO dual-stack:

```
$ aws configure sso-session
SSO session name: my-sso
SSO start URL [None]: https://ssoins-1234567890abcdef.portal.us-east-1.app.aws
SSO region [None]: us-east-1
SSO registration scopes [None]: sso:account:access
```

Configurazione manuale utilizzando il file `config`

Le informazioni di configurazione di Centro identità IAM sono archiviate nel file `config` e possono essere modificate utilizzando un editor di testo. Per aggiungere manualmente il supporto di Centro identità IAM a un profilo denominato, devi aggiungere chiavi e valori al file `config`.

File di configurazione per Centro identità IAM

La `sso-session` sezione del `config` file viene utilizzata per raggruppare le variabili di configurazione per l'acquisizione di token di accesso SSO, che possono quindi essere utilizzati per acquisire credenziali. AWS Vengono utilizzate le impostazioni seguenti:

- (Obbligatorio) `sso_start_url`
- (Obbligatorio) `sso_region`
- `sso_account_id`
- `sso_role_name`
- `sso_registration_scopes`

È necessario definire una sezione `sso-session` e associarla a un profilo. Le impostazioni `sso_region` e `sso_start_url` devono essere configurate all'interno della sezione `sso-session`. In genere, le impostazioni `sso_account_id` e `sso_role_name` devono essere configurate nella sezione `profile` in modo che l'SDK possa richiedere le credenziali SSO.

L'esempio seguente configura l'SDK in modo da richiedere le credenziali SSO e supporta l'aggiornamento automatico dei token:

```
[profile dev]
sso_session = my-sso
sso_account_id = 111122223333
sso_role_name = SampleRole

[sso-session my-sso]
sso_region = us-east-1
sso_start_url = https://my-sso-portal.awsapps.com/start
```

Per il supporto dual-stack, usa il formato URL di avvio SSO dual-stack:

```
[profile dev]
sso_session = my-sso
sso_account_id = 111122223333
sso_role_name = SampleRole

[sso-session my-sso]
sso_region = us-east-1
sso_start_url = https://ssoins-1234567890abcdef.portal.us-east-1.app.aws
```

In questo modo, è possibile riutilizzare le configurazioni `sso-session` su più profili:

```
[profile dev]
sso_session = my-sso
sso_account_id = 111122223333
sso_role_name = SampleRole

[profile prod]
sso_session = my-sso
sso_account_id = 111122223333
sso_role_name = SampleRole2

[sso-session my-sso]
sso_region = us-east-1
sso_start_url = https://my-sso-portal.awsapps.com/start
```

Per il supporto dual-stack, usa il formato URL di avvio SSO dual-stack:

```
[profile dev]
```

```
sso_session = my-sso
sso_account_id = 111122223333
sso_role_name = SampleRole

[profile prod]
sso_session = my-sso
sso_account_id = 111122223333
sso_role_name = SampleRole2

[sso-session my-sso]
sso_region = us-east-1
sso_start_url = https://ssoins-1234567890abcdef.portal.us-east-1.app.aws
```

Tuttavia, le impostazioni `sso_account_id` e `sso_role_name` non sono necessarie per tutti gli scenari di configurazione dei token SSO. Se l'applicazione utilizza solo AWS servizi che supportano l'autenticazione al portatore, non sono necessarie le credenziali tradizionali. AWS Questo tipo di autenticazione è uno schema di autenticazione HTTP che utilizza token di sicurezza noti come token di connessione. In questo scenario le impostazioni `sso_account_id` e `sso_role_name` non sono obbligatorie. Consultate la guida individuale del vostro AWS servizio per determinare se supporta l'autorizzazione con token al portatore.

Inoltre, gli ambiti di registrazione possono essere configurati all'interno di un parametro `sso-session`. Scope è un meccanismo della OAuth versione 2.0 per limitare l'accesso di un'applicazione all'account di un utente. Un'applicazione può richiedere uno o più ambiti e il token di accesso emesso all'applicazione sarà limitato agli ambiti consentiti. Questi ambiti definiscono le autorizzazioni richieste per l'autorizzazione per il client OIDC registrato e i token di accesso recuperati dal client. L'esempio seguente imposta `sso_registration_scopes` in modo da fornire l'accesso necessario per elencare account/ruoli:

```
[sso-session my-sso]
sso_region = us-east-1
sso_start_url = https://my-sso-portal.awsapps.com/start
sso_registration_scopes = sso:account:access
```

Per il supporto dual-stack:

```
[sso-session my-sso]
sso_region = us-east-1
sso_start_url = https://ssoins-1234567890abcdef.portal.us-east-1.app.aws
sso_registration_scopes = sso:account:access
```

Il token di autenticazione viene memorizzato nella cache su disco all'interno della directory `sso/cache` con un nome di file basato sul nome della sessione.

File di configurazione per la configurazione legacy di Centro identità IAM

> **Note**
>
> L'aggiornamento automatico dei token non è supportato utilizzando la configurazione legacy non aggiornabile. Consigliamo di utilizzare la configurazione dei token SSO.

Per aggiungere manualmente il supporto di Centro identità IAM a un profilo denominato, devi aggiungere le chiavi e i valori seguenti alla definizione del profilo nel file `config`.

- `sso_start_url`
- `sso_region`
- `sso_account_id`
- `sso_role_name`

Puoi includere anche altri valori e chiavi che sono validi nel file `.aws/config`. L'esempio seguente è un profilo Centro identità IAM:

```
[profile my-sso-profile]
sso_start_url = https://my-sso-portal.awsapps.com/start
sso_region = us-west-2
sso_account_id = 111122223333
sso_role_name = SSOReadOnlyRole
region = us-west-2
output = json
```

Per il supporto dual-stack:

```
[profile my-sso-profile]
sso_start_url = https://ssoins-1234567890abcdef.portal.us-east-1.app.aws
sso_region = us-west-2
sso_account_id = 111122223333
sso_role_name = SSOReadOnlyRole
region = us-west-2
```

```
output = json
```

Prima di tutto, devi utilizzare the section called "Accedere a una sessione di Centro identità IAM" per richiedere e recuperare le credenziali temporanee.

Per ulteriori informazioni sui file `config` e `credentials`, consulta the section called "Impostazioni di configurazione".

Accedere a una sessione di Centro identità IAM

> **Note**
>
> La procedura di accesso potrebbe richiedere all'utente di consentire l' AWS CLI accesso ai dati. Poiché AWS CLI è basato sull'SDK per Python, i messaggi di autorizzazione possono contenere variazioni del `botocore` nome.

Per recuperare e memorizzare nella cache un set di credenziali per Centro identità IAM, esegui il comando seguente per AWS CLI per aprire il browser predefinito e verificare l'accesso a Centro identità IAM.

```
$ aws sso login --profile my-dev-profile
SSO authorization page has automatically been opened in your default browser.
Follow the instructions in the browser to complete this authorization request.
Successfully logged into Start URL: https://my-sso-portal.awsapps.com/start
```

Le credenziali di sessione di IAM Identity Center vengono memorizzate nella cache e vengono AWS CLI utilizzate per recuperare in modo sicuro AWS le credenziali per il ruolo IAM specificato nel profilo.

Se non riescono ad aprire il browser AWS CLI

Se AWS CLI non è possibile aprire automaticamente il browser, vengono visualizzate le istruzioni per avviare manualmente la procedura di accesso in base al tipo di autorizzazione utilizzato.

PKCE authorization

L'autorizzazione Proof Key for Code Exchange (PKCE) viene utilizzata per impostazione predefinita AWS CLI a partire dalla versione 2.22.0. L'URL visualizzato è un URL univoco che inizia con:

- IPv4: *https://oidc.us-east-1.amazonaws.com/authorize*
- Doppio stack: *https://oidc.us-east-1.api.aws/authorize*

L'autorizzazione PKCE URLs deve essere aperta sullo stesso dispositivo con cui si effettua l'accesso e deve essere utilizzata per un dispositivo dotato di browser.

```
Attempting to automatically open the SSO authorization
 page in your
default browser.
If the browser does not open or you wish to use a
 different device to
authorize the request, open the following URL:

https://oidc.us-east-1.amazonaws.com/authorize?
<abbreviated>
```

Device authorization

L'autorizzazione del dispositivo OAuth 2.0 viene utilizzata AWS CLI per le versioni precedenti alla 2.22.0. Puoi abilitare questo metodo sulle versioni più recenti utilizzando l'opzione `--use-device-code`.

L'autorizzazione del dispositivo URLs non deve essere aperta sullo stesso dispositivo a cui stai effettuando l'accesso e può essere utilizzata per un dispositivo con o senza browser.

```
If the browser does not open or you wish to use a
 different device to
authorize this request, open the following URL:
https://device.sso.us-west-2.amazonaws.com/

Then enter the code:
QCFK-N451
```

Puoi anche specificare quale profilo `sso-session` utilizzare per l'accesso tramite il parametro `--sso-session` del comando `aws sso login`. L'opzione `sso-session` non è disponibile per la configurazione legacy di Centro identità IAM.

```
$ aws sso login --sso-session my-dev-session
```

A partire dalla versione 2.22.0, l'autorizzazione PKCE è l'impostazione predefinita. Per utilizzare l'autorizzazione del dispositivo per l'accesso, aggiungi l'opzione `--use-device-code`.

```
$ aws sso login --profile my-dev-profile --use-device-code
```

Il token di autenticazione viene memorizzato nella cache su disco all'interno della directory `~/.aws/sso/cache` con un nome di file basato su `sso_start_url`.

Eseguire un comando con il profilo Centro identità IAM

Una volta effettuato l'accesso, puoi utilizzare le tue credenziali per richiamare AWS CLI comandi con il profilo denominato associato. L'esempio seguente mostra un comando che utilizza un profilo:

```
$ aws sts get-caller-identity --profile my-dev-profile
```

Se hai effettuato l'accesso a IAM Identity Center e le credenziali memorizzate nella cache non sono scadute, le credenziali scadute vengono rinnovate AWS CLI automaticamente quando necessario. AWS Tuttavia, se le credenziali per Centro identità IAM scadono, devi rinnovarle in modo esplicito accedendo di nuovo all'account Centro identità IAM.

Disconnettersi dalle sessioni in Centro identità IAM

Dopo aver terminato di utilizzare il profilo Centro identità IAM, puoi lasciare scadere le credenziali o eseguire il comando seguente per eliminare le credenziali memorizzate nella cache.

```
$ aws sso logout
Successfully signed out of all SSO profiles.
```

Risoluzione dei problemi

Se riscontri problemi durante l'utilizzo di, consulta la procedura per la risoluzione dei problemi. AWS CLIthe section called "Risoluzione degli errori"

Risorse correlate

Di seguito sono elencate alcune risorse aggiuntive.

- the section called "Concetti sul Centro identità IAM"

- the section called “Tutorial: AWS IAM Identity Center e Amazon S3”
- the section called “Installazione o aggiornamento”
- the section called “Impostazioni di configurazione”
- `aws configure sso` in AWS CLI version 2 Reference
- `aws configure sso-session` in AWS CLI version 2 Reference
- `aws sso login` in AWS CLI version 2 Reference
- `aws sso logout` in AWS CLI version 2 Reference
- Configurazione per l'utilizzo di AWS CLI with CodeCatalyst nella Amazon CodeCatalyst User Guide
- OAuth 2.0 Ambiti di accesso nella Guida per l'utente di IAM Identity Center
- Tutorial introduttivi nella Guida per l'utente del Centro identità IAM

Concetti diAWS IAM Identity Center per la AWS CLI

Questo argomento descrive i concetti chiave di AWS IAM Identity Center (Centro identità IAM). Il Centro identità IAM è un servizio IAM basato sul cloud che semplifica la gestione degli accessi degli utenti su più Account AWS, applicazioni, SDK e strumenti integrandosi con i gestori dell'identità digitale (IdP) esistenti. Consente il single sign-on sicuro, la gestione delle autorizzazioni e il controllo attraverso un portale utenti centralizzato, semplificando la governance delle identità e degli accessi per le organizzazioni.

Argomenti

- Cos'è IAM Identity Center?
- Termini
- Come funziona il Centro identità IAM
- Risorse aggiuntive

Cos'è IAM Identity Center?

Il Centro identità IAM è un servizio di gestione delle identità e degli accessi (IAM) basato sul cloud che consente di gestire centralmente l'accesso a più Account AWS e applicazioni aziendali.

Fornisce un portale utenti in cui gli utenti autorizzati possono accedere alle applicazioni per gli Account AWS che hanno ottenuto l'autorizzazione, utilizzando le credenziali aziendali esistenti. Ciò

consente alle organizzazioni di applicare policy di sicurezza coerenti e semplificare la gestione degli accessi degli utenti.

Indipendentemente dall'IdP utilizzato, il Centro identità IAM astrae queste distinzioni. Ad esempio, è possibile connettere Microsoft Azure AD come descritto nell'articolo del blog The Next Evolution in IAM Identity Center.

> **Note**
>
> Per informazioni sull'utilizzo dell'autenticazione tramite token di connessione, che non impiega alcun ID account e ruolo, consulta Configurazione per l'utilizzo di AWS CLI con CodeCatalyst nella Guida per l'utente di Amazon CodeCatalyst.

Termini

I termini comuni per l'utilizzo del Centro identità IAM sono i seguenti:

Provider di identità

: Un sistema di gestione delle identità come il Centro identità IAM, Microsoft Azure AD, Okta o il tuo servizio di directory aziendale.

AWS IAM Identity Center

: Il Centro identità IAM è il servizio IdP proprietario di AWS. Precedentemente noti come AWS Single Sign-On, gli SDK e gli strumenti mantengono i namespace delle API `sso` per garantire la compatibilità con le versioni precedenti. Per ulteriori informazioni, consulta IAM Identity Center rename nella Guida per l'utente di AWS IAM Identity Center.

URL, URL di avvio SSO, URL di avvio del Portale di accesso AWS

: L'URL univoco del Centro identità IAM della tua organizzazione per accedere agli Account AWS, ai servizi e alle risorse autorizzati.

URL emittente

: L'URL univoco del Centro identità IAM emittente della tua organizzazione per l'accesso programmatico agli Account AWS, ai servizi e alle risorse autorizzati. A partire dalla versione 2.22.0 della AWS CLI, l'URL dell'emittente può essere utilizzato in modo intercambiabile con l'URL iniziale.

Federazione

Il processo di creazione dell'attendibilità tra il Centro identità IAM e un gestore dell'identità digitale per abilitare il Single Sign-On (SSO).

Account AWS

Gli Account AWS per i quali fornisci l'accesso agli utenti tramite AWS IAM Identity Center.

Set di autorizzazioni, credenziali AWS, credenziali sigv4

Raccolte predefinite di autorizzazioni che possono essere assegnate a utenti o gruppi a cui concedere l'accesso agli Servizi AWS.

Ambiti di registrazione, ambiti di accesso, ambiti

Gli ambiti sono un meccanismo di OAuth 2.0 per limitare l'accesso di un'applicazione a un account utente. Un'applicazione può richiedere uno o più ambiti e il token di accesso emesso all'applicazione è limitato agli ambiti consentiti. Per informazioni sugli ambiti, consulta Access scopes nella Guida per l'utente del Centro identità IAM.

Token, token di aggiornamento, token di accesso

I token sono credenziali di sicurezza temporanee che ti vengono rilasciate al momento dell'autenticazione. Questi token contengono informazioni sulla tua identità e sulle autorizzazioni che ti sono state concesse.

Quando accedi a una risorsa o a un'applicazione AWS tramite il portale Centro identità IAM, il token viene sottoposto all'autenticazione e all'autorizzazione di AWS. Ciò consente ad AWS di verificare l'identità dell'utente e avere la certezza di disporre delle autorizzazioni necessarie per eseguire le azioni richieste.

Il token di autenticazione viene memorizzato nella cache su disco all'interno della directory `~/.aws/sso/cache` con un nome di file JSON basato sul nome della sessione.

Sessione

Una sessione del Centro identità IAM si riferisce al periodo di tempo in cui un utente è autenticato e autorizzato ad accedere a risorse o applicazioni AWS. Quando un utente accede al portale del Centro identità IAM, viene stabilita una sessione e il token dell'utente è valido per una durata specificata. Per ulteriori informazioni sull'impostazione della durata delle sessioni, consulta Impostazione della durata della sessione nella Guida per l'utente di AWS IAM Identity Center.

Durante la sessione, puoi passare tra diversi account e applicazioni AWS senza dover effettuare nuovamente l'autenticazione, purché la sessione rimanga attiva. Quando la sessione scade, accedi nuovamente per rinnovare l'accesso.

Le sessioni del Centro identità IAM aiutano a fornire un'esperienza utente senza interruzioni, rafforzando al contempo le best practice di sicurezza limitando la validità delle credenziali di accesso degli utenti.

Concessione del codice di autorizzazione con PKCE, PKCE, Proof Key for Code Exchange

A partire dalla versione 2.22.0, Proof Key for Code Exchange (PKCE) è un flusso di autorizzazione all'autenticazione OAuth 2.0 per dispositivi dotati di browser. Il PKCE è un modo semplice e sicuro per autenticarsi e ottenere il consenso all'accesso alle risorse AWS da desktop e dispositivi mobili con browser web. Questo è il comportamento di autorizzazione per impostazione predefinita. Per ulteriori informazioni su PKCE, consulta Authorization Code Grant with PKCE nella Guida per l'utente di AWS IAM Identity Center.

Concessione dell'autorizzazione a un dispositivo

Un flusso di autorizzazione all'autenticazione OAuth 2.0 per dispositivi con o senza browser web. Per ulteriori informazioni sull'impostazione della durata delle sessioni, consulta Concessione dell'autorizzazione a un dispositivo nella Guida per l'utente di AWS IAM Identity Center.

Come funziona il Centro identità IAM

Il Centro identità IAM si integra con il gestore dell'identità digitale della tua organizzazione, come il Centro identità IAM, Microsoft Azure AD oppure Okta. Gli utenti si autenticano con questo gestore dell'identità digitale e il Centro identità IAM associa quindi tali identità alle autorizzazioni e agli accessi appropriati all'interno dell'ambiente AWS.

Il seguente flusso di lavoro del Centro identità IAM presuppone che tu abbia già configurato la AWS CLI per utilizzare il Centro identità IAM:

1. Nel terminale preferito, esegui il comando `aws sso login`.
2. Accedi al tuo account Portale di accesso AWS per avviare una nuova sessione.
   - Quando avvii una nuova sessione, ricevi un token di aggiornamento e un token di accesso che vengono memorizzati nella cache.
   - Se hai già una sessione attiva, la sessione esistente viene riutilizzata e scade quando scade la sessione esistente.

3. In base al profilo che hai impostato nel tuo file `config`, il Centro identità IAM presuppone i set di autorizzazioni appropriati, garantendo l'accesso agli Account AWS e alle applicazioni pertinenti.
4. La AWS CLI, gli SDK e gli strumenti utilizzano il ruolo IAM che assumi per effettuare chiamate ai Servizi AWS come la creazione di bucket Amazon S3, fino alla scadenza della sessione.
5. Il token di accesso del Centro identità IAM viene controllato ogni ora e aggiornato automaticamente utilizzando il token di aggiornamento.
   - Se il token di accesso è scaduto, l'SDK o lo strumento utilizza il token di aggiornamento per ottenere un nuovo token di accesso. Le durate delle sessioni di questi token vengono quindi confrontate e, se il token di aggiornamento non è scaduto, il Centro identità IAM ne fornisce uno nuovo.
   - Se il token di aggiornamento è scaduto, non vengono forniti nuovi token di accesso e la sessione è terminata.
6. Le sessioni terminano dopo la scadenza dei token di aggiornamento o quando ti disconnetti manualmente utilizzando il comando `aws sso logout`. Le credenziali memorizzate nella cache vengono rimosse. Per continuare ad accedere ai servizi utilizzando il Centro identità IAM, è necessario avviare una nuova sessione con il comando `aws sso login`.

Risorse aggiuntive

Di seguito sono elencate alcune risorse aggiuntive.

- the section called "Autenticazione in Centro identità IAM"
- the section called "Tutorial: AWS IAM Identity Center e Amazon S3"
- the section called "Installazione o aggiornamento"
- the section called "Impostazioni di configurazione"
- `aws configure sso` in AWS CLI version 2 Reference
- `aws configure sso-session` in AWS CLI version 2 Reference
- `aws sso login` in AWS CLI version 2 Reference
- `aws sso logout` in AWS CLI version 2 Reference
- Configurazione per l'utilizzo di AWS CLI con CodeCatalyst nella Guida per l'utente di Amazon CodeCatalyst
- IAM Identity Center rename nella Guida per l'utente di AWS IAM Identity Center
- Ambiti di accesso OAuth 2.0 nella Guida per l'utente del Centro identità IAM

- Set session duration nella Guida per l'utente di AWS IAM Identity Center
- Tutorial introduttivi nella Guida per l'utente del Centro identità IAM

Tutorial: Utilizzo di IAM Identity Center per eseguire i comandi Amazon S3 nel AWS CLI

Questo argomento descrive come configurare l'autenticazione degli utenti con la AWS CLI versione corrente AWS IAM Identity Center (IAM Identity Center) per recuperare le credenziali per i comandi run AWS Command Line Interface (AWS CLI) per (Amazon Amazon Simple Storage Service S3).

Argomenti

- Fase 1: autenticazione tramite il Centro identità IAM
- Fase 2: raccogliere le informazioni del Centro identità IAM
- Fase 3: creazione dei bucket Amazon S3
- Passaggio 4: installa il AWS CLI
- Fase 5: configurazione del profilo della AWS CLI
- Fase 6: accesso al Centro identità IAM
- Fase 7: esecuzione dei comandi Amazon S3
- Fase 8: disconnessione dal Centro identità IAM
- Fase 9: eliminazione delle risorse
- Risoluzione dei problemi
- Risorse aggiuntive

Fase 1: autenticazione tramite il Centro identità IAM

Ottieni l'accesso all'autenticazione SSO con il Centro identità IAM. Scegli uno dei seguenti metodi per accedere alle tue credenziali. AWS

Non ho stabilito l'accesso tramite Centro identità IAM

Segui le istruzioni riportate in Nozioni di base nella Guida per l'utente di AWS IAM Identity Center . Questo processo attiva Centro identità IAM, crea un utente amministratore e aggiunge un set di autorizzazioni con privilegi minimi appropriato.

> **Note**
>
> Crea un set di autorizzazioni per l'applicazione di autorizzazioni con privilegio minimo. Consigliamo di utilizzare il set di autorizzazioni `PowerUserAccess` predefinito, a meno che il datore di lavoro non ne abbia creato uno personalizzato a questo scopo.

Esci dal portale e accedi nuovamente per visualizzare i Account AWS dettagli di accesso programmatici e le opzioni per `Administrator` o. `PowerUserAccess` Seleziona `PowerUserAccess` quando utilizzi l'SDK.

Ho già accesso AWS tramite un provider di identità federato gestito dal mio datore di lavoro (come Azure AD o Okta)

Accedi AWS tramite il portale del tuo provider di identità. Se il tuo amministratore cloud ti ha concesso le autorizzazioni `PowerUserAccess` (sviluppatore), vedi quelle a Account AWS cui hai accesso e il tuo set di autorizzazioni. Accanto al nome del set di autorizzazioni, vengono visualizzate le opzioni per accedere agli account manualmente o a livello di programmazione utilizzando il set di autorizzazioni.

Le implementazioni personalizzate possono dare luogo a esperienze diverse, ad esempio nomi di set di autorizzazioni diversi. In caso di dubbi su quale set di autorizzazioni utilizzare, contatta il team IT per assistenza.

Ho già accesso AWS tramite il portale di AWS accesso gestito dal mio datore di lavoro

Accedi AWS tramite il tuo portale di AWS accesso. Se il tuo amministratore cloud ti ha concesso le autorizzazioni `PowerUserAccess` (sviluppatore), vedi quelle a Account AWS cui hai accesso e il tuo set di autorizzazioni. Accanto al nome del set di autorizzazioni, vengono visualizzate le opzioni per accedere agli account manualmente o a livello di programmazione utilizzando il set di autorizzazioni.

Ho già accesso AWS tramite un provider di identità personalizzato federato gestito dal mio datore di lavoro

Contatta il team IT per assistenza.

Fase 2: raccogliere le informazioni del Centro identità IAM

Dopo aver ottenuto l'accesso a AWS, raccogli le informazioni del tuo IAM Identity Center effettuando le seguenti operazioni:

1. Recupera i valori `SSO Start URL` e `SSO Region` necessari per eseguire `aws configure sso`

   a. Nel tuo portale di AWS accesso, seleziona il set di autorizzazioni che utilizzi per lo sviluppo e seleziona il link Access keys.

   b. Nella finestra di dialogo Ottieni le credenziali seleziona la scheda corrispondente al tuo sistema operativo.

   c. Scegli il metodo Credenziali Centro identità IAM per ottenere i valori `SSO Start URL` e `SSO Region`.

2. In alternativa, a partire dalla versione 2.22.0 puoi utilizzare l'URL del nuovo emittente anziché l'URL di avvio. L'URL dell'emittente si trova nella AWS IAM Identity Center console in una delle seguenti posizioni:
   - Nella pagina Dashboard l'URL dell'emittente si trova nel riepilogo delle impostazioni.
   - Nella pagina Impostazioni l'URL dell'emittente si trova nelle impostazioni in Origine identità.
3. Per informazioni sul valore degli ambiti da registrare, consulta OAuth 2.0 Access scopes nella Guida per l'utente di IAM Identity Center.

Fase 3: creazione dei bucket Amazon S3

Accedi a Console di gestione AWS e apri la console Amazon S3 all'indirizzo. https://console.aws.amazon.com/s3/

Per questo tutorial, crea alcuni bucket da recuperare successivamente in un elenco.

Passaggio 4: installa il AWS CLI

Installa AWS CLI le seguenti istruzioni per il tuo sistema operativo. Per ulteriori informazioni, consulta the section called "Installazione o aggiornamento".

Una volta installato, puoi verificare l'installazione aprendo il tuo terminale preferito ed eseguendo il seguente comando. Dovrebbe essere visualizzata la versione installata di AWS CLI.

```
$ aws --version
```

Fase 5: configurazione del profilo della AWS CLI

Configurare il profilo utilizzando uno dei seguenti metodi

Configura il tuo profilo con la procedura guidata **`aws configure sso`**

La `sso-session` sezione del `config` file viene utilizzata per raggruppare le variabili di configurazione per l'acquisizione di token di accesso SSO, che possono quindi essere utilizzati per acquisire le credenziali. AWS Vengono utilizzate le impostazioni seguenti:

- (Obbligatorio) `sso_start_url`
- (Obbligatorio) `sso_region`
- `sso_account_id`
- `sso_role_name`
- `sso_registration_scopes`

È necessario definire una sezione `sso-session` e associarla a un profilo. Le impostazioni `sso_region` e `sso_start_url` devono essere configurate all'interno della sezione `sso-session`. In genere, le impostazioni `sso_account_id` e `sso_role_name` devono essere configurate nella sezione `profile` in modo che l'SDK possa richiedere le credenziali SSO.

L'esempio seguente configura l'SDK in modo da richiedere le credenziali SSO e supporta l'aggiornamento automatico dei token:

```
$ aws configure sso
SSO session name (Recommended): my-sso
SSO start URL [None]: https://my-sso-portal.awsapps.com/start
SSO region [None]: us-east-1
SSO registration scopes [None]: sso:account:access
```

Per il supporto dual-stack, puoi utilizzare il formato URL di avvio SSO dual-stack:

```
$ aws configure sso
SSO session name (Recommended): my-sso
SSO start URL [None]: https://ssoins-1234567890abcdef.portal.us-east-1.app.aws
SSO region [None]: us-east-1
SSO registration scopes [None]: sso:account:access
```

Per impostazione predefinita, viene utilizzata l'autorizzazione Proof Key for Code Exchange (PKCE) per AWS CLI a partire dalla versione 2.22.0 e questa deve essere utilizzata su dispositivi dotati di browser. Per continuare a utilizzare l'autorizzazione del dispositivo, aggiungi l'opzione `--use-device-code`.

```
$ aws configure sso --use-device-code
```

Configurazione manuale utilizzando il file **`config`**

La `sso-session` sezione del `config` file viene utilizzata per raggruppare le variabili di configurazione per l'acquisizione di token di accesso SSO, che possono quindi essere utilizzati per acquisire le credenziali. AWS Vengono utilizzate le impostazioni seguenti:

- (Obbligatorio) `sso_start_url`
- (Richiesto) `sso_region`
- `sso_account_id`
- `sso_role_name`
- `sso_registration_scopes`

È necessario definire una sezione `sso-session` e associarla a un profilo. `sso_region` e `sso_start_url` devono essere impostati all'interno della sezione `sso-session`. In genere, le impostazioni `sso_account_id` e `sso_role_name` devono essere configurate nella sezione `profile` in modo che l'SDK possa richiedere le credenziali SSO.

L'esempio seguente configura l'SDK in modo da richiedere le credenziali SSO e supporta l'aggiornamento automatico dei token:

```
[profile my-dev-profile]
sso_session = my-sso
sso_account_id = 111122223333
sso_role_name = SampleRole

[sso-session my-sso]
sso_region = us-east-1
sso_start_url = https://my-sso-portal.awsapps.com/start
sso_registration_scopes = sso:account:access
```

Per il supporto dual-stack, usa il formato URL di avvio SSO dual-stack:

```
[profile my-dev-profile]
sso_session = my-sso
sso_account_id = 111122223333
sso_role_name = SampleRole
```

```
[sso-session my-sso]
sso_region = us-east-1
sso_start_url = https://ssoins-1234567890abcdef.portal.us-east-1.app.aws
sso_registration_scopes = sso:account:access
```

Il token di autenticazione viene memorizzato nella cache su disco all'interno della directory `~/.aws/sso/cache` con un nome di file basato sul nome della sessione.

Fase 6: accesso al Centro identità IAM

> **Note**
>
> La procedura di accesso potrebbe richiedere all'utente di consentire l'accesso ai dati. AWS CLI Poiché AWS CLI è basato sull'SDK per Python, i messaggi di autorizzazione possono contenere variazioni del `botocore` nome.

Per recuperare e memorizzare nella cache le credenziali per il Centro identità IAM, esegui il comando seguente per AWS CLI per aprire il browser predefinito e verificare l'accesso al Centro identità IAM.

```
$ aws sso login --profile my-dev-profile
```

A partire dalla versione 2.22.0, l'autorizzazione PKCE è l'impostazione predefinita. Per utilizzare l'autorizzazione del dispositivo per l'accesso, aggiungi l'opzione `--use-device-code`.

```
$ aws sso login --profile my-dev-profile --use-device-code
```

Fase 7: esecuzione dei comandi Amazon S3

Per elencare i bucket che hai creato in precedenza, utilizza il comando `aws s3 ls`. L'esempio seguente elenca tutti i tuoi bucket Amazon S3.

```
$ aws s3 ls
2018-12-11 17:08:50 my-bucket
2018-12-14 14:55:44 my-bucket2
```

Fase 8: disconnessione dal Centro identità IAM

Dopo aver terminato di utilizzare il profilo Centro identità IAM, esegui il comando seguente per eliminare le credenziali memorizzate nella cache.

```
$ aws sso logout
Successfully signed out of all SSO profiles.
```

Fase 9: eliminazione delle risorse

Al termine di questo tutorial, esegui la pulizia di tutte le risorse che hai creato durante questo tutorial che non ti servono più, inclusi i bucket Amazon S3.

Risoluzione dei problemi

Se riscontri problemi con l'utilizzo di AWS CLI, consulta le procedure comuni the section called "Risoluzione degli errori" per la risoluzione dei problemi.

Risorse aggiuntive

Di seguito sono elencate alcune risorse aggiuntive.

- the section called "Concetti sul Centro identità IAM"
- the section called "Autenticazione in Centro identità IAM"
- the section called "Installazione o aggiornamento"
- the section called "Impostazioni di configurazione"
- `aws configure sso` in AWS CLI version 2 Reference
- `aws configure sso-session` in AWS CLI version 2 Reference
- `aws sso login` in AWS CLI version 2 Reference
- `aws sso logout` in AWS CLI version 2 Reference
- Configurazione per l'utilizzo di AWS CLI with CodeCatalyst nella Amazon CodeCatalyst User Guide
- OAuth 2.0 Ambiti di accesso nella Guida per l'utente di IAM Identity Center
- Tutorial introduttivi nella Guida per l'utente del Centro identità IAM

Autenticazione con credenziali a breve termine per la AWS CLI

Ti suggeriamo di configurare l'SDK o lo strumento per utilizzare l'autenticazione Centro identità IAM con opzioni di durata della sessione estesa. Tuttavia, puoi copiare e utilizzare le credenziali temporanee disponibili nel portale di accesso AWS. Le nuove credenziali dovranno essere copiate

quando scadono. È possibile utilizzare le credenziali temporanee in un profilo o utilizzarle come valori per le proprietà di sistema e le variabili di ambiente.

1. Accedi al portale di accesso AWS.
2. Segui queste istruzioni per copiare le credenziali del ruolo IAM dal portale di accesso AWS.
    1. Per la fase 2 delle istruzioni collegate, scegli l'account AWS e il nome del ruolo IAM che garantiscono l'accesso per le tue esigenze di sviluppo. In genere, questo ruolo presenta un nome come PowerUserAccess o Developer.
    2. Per la fase 4, seleziona l'opzione Aggiungi un profilo al file delle credenziali AWS e copia il contenuto.
3. Crea o apri il file delle credenziali `credentials` condiviso. Questo file è denominato `~/.aws/credentials` su sistemi Linux e macOS e `%USERPROFILE%\.aws\credentials` su Windows. Per ulteriori informazioni, consulta the section called "Impostazioni di configurazione".
4. Aggiungi il testo seguente al file `credentials` condiviso. Sostituisci i valori di esempio con le credenziali che hai copiato.

```
[default]
aws_access_key_id = AKIAIOSFODNN7EXAMPLE
aws_secret_access_key = wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY
aws_session_token =
 IQoJb3JpZ2luX2IQoJb3JpZ2luX2IQoJb3JpZ2luX2IQoJb3JpZ2luX2IQoJb3JpZVERYLONGSTRINGEXAMPLE
```

5. Aggiungi la Regione e il formato predefiniti preferiti al file `config` condiviso.

```
[default]
region=us-west-2
output=json

[profile user1]
region=us-east-1
output=text
```

Quando l'SDK crea un client di servizio, accederà a queste credenziali temporanee e le utilizzerà per ogni richiesta. Le impostazioni per il ruolo IAM scelto nella fase 2a determinano per quanto tempo le credenziali temporanee sono valide. La durata massima è di dodici ore.

Ripeti queste fasi ogni volta che scadono le tue credenziali.

Utilizzo di un ruolo IAM in AWS CLI

Un ruolo AWS Identity and Access Management (IAM) è uno strumento di autorizzazione che consente a un utente di ottenere autorizzazioni aggiuntive (o diverse) o di ottenere le autorizzazioni per eseguire azioni in un account diverso AWS .

Argomenti

- Prerequisiti
- Panoramica sull'utilizzo dei ruoli IAM
- Configurazione e utilizzo di un ruolo
- Utilizzo dell'autenticazione a più fattori
- Ruoli per più account e ID esterno
- Specifica di un nome di sessione del ruolo per semplificare l'audit
- Assunzione di un ruolo con un'identità Web
- Cancellazione delle credenziali nella cache

Prerequisiti

Per eseguire i comandi `iam`, è necessario installare e configurare la AWS CLI. Questo include l'impostazione di un profilo configurato, ad esempio presupponendo che un ruolo sia associato a un altro metodo di credenziali. Per ulteriori informazioni, consulta the section called "Installazione o aggiornamento".

Panoramica sull'utilizzo dei ruoli IAM

Puoi configurare AWS Command Line Interface (AWS CLI) per utilizzare un ruolo IAM definendo un profilo per il ruolo nel file. `~/.aws/config`

L'esempio seguente mostra un profilo del ruolo denominato `marketingadmin`. Se esegui comandi con `--profile marketingadmin` (o li specifichi con la variabile di AWS_PROFILE ambiente), AWS CLI utilizza le credenziali definite in un profilo separato `user1` per assumere il ruolo con Amazon Resource Name (`arn:aws:iam::123456789012:role/marketingadminrole`ARN). Puoi eseguire tutte le operazioni consentite dalle autorizzazioni assegnate a tale ruolo.

```
[profile marketingadmin]
```

```
role_arn = arn:aws:iam::123456789012:role/marketingadminrole
source_profile = user1
```

Puoi specificare un elemento `source_profile` che punti a un profilo denominato separato contenente le credenziali utente con l'autorizzazione per utilizzare il ruolo. Nell'esempio precedente il profilo `marketingadmin` utilizza le credenziali del profilo `user1`. Quando specifichi che un AWS CLI comando deve utilizzare il profilo`marketingadmin`, cerca AWS CLI automaticamente le credenziali per il `user1` profilo collegato e le usa per richiedere credenziali temporanee per il ruolo IAM specificato. La CLI utilizza l'AssumeRoleoperazione sts: in background per eseguire questa operazione. Tali credenziali temporanee vengono quindi utilizzate per eseguire il comando AWS CLI richiesto. Al ruolo specificato devono essere associate politiche di autorizzazione IAM che consentano l'esecuzione del AWS CLI comando richiesto.

Per eseguire un AWS CLI comando da un'istanza Amazon Elastic Compute Cloud (Amazon EC2) o da un contenitore Amazon Elastic Container Service (Amazon ECS), puoi utilizzare un ruolo IAM collegato al profilo dell'istanza o al contenitore. Se non specifichi alcun profilo o non imposti variabili di ambiente, tale ruolo viene utilizzato direttamente. In questo modo, potrai evitare di archiviare chiavi di accesso di lunga durata nelle istanze. Puoi anche utilizzare i ruoli dell'istanza o del container solo per ottenere le credenziali per un altro ruolo. A tale scopo, usa `credential_source` (anziché `source_profile`) per specificare come trovare le credenziali. L'attributo `credential_source` supporta i seguenti valori:

- `Environment`: recupera le credenziali di origine dalle variabili di ambiente.
- `Ec2InstanceMetadata`— Utilizza il ruolo IAM collegato al profilo dell' EC2istanza Amazon.
- `EcsContainer`: utilizza il ruolo IAM collegato al container Amazon ECS.

L'esempio seguente mostra lo stesso `marketingadminrole` ruolo utilizzato facendo riferimento a un profilo di EC2 istanza Amazon.

```
[profile marketingadmin]
role_arn = arn:aws:iam::123456789012:role/marketingadminrole
credential_source = Ec2InstanceMetadata
```

Quando si richiama un ruolo, sono disponibili opzioni aggiuntive che è possibile richiedere, ad esempio l'utilizzo dell'autenticazione a più fattori e di un ID esterno (utilizzato da società terze per accedere alle risorse dei client). Puoi anche specificare nomi di sessione di ruolo univoci che possono essere controllati più facilmente nei AWS CloudTrail log.

Configurazione e utilizzo di un ruolo

Quando esegui comandi utilizzando un profilo che specifica un ruolo IAM, AWS CLI utilizza le credenziali del profilo di origine per chiamare AWS Security Token Service (AWS STS) e richiedere credenziali temporanee per il ruolo specificato. L'utente nel profilo di origine deve disporre dell'autorizzazione per chiamare `sts:assume-role` per il ruolo nel profilo specificato. Il ruolo deve disporre di una relazione di trust che consente all'utente nel profilo di origine di usare il ruolo. Il processo di recupero e quindi di utilizzo delle credenziali temporanee per un ruolo è spesso definito come assunzione del ruolo.

Puoi creare un ruolo in IAM con le autorizzazioni che gli utenti devono assumere seguendo la procedura descritta in Creating a Role to Delegate Permissions to an IAM user nella Guida per l'utente di AWS Identity and Access Management . Se il ruolo e l'utente del profilo di origine appartengono allo stesso account, puoi immettere il tuo ID account durante la configurazione della relazione di trust del ruolo.

Dopo aver creato il ruolo, modifica la relazione di fiducia per consentire all'utente di assumerlo.

L'esempio seguente mostra una policy di trust che può essere collegata a un ruolo. La policy fa in modo che il ruolo possa essere assunto da qualsiasi utente nell'account 123456789012, se l'amministratore di tale account concede esplicitamente l'autorizzazione `sts:AssumeRole` all'utente.

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::123456789012:root"
      },
      "Action": "sts:AssumeRole"
    }
  ]
}
```

La policy di trust non concede attualmente autorizzazioni. L'amministratore dell'account deve delegare l'autorizzazione per assumere il ruolo a singoli utenti collegando una policy con le

autorizzazioni appropriate. Nell'esempio seguente viene illustrata una policy che puoi collegare a un utente e che permette a tale utente di assumere solo il ruolo `marketingadminrole`. Per ulteriori informazioni su come concedere a un utente l'accesso per assumere un ruolo, consulta Concessione di autorizzazioni agli utenti per il cambio di ruoli nella Guida per l'utente di IAM.

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "sts:AssumeRole",
      "Resource": "arn:aws:iam::123456789012:role/marketingadminrole"
    }
  ]
}
```

L'utente non deve disporre di autorizzazioni aggiuntive per eseguire AWS CLI i comandi utilizzando il profilo del ruolo. Al contrario, le autorizzazioni per eseguire il comando provengono da quelle associate al ruolo. Si allegano politiche di autorizzazione al ruolo per specificare quali azioni possono essere eseguite su quali AWS risorse. Per ulteriori informazioni su come collegare le autorizzazioni a un ruolo, il cui funzionamento è del tutto identico a quello di un utente, consulta Modifica delle autorizzazioni per un utente IAM nella Guida per l'utente di IAM.

Ora che il profilo del ruolo, le autorizzazioni del ruolo, la relazione di trust del ruolo e le autorizzazioni utente sono configurate correttamente, puoi utilizzare il ruolo nella riga di comando richiamando l'opzione `--profile`. Ad esempio, il comando seguente chiama il comando `ls` di Amazon S3 utilizzando le autorizzazioni collegate al ruolo `marketingadmin`, come definito dall'esempio all'inizio di questo argomento.

```
$ aws s3 ls --profile marketingadmin
```

Per utilizzare il ruolo per diverse chiamate, puoi impostare la variabile di ambiente `AWS_PROFILE` per la sessione corrente dalla riga di comando. Sebbene la variabile di ambiente sia definita, non è necessario specificare l'opzione `--profile` su ogni comando.

Linux o macOS

```
$ export AWS_PROFILE=marketingadmin
```

Windows

```
C:\> setx AWS_PROFILE marketingadmin
```

Per ulteriori informazioni sulla configurazione di utenti e ruoli, consulta Identità IAM (utenti, gruppi e ruoli) e Ruoli IAM nella Guida per l'utente di IAM.

Utilizzo dell'autenticazione a più fattori

Per una maggiore sicurezza, puoi richiedere agli utenti di fornire una chiave monouso generata da un dispositivo di autenticazione a più fattori (MFA), da un dispositivo U2F o da un'app per dispositivi mobili durante il tentativo di effettuare una chiamata utilizzando il profilo del ruolo.

In primo luogo, puoi scegliere di modificare la relazione di trust sul ruolo IAM per richiedere la MFA. Ciò impedisce a chiunque di utilizzare il ruolo senza prima autenticarsi utilizzando MFA. Per un esempio, consulta la riga `Condition` nel seguente esempio. Questa policy consente all'utente denominato `anika` di assumere il ruolo collegato alla policy, ma solo se esegue l'autenticazione utilizzando MFA.

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Sid": "",
      "Effect": "Allow",
      "Principal": { "AWS": "arn:aws:iam::123456789012:user/anika" },
      "Action": "sts:AssumeRole",
      "Condition": { "Bool": { "aws:multifactorAuthPresent": true } }
    }
  ]
}
```

Quindi, aggiungi una riga al profilo del ruolo che specifica l'ARN del dispositivo MFA dell'utente. Le seguenti voci del file `config` di esempio illustrano due profili di ruolo che utilizzano le chiavi di

accesso per l'utente `anika` per richiedere le credenziali provvisorie per il ruolo `cli-role`. L'utente `anika` ha le autorizzazioni per assumere quel ruolo concesse dalla policy di trust del ruolo.

```
[profile role-without-mfa]
region = us-west-2
role_arn= arn:aws:iam::128716708097:role/cli-role
source_profile=cli-user

[profile role-with-mfa]
region = us-west-2
role_arn= arn:aws:iam::128716708097:role/cli-role
source_profile = cli-user
mfa_serial = arn:aws:iam::128716708097:mfa/cli-user

[profile cli-user]
region = us-west-2
output = json
```

L'impostazione `mfa_serial` può richiedere un ARN, come illustrato, o il numero di serie di un token hardware MFA.

Il primo profilo, `role-without-mfa`, non richiede MFA. Tuttavia, poiché la precedente policy di trust di esempio associata al ruolo richiede MFA, qualsiasi tentativo di eseguire un comando con questo profilo non riesce.

```
$ aws iam list-users --profile role-without-mfa

An error occurred (AccessDenied) when calling the AssumeRole operation: Access denied
```

Il secondo profilo `role-with-mfa`, identifica un dispositivo MFA da utilizzare. Quando l'utente tenta di eseguire un AWS CLI comando con questo profilo, AWS CLI richiede all'utente di inserire la password monouso (OTP) fornita dal dispositivo MFA. Se l'autenticazione MFA ha esito positivo, il comando esegue l'operazione richiesta. La password monouso non viene visualizzata sullo schermo.

```
$ aws iam list-users --profile role-with-mfa
Enter MFA code for arn:aws:iam::123456789012:mfa/cli-user:
{
    "Users": [
        {
            ...
```

Ruoli per più account e ID esterno

Puoi abilitare gli utenti in modo che assumano ruoli appartenenti ad account diversi configurando il ruolo come ruolo per più account. Durante la creazione del ruolo, imposta il tipo di ruolo su Another AWS account, come descritto in Creating a Role to Delegate Permissions to an IAM user. Facoltativamente, seleziona Require MFA (Richiedi MFA). Require MFA (Richiedi MFA) configura la condizione appropriata nella relazione di trust, come descritto in Utilizzo dell'autenticazione a più fattori.

Se utilizzi un ID esterno per fornire ulteriore controllo su chi può usare un ruolo per diversi account, devi aggiungere il parametro `external_id` al profilo del ruolo. Ciò viene utilizzato in genere solo quando l'altro account è controllato da qualcuno esterno all'azienda o organizzazione.

```
[profile crossaccountrole]
role_arn = arn:aws:iam::234567890123:role/SomeRole
source_profile = default
mfa_serial = arn:aws:iam::123456789012:mfa/saanvi
external_id = 123456
```

Specifica di un nome di sessione del ruolo per semplificare l'audit

Quando molti individui condividono un ruolo, l'audit diventa più difficile. Si supponga che tu voglia associare ogni operazione richiamata alla persona che ha richiamato l'operazione. Tuttavia, quando la persona utilizza un ruolo, l'assunzione del ruolo da parte dell'individuo è un'operazione distinta rispetto alla chiamata di un'operazione ed è pertanto necessario correlare manualmente le due cose.

Puoi semplificare questa operazione specificando nomi univoci delle sessioni del ruolo quando gli utenti assumono un ruolo. A tale scopo, aggiungi un parametro `role_session_name` a ogni profilo denominato nel file `config` che specifica un ruolo. Il valore `role_session_name` viene passato all'operazione `AssumeRole` e diventa parte dell'ARN della sessione del ruolo. È inoltre inclusa nei AWS CloudTrail registri di tutte le operazioni registrate.

Ad esempio, puoi creare un profilo basato su ruoli come segue.

```
[profile namedsessionrole]
role_arn = arn:aws:iam::234567890123:role/SomeRole
source_profile = default
role_session_name = Session_Maria_Garcia
```

Di conseguenza, la sessione del ruolo ha il seguente ARN.

```
arn:aws:iam::234567890123:assumed-role/SomeRole/Session_Maria_Garcia
```

Inoltre, tutti i AWS CloudTrail registri includono il nome della sessione di ruolo nelle informazioni acquisite per ogni operazione.

Assunzione di un ruolo con un'identità Web

È possibile configurare un profilo per indicare che AWS CLI devono assumere un ruolo utilizzando la federazione delle identità Web e Open ID Connect (OIDC). Quando lo specificate in un profilo, effettua AWS CLI automaticamente la AWS STS `AssumeRoleWithWebIdentity` chiamata corrispondente per voi.

> **Note**
>
> Quando specificate un profilo che utilizza un ruolo IAM, AWS CLI effettua le chiamate appropriate per recuperare le credenziali temporanee. Queste credenziali sono archiviate in `~/.aws/cli/cache`. AWS CLI I comandi successivi che specificano lo stesso profilo utilizzano le credenziali temporanee memorizzate nella cache fino alla scadenza. A quel punto, aggiorna AWS CLI automaticamente le credenziali.

Per recuperare e utilizzare le credenziali temporanee utilizzando la federazione delle identità Web, puoi specificare i seguenti valori di configurazione in un profilo condiviso.

role_arn

Specificare l'ARN del ruolo da assumere.

web_identity_token_file

Speciifica il percorso di un file che contiene un token di accesso OAuth 2.0 o un token OpenID Connect ID fornito dal provider di identità. L' AWS CLI carica questo file e passa il relativo contenuto come argomento `WebIdentityToken` all'operazione `AssumeRoleWithWebIdentity`.

role_session_name

Specifica un nome facoltativo applicato a questa sessione assume-role.

Di seguito è riportato un esempio di configurazione per la quantità minima di configurazione necessaria per configurare un profilo del ruolo assunto con un'identità Web:

```
# In ~/.aws/config

[profile web-identity]
role_arn=arn:aws:iam:123456789012:role/RoleNameToAssume
web_identity_token_file=/path/to/a/token
```

Puoi anche fornire questa configurazione utilizzando variabili di ambiente:

AWS_ROLE_ARN

L'ARN del ruolo da assumere.

AWS_WEBFILE_TOKEN DI _IDENTITÀ

Il percorso del file del token dell'identità Web.

AWS_ROLE_NOME_SESSIONE

Il nome applicato a questa sessione assume-role.

> **Note**
>
> Queste variabili di ambiente attualmente si applicano solo al ruolo assumere con provider di identità Web. Non si applicano alla configurazione generale del provider dell'assunzione di ruoli.

Cancellazione delle credenziali nella cache

Quando si utilizza un ruolo, memorizza nella AWS CLI cache locale le credenziali temporanee fino alla loro scadenza. La prossima volta che provi a utilizzarle, AWS CLI tenta di rinnovarle per tuo conto.

Se le credenziali temporanee del ruolo vengono revocate, non vengono rinnovate automaticamente e i tentativi di utilizzarle non vanno a buon fine. Tuttavia, puoi eliminare la cache per AWS CLI forzare il recupero di nuove credenziali.

Linux o macOS

```
$ rm -r ~/.aws/cli/cache
```

Windows

```
C:\> del /s /q %UserProfile%\.aws\cli\cache
```

Autenticazione tramite credenziali utente IAM per AWS CLI

> **Warning**
>
> Per evitare rischi per la sicurezza, non utilizzare gli utenti IAM per l'autenticazione quando sviluppi software creato ad hoc o lavori con dati reali. Utilizza invece la federazione con un provider di identità come AWS IAM Identity Center.

In questa sezione viene descritto come configurare le impostazioni di base con un utente IAM. Queste includono le credenziali di sicurezza che utilizzano i file `credentials` e `config`. Per vedere invece le istruzioni di configurazione per AWS IAM Identity Center, consulta the section called "Autenticazione in Centro identità IAM".

Argomenti

- Fase 1: creare un utente IAM
- Fase 2: ottenere le chiavi di accesso
- Passaggio 3: configura il AWS CLI
  - Uso di aws configure
  - Importazione delle chiavi di accesso tramite un file .CSV
  - Modifica diretta dei file credentials e config
- (Facoltativo) Utilizzo dell'autenticazione a più fattori con le credenziali utente IAM

Fase 1: creare un utente IAM

Crea il tuo utente IAM seguendo la procedura Creazione di utenti IAM (console) nella Guida per l'utente di IAM.

- Per le opzioni di autorizzazione, scegli Collega direttamente le policy esistenti in base al modo in cui desideri assegnare le autorizzazioni a questo utente.
- La maggior parte dei tutorial SDK "Nozioni di base" utilizza il servizio Amazon S3 come esempio. Per fornire alla tua applicazione l'accesso completo ad Amazon S3, seleziona la policy `AmazonS3FullAccess` da allegare a questo utente.

Fase 2: ottenere le chiavi di accesso

1. Accedi Console di gestione AWS e apri la console IAM all'indirizzo https://console.aws.amazon.com/iam/.
2. Nel riquadro di navigazione della console IAM, seleziona Utenti, quindi seleziona il **`User name`** dell'utente che hai creato in precedenza.
3. Nella pagina dell'utente, scegli la scheda Credenziali di sicurezza. Quindi, in Chiavi di accesso, seleziona Crea chiave di accesso.
4. Per Crea chiave di accesso (fase 1), scegli Interfaccia a riga di comando (CLI).
5. Per Crea chiave di accesso (fase 2), inserisci facoltativamente un tag e scegli Avanti.
6. Per Crea chiave di accesso (fase 3), seleziona Scarica il file.csv per salvare un file `.csv` con la chiave di accesso e la chiave di accesso segreta dell'utente IAM. Queste informazioni serviranno in seguito.
7. Seleziona Done (Fatto)

Passaggio 3: configura il AWS CLI

Per uso generale, sono AWS CLI necessarie le seguenti informazioni:

- ID chiave di accesso
- Chiave di accesso segreta
- AWS Regione
- Formato di output

AWS CLI Memorizza queste informazioni in un profilo (una raccolta di impostazioni) denominato `default` nel `credentials` file. Per impostazione predefinita, le informazioni in questo profilo vengono utilizzate quando si esegue un AWS CLI comando che non specifica esplicitamente un

profilo da utilizzare. Per ulteriori informazioni sul file `credentials`, consulta Impostazioni dei file di configurazione e delle credenziali nella AWS CLI.

Per configurare AWS CLI, utilizzare una delle seguenti procedure:

Argomenti

- Uso di aws configure
- Importazione delle chiavi di accesso tramite un file .CSV
- Modifica diretta dei file credentials e config

Uso di **`aws configure`**

Per uso generale, il `aws configure` comando è il modo più veloce per configurare AWS CLI l'installazione. Questa procedura guidata di configurazione richiede ogni informazione necessaria per iniziare. Se non diversamente specificato utilizzando l'`--profile`opzione, AWS CLI memorizza queste informazioni nel `default` profilo.

L'esempio seguente configura un profilo `default` utilizzando valori di esempio. Sostituirli con i propri valori come descritto nelle sezioni seguenti.

```
$ aws configure
AWS Access Key ID [None]: AKIAIOSFODNN7EXAMPLE
AWS Secret Access Key [None]: wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY
Default region name [None]: us-west-2
Default output format [None]: json
```

L'esempio seguente configura un profilo denominato `userprod` utilizzando valori di esempio. Sostituirli con i propri valori come descritto nelle sezioni seguenti.

```
$ aws configure --profile userprod
AWS Access Key ID [None]: AKIAIOSFODNN7EXAMPLE
AWS Secret Access Key [None]: wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY
Default region name [None]: us-west-2
Default output format [None]: json
```

Importazione delle chiavi di accesso tramite un file .CSV

Invece di utilizzare `aws configure` per inserire le chiavi di accesso, puoi importare il file di testo semplice `.csv` che hai scaricato dopo aver creato le chiavi di accesso.

Il file `.csv` deve contenere le seguenti intestazioni.

- Nome utente: questa colonna deve essere aggiunta al tuo `.csv`. Viene utilizzato per creare il nome del profilo presente nei file `credentials` e `config` durante l'importazione.
- ID chiave di accesso
- Chiave di accesso segreta

> **Note**
>
> Durante la creazione iniziale delle chiavi di accesso, una volta chiusa la finestra di dialogo Scarica il file.csv, non è possibile accedere alla chiave di accesso segreta dopo aver chiuso la finestra di dialogo. Se hai bisogno di un file `.csv`, dovrai crearne uno personalmente con le intestazioni richieste e le informazioni sulle chiavi di accesso memorizzate. Se non hai accesso alle informazioni sulle chiavi di accesso, devi crearne di nuove.

Per importare il file `.csv`, utilizza il comando `aws configure import` con l'opzione `--csv` come segue:

```
$ aws configure import --csv file://credentials.csv
```

Per ulteriori informazioni, consulta `aws_configure_import`.

Modifica diretta dei file **`credentials`** e **`config`**

Per modificare direttamente i file `credentials` e `config`, effettua le seguenti operazioni.

1. Crea o apri il file delle credenziali AWS `credentials` condiviso. Questo file è denominato `~/.aws/credentials` su sistemi Linux e macOS e `%USERPROFILE%\.aws\credentials` su Windows. Per ulteriori informazioni, consulta the section called "Impostazioni di configurazione".
2. Aggiungi il testo seguente al file `credentials` condiviso. Sostituisci i valori di esempio nel file `.csv` scaricato in precedenza e salva il file.

   ```
   [default]
   aws_access_key_id = AKIAIOSFODNN7EXAMPLE
   aws_secret_access_key = wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY
   ```

(Facoltativo) Utilizzo dell'autenticazione a più fattori con le credenziali utente IAM

Per una maggiore sicurezza, puoi utilizzare una chiave monouso generata da un dispositivo di autenticazione a più fattori (MFA), un dispositivo U2F o un'app mobile quando tenti di effettuare una chiamata.

Con il tuo utente IAM abilitato all'MFA, esegui il `aws configure mfa-login`comando per configurare un nuovo profilo da utilizzare con l'autenticazione a più fattori (MFA) per il profilo specificato. Se non viene specificato alcun profilo, l'MFA si basa sul `default` profilo. Se non è configurato alcun profilo predefinito, il `mfa-login` comando richiede AWS le credenziali prima di richiedere le informazioni sull'MFA. Il seguente esempio di comando utilizza la configurazione predefinita e crea un profilo MFA.

```
$ aws configure mfa-login
MFA serial number or ARN: arn:aws:iam::123456789012:mfa/MFADeviceName
MFA token code: 123456
Profile to update [session-MFADeviceName]:
Temporary credentials written to profile 'session-MFADeviceName'
Credentials will expire at 2023-05-19 18:06:10 UTC
To use these credentials, specify --profile session-MFADeviceName when running AWS CLI
 commands
```

Per aggiornare un profilo esistente, utilizzare il `--update-profile` parametro.

```
$ aws configure mfa-login --profile myprofile --update-profile mfaprofile
MFA token code: 123456
Temporary credentials written to profile 'mfaprofile'
Credentials will expire at 2023-05-19 18:06:10 UTC
To use these credentials, specify --profile mfaprofile when running AWS CLI commands
```

Attualmente questo comando supporta solo autenticatori OTP (One-Time Password) basati su hardware o software. Le password e i dispositivi U2F non sono attualmente supportati con questo comando.

Per utilizzare il tuo profilo MFA, usa l'`--profile`opzione con i tuoi comandi.

```
$ aws s3 ls --profile mfaprofile
```

Per ulteriori informazioni sull'utilizzo della MFA con IAM, incluso come assegnare l'MFA a un utente IAM, AWS consulta Autenticazione a più fattori in IAM nella Guida per l'utente.AWS Identity and Access Management

Utilizzo dei metadati dell'istanza Amazon EC2 come credenziali nella AWS CLI

Quando esegui la AWS CLI da un'istanza Amazon Elastic Compute Cloud (Amazon EC2), puoi semplificare la fornitura di credenziali ai comandi. Ogni istanza di Amazon EC2 contiene metadati per i quali la AWS CLI può eseguire direttamente una query delle credenziali temporanee. Quando un ruolo IAM viene associato all'istanza, la AWS CLI recupera in maniera automatica e sicura le credenziali dai metadati dell'istanza.

Per disabilitare questo servizio, usa la variabile di ambiente AWS_EC2_METADATA_DISABLED.

Argomenti

- Prerequisiti
- Configurazione di un profilo per i metadati Amazon EC2

Prerequisiti

Per utilizzare le credenziali di Amazon EC2 con la AWS CLI, è necessario completare quanto segue:

- Istalla e configura la AWS CLI. Per ulteriori informazioni, consultare the section called "Installazione o aggiornamento" e Credenziali di autenticazione e accesso per la AWS CLI.
- Conosci i file di configurazione e i profili denominati. Per ulteriori informazioni, consulta Impostazioni dei file di configurazione e delle credenziali nella AWS CLI.
- A questo scopo, crea un ruolo (IAM) AWS Identity and Access Management che dispone dell'accesso alle risorse necessarie e collega tale ruolo all'istanza Amazon EC2 quando viene avviata. Per ulteriori informazioni, consulta Policy IAM per Amazon EC2 nella Guida per l'utente di Amazon EC2 e Granting Applications That Run on Amazon EC2 Instances Access to AWS Resources nella Guida per l'utente di IAM.

Configurazione di un profilo per i metadati Amazon EC2

Per specificare di voler utilizzare le credenziali disponibili nel profilo dell'istanza di Amazon EC2 di hosting, inserisci la seguente sintassi nel profilo denominato nel file di configurazione. Per istruzioni dettagliate, fai riferimento alle fasi seguenti:

```
[profile profilename]
role_arn = arn:aws:iam::123456789012:role/rolename
credential_source = Ec2InstanceMetadata
region = region
```

1. Crea un profilo nel file di configurazione.

   ```
   [profile profilename]
   ```

2. Aggiungi il ruolo IAM arn che ha accesso alle risorse necessarie.

   ```
   role_arn = arn:aws:iam::123456789012:role/rolename
   ```

3. Specifica `Ec2InstanceMetadata` come fonte delle credenziali.

   ```
   credential_source = Ec2InstanceMetadata
   ```

4. Imposta la tua Regione.

   ```
   region = region
   ```

Esempio

L'esempio seguente assume il ruolo *marketingadminrole* e utilizza la Regione *us-west-2* in un profilo dell'istanza Amazon EC2 denominato *marketingadmin*.

```
[profile marketingadmin]
role_arn = arn:aws:iam::123456789012:role/marketingadminrole
credential_source = Ec2InstanceMetadata
region = us-west-2
```

Acquisizione di credenziali con un processo esterno nella CLI

> **Warning**
>
> Il seguente argomento illustra le credenziali di approvvigionamento da un processo esterno. Potrebbe insorgere un rischio per la sicurezza se il comando per generare le credenziali diventa accessibile a utenti o processi non riconosciuti. Ti consigliamo di utilizzare le soluzioni alternative sicure e supportate offerte dall'AWS CLI e AWS per ridurre il rischio di compromettere le tue credenziali. Assicurati di proteggere il file `config` e altri strumenti e file di supporto per evitare la divulgazione.
> Assicurati che lo strumento delle credenziali personalizzate non scriva informazioni segrete in `StdErr` perché gli SDK e la AWS CLI possono acquisire e registrare tali informazioni, esponendole potenzialmente a utenti non autorizzati.

Se disponi di un metodo per generare o cercare le credenziali non direttamente supportato dalla AWS CLI, puoi fare in modo che venga utilizzato dalla AWS CLI configurando l'impostazione `credential_process` nel file `config`.

Puoi ad esempio includere una voce simile al seguente file `config`.

```
[profile developer]
credential_process = /opt/bin/awscreds-custom --username helen
```

Sintassi

Per creare questa stringa in modo compatibile con qualsiasi sistema operativo, segui queste regole:

- Se il percorso o il nome del file contiene uno spazio, circondare il percorso completo e il nome del file con virgolette doppie (" "). Il percorso e il nome del file possono essere composti solo dai caratteri: A-Z a-z 0-9 - _ . spazio
- Se un nome di parametro o un valore di parametro contiene uno spazio, circondare tale elemento con virgolette doppie (" "). È possibile racchiudere solo il nome o il valore, non la coppia.
- Non includere variabili di ambiente nelle stringhe. Ad esempio, non puoi includere `$HOME` o `%USERPROFILE%`.
- Non specificare la cartella home come ~. Devi specificare il percorso completo.

Esempio per Windows

```
credential_process = "C:\Path\To\credentials.cmd" parameterWithoutSpaces "parameter
 with spaces"
```

Esempio per Linux o macOS

```
credential_process = "/Users/Dave/path/to/credentials.sh" parameterWithoutSpaces
 "parameter with spaces"
```

Output previsto dal programma Credentials

L'AWS CLI esegue il comando esattamente come specificato nel profilo, quindi legge i dati da `STDOUT`. Il comando specificato deve generare in `STDOUT` l'output JSON corrispondente alla sintassi seguente:

```
{
  "Version": 1,
  "AccessKeyId": "an AWS access key",
  "SecretAccessKey": "your AWS secret access key",
  "SessionToken": "the AWS session token for temporary credentials",
  "Expiration": "ISO8601 timestamp when the credentials expire"
}
```

> **Note**
>
> Al momento della stesura del presente documento, la chiave `Version` deve essere configurata su 1. Questo valore potrebbe incrementare nel tempo, man mano che la struttura evolve.

La chiave `Expiration` è un timestamp formattato ISO8601. Se la chiave `Expiration` non è presente nell'output dello strumento, l'interfaccia a riga di comando presuppone che le credenziali siano a lungo termine, non soggette ad aggiornamento. In caso contrario, le credenziali vengono considerate provvisorie e aggiornate automaticamente mediante una nuova esecuzione del comando `credential_process` prima della relativa scadenza.

> **Note**
>
> AWS CLI non esegue il caching delle credenziali dei processi esterni, come nel caso delle credenziali di assunzione del ruolo. Se il caching è necessario, dovrai implementarlo nel processo esterno.

Il processo esterno può restituire un codice diverso da zero per indicare che si è verificato un errore durante il recupero delle credenziali.

Utilizzo di AWS CLI

Questa sezione fornisce una panoramica completa dell'uso generale, delle funzionalità comuni e delle opzioni disponibili in AWS Command Line Interface (AWS CLI), oltre ai dettagli trattati nella sezione Configurazione di the section called "Endpoints".

Questa guida approfondisce gli aspetti fondamentali della scrittura dei comandi della AWS CLI, incluse la struttura di base, la formattazione e le funzionalità di filtraggio. A partire da questi elementi fondamentali, è possibile creare comandi che indirizzano con precisione le risorse e le azioni necessarie, senza bisogno di navigare su complesse console basate sul web.

Inoltre, questo evidenzia il contenuto della guida e la documentazione disponibili per AWS CLI. Dalla guida della riga di comando integrata alla AWS CLI version 2 reference guide completa, avrai accesso alle informazioni per aiutarti a esplorare le caratteristiche e le funzionalità della AWS CLI.

Per esempi e casi d'uso specifici della Servizio AWS, consulta Esempi di AWS CLI o la AWS CLI version 2 reference guide. Questa documentazione fornisce informazioni specifiche sui comandi e mostra degli esempi su come sfruttare la AWS CLI per vari Servizi AWS.

> **Note**
>
> Per impostazione predefinita, la AWS CLI invia le richieste ai Servizi AWS utilizzando l'HTTPS sulla porta TCP 443. Per utilizzare la AWS CLI correttamente, devi essere in grado di eseguire connessioni in uscita su questa porta.

Argomenti in questa guida

- Accesso all'aiuto e alle risorse per AWS CLI
- Struttura di comando in AWS CLI
- Specificare dei valori di parametro nella AWS CLI
- Attivazione e utilizzo dei prompt dei comandi in AWS CLI
- Controllo dell'output dei comandi nella AWS CLI
- Codici restituiti dalla riga di comando in AWS CLI
- Utilizzo di procedure guidate personalizzate per eseguire comandi interattivi nella AWS CLI
- Creazione e utilizzo di alias in AWS CLI
- Risoluzione degli errori per AWS CLI

Accesso all'aiuto e alle risorse per AWS CLI

Questo argomento descrive come accedere ai contenuti di aiuto per AWS Command Line Interface (AWS CLI).

Argomenti

- Comando di aiuto per la AWS CLI integrato
- AWS CLI guida di riferimento
- Documentazione API
- Risoluzione degli errori
- Aiuto aggiuntivo

Comando di aiuto per la AWS CLI integrato

È possibile ottenere assistenza con qualsiasi comando quando si utilizza il comando AWS Command Line Interface (AWS CLI). Per farlo, basta digitare `help` al termine del nome di un comando.

Ad esempio, il comando seguente mostra la guida per le AWS CLI opzioni generali e i comandi di primo livello disponibili.

```
$ aws help
```

Il comando seguente mostra i comandi specifici di Amazon Elastic Compute Cloud (Amazon EC2) disponibili.

```
$ aws ec2 help
```

L'esempio seguente mostra una guida dettagliata per il EC2 `DescribeInstances` funzionamento di Amazon. L'assistenza include le descrizioni dei parametri di input, i filtri disponibili e tutto ciò che è incluso nell'output. Include anche esempi che mostrano come digitare varianti comuni del comando.

```
$ aws ec2 describe-instances help
```

A partire dalla versione `2.31.0` La visualizzazione del comando `help` è configurata dall'impostazione `cli_help_output` e presenta i seguenti valori:

- (impostazione predefinita) `terminal` - Apre la pagina man nel terminale.

- `browser` - Apre la pagina man come file HTML locale nel browser predefinito. Viene stampato un avviso sul terminale quando viene aperto il browser predefinito e un messaggio di errore se AWS CLI non è possibile aprire il browser.
- `url`- Stampa l'URL della Guida AWS CLI di riferimento online per la versione di AWS CLI cui hai installato. Le impostazioni per la paginazione lato client, come la variabile di `AWS_PAGER` ambiente, vengono rispettate.

Il contenuto della Guida per ciascun comando è suddiviso in sei sezioni:

Name

Nome del comando.

```
NAME
       describe-instances -
```

Description

Descrizione dell'operazione API richiamata dal comando.

```
DESCRIPTION
       Describes one or more of your instances.

       If you specify one or more instance IDs, Amazon EC2 returns information
       for those instances. If you do not specify  instance  IDs,  Amazon  EC2
       returns  information  for  all  relevant  instances.  If you specify an
       instance ID that is not valid, an error is returned. If you specify  an
       instance  that  you  do  not  own,  it  is not included in the returned
       results.
...
```

Riepilogo

Sintassi di base per l'utilizzo del comando e delle relative opzioni. Le opzioni tra parentesi quadre sono facoltative, hanno un valore predefinito oppure hanno un'opzione che può essere utilizzata in alternativa.

```
SYNOPSIS
            describe-instances
          [--dry-run | --no-dry-run]
          [--instance-ids <value>]
```

```
          [--filters <value>]
          [--cli-input-json <value>]
          [--starting-token <value>]
          [--page-size <value>]
          [--max-items <value>]
          [--generate-cli-skeleton]
```

Ad esempio, `describe-instances` ha un comportamento predefinito che descrive tutte le istanze nell'account corrente e nella regione. AWS Puoi specificare, a tua discrezione, un elenco di `instance-ids` per descrivere una o più istanze. `dry-run` è un flag booleano opzionale che non utilizza un valore. Per usare un flag booleano specifica il valore mostrato, in questo caso `--dry-run` o `--no-dry-run`. Anche `--generate-cli-skeleton` non utilizza un valore. Se per l'utilizzo di un'opzione sono previste condizioni, queste sono descritte nella sezione `OPTIONS` o mostrate negli esempi.

Opzioni

Descrizione di ogni singola opzione mostrata nella sinossi.

```
OPTIONS
       --dry-run | --no-dry-run (boolean)
          Checks whether you have the required  permissions  for  the  action,
          without actually making the request, and provides an error response.
          If you have the required permissions, the error response is  DryRun-
          Operation . Otherwise, it is UnauthorizedOperation .

       --instance-ids (list)
          One or more instance IDs.

          Default: Describes all your instances.
...
```

Esempi

Esempi che mostrano l'utilizzo del comando e delle relative opzioni. Se non è disponibile alcun esempio per un comando o un caso d'uso di cui hai bisogno, richiedine uno utilizzando il link di feedback in questa pagina o nel riferimento al AWS CLI comando nella pagina di aiuto del comando.

```
    EXAMPLES
    To describe an Amazon EC2 instance
```

```
    Command:

    aws ec2 describe-instances --instance-ids i-5203422c

    To describe all instances with the instance type m1.small

    Command:

    aws ec2 describe-instances --filters "Name=instance-type,Values=m1.small"

    To describe all instances with an Owner tag

    Command:

    aws ec2 describe-instances --filters "Name=tag-key,Values=Owner"
...
```

Output

Descrizioni di ciascuno dei campi e dei tipi di dati inclusi nella risposta di AWS.

Per `describe-instances`, l'output è un elenco di oggetti prenotazione, ciascuno dei quali contiene diversi campi e oggetti che includono informazioni sulle istanze a essi associate. Queste informazioni provengono dalla documentazione dell'API per il tipo di dati di prenotazione utilizzato da Amazon EC2.

```
OUTPUT
       Reservations -> (list)
          One or more reservations.

          (structure)
              Describes a reservation.

              ReservationId -> (string)
                 The ID of the reservation.

              OwnerId -> (string)
                 The ID of the AWS account that owns the reservation.

              RequesterId -> (string)
                 The ID of the requester that launched the instances  on  your
                 behalf (for example, AWS Management Console or Auto Scaling).
```

```
               Groups -> (list)
                  One or more security groups.

                  (structure)
                      Describes a security group.

                      GroupName -> (string)
                         The name of the security group.

                      GroupId -> (string)
                         The ID of the security group.

               Instances -> (list)
                  One or more instances.

                  (structure)
                      Describes an instance.

                      InstanceId -> (string)
                         The ID of the instance.

                      ImageId -> (string)
                         The ID of the AMI used to launch the instance.

                      State -> (structure)
                         The current state of the instance.

                         Code -> (integer)
                             The  low  byte represents the state. The high byte
                             is an opaque internal value and should be ignored.
...
```

Quando AWS CLI esegue il rendering dell'output in JSON, diventa una matrice di oggetti di prenotazione, simile all'esempio seguente.

```
{
    "Reservations": [
        {
            "OwnerId": "012345678901",
            "ReservationId": "r-4c58f8a0",
            "Groups": [],
            "RequesterId": "012345678901",
            "Instances": [
```

```
                {
                    "Monitoring": {
                        "State": "disabled"
                    },
                    "PublicDnsName": "ec2-52-74-16-12.us-
west-2.compute.amazonaws.com",
                    "State": {
                        "Code": 16,
                        "Name": "running"
                    },
...
```

Ogni oggetto prenotazione contiene campi che descrivono la prenotazione e una serie di oggetti istanza, ognuno con specifici campi (ad esempio, `PublicDnsName`) e oggetti (ad esempio, `State`) che lo descrivono.

> **Utenti Windows**
>
> Puoi eseguire il piping (|) dell'output del comando help su `more` per visualizzare il file della guida una pagina alla volta. Premete la barra spaziatrice o PgDnper visualizzare altre parti del documento e **q** per uscire.
>
> ```
> C:\> aws ec2 describe-instances help | more
> ```

AWS CLI guida di riferimento

I file della guida contengono collegamenti che non possono essere visualizzati o esplorati dalla riga di comando. È possibile visualizzare e interagire con questi collegamenti utilizzando la guida di di riferimento AWS CLI versione 2). Il riferimento contiene anche il contenuto della guida per tutti AWS CLI i comandi. Le descrizioni sono presentate in modo da semplificare la navigazione e la visualizzazione su dispositivi mobili, tablet o schermi del desktop.

Documentazione API

Tutti i comandi in AWS CLI corrispondono alle richieste effettuate all'API pubblica di un AWS servizio. Ogni servizio con un'API pubblica include un riferimento API, disponibile nella home page del servizio nel sito web della documentazione AWS. Il contenuto di un riferimento API varia in base al modo in cui è costruita l'API e al protocollo utilizzato. Di solito, un riferimento API contiene informazioni

dettagliate sulle operazioni supportate dall'API, sui dati inviati da e verso il servizio ed su eventuali condizioni di errore riportate dal servizio.

Sezioni della documentazione API

- Operazioni: informazioni dettagliate su ogni operazione e sui relativi parametri (inclusi i limiti di lunghezza o di contenuto e i valori predefiniti). Vengono elencati gli errori che possono verificarsi per questa operazione. Ogni operazione corrisponde a un sottocomando in. AWS CLI
- Tipi di dati: informazioni dettagliate sulle strutture che un comando potrebbe richiedere come parametro o restituire in risposta a una richiesta.
- Parametri comuni: informazioni dettagliate sui parametri condivisi da tutte le operazioni per il servizio.
- Errori comuni: informazioni dettagliate sugli errori che possono essere restituiti da qualsiasi operazione del servizio.

Il nome e la disponibilità di ogni sezione possono variare a seconda del servizio.

> **Specifico del servizio CLIs**
>
> Alcuni servizi dispongono di una CLI separata che risale a prima della creazione di una CLI singola AWS CLI per funzionare con tutti i servizi. Questi servizi specifici CLIs hanno una documentazione separata che è collegata alla pagina della documentazione del servizio. La documentazione relativa ai servizi specifici CLIs non si applica a. AWS CLI

Risoluzione degli errori

Per informazioni sulla diagnosi e la correzione degli AWS CLI errori, vedere. the section called "Risoluzione degli errori"

Aiuto aggiuntivo

Per ulteriore assistenza con i tuoi AWS CLI problemi, visita la AWS CLI community su GitHub.

Struttura di comando in AWS CLI

Questo argomento illustra come è strutturato il comando AWS Command Line Interface (AWS CLI) e come utilizzare i comandi wait.

Argomenti

- Struttura di comando
- Comandi wait

Struttura di comando

La AWS CLI utilizza una struttura in più parti nella riga di comando che deve essere specificata in questo ordine:

1. La chiamata di base al programma `aws`.
2. Il comando di livello superiore, che in genere corrisponde a un servizio AWS supportato dall'AWS CLI.
3. Il sottocomando che specifica l'operazione da eseguire.
4. Le opzioni o i parametri della AWS CLI generali necessari per l'operazione. Possono essere specificati in qualsiasi ordine, purché seguano le prime tre parti. Se un parametro esclusivo viene specificato più volte, si applica solo l'ultimo valore.

```
$ aws <command> <subcommand> [options and parameters]
```

I parametri possono utilizzare vari tipi di valori di input, ad esempio numeri, stringhe, elenchi, mappe e strutture JSON. Quanto supportato dipende dal comando e dal sottocomando specificati.

Esempi

Amazon S3

L'esempio seguente elenca tutti i tuoi bucket Amazon S3.

```
$ aws s3 ls
2018-12-11 17:08:50 amzn-s3-demo-bucket1
2018-12-14 14:55:44 amzn-s3-demo-bucket2
```

Per ulteriori informazioni sull'utilizzo dei comandi Amazon S3, consulta `aws s3` in AWS CLI Command Reference.

AWS CloudFormation

Il seguente esempio del comando `create-change-set` modifica il nome dello stack in *my-change-set*.

```
$ aws cloudformation create-change-set --stack-name my-stack --change-set-name my-change-set
```

Per ulteriori informazioni sui comandi AWS CloudFormation, consulta `aws cloudformation` in AWS CLI Command Reference.

Comandi wait

Alcuni servizi AWS dispongono dei comandi `wait`. Qualsiasi comando che utilizza `aws wait` in genere attende il completamento di un comando prima di passare alla fase successiva. Ciò è particolarmente utile per i comandi multiparte o gli script, poiché è possibile utilizzare un comando wait per evitare di passare alle fasi successive se il comando wait ha esito negativo.

La AWS CLI utilizza una struttura in più parti nella riga di comando per il comando `wait` che deve essere specificata in questo ordine:

1. La chiamata di base al programma `aws`.
2. Il comando di livello superiore, che in genere corrisponde a un servizio AWS supportato dall'AWS CLI.
3. Il comando `wait`.
4. Il sottocomando che specifica l'operazione da eseguire.
5. Le opzioni o i parametri dell'interfaccia a riga di comando necessari per l'operazione. Possono essere specificati in qualsiasi ordine, purché seguano le prime tre parti. Se un parametro esclusivo viene specificato più volte, si applica solo l'ultimo valore.

```
$ aws <command> wait <subcommand> [options and parameters]
```

I parametri possono utilizzare vari tipi di valori di input, ad esempio numeri, stringhe, elenchi, mappe e strutture JSON. Quanto supportato dipende dal comando e dal sottocomando specificati.

> **Note**
>
> Non tutti i servizi AWS supportano i comandi `wait`. Consulta la AWS CLI version 2 reference guide per verificare se il tuo servizio supporta i comandi `wait`.

Esempi

AWS CloudFormation

I seguenti esempi di comandi `wait change-set-create-complete` si interrompono e riprendono solo dopo aver confermato che il set di modifiche *my-change-set* nello stack *my-stack* è pronto per l'esecuzione.

```
$ aws cloudformation wait change-set-create-complete --stack-name my-stack --change-set-name my-change-set
```

Per ulteriori informazioni sui comandi AWS CloudFormation `wait`, consulta `wait` in AWS CLI Command Reference.

AWS CodeDeploy

I seguenti esempi di comando `wait deployment-successful` si mettono in pausa finché l'implementazione *d-A1B2C3111* viene completata correttamente.

```
$ aws deploy wait deployment-successful --deployment-id d-A1B2C3111
```

Per ulteriori informazioni sui comandi AWS CodeDeploy `wait`, consulta `wait` in AWS CLI Command Reference.

Specificare dei valori di parametro nella AWS CLI

Molti parametri utilizzati in AWS Command Line Interface (AWS CLI) sono semplici valori stringa o numerici, ad esempio il nome della coppia di chiavi `my-key-pair` nell'esempio del comando `aws ec2 create-key-pair` seguente.

```
$ aws ec2 create-key-pair --key-name my-key-pair
```

La formattazione del comando può variare tra i terminali. Ad esempio, la maggior parte dei terminali distingue tra maiuscole e minuscole, ma Powershell non fa distinzione tra maiuscole e minuscole. Ciò significa che i due esempi di comandi seguenti produrrebbero risultati diversi per i terminali con distinzione tra maiuscole e minuscole in quanto visualizzano `MyFile*.txt` e `myfile*.txt` come parametri diversi.

Tuttavia, PowerShell potrebbe elaborare queste richieste allo stesso modo in cui le vedono `MyFile*.txt` e `myfile*.txt` con gli stessi parametri. Il comando di esempio seguente dimostra questi parametri che utilizzano il comando `aws s3 cp`:

```
$ aws s3 cp . s3://amzn-s3-demo-bucket/path --include "MyFile*.txt"
$ aws s3 cp . s3://amzn-s3-demo-bucket/path --include "myfile*.txt"
```

Per ulteriori informazioni sulla mancanza di distinzione tra maiuscole e minuscole di PowerShell, consulta about_Case-Sensitivity nella documentazione di PowerShell.

A volte è necessario utilizzare virgolette o caratteri letterali attorno a stringhe che includono caratteri speciali o di spazio. Le regole relative a questa formattazione possono variare anche tra i terminali. Per ulteriori informazioni sull'utilizzo di virgolette attorno a parametri complessi, consulta Utilizzo di virgolette e valori letterali con stringhe nella AWS CLI.

Questi argomenti riguardano le regole di formattazione dei terminali più comuni. Se riscontri dei problemi con il riconoscimento dei valori dei parametri da parte del terminale, assicurati di consultare gli argomenti di questa sezione e di consultare anche la documentazione del terminale per le relative regole di sintassi specifiche.

Argomenti del parametro

- Tipi di parametri comuni nella AWS CLI
- Utilizzo di virgolette e valori letterali con stringhe nella AWS CLI
- Caricamento di un parametro da un file nella AWS CLI
- Skeleton e file di input della AWS CLI nella AWS CLI
- Utilizzo della sintassi abbreviata in AWS CLI

Tipi di parametri comuni nella AWS CLI

Questa sezione descrive alcuni tipi di parametri comuni e il formato tipico richiesto.

In caso di problemi con la formattazione di un parametro per un comando specifico, consulta la guida digitando **`help`** dopo il nome del comando. L'aiuto per ogni sottocomando include il nome e la descrizione di un'opzione. Il tipo di parametro dell'opzione è elencato tra parentesi. Per ulteriori informazioni sulla visualizzazione della guida, consulta the section called "Chiedere aiuto".

I tipi di parametri includono:

- Stringa

- Timestamp
- Elenco
- Booleano
- Numero intero
- Binary / blob (oggetto binario di grandi dimensioni) e blob in streaming
- Eseguire la mappatura
- Documento

Stringa

I parametri stringa possono contenere caratteri alfanumerici, simboli e spazi vuoti del set di caratteri ASCII. Le stringhe che contengono spazi vuoti devono essere racchiuse tra virgolette. Ti consigliamo di non utilizzare simboli o spazi bianchi diversi dal carattere di spazio standard e di osservare le regole di citazione del tuo terminale per evitare risultati imprevisti.

Alcuni parametri di stringa accettano i dati binari da un file. Consulta File binari per un esempio.

Timestamp

I timestamp sono formattati in base allo standard ISO 8601. Questi sono spesso indicati come parametri "`DateTime`" o "`Date`".

```
$ aws ec2 describe-spot-price-history --start-time 2014-10-13T19:00:00Z
```

I formati accettabili includono:

- *AAAA*-*MM*-*GG*T*hh*:*mm*:*ss.sssTZD (UTC)*, ad esempio 2014-10-01T20:30:00.000Z
- *AAAA*-*MM*-*GG*T*hh*:*mm*:*ss.sssTZD (con offset)*, ad esempio 2014-10-01T12:30:00.000-08:00
- *YYYY*-*MM*-*DD*, ad esempio, 2014-10-01
- Tempo Unix in secondi, ad esempio 1412195400. A volte, viene chiamato epoca Unix (Unix epoch) e rappresenta il numero di secondi dalla mezzanotte del 1 gennaio 1970 UTC.

Per impostazione predefinita, la AWS CLI versione 2 traduce tutti i valori DateTime risposta nel formato ISO 8601.

È possibile impostare il formato del timestamp utilizzando l'impostazione del file `cli_timestamp_format`.

Elenco

Una o più stringhe separate da spazi. Se uno degli elementi della stringa contiene uno spazio, devi racchiuderlo tra virgolette. Osserva le regole di citazione del tuo terminale per evitare risultati imprevisti.

```
$ aws ec2 describe-spot-price-history --instance-types m1.xlarge m1.medium
```

Booleano

Flag binario che attiva o disattiva un'opzione. Ad esempio, `ec2 describe-spot-price-history` ha un parametro `--dry-run` booleano che, quando specificato, consente di convalidare la query con il servizio anche senza eseguire la query.

```
$ aws ec2 describe-spot-price-history --dry-run
```

L'output indica se il formato del comando è valido. Questo comando include anche una versione `--no-dry-run` del parametro, che puoi utilizzare per indicare esplicitamente che il comando deve essere eseguito normalmente, anche se non è necessario includerlo poiché questa è l'impostazione predefinita.

Numero intero

Numero intero senza firma.

```
$ aws ec2 describe-spot-price-history --max-items 5
```

Binary / blob (oggetto binario di grandi dimensioni) e blob in streaming

Nella AWS CLI, è possibile passare un valore binario come stringa direttamente sulla riga di comando. Esistono due tipi di blob:

- Blob
- Blob di streaming

Blob

Per passare un valore a un parametro con tipo `fileb://`, è necessario specificare un percorso di un file locale che contiene i dati binari utilizzando il prefisso `blob`. I file a cui si fa riferimento utilizzando il

prefisso `fileb://` vengono sempre trattati come binario non codificato. Il percorso specificato viene interpretato come relativo alla directory di lavoro. Ad esempio, il parametro `--plaintext` per `aws kms encrypt` è un blob.

```
$ aws kms encrypt \
    --key-id 1234abcd-12ab-34cd-56ef-1234567890ab \
    --plaintext fileb://ExamplePlaintextFile \
    --output text \
    --query CiphertextBlob | base64 \
    --decode > ExampleEncryptedFile
```

> **Note**
>
> Per la compatibilità con le versioni precedenti, puoi utilizzare il prefisso `file://`. Esistono due formati utilizzati in base all'impostazione del file `cli_binary_format` o all'opzione della riga di comando `--cli-binary-format`:
>
> - Impostazione predefinita per la AWS CLI versione 2. Se il valore dell'impostazione è `base64`, i file a cui si fa riferimento utilizzando il prefisso `file://` vengono trattati come testo con codifica Base64.
> - Impostazione predefinita per la AWS CLI versione 1. Se il valore dell'impostazione è `raw-in-base64-out`, i file a cui si fa riferimento utilizzando il prefisso `file://` vengono letti come testo e quindi la AWS CLI tenta di codificarli in codice binario.
>
> Per ulteriori informazioni, consulta l'impostazione del file `cli_binary_format` o l'opzione della riga di comando `--cli-binary-format`.

Blob di streaming

I blob di streaming, ad esempio `aws cloudsearchdomain upload-documents` non utilizza prefissi. Invece, i parametri dello streaming blob vengono formattati utilizzando il percorso diretto del file. L'esempio seguente utilizza il percorso diretto del file `document-batch.json` per il comando `aws cloudsearchdomain upload-documents`:

```
$ aws cloudsearchdomain upload-documents \
    --endpoint-url https://doc-my-domain.us-west-1.cloudsearch.amazonaws.com \
    --content-type application/json \
```

```
    --documents document-batch.json
```

Eseguire la mappatura

Un set di coppie chiave-valore specificato in formato JSON o utilizzando la sintassi abbreviata dell'interfaccia a riga di comando (CLI). L'esempio JSON seguente legge un elemento da una tabella Amazon DynamoDB denominata my-table con un parametro mappa, `--key`. Il parametro specifica la chiave primaria denominata id con un valore numerico 1 in una struttura JSON nidificata.

Per un utilizzo JSON più avanzato in una riga di comando, prendi in considerazione l'utilizzo di un processore JSON a riga di comando, ad esempio `jq`, per creare stringhe JSON. Per ulteriori informazioni su `jq`, consulta jq repository in GitHub.

```
$ aws dynamodb get-item --table-name my-table --key '{"id": {"N":"1"}}'

{
    "Item": {
        "name": {
            "S": "John"
        },
        "id": {
            "N": "1"
        }
    }
}
```

Documento

> **Note**
>
> La sintassi abbreviata non è compatibile con i tipi di documento.

I tipi di documento vengono utilizzati per inviare dati senza dover incorporare JSON nelle stringhe. Il tipo di documento consente ai servizi di fornire schemi arbitrari per l'utilizzo di tipi di dati più flessibili.

Ciò consente di inviare dati JSON senza la necessità di eseguire l'escape dei valori. Ad esempio, invece di utilizzare il seguente input JSON con escape:

```
{"document": "{\"key\":true}"}
```

Puoi utilizzare il seguente tipo di documento:

```
{"document": {"key": true}}
```

I valori validi per i tipi di documento

A causa della natura flessibile dei tipi di documento, esistono più tipi di valori validi. I valori validi includono i seguenti:

Stringa

```
--option '"value"'
```

Numero

```
--option 123
--option 123.456
```

Booleano

```
--option true
```

Null

```
--option null
```

Array

```
--option '["value1", "value2", "value3"]'
--option '["value", 1, true, null, ["key1", 2.34], {"key2": "value2"}]'
```

Oggetto

```
--option '{"key": "value"}'
--option '{"key1": "value1", "key2": 123, "key3": true, "key4": null, "key5":
 ["value3", "value4"], "key6": {"value5": "value6"}'
```

Utilizzo di virgolette e valori letterali con stringhe nella AWS CLI

Esistono principalmente due modi in cui vengono utilizzate le virgolette singole e doppie nella AWS CLI.

- Utilizzo delle virgolette attorno a stringhe che contengono spazi bianchi
- Utilizzo delle virgolette all'interno delle stringhe

Utilizzo delle virgolette attorno a stringhe che contengono spazi bianchi

I nomi dei parametri e i relativi valori sono separati da spazi sulla riga di comando. Se un valore stringa contiene uno spazio incorporato, è necessario circondare l'intera stringa con virgolette per evitare che l'AWS CLI interpreti erroneamente lo spazio come divisore tra il valore e il nome del parametro successivo. Il tipo di virgolette utilizzate dipende dal sistema operativo su cui si sta eseguendo la AWS CLI.

Linux and macOS

Utilizzo di virgolette singole `' '`

```
$ aws ec2 create-key-pair --key-name 'my key pair'
```

Per ulteriori informazioni sull'utilizzo delle virgolette, consulta la documentazione per l'utente della shell che preferisci.

PowerShell

Virgolette singole (scelta consigliata)

Le virgolette singole `' '` sono chiamate stringhe `verbatim`. La stringa viene passata al comando esattamente durante la digitazione, e ciò significa che le variabili PowerShell non verranno trasmesse.

```
PS C:\> aws ec2 create-key-pair --key-name 'my key pair'
```

Virgolette doppie

Le virgolette doppie `" "` sono chiamate stringhe `expandable`. Le variabili possono essere trasmesse nelle stringhe espandibili.

```
PS C:\> aws ec2 create-key-pair --key-name "my key pair"
```

Per ulteriori informazioni sull'utilizzo delle virgolette, consulta About Quoting Rules in Microsoft PowerShell Docs.

Windows command prompt

Utilizzo delle virgolette doppie " ".

```
C:\> aws ec2 create-key-pair --key-name "my key pair"
```

Facoltativamente, è possibile separare il nome del parametro dal valore con un segno di uguale (=) invece di uno spazio. In genere è necessario solo se il valore del parametro inizia con un trattino.

```
$ aws ec2 delete-key-pair --key-name=-mykey
```

Utilizzo delle virgolette all'interno delle stringhe

Le stringhe potrebbero contenere virgolette e la shell potrebbe richiedere le virgolette di escape perché funzionino correttamente. Uno dei tipi di valore di parametro comuni è una stringa JSON. Questa operazione è complessa poiché include spazi e virgolette doppie " " attorno al nome e al valore di ogni elemento nella struttura JSON. Il modo in cui si immettono parametri in formato JSON alla riga di comando varia a seconda del sistema operativo.

Per un utilizzo più avanzato del codice JSON su una riga di comando, prendi in considerazione l'utilizzo di un processore JSON a riga di comando, ad esempio `jq`, per creare stringhe JSON. Per ulteriori informazioni su `jq`, consulta jq repository in GitHub.

Linux and macOS

Affinché Linux e macOS interpretino le stringhe, usa letteralmente le virgolette singole ' ' per racchiudere la struttura dei dati JSON, come nell'esempio seguente. Non è necessario utilizzare caratteri di escape per le virgolette doppie incorporate nella stringa JSON, poiché vengono trattate letteralmente. Poiché il codice JSON è racchiuso tra virgolette singole, sarà necessario eliminare tutte le virgolette singole nella stringa, in genere utilizzando una barra rovesciata prima delle virgolette singole `\'`.

```
$ aws ec2 run-instances \
```

```
    --image-id ami-12345678 \
    --block-device-mappings '[{"DeviceName":"/dev/sdb","Ebs":
{"VolumeSize":20,"DeleteOnTermination":false,"VolumeType":"standard"}}]'
```

Per ulteriori informazioni sull'utilizzo delle virgolette, consulta la documentazione per l'utente della shell che preferisci.

PowerShell

Utilizzo di virgolette singole ' ' o doppie " ".

Virgolette singole (scelta consigliata)

Le virgolette singole ' ' sono chiamate stringhe `verbatim`. La stringa viene passata al comando esattamente durante la digitazione, e ciò significa che le variabili PowerShell non verranno trasmesse.

Poiché le strutture di dati JSON includono virgolette doppie, suggeriamo di racchiuderle tra virgolette singole ' '. Se si utilizzano virgolette singole, non è necessario eseguire l'escape delle virgolette doppie incorporate nella stringa JSON. Tuttavia, è necessario eseguire l'escape per ogni singola virgoletta inserendo un segno di spunta ` all'interno della struttura JSON.

```
PS C:\> aws ec2 run-instances `
    --image-id ami-12345678 `
    --block-device-mappings '[{"DeviceName":"/dev/sdb","Ebs":
{"VolumeSize":20,"DeleteOnTermination":false,"VolumeType":"standard"}}]'
```

Virgolette doppie

Le virgolette doppie " " sono chiamate stringhe `expandable`. Le variabili possono essere trasmesse nelle stringhe espandibili.

Se si utilizzano le virgolette doppie, non è necessario eseguire l'escape delle virgolette singole incorporate nella stringa JSON. Tuttavia, è necessario eseguire l'escape per ogni virgoletta doppia con un segno di spunta ` all'interno della struttura JSON, come nell'esempio seguente.

```
PS C:\> aws ec2 run-instances `
    --image-id ami-12345678 `
    --block-device-mappings "[{`"DeviceName`":`"/dev/sdb`",`"Ebs`":
{`"VolumeSize`":20,`"DeleteOnTermination`":false,`"VolumeType`":`"standard`"}}]"
```

Per ulteriori informazioni sull'utilizzo delle virgolette, consulta About Quoting Rules in Microsoft PowerShell Docs.

> **Warning**
>
> Prima che PowerShell invii un comando a AWS CLI, determina se il comando viene interpretato utilizzando le tipiche regole di PowerShell o delle virgolette `CommandLineToArgvW`. Quando i processi PowerShell utilizzano `CommandLineToArgvW`, è necessario eseguire l'escape dei caratteri con una barra rovesciata `\`.
> Per ulteriori informazioni su `CommandLineToArgvW` in PowerShell, consulta What's up with the strange treatment of quotation marks and backslashes by CommandLineToArgvW in Microsoft DevBlogs, Everyone quotes command line arguments the wrong way in Microsoft Docs Blog e CommandLineToArgvW function in Microsoft Docs.
> Virgolette singole
> Le virgolette singole `' '` sono chiamate stringhe `verbatim`. La stringa viene passata al comando esattamente durante la digitazione, e ciò significa che le variabili PowerShell non verranno trasmesse. Esegui l'escape dei caratteri con una barra rovesciata `\`.

```
PS C:\> aws ec2 run-instances `
    --image-id ami-12345678 `
    --block-device-mappings '[{\"DeviceName\":\"/dev/sdb\",\"Ebs\":
{\"VolumeSize\":20,\"DeleteOnTermination\":false,\"VolumeType\":\"standard\"}}]'
```

> Virgolette doppie
> Le virgolette doppie `" "` sono chiamate stringhe `expandable`. Le variabili possono essere trasmesse nelle stringhe `expandable`. Per le stringhe tra virgolette doppie devi eseguire l'escape due volte usando `` `\ `` per ogni virgoletta invece di utilizzare solo un carattere di apice. Il carattere di apice evita la barra rovesciata, e quindi la barra rovesciata viene utilizzata come carattere di escape per il processo `CommandLineToArgvW`.

```
PS C:\> aws ec2 run-instances `
    --image-id ami-12345678 `
    --block-device-mappings "[{`\"DeviceName`\":`\"/dev/sdb`\",`\"Ebs`\":
{`\"VolumeSize`\":20,`\"DeleteOnTermination`\":false,`\"VolumeType`\":`
\"standard`\"}}]"
```

Blobs (scelta consigliata)
Per ignorare le regole delle virgolette di PowerShell per l'input di dati JSON, utilizza Blobs per passare i dati JSON direttamente a AWS CLI. Per ulteriori informazioni sui Blob, consulta Blob.

Windows command prompt

Il prompt dei comandi di Windows richiede le virgolette doppie " " per racchiudere la struttura di dati JSON. Inoltre, per impedire al processore di comandi di interpretare erroneamente le virgolette doppie incorporate nel JSON, è necessario anche evitare (inserendo un carattere barra rovesciata \) l'uso di virgolette doppie " all'interno della struttura dati JSON stessa, come nell'esempio seguente.

```
C:\> aws ec2 run-instances ^
    --image-id ami-12345678 ^
    --block-device-mappings "[{\"DeviceName\":\"/dev/sdb\",\"Ebs\":
{\"VolumeSize\":20,\"DeleteOnTermination\":false,\"VolumeType\":\"standard\"}}]"
```

Soltanto le virgolette doppie più esterne non richiedono il carattere di escape.

Caricamento di un parametro da un file nella AWS CLI

Alcuni parametri si aspettano nomi di file come argomenti, grazie ai quali l'AWS CLI carica i dati. Altri parametri consentono di specificare il valore del parametro come testo digitato sulla riga di comando o letto da un file. Se un file è obbligatorio o facoltativo, è necessario codificare correttamente il file in modo che l'AWS CLI possa comprenderlo. La codifica del file deve corrispondere alle impostazioni locali predefinite del sistema di lettura. È possibile determinarle usando il metodo Python `locale.getpreferredencoding()`.

Questo metodo serve per caricare un file per un singolo parametro. Per informazioni sul caricamento di più parametri con un singolo file, consultathe section called "Generazione di un modello dello skeleton della CLI".

> **Note**
>
> Per impostazione predefinita, Windows PowerShell restituisce il testo in formato UTF-16, che è in conflitto con la codifica UTF-8 utilizzata dai file JSON e da molti sistemi Linux. Ti

consigliamo di utilizzare `-Encoding ascii` con i comandi PowerShell `Out-File` per essere sicuro che il file risultante possa essere letto dall'AWS CLI.

Argomenti

- Come caricare un parametro da un file
- File binari
- Caricamento di un file come valore di sintassi abbreviata

Come caricare un parametro da un file

In alcuni casi, ad esempio quando il parametro è una stringa JSON complessa, è utile caricare il valore di un parametro da un file anziché provare a digitarlo come valore di parametro della riga di comando. Per specificare un file che contiene il valore, specifica un URL di file nel formato seguente:

```
file://complete/path/to/file
```

- I primi due caratteri di barra" /" fanno parte della specifica. Se il percorso richiesto inizia con "/", il risultato conterrà tre caratteri di barra: `file:///folder/file`.
- L'URL fornisce il percorso del file con il contenuto del parametro effettivo.
- Quando utilizzi dei file con spazi o caratteri speciali, segui le regole di escape e di utilizzo delle virgolette del tuo terminale.

I percorsi di file specificati nei seguenti esempi vengono interpretati come relativi alla directory di lavoro corrente.

Linux or macOS

```
// Read from a file in the current directory
$ aws ec2 describe-instances --filters file://filter.json

// Read from a file in /tmp
$ aws ec2 describe-instances --filters file:///tmp/filter.json

// Read from a file with a filename with whitespaces
$ aws ec2 describe-instances --filters 'file://filter content.json'
```

Windows command prompt

```
// Read from a file in C:\temp
C:\> aws ec2 describe-instances --filters file://C:\temp\filter.json

// Read from a file with a filename with whitespaces
C:\> aws ec2 describe-instances --filters "file://C:\temp\filter content.json"
```

L'opzione del prefisso `file://` supporta le espansioni in stile Unix, tra cui "~/", "`./`" e "`../`". In Windows l'espressione "~/" si espande alla directory utente, archiviata nella variabile di ambiente `%USERPROFILE%`. Ad esempio, in Windows 10 in genere è presente una directory utente in `%USERPROFILE%`.

Nei documenti JSON incorporati come valore di un altro documento JSON deve essere comunque inserito il carattere di escape.

```
$ aws sqs create-queue --queue-name my-queue --attributes file://attributes.json
```

attributes.json

```
{
  "RedrivePolicy": "{\"deadLetterTargetArn\":\"arn:aws:sqs:us-
west-2:0123456789012:deadletter\", \"maxReceiveCount\":\"5\"}"
}
```

File binari

Per i comandi che richiedono dati binari come parametro, è necessario specificare che i dati sono contenuto binario mediante il prefisso `fileb://`. I comandi che accettano i dati binari includono:

- Parametro `--user-data` **`aws ec2 run-instances:`**.
- Parametro `--sse-customer-key` **`aws s3api put-object:`**.
- Parametro `--ciphertext-blob` **`aws kms decrypt:`**.

Nell'esempio seguente viene generata una chiave AES binaria da 256 bit mediante uno strumento a riga di comando Linux che vien poi fornita ad Amazon S3 per crittografare un file caricato sul lato server.

```
$ dd if=/dev/urandom bs=1 count=32 > sse.key
32+0 records in
32+0 records out
32 bytes (32 B) copied, 0.000164441 s, 195 kB/s
$ aws s3api put-object \
    --bucket amzn-s3-demo-bucket \
    --key test.txt \
    --body test.txt \
    --sse-customer-key fileb://sse.key \
    --sse-customer-algorithm AES256
{
    "SSECustomerKeyMD5": "iVg8oWa8sy714+FjtesrJg==",
    "SSECustomerAlgorithm": "AES256",
    "ETag": "\"a6118e84b76cf98bf04bbe14b6045c6c\""
}
```

Per un altro esempio che fa riferimento a un file contenente parametri in formato JSON più complessi, consulta Collegamento di una policy IAM gestita a un utente.

Caricamento di un file come valore di sintassi abbreviata

Quando si utilizza una sintassi abbreviata in cui un valore è grande o complesso, è spesso più semplice caricarlo in un file come valore. Per caricare un file come valore di sintassi abbreviata, la formattazione cambierà leggermente. Invece di `key=value`, utilizzerai l'operatore `@=` al posto dell'operatore `=`. `@=` significa per la AWS CLI che il valore deve essere letto come un percorso di file e non come una stringa. L'esempio seguente mostra una coppia chiave-valore che carica un file per il relativo valore.

Linux or macOS

```
--option key@=file://template.txt
```

Windows

```
--option "key1@=file://template.txt"
```

L'esempio seguente mostra il caricamento di un file di certificato per il comando `aws rolesanywhere create-trust-anchor`.

```
$ aws rolesanywhere create-trust-anchor --name TrustAnchor \
```

```
    --source sourceData={x509CertificateData@=file://root-
ca.crt},sourceType="CERTIFICATE_BUNDLE"  \
    --enabled
```

Per ulteriori informazioni sulla sintassi abbreviata, consulta the section called "Sintassi abbreviata".

Skeleton e file di input della AWS CLI nella AWS CLI

La maggior parte dei comandi della AWS CLI accetta l'importazione di input di parametri da un file. Questi modelli possono essere generati utilizzando l'opzione `generate-cli-skeleton` e quindi importati utilizzando i parametri `--cli-input-json` e `--cli-input-yaml`.

Argomenti

- Informazioni sugli skeleton e sui file di input della AWS CLI
- Generazione e importazione di uno skeleton di comandi
- Combinazione di file di input e parametri della riga di comando

Informazioni sugli skeleton e sui file di input della AWS CLI

La maggior parte dei comandi di AWS Command Line Interface (AWS CLI) consente di accettare tutti gli input di parametri da un file utilizzando i parametri `--cli-input-json` e `--cli-input-yaml`.

Questi stessi comandi utilizzano il parametro `--generate-cli-skeleton` per generare un file in formato JSON o YAML con tutti i parametri che può essere modificato e compilato. Quindi puoi eseguire il comando con il parametro `--cli-input-json` o `--cli-input-yaml` e scegliere il file compilato.

> **Important**
>
> I comandi della AWS CLI personalizzati, come i `aws s3` non supportano né `--generate-cli-skeleton` né `--cli-input-json` e i parametri `--cli-input-yaml` descritti in questo argomento. Per verificare se un comando specifico supporta questi parametri, esegui il `comando help` relativo al comando che desideri utilizzare o consulta la AWS CLI version 2 reference guide.

Il comando `--generate-cli-skeleton` genera e visualizza un modello di parametri che è possibile personalizzare e utilizzare come input per un comando. Il modello generato include tutti i parametri supportati dal comando.

Il parametro `--generate-cli-skeleton` accetta uno dei seguenti valori:

- `input` - Il modello generato include tutti i parametri di input formattati come JSON. Si tratta del valore di default.
- `yaml-input` - Il modello generato include tutti i parametri di input formattati come YAML.
- `output` - Il modello generato include tutti i parametri di output formattati come JSON. Al momento non è possibile richiedere parametri di output come YAML.

Poiché l'AWS CLI è essenzialmente un "wrapper" per l'API del servizio, il file di skeleton prevede che tu faccia riferimento a tutti i parametri in base ai nomi del parametro API sottostanti. Questo è probabilmente diverso dal nome del parametro AWS CLI. Ad esempio, un parametro AWS CLI denominato `user-name` potrebbe essere mappato al parametro API del servizio AWS denominato `UserName` (nota le maiuscole e il trattino mancante). Ti consigliamo di utilizzare l'opzione `--generate-cli-skeleton` per generare il modello con i nomi di parametro "corretti" per evitare errori. Puoi fare riferimento alla Guida di riferimento delle API per il servizio per visualizzare i nomi dei parametri previsti. Puoi eliminare dal modello tutti i parametri che non sono obbligatori e per i quali non desideri fornire un valore.

Ad esempio, se esegui questo comando, viene generato il modello di parametri per il comando run-instances di Amazon Elastic Compute Cloud (Amazon EC2).

JSON

Nell'esempio seguente viene illustrato come generare un modello formattato in JSON utilizzando il valore predefinito (`input`) per il parametro `--generate-cli-skeleton`.

```
$ aws ec2 run-instances --generate-cli-skeleton
```

```
{
    "DryRun": true,
    "ImageId": "",
    "MinCount": 0,
    "MaxCount": 0,
    "KeyName": "",
    "SecurityGroups": [
```

```
        ""
    ],
    "SecurityGroupIds": [
        ""
    ],
    "UserData": "",
    "InstanceType": "",
    "Placement": {
        "AvailabilityZone": "",
        "GroupName": "",
        "Tenancy": ""
    },
    "KernelId": "",
    "RamdiskId": "",
    "BlockDeviceMappings": [
        {
            "VirtualName": "",
            "DeviceName": "",
            "Ebs": {
                "SnapshotId": "",
                "VolumeSize": 0,
                "DeleteOnTermination": true,
                "VolumeType": "",
                "Iops": 0,
                "Encrypted": true
            },
            "NoDevice": ""
        }
    ],
    "Monitoring": {
        "Enabled": true
    },
    "SubnetId": "",
    "DisableApiTermination": true,
    "InstanceInitiatedShutdownBehavior": "",
    "PrivateIpAddress": "",
    "ClientToken": "",
    "AdditionalInfo": "",
    "NetworkInterfaces": [
        {
            "NetworkInterfaceId": "",
            "DeviceIndex": 0,
            "SubnetId": "",
            "Description": "",
```

```
            "PrivateIpAddress": "",
            "Groups": [
                ""
            ],
            "DeleteOnTermination": true,
            "PrivateIpAddresses": [
                {
                    "PrivateIpAddress": "",
                    "Primary": true
                }
            ],
            "SecondaryPrivateIpAddressCount": 0,
            "AssociatePublicIpAddress": true
        }
    ],
    "IamInstanceProfile": {
        "Arn": "",
        "Name": ""
    },
    "EbsOptimized": true
}
```

YAML

Nell'esempio seguente viene illustrato come generare un modello formattato in YAML utilizzando il valore `yaml-input` per il parametro `--generate-cli-skeleton`.

```
$ aws ec2 run-instances --generate-cli-skeleton yaml-input
```

```
BlockDeviceMappings:  # The block device mapping entries.
- DeviceName: ''  # The device name (for example, /dev/sdh or xvdh).
  VirtualName: '' # The virtual device name (ephemeralN).
  Ebs: # Parameters used to automatically set up Amazon EBS volumes when the
 instance is launched.
    DeleteOnTermination: true  # Indicates whether the EBS volume is deleted on
 instance termination.
    Iops: 0 # The number of I/O operations per second (IOPS) that the volume
 supports.
    SnapshotId: '' # The ID of the snapshot.
    VolumeSize: 0 # The size of the volume, in GiB.
    VolumeType: st1 # The volume type. Valid values are: standard, io1, gp2, sc1,
 st1.
```

```
    Encrypted: true # Indicates whether the encryption state of an EBS volume is
 changed while being restored from a backing snapshot.
    KmsKeyId: '' # Identifier (key ID, key alias, ID ARN, or alias ARN) for a
 customer managed KMS key under which the EBS volume is encrypted.
  NoDevice: '' # Suppresses the specified device included in the block device
 mapping of the AMI.
ImageId: '' # The ID of the AMI.
InstanceType: c4.4xlarge # The instance type. Valid values are: t1.micro, t2.nano,
 t2.micro, t2.small, t2.medium, t2.large, t2.xlarge, t2.2xlarge, t3.nano, t3.micro,
 t3.small, t3.medium, t3.large, t3.xlarge, t3.2xlarge, t3a.nano, t3a.micro,
 t3a.small, t3a.medium, t3a.large, t3a.xlarge, t3a.2xlarge, m1.small, m1.medium,
 m1.large, m1.xlarge, m3.medium, m3.large, m3.xlarge, m3.2xlarge, m4.large,
 m4.xlarge, m4.2xlarge, m4.4xlarge, m4.10xlarge, m4.16xlarge, m2.xlarge, m2.2xlarge,
 m2.4xlarge, cr1.8xlarge, r3.large, r3.xlarge, r3.2xlarge, r3.4xlarge, r3.8xlarge,
 r4.large, r4.xlarge, r4.2xlarge, r4.4xlarge, r4.8xlarge, r4.16xlarge, r5.large,
 r5.xlarge, r5.2xlarge, r5.4xlarge, r5.8xlarge, r5.12xlarge, r5.16xlarge,
 r5.24xlarge, r5.metal, r5a.large, r5a.xlarge, r5a.2xlarge, r5a.4xlarge,
 r5a.8xlarge, r5a.12xlarge, r5a.16xlarge, r5a.24xlarge, r5d.large, r5d.xlarge,
 r5d.2xlarge, r5d.4xlarge, r5d.8xlarge, r5d.12xlarge, r5d.16xlarge, r5d.24xlarge,
 r5d.metal, r5ad.large, r5ad.xlarge, r5ad.2xlarge, r5ad.4xlarge, r5ad.8xlarge,
 r5ad.12xlarge, r5ad.16xlarge, r5ad.24xlarge, x1.16xlarge, x1.32xlarge, x1e.xlarge,
 x1e.2xlarge, x1e.4xlarge, x1e.8xlarge, x1e.16xlarge, x1e.32xlarge, i2.xlarge,
 i2.2xlarge, i2.4xlarge, i2.8xlarge, i3.large, i3.xlarge, i3.2xlarge, i3.4xlarge,
 i3.8xlarge, i3.16xlarge, i3.metal, i3en.large, i3en.xlarge, i3en.2xlarge,
 i3en.3xlarge, i3en.6xlarge, i3en.12xlarge, i3en.24xlarge, i3en.metal, hi1.4xlarge,
 hs1.8xlarge, c1.medium, c1.xlarge, c3.large, c3.xlarge, c3.2xlarge, c3.4xlarge,
 c3.8xlarge, c4.large, c4.xlarge, c4.2xlarge, c4.4xlarge, c4.8xlarge, c5.large,
 c5.xlarge, c5.2xlarge, c5.4xlarge, c5.9xlarge, c5.12xlarge, c5.18xlarge,
 c5.24xlarge, c5.metal, c5d.large, c5d.xlarge, c5d.2xlarge, c5d.4xlarge,
 c5d.9xlarge, c5d.18xlarge, c5n.large, c5n.xlarge, c5n.2xlarge, c5n.4xlarge,
 c5n.9xlarge, c5n.18xlarge, cc1.4xlarge, cc2.8xlarge, g2.2xlarge, g2.8xlarge,
 g3.4xlarge, g3.8xlarge, g3.16xlarge, g3s.xlarge, g4dn.xlarge, g4dn.2xlarge,
 g4dn.4xlarge, g4dn.8xlarge, g4dn.12xlarge, g4dn.16xlarge, cg1.4xlarge, p2.xlarge,
 p2.8xlarge, p2.16xlarge, p3.2xlarge, p3.8xlarge, p3.16xlarge, p3dn.24xlarge,
 d2.xlarge, d2.2xlarge, d2.4xlarge, d2.8xlarge, f1.2xlarge, f1.4xlarge, f1.16xlarge,
 m5.large, m5.xlarge, m5.2xlarge, m5.4xlarge, m5.8xlarge, m5.12xlarge, m5.16xlarge,
 m5.24xlarge, m5.metal, m5a.large, m5a.xlarge, m5a.2xlarge, m5a.4xlarge,
 m5a.8xlarge, m5a.12xlarge, m5a.16xlarge, m5a.24xlarge, m5d.large, m5d.xlarge,
 m5d.2xlarge, m5d.4xlarge, m5d.8xlarge, m5d.12xlarge, m5d.16xlarge, m5d.24xlarge,
 m5d.metal, m5ad.large, m5ad.xlarge, m5ad.2xlarge, m5ad.4xlarge, m5ad.8xlarge,
 m5ad.12xlarge, m5ad.16xlarge, m5ad.24xlarge, h1.2xlarge, h1.4xlarge, h1.8xlarge,
 h1.16xlarge, z1d.large, z1d.xlarge, z1d.2xlarge, z1d.3xlarge, z1d.6xlarge,
 z1d.12xlarge, z1d.metal, u-6tb1.metal, u-9tb1.metal, u-12tb1.metal, u-18tb1.metal,
 u-24tb1.metal, a1.medium, a1.large, a1.xlarge, a1.2xlarge, a1.4xlarge, a1.metal,
```

```
 m5dn.large, m5dn.xlarge, m5dn.2xlarge, m5dn.4xlarge, m5dn.8xlarge, m5dn.12xlarge,
 m5dn.16xlarge, m5dn.24xlarge, m5n.large, m5n.xlarge, m5n.2xlarge, m5n.4xlarge,
 m5n.8xlarge, m5n.12xlarge, m5n.16xlarge, m5n.24xlarge, r5dn.large, r5dn.xlarge,
 r5dn.2xlarge, r5dn.4xlarge, r5dn.8xlarge, r5dn.12xlarge, r5dn.16xlarge,
 r5dn.24xlarge, r5n.large, r5n.xlarge, r5n.2xlarge, r5n.4xlarge, r5n.8xlarge,
 r5n.12xlarge, r5n.16xlarge, r5n.24xlarge.
Ipv6AddressCount: 0 # [EC2-VPC] The number of IPv6 addresses to associate with the
 primary network interface.
Ipv6Addresses: # [EC2-VPC] The IPv6 addresses from the range of the subnet to
 associate with the primary network interface.
- Ipv6Address: ''  # The IPv6 address.
KernelId: '' # The ID of the kernel.
KeyName: '' # The name of the key pair.
MaxCount: 0 # [REQUIRED] The maximum number of instances to launch.
MinCount: 0 # [REQUIRED] The minimum number of instances to launch.
Monitoring: # Specifies whether detailed monitoring is enabled for the instance.
  Enabled: true  # [REQUIRED] Indicates whether detailed monitoring is enabled.
Placement: # The placement for the instance.
  AvailabilityZone: ''  # The Availability Zone of the instance.
  Affinity: '' # The affinity setting for the instance on the Dedicated Host.
  GroupName: '' # The name of the placement group the instance is in.
  PartitionNumber: 0 # The number of the partition the instance is in.
  HostId: '' # The ID of the Dedicated Host on which the instance resides.
  Tenancy: dedicated # The tenancy of the instance (if the instance is running in a
 VPC). Valid values are: default, dedicated, host.
  SpreadDomain: '' # Reserved for future use.
RamdiskId: '' # The ID of the RAM disk to select.
SecurityGroupIds: # The IDs of the security groups.
- ''
SecurityGroups: # [default VPC] The names of the security groups.
- ''
SubnetId: '' # [EC2-VPC] The ID of the subnet to launch the instance into.
UserData: '' # The user data to make available to the instance.
AdditionalInfo: '' # Reserved.
ClientToken: '' # Unique, case-sensitive identifier you provide to ensure the
 idempotency of the request.
DisableApiTermination: true # If you set this parameter to true, you can't terminate
 the instance using the Amazon EC2 console, CLI, or API; otherwise, you can.
DryRun: true # Checks whether you have the required permissions for the action,
 without actually making the request, and provides an error response.
EbsOptimized: true # Indicates whether the instance is optimized for Amazon EBS I/O.
IamInstanceProfile: # The IAM instance profile.
  Arn: ''  # The Amazon Resource Name (ARN) of the instance profile.
  Name: '' # The name of the instance profile.
```

```
InstanceInitiatedShutdownBehavior: stop # Indicates whether an instance stops or
 terminates when you initiate shutdown from the instance (using the operating system
 command for system shutdown). Valid values are: stop, terminate.
NetworkInterfaces: # The network interfaces to associate with the instance.
- AssociatePublicIpAddress: true  # Indicates whether to assign a public IPv4
 address to an instance you launch in a VPC.
  DeleteOnTermination: true # If set to true, the interface is deleted when the
 instance is terminated.
  Description: '' # The description of the network interface.
  DeviceIndex: 0 # The position of the network interface in the attachment order.
  Groups: # The IDs of the security groups for the network interface.
  - ''
  Ipv6AddressCount: 0 # A number of IPv6 addresses to assign to the network
 interface.
  Ipv6Addresses: # One or more IPv6 addresses to assign to the network interface.
  - Ipv6Address: ''  # The IPv6 address.
  NetworkInterfaceId: '' # The ID of the network interface.
  PrivateIpAddress: '' # The private IPv4 address of the network interface.
  PrivateIpAddresses: # One or more private IPv4 addresses to assign to the network
 interface.
  - Primary: true  # Indicates whether the private IPv4 address is the primary
 private IPv4 address.
    PrivateIpAddress: '' # The private IPv4 addresses.
  SecondaryPrivateIpAddressCount: 0 # The number of secondary private IPv4
 addresses.
  SubnetId: '' # The ID of the subnet associated with the network interface.
  InterfaceType: '' # The type of network interface.
PrivateIpAddress: '' # [EC2-VPC] The primary IPv4 address.
ElasticGpuSpecification: # An elastic GPU to associate with the instance.
- Type: ''  # [REQUIRED] The type of Elastic Graphics accelerator.
ElasticInferenceAccelerators: # An elastic inference accelerator to associate with
 the instance.
- Type: ''  # [REQUIRED]  The type of elastic inference accelerator.
TagSpecifications: # The tags to apply to the resources during launch.
- ResourceType: network-interface  # The type of resource to tag. Valid values
 are: client-vpn-endpoint, customer-gateway, dedicated-host, dhcp-options, elastic-
ip, fleet, fpga-image, host-reservation, image, instance, internet-gateway,
 launch-template, natgateway, network-acl, network-interface, reserved-instances,
 route-table, security-group, snapshot, spot-instances-request, subnet, traffic-
mirror-filter, traffic-mirror-session, traffic-mirror-target, transit-gateway,
 transit-gateway-attachment, transit-gateway-route-table, volume, vpc, vpc-peering-
connection, vpn-connection, vpn-gateway.
  Tags: # The tags to apply to the resource.
  - Key: ''  # The key of the tag.
```

```
    Value: '' # The value of the tag.
LaunchTemplate: # The launch template to use to launch the instances.
  LaunchTemplateId: ''  # The ID of the launch template.
  LaunchTemplateName: '' # The name of the launch template.
  Version: '' # The version number of the launch template.
InstanceMarketOptions: # The market (purchasing) option for the instances.
  MarketType: spot  # The market type. Valid values are: spot.
  SpotOptions: # The options for Spot Instances.
    MaxPrice: ''  # The maximum hourly price you're willing to pay for the Spot
 Instances.
    SpotInstanceType: one-time # The Spot Instance request type. Valid values are:
 one-time, persistent.
    BlockDurationMinutes: 0 # The required duration for the Spot Instances (also
 known as Spot blocks), in minutes.
    ValidUntil: 1970-01-01 00:00:00 # The end date of the request.
    InstanceInterruptionBehavior: terminate # The behavior when a Spot Instance is
 interrupted. Valid values are: hibernate, stop, terminate.
CreditSpecification: # The credit option for CPU usage of the T2 or T3 instance.
  CpuCredits: ''  # [REQUIRED] The credit option for CPU usage of a T2 or T3
 instance.
CpuOptions: # The CPU options for the instance.
  CoreCount: 0  # The number of CPU cores for the instance.
  ThreadsPerCore: 0 # The number of threads per CPU core.
CapacityReservationSpecification: # Information about the Capacity Reservation
 targeting option.
  CapacityReservationPreference: none  # Indicates the instance's Capacity
 Reservation preferences. Valid values are: open, none.
  CapacityReservationTarget: # Information about the target Capacity Reservation.
    CapacityReservationId: ''  # The ID of the Capacity Reservation.
HibernationOptions: # Indicates whether an instance is enabled for hibernation.
  Configured: true  # If you set this parameter to true, your instance is enabled
 for hibernation.
LicenseSpecifications: # The license configurations.
- LicenseConfigurationArn: ''  # The Amazon Resource Name (ARN) of the license
 configuration.
```

Generazione e importazione di uno skeleton di comandi

Per generare e utilizzare un file scheletro dei parametri

1. Eseguire il comando con il parametro `--generate-cli-skeleton` per produrre JSON o YAML e indirizzare l'output a un file per salvarlo.

JSON

```
$ aws ec2 run-instances --generate-cli-skeleton input > ec2runinst.json
```

YAML

```
$ aws ec2 run-instances --generate-cli-skeleton yaml-input > ec2runinst.yaml
```

2. Apri il file scheletro dei parametri nell'editor di testo e rimuovi qualsiasi parametro non necessario. Ad esempio, è possibile eliminare il modello fino al seguente. Conferma che il file sia ancora nel formato JSON o YAML valido dopo aver rimosso gli elementi non necessari.

   JSON

   ```
   {
       "DryRun": true,
       "ImageId": "",
       "KeyName": "",
       "SecurityGroups": [
           ""
       ],
       "InstanceType": "",
       "Monitoring": {
           "Enabled": true
       }
   }
   ```

   YAML

   ```
   DryRun: true
   ImageId: ''
   KeyName: ''
   SecurityGroups:
   - ''
   InstanceType:
   Monitoring:
     Enabled: true
   ```

In questo esempio, il parametro `DryRun` viene lasciato impostato su `true` per utilizzare la funzionalità di test EC2. Questa funzione consente di testare in modo sicuro il comando senza creare o modificare effettivamente alcuna risorsa.

3. Compila i valori rimanenti con valori appropriati per il tuo scenario. In questo esempio, forniamo il tipo di istanza, il nome della chiave, il gruppo di sicurezza e l'identificatore dell'Amazon Machine Image (AMI) da utilizzare. Questo esempio prevede l'utilizzo della Regione AWS. L'AMI `ami-dfc39aef` è un'immagine a 64 bit Amazon Linux ospitata nella Regione `us-west-2`. Se utilizzi un'altra regione, devi trovare l'ID AMI corretto da utilizzare.

   JSON

   ```
   {
       "DryRun": true,
       "ImageId": "ami-dfc39aef",
       "KeyName": "mykey",
       "SecurityGroups": [
           "my-sg"
       ],
       "InstanceType": "t2.micro",
       "Monitoring": {
           "Enabled": true
       }
   }
   ```

   YAML

   ```
   DryRun: true
   ImageId: 'ami-dfc39aef'
   KeyName: 'mykey'
   SecurityGroups:
   - 'my-sg'
   InstanceType: 't2.micro'
   Monitoring:
     Enabled: true
   ```

4. Eseguire il comando con i parametri completati passando il file modello completato al parametro `--cli-input-json` o `--cli-input-yaml` utilizzando il prefisso `file://`. La AWS CLI interpreta il percorso come relativo alla directory di lavoro corrente. L'esempio AWS CLI seguente cerca il file nella directory di lavoro corrente.

JSON

```
$ aws ec2 run-instances --cli-input-json file://ec2runinst.json
```

```
A client error (DryRunOperation) occurred when calling the RunInstances
 operation: Request would have succeeded, but DryRun flag is set.
```

YAML

```
$ aws ec2 run-instances --cli-input-yaml file://ec2runinst.yaml
```

```
A client error (DryRunOperation) occurred when calling the RunInstances
 operation: Request would have succeeded, but DryRun flag is set.
```

L'errore in fase di test indica che la formattazione del JSON o YAML è corretta e i valori dei parametri sono validi. Se nell'output vengono segnalati altri problemi, correggerli e ripetere la fase precedente fino a quando non viene visualizzato il messaggio"`Request would have succeeded`".

5. Ora puoi impostare il parametro `DryRun` su `false` per disabilitare il test.

   JSON

```
{
    "DryRun": false,
    "ImageId": "ami-dfc39aef",
    "KeyName": "mykey",
    "SecurityGroups": [
        "my-sg"
    ],
    "InstanceType": "t2.micro",
    "Monitoring": {
        "Enabled": true
    }
}
```

YAML

```
DryRun: false
ImageId: 'ami-dfc39aef'
KeyName: 'mykey'
SecurityGroups:
- 'my-sg'
InstanceType: 't2.micro'
Monitoring:
  Enabled: true
```

6. Eseguire il comando, e `run-instances` avvia effettivamente un'istanza Amazon EC2 e visualizza tutte le informazioni generate dal lancio corretto. Il formato dell'output è controllato dal parametro `--output`, separatamente dal formato del modello di parametro di input.

   JSON

   ```
   $ aws ec2 run-instances --cli-input-json file://ec2runinst.json --output json
   ```

   ```
   {
       "OwnerId": "123456789012",
       "ReservationId": "r-d94a2b1",
       "Groups": [],
       "Instances": [
   ...
   ```

   YAML

   ```
   $ aws ec2 run-instances --cli-input-yaml file://ec2runinst.yaml --output yaml
   ```

   ```
   OwnerId: '123456789012'
   ReservationId: 'r-d94a2b1',
   Groups":
   - ''
   Instances:
   ...
   ```

Combinazione di file di input e parametri della riga di comando

Un file di input può essere utilizzato per tutti i parametri o può essere combinato con i parametri specificati in AWS CLI. È possibile utilizzare questa funzionalità per le impostazioni che si riutilizzano frequentemente in un file di input, mantenendo le impostazioni individuali nel comando stesso.

Negli esempi seguenti `aws ec2 run-instances` combinano l'uso di un file di input e di parametri. Forniamo il tipo di istanza, il nome della chiave, il gruppo di sicurezza, l'identificatore dell'Amazon Machine Image (AMI) da utilizzare e assegniamo la Regione AWS predefinita. L'AMI `ami-dfc39aef` è un'immagine a 64 bit Amazon Linux ospitata nella Regione `us-west-2`. Se utilizzi un'altra regione, devi trovare l'ID AMI corretto da utilizzare.

JSON

Contenuto del file JSON:

```
{
    "ImageId": "ami-dfc39aef",
    "KeyName": "mykey",
    "SecurityGroups": [
        "my-sg"
    ],
    "InstanceType": "t2.micro",
    "Monitoring": {
        "Enabled": true
    }
}
```

YAML

Contenuto del file YAML:

```
ImageId: 'ami-dfc39aef'
KeyName: 'mykey'
SecurityGroups:
- 'my-sg'
InstanceType: 't2.micro'
Monitoring:
  Enabled: true
```

L'esempio seguente utilizza il file di input in combinazione con il parametro `--dry-run` per eseguire una prova del comando per confermare se si dispone delle autorizzazioni richieste e se il file è stato compilato con valori validi.

JSON

```
$ aws ec2 run-instances --cli-input-json file://ec2runinst.json --dry-run
```

```
A client error (DryRunOperation) occurred when calling the RunInstances operation:
 Request would have succeeded, but DryRun flag is set.
```

YAML

```
$ aws ec2 run-instances --cli-input-yaml file://ec2runinst.yaml --dry-run
```

```
A client error (DryRunOperation) occurred when calling the RunInstances operation:
 Request would have succeeded, but DryRun flag is set.
```

L'esempio seguente utilizza quindi lo stesso file di input, ma con il parametro `--no-dry-run` per eseguire il comando per intero.

JSON

```
$ aws ec2 run-instances --cli-input-json file://ec2runinst.json --no-dry-run --
output json
```

```
{
    "OwnerId": "123456789012",
    "ReservationId": "r-d94a2b1",
    "Groups": [],
    "Instances": [
...
```

YAML

```
$ aws ec2 run-instances --cli-input-yaml file://ec2runinst.yaml --no-dry-run --
output yaml
```

```
OwnerId: '123456789012'
ReservationId: 'r-d94a2b1',
Groups":
- ''
Instances:
...
```

Utilizzo della sintassi abbreviata in AWS CLI

AWS Command Line Interface (AWS CLI) può accettare molti parametri delle opzioni in formato JSON. Tuttavia, può essere noioso immettere lunghi elenchi o strutture JSON nella riga di comando. Per agevolare l'operazione, AWS CLI supporta anche una sintassi abbreviata che consente una rappresentazione più semplice dei parametri delle opzioni rispetto a quella che prevede l'utilizzo del formato JSON completo.

Argomenti

- Parametri della struttura con coppie chiave-valore
- Caricamento di un file come valore di sintassi abbreviata
- Utilizzo della sintassi abbreviata con l'AWS CLI

Parametri della struttura con coppie chiave-valore

La sintassi abbreviata nell'AWS CLI rende più semplice per gli utenti immettere parametri piatti (strutture non nidificate). Il formato è un elenco separato da virgole di coppie chiave-valore. Assicurati di utilizzare le regole di citazione e di escape appropriate per il tuo terminale poiché la sintassi abbreviata è costituita da stringhe.

Linux or macOS

```
--option key1=value1,key2=value2,key3=value3
```

È l'equivalente dell'esempio seguente in formato JSON.

```
--option '{"key1":"value1","key2":"value2","key3":"value3"}'
```

Windows

```
--option "key1=value1,key2=value2,key3=value3"
```

È l'equivalente dell'esempio seguente in formato JSON.

```
--option '{"key1":"value1","key2":"value2","key3":"value3"}'
```

Le coppie chiave-valore separate da virgole non devono contenere spazi vuoti. Di seguito è riportato un esempio del comando `update-table` di Amazon DynamoDB con l'opzione `--provisioned-throughput` specificata in formato abbreviato.

```
$ aws dynamodb update-table \
    --provisioned-throughput ReadCapacityUnits=15,WriteCapacityUnits=10 \
    --table-name MyDDBTable
```

È l'equivalente dell'esempio seguente in formato JSON.

```
$ aws dynamodb update-table \
    --provisioned-throughput '{"ReadCapacityUnits":15,"WriteCapacityUnits":10}' \
    --table-name MyDDBTable
```

Caricamento di un file come valore di sintassi abbreviata

Quando un valore è grande o complesso, è spesso più facile caricarlo come valore. Per caricare un file come valore di sintassi abbreviata, la formattazione cambierà leggermente. Invece di `key=value`, utilizzerai l'operatore `@=` al posto dell'operatore `=`. `@=` significa per la AWS CLI che il valore deve essere letto come un percorso di file e non come una stringa. Quando si caricano dei file in sintassi abbreviata, si applicano le normali regole di formattazione dei file AWS CLI. L'esempio seguente mostra una coppia chiave-valore che carica un file per il relativo valore.

Linux or macOS

```
--option key@=file://template.txt
```

Windows

```
--option "key1@=file://template.txt"
```

L'esempio seguente mostra il caricamento di un file di certificato per il comando `aws rolesanywhere create-trust-anchor`.

```
$ aws rolesanywhere create-trust-anchor --name TrustAnchor \
    --source sourceData={x509CertificateData@=file://root-ca.crt},sourceType="CERTIFICATE_BUNDLE"  \
    --enabled
```

Utilizzo della sintassi abbreviata con l'AWS CLI

Puoi specificare i parametri di input in formato elenco in due modi: JSON o abbreviato. La sintassi abbreviata di AWS CLI è stata progettata per rendere più facile il trasferimento di elenchi di numeri, di stringhe o di strutture non nidificate.

Qui viene mostrato il formato di base,in cui i valori nell'elenco sono separati da un solo spazio.

```
--option value1 value2 value3
```

È l'equivalente dell'esempio seguente in formato JSON.

```
--option '[value1,value2,value3]'
```

Come accennato in precedenza, è possibile specificare un elenco di numeri, un elenco di stringhe o un elenco di strutture non nidificate in forato abbreviato. Di seguito viene riportato un esempio del comando `stop-instances` per Amazon Elastic Compute Cloud (Amazon EC2), dove il parametro di input (elenco di stringhe) per l'opzione `--instance-ids` viene specificato in formato abbreviato.

```
$ aws ec2 stop-instances \
    --instance-ids i-1486157a i-1286157c i-ec3a7e87
```

È l'equivalente dell'esempio seguente in formato JSON.

```
$ aws ec2 stop-instances \
    --instance-ids '["i-1486157a","i-1286157c","i-ec3a7e87"]'
```

L'esempio successivo mostra il comando `create-tags` di Amazon EC2 che utilizza un elenco di strutture non nidificate per l'opzione `--tags`. L'opzione `--resources` specifica l'ID dell'istanza da contrassegnare.

```
$ aws ec2 create-tags \
```

```
    --resources i-1286157c \
    --tags Key=My1stTag,Value=Value1 Key=My2ndTag,Value=Value2
 Key=My3rdTag,Value=Value3
```

È l'equivalente dell'esempio seguente in formato JSON. Il parametro JSON viene scritto su più righe per maggiore leggibilità.

```
$ aws ec2 create-tags \
    --resources i-1286157c \
    --tags '[
        {"Key": "My1stTag", "Value": "Value1"},
        {"Key": "My2ndTag", "Value": "Value2"},
        {"Key": "My3rdTag", "Value": "Value3"}
    ]'
```

Attivazione e utilizzo dei prompt dei comandi in AWS CLI

È possibile fare in modo che la versione 2 della AWS CLI richieda comandi, parametri e risorse quando si esegue un comando `aws`.

Argomenti

- Come funziona
- Funzionalità di prompt automatico
- Modalità di prompt automatico
- Configurare il prompt automatico

Come funziona

Se abilitato, il prompt automatico consente di utilizzare il tasto INVIO per completare un comando immesso parzialmente. Dopo aver premuto il tasto INVIO, vengono suggeriti comandi, parametri e risorse in base a ciò che si continua a digitare. I suggerimenti elencano il nome del comando, del parametro o della risorsa a sinistra e una descrizione a destra. Per selezionare e utilizzare un suggerimento, utilizza i tasti freccia per evidenziare una riga, quindi premi il tasto SPAZIO. Dopo aver immesso il comando, premi INVIO per utilizzare il comando. L'esempio seguente mostra l'aspetto di un elenco suggerito dal prompt automatico.

```
$ aws
> aws a
```

```
      accessanalyzer                Access Analyzer
      acm                           AWS Certificate Manager
      acm-pca                       AWS Certificate Manager Private Certificate
Authority
      alexaforbusiness              Alexa For Business
      amplify                       AWS Amplify
```

Funzionalità di prompt automatico

Il prompt automatico contiene le seguenti utili funzionalità:

Pannello della documentazione

Fornisce la documentazione di aiuto per il comando corrente. Per aprire la documentazione, premi il tasto F3.

Completamento dei comandi

Suggerisce i comandi `aws` da utilizzare. Per visualizzare un elenco, inserisci parzialmente il comando. L'esempio seguente è la ricerca di un servizio che inizia con la lettera `a`.

```
$ aws
> aws a
       accessanalyzer                Access Analyzer
       acm                           AWS Certificate Manager
       acm-pca                       AWS Certificate Manager Private Certificate
 Authority
       alexaforbusiness              Alexa For Business
       amplify                       AWS Amplify
```

Completamento dei parametri

Dopo aver digitato un comando, il prompt automatico inizia a suggerire i parametri. Le descrizioni dei parametri includono il tipo di valore e una descrizione del parametro. I parametri obbligatori vengono elencati per primi e sono etichettati come obbligatori. L'esempio seguente mostra l'elenco di parametri con prompt automatico per `aws dynamodb describe-table`.

```
$ aws dynamodb describe-table
> aws dynamodb describe-table
                              --table-name (required)  [string] The name of the
 table to describe.
                               --cli-input-json         [string] Reads arguments
 from the JSON string provided. The JSON string follows the format provide...
```

```
                                --cli-input-yaml        [string] Reads arguments
 from the YAML string provided. The YAML string follows the format provide...
                                --generate-cli-skeleton  [string] Prints a JSON
 skeleton to standard output without sending an API request. If provided wit...
```

Completamento delle risorse

Il prompt automatico effettua chiamate API AWS utilizzando le proprietà delle risorse AWS disponibili per suggerire i valori delle risorse. Ciò consente al prompt automatico di suggerire possibili risorse di tua proprietà quando inserisci i parametri. Nell'esempio seguente, il prompt automatico elenca i nomi delle tabelle durante la compilazione del parametro `--table-name` per il comando `aws dynamodb describe-table`.

```
$ aws dynamodb describe-table
> aws dynamodb describe-table --table-name
                                                Table1
                                                Table2
                                                Table3
```

Completamento abbreviato

Per i parametri che utilizzano una sintassi abbreviata, il prompt automatico suggerisce i valori da utilizzare. Nell'esempio seguente, il prompt automatico elenca i valori di sintassi abbreviata per il parametro `--placement` nel comando `aws ec2 run-instances`.

```
$ aws ec2 run-instances
> aws ec2 run-instances --placement
 AvailabilityZone=      [string] The Availability Zone of the instance. If not
 specified, an Availability Zone wil...
 Affinity=              [string] The affinity setting for the instance on the
 Dedicated Host. This parameter is no...
 GroupName=             [string] The name of the placement group the instance is in.
 PartitionNumber=       [integer] The number of the partition the instance is in.
 Valid only if the placement grou...
```

Completamento del file

Durante la compilazione dei parametri nei comandi `aws`, il completamento automatico suggerisce i nomi di file locali dopo aver usato il prefisso `file://` o `fileb://`. Nell'esempio seguente, il prompt automatico suggerisce i file locali dopo aver immesso `--item file://` per il comando `aws ec2 run-instances`.

```
$ aws ec2 run-instances
> aws ec2 run-instances --item file://
                        item1.txt
                        file1.json
                        file2.json
```

Completamento della Regione

Quando si utilizza il parametro globale `--region`, il prompt automatico elenca le possibili Regioni tra cui scegliere. Nell'esempio seguente, il prompt automatico suggerisce le Regioni in ordine alfabetico dopo aver immesso `--region` per il comando `aws dynamodb list-tables`.

```
$ aws dynamodb list-tables
> aws dynamodb list-tables --region
                                  af-south-1
                                  ap-east-1
                                  ap-northeast-1
                                  ap-northeast-2
```

Completamento del profilo

Quando si utilizza il parametro globale `--profile`, il prompt automatico elenca i profili. Nell'esempio seguente, il prompt automatico suggerisce i profili dopo aver immesso `--profile` per il comando `aws dynamodb list-tables`.

```
$ aws dynamodb list-tables
> aws dynamodb list-tables --profile
                                  profile1
                                  profile2
                                  profile3
```

Ricerche fuzzy

Comandi e valori completi che contengono un set di caratteri specifico. Nell'esempio seguente, il prompt automatico suggerisce le Regioni che contengono `eu` dopo aver immesso `--region eu` per il comando `aws dynamodb list-tables`.

```
$ aws dynamodb list-tables
> aws dynamodb list-tables --region west
                                       eu-west-1
```

```
                                        eu-west-2
                                        eu-west-3
                                        us-west-1
```

Cronologia

Per visualizzare ed eseguire i comandi utilizzati in precedenza in modalità prompt automatico, premere CTRL + R. La cronologia elenca i comandi precedenti che è possibile selezionare utilizzando i tasti freccia. Nell'esempio seguente, viene visualizzata la cronologia della modalità di prompt automatico.

```
$ aws
> aws
        dynamodb list-tables
        s3 ls
```

Modalità di prompt automatico

Il prompt automatico per la versione 2 della AWS CLI dispone di 2 modalità configurabili:

- Modalità completa: utilizza il prompt automatico ogni volta che tenti di eseguire un comando `aws`, sia che lo si richiami manualmente utilizzando il parametro `--cli-auto-prompt` sia che venga abilitato in modo permanente. Include il dover premere INVIO dopo un comando completo o incompleto.
- Modalità parziale: utilizza il prompt automatico se un comando è incompleto o non può essere eseguito a causa di errori di convalida lato client. Questa modalità è particolarmente utile se sono disponibili script e runbook preesistenti o per ricevere un prompt automatico solo per i comandi che non si conoscono piuttosto che per ogni comando.

Configurare il prompt automatico

Per configurare il prompt automatico puoi utilizzare i seguenti metodi in ordine di precedenza:

- Opzioni della riga di comando abilitano o disabilitano il prompt automatico per un singolo comando. Utilizza `--cli-auto-prompt` per chiamare il prompt automatico e `--no-cli-auto-prompt` per disabilitare il prompt automatico.
- Le variabili di ambiente utilizzano la variabile `aws_cli_auto_prompt`.

- I file di configurazione condivisi utilizzano l'impostazione `cli_auto_prompt`.

Controllo dell'output dei comandi nella AWS CLI

In questa sezione vengono descritti i diversi modi per controllare l'output dall'AWS Command Line Interface (AWS CLI). La personalizzazione dell'output della AWS CLI nel terminale può migliorare la leggibilità, semplificare l'automazione degli script e facilitare la navigazione tra set di dati più grandi.

La AWS CLI supporta diversi formati di output, tra cui `json`, `text`, `yaml`, e `table`. Alcuni servizi prevedono la paginazione lato server per i relativi dati e la AWS CLI fornisce funzionalità lato client proprie per opzioni di paginazione aggiuntive.

Infine, la AWS CLI dispone di filtri sia lato server che lato client utilizzabili singolarmente o insieme per filtrare l'output della AWS CLI.

Argomenti

- Output sensibili
- Opzioni di output lato server o lato client
- Impostazione del formato di output della AWS CLI
- Utilizzo delle opzioni di paginazione della AWS CLI
- Uscita di filtraggio nel AWS CLI

Output sensibili

Alcune operazioni della AWS CLI potrebbero restituire informazioni potenzialmente riservate, incluse le informazioni provenienti da variabili di ambiente. L'esposizione di queste informazioni potrebbe rappresentare un rischio per la sicurezza in determinati scenari; ad esempio, le informazioni potrebbero essere incluse nei log di integrazione e implementazione continua (CI/CD). È quindi importante verificare quando includere tali output nei log e sopprimerli quando non sono necessari.

Per ulteriori informazioni sulla protezione dei dati sensibili, consulta the section called "Protezione dei dati".

Prendi in considerazione le seguenti best practice:

- Prendi in considerazione la possibilità di recuperare a livello di codice i tuoi segreti da un archivio di segreti, ad esempio Gestione dei segreti AWS.

- Effettua una revisione dei contenuti dei log di compilazione per assicurarti che non contengano informazioni riservate. Prendi in considerazione approcci come il piping in `/dev/null` o l'acquisizione dell'output come variabile bash o PowerShell per sopprimere gli output dei comandi.

  Quello che segue è un esempio di bash per reindirizzare l'output, ma non gli errori, a `/dev/null`:

  ```
  $ aws s3 ls > /dev/null
  ```

  Per informazioni specifiche sulla soppressione dell'output per il terminale, consulta la documentazione per l'utente del terminale che utilizzi.
- Prendi in considerazione l'accesso ai tuoi log e definisci l'accesso in modo appropriato per il tuo caso d'uso.

Opzioni di output lato server o lato client

La AWS CLI dispone di filtri sia lato server che lato client utilizzabili singolarmente o insieme per filtrare l'output della AWS CLI. Il filtraggio lato server viene elaborato per primo e restituisce l'output per il filtraggio lato client. Il filtraggio lato server è supportato dall'API del servizio. Il filtraggio lato client è supportato dal client della con AWS CLI il parametro `--query`.

Le opzioni di output lato server sono funzionalità supportate direttamente dall'API Servizio AWS. Tutti i dati filtrati o paginati non vengono inviati al client, eventualmente accelerando i tempi di risposta HTTP e migliorando la larghezza di banda per set di dati più grandi.

Le opzioni di output lato client sono funzionalità create da AWS CLI. Tutti i dati vengono inviati al client, quindi la AWS CLI filtra o suddivide in pagine il contenuto visualizzato. Le operazioni lato client non consentono di risparmiare sulla velocità o sulla larghezza di banda per set di dati più grandi.

Quando le opzioni lato server e lato client vengono utilizzate insieme, le operazioni lato server vengono prima completate e poi inviate al client per le operazioni lato client. In questo modo si sfruttano i potenziali risparmi di velocità e larghezza di banda delle opzioni lato server, utilizzando al contempo le funzionalità aggiuntive della AWS CLI per ottenere l'output desiderato.

Impostazione del formato di output della AWS CLI

Questo argomento descrive i diversi formati di output per AWS Command Line Interface (AWS CLI). La AWS CLI supporta i seguenti formati di output:

- **`json`**: l'output è formattato come una stringa JSON.

- **`yaml`**: l'output è formattato come una stringa YAML.
- **`yaml-stream`**: l'output viene inviato in streaming e formattato come una stringa YAML. Lo streaming consente una gestione più rapida di tipi di dati di grandi dimensioni.
- **`text`**: l'output è formattato come più righe di valori di stringa separati da tabulazioni. Questa formattazione può essere utile per passare l'output a un elaboratore di testi, ad esempio `grep`, `sed` o `awk`.
- **`table`**: l'output è formattato come una tabella utilizzando i caratteri +|- per formare i bordi delle celle. In genere presenta le informazioni in un formato comprensibile molto più semplice da leggere rispetto ad altri, ma non altrettanto utile a livello programmatico.

Come scegliere il formato di output

Come illustrato nell'argomento relativo alla configurazione, puoi specificare il formato di output in tre modi diversi:

- Utilizzo dell'opzione **`output`** in un profilo denominato nel file **`config`**: l'esempio seguente imposta il formato di output predefinito su `text`.

```
[default]
output=text
```

- Utilizzo della variabile d'ambiente **`AWS_DEFAULT_OUTPUT`**: l'output seguente imposta il formato su `table` per i comandi di questa sessione dalla riga di comando fino a quando la variabile viene modificata o la sessione termina. L'utilizzo di questa variabile di ambiente sostituisce il valore impostato nel file `config`.

```
$ export AWS_DEFAULT_OUTPUT="table"
```

- Utilizzo dell'opzione **`--output`** sulla riga di comando: l'esempio seguente imposta l'output di un solo comando su `json`. L'utilizzo di questa opzione con il comando sostituisce eventuali variabili di ambiente attualmente impostate o il valore nel file `config`.

```
$ aws swf list-domains --registration-status REGISTERED --output json
```

> **Important**
>
> Il tipo di output specificato influenza il funzionamento dell'opzione `--query`.

- Se si specifica `--output text`, l'output viene paginato prima dell'applicazione del filtro `--query` e l'AWS CLI esegue la query una volta in ogni pagina dell'output. Per questo motivo, la query include il primo elemento corrispondente in ogni pagina, eventualmente generando un output aggiuntivo inaspettato. È inoltre possibile utilizzare altri strumenti a riga di comando, ad esempio `head` o `tail` per filtrare ulteriormente l'output.
- Se si specifica `--output json`, `--output yaml` o `--output yaml-stream`, l'output viene completamente elaborato come una struttura singola nativa prima dell'applicazione del filtro `--query`. La AWS CLI esegue la query solo una volta sull'intera struttura, producendo un risultato filtrato che viene quindi emesso.

Formato di output JSON

JSON è il formato di output predefinito dell'AWS CLI. La maggior parte dei linguaggi di programmazione può decodificare facilmente le stringhe JSON utilizzando funzioni integrate o con librerie disponibili pubblicamente. È possibile combinare l'output JSON con l' opzione --query in modi potenti per filtrare e formattare l'output formattato JSON dell'AWS CLI.

Per un filtro più avanzato che potrebbe non essere eseguibile con `--query`, è possibile prendere in considerazione `jq`, un processore JSON a riga di comando. Il tutorial ufficiale è disponibile per il download all'indirizzo http://stedolan.github.io/jq/.

Di seguito è riportato un esempio di output JSON:

```
$ aws iam list-users --output json
```

```
{
    "Users": [
        {
            "Path": "/",
            "UserName": "Admin",
            "UserId": "AIDA1111111111EXAMPLE",
            "Arn": "arn:aws:iam::123456789012:user/Admin",
            "CreateDate": "2014-10-16T16:03:09+00:00",
            "PasswordLastUsed": "2016-06-03T18:37:29+00:00"
        },
        {
            "Path": "/backup/",
            "UserName": "backup-user",
```

```
            "UserId": "AIDA2222222222EXAMPLE",
            "Arn": "arn:aws:iam::123456789012:user/backup/backup-user",
            "CreateDate": "2019-09-17T19:30:40+00:00"
        },
        {
            "Path": "/",
            "UserName": "cli-user",
            "UserId": "AIDA3333333333EXAMPLE",
            "Arn": "arn:aws:iam::123456789012:user/cli-user",
            "CreateDate": "2019-09-17T19:11:39+00:00"
        }
    ]
}
```

Il formato YAML dell'output

YAML è una buona scelta per gestire l'output a livello di codice con strumenti e servizi che emettono o consumano stringhe formattate YAML, ad esempio CloudFormation, con il supporto per modelli formattati YAML.

Per un filtro più avanzato che potrebbe non essere possibile eseguire con `--query`, è possibile prendere in considerazione `yq`, un processore YAML della riga di comando. Puoi scaricare `yq` nel repository yq su GitHub.

Segue un esempio di output YAML.

```
$ aws iam list-users --output yaml
```

```
Users:
- Arn: arn:aws:iam::123456789012:user/Admin
  CreateDate: '2014-10-16T16:03:09+00:00'
  PasswordLastUsed: '2016-06-03T18:37:29+00:00'
  Path: /
  UserId: AIDA1111111111EXAMPLE
  UserName: Admin
- Arn: arn:aws:iam::123456789012:user/backup/backup-user
  CreateDate: '2019-09-17T19:30:40+00:00'
  Path: /backup/
  UserId: AIDA2222222222EXAMPLE
  UserName: arq-45EFD6D1-CE56-459B-B39F-F9C1F78FBE19
- Arn: arn:aws:iam::123456789012:user/cli-user
  CreateDate: '2019-09-17T19:30:40+00:00'
```

```
  Path: /
  UserId: AIDA3333333333EXAMPLE
  UserName: cli-user
```

Formato di output dei flussi YAML

Il formato `yaml-stream` sfrutta il formato YAML fornendo al contempo una visualizzazione più reattiva/veloce di set di dati di grandi dimensioni mediante lo streaming dei dati. È possibile iniziare a visualizzare e utilizzare i dati YAML prima del download dell'intera query.

Per un filtro più avanzato che potrebbe non essere possibile eseguire con `--query`, è possibile prendere in considerazione `yq`, un processore YAML della riga di comando. Puoi scaricare `yq` nel repository yq su GitHub.

Segue un esempio di output `yaml-stream`.

```
$ aws iam list-users --output yaml-stream
```

```
- IsTruncated: false
  Users:
  - Arn: arn:aws:iam::123456789012:user/Admin
    CreateDate: '2014-10-16T16:03:09+00:00'
    PasswordLastUsed: '2016-06-03T18:37:29+00:00'
    Path: /
    UserId: AIDA1111111111EXAMPLE
    UserName: Admin
  - Arn: arn:aws:iam::123456789012:user/backup/backup-user
    CreateDate: '2019-09-17T19:30:40+00:00'
    Path: /backup/
    UserId: AIDA2222222222EXAMPLE
    UserName: arq-45EFD6D1-CE56-459B-B39F-F9C1F78FBE19
  - Arn: arn:aws:iam::123456789012:user/cli-user
    CreateDate: '2019-09-17T19:30:40+00:00'
    Path: /
    UserId: AIDA3333333333EXAMPLE
    UserName: cli-user
```

Di seguito è riportato un esempio di output `yaml-stream` combinato con l'utilizzo del parametro `--page-size` per paginare il contenuto YAML in streaming.

```
$ aws iam list-users --output yaml-stream --page-size 2
```

```
- IsTruncated: true
  Marker: ab1234cdef5ghi67jk8lmo9p/
q012rs3t445uv6789w0x1y2z/345a6b78c9d00/1efgh234ij56klmno78pqrstu90vwxyx
  Users:
  - Arn: arn:aws:iam::123456789012:user/Admin
    CreateDate: '2014-10-16T16:03:09+00:00'
    PasswordLastUsed: '2016-06-03T18:37:29+00:00'
    Path: /
    UserId: AIDA1111111111EXAMPLE
    UserName: Admin
  - Arn: arn:aws:iam::123456789012:user/backup/backup-user
    CreateDate: '2019-09-17T19:30:40+00:00'
    Path: /backup/
    UserId: AIDA2222222222EXAMPLE
    UserName: arq-45EFD6D1-CE56-459B-B39F-F9C1F78FBE19
- IsTruncated: false
  Users:
  - Arn: arn:aws:iam::123456789012:user/cli-user
    CreateDate: '2019-09-17T19:30:40+00:00'
    Path: /
    UserId: AIDA3333333333EXAMPLE
    UserName: cli-user
```

Formato di output di testo

Il formato `text` organizza l'output dell'AWS CLI in righe delimitate da tabulazione. Il formato funziona in modo ottimale con gli strumenti di testo Unix tradizionali, ad esempio `grep`, `sed` e `awk`, e con l'elaborazione del testo eseguita da PowerShell.

Il formato di output `text` segue la struttura di base riportata di seguito. Le colonne sono disposte in ordine alfabetico in base ai nomi di chiave corrispondenti dell'oggetto JSON sottostante.

```
IDENTIFIER  sorted-column1 sorted-column2
IDENTIFIER2 sorted-column1 sorted-column2
```

Segue un esempio di output `text`. Ogni campo è una scheda separata dalle altre, con una scheda aggiuntiva dove c'è un campo vuoto.

```
$ aws iam list-users --output text
```

```
USERS   arn:aws:iam::123456789012:user/Admin                2014-10-16T16:03:09+00:00
 2016-06-03T18:37:29+00:00   /          AIDA1111111111EXAMPLE   Admin
USERS   arn:aws:iam::123456789012:user/backup/backup-user   2019-09-17T19:30:40+00:00
                            /backup/   AIDA2222222222EXAMPLE   backup-user
USERS   arn:aws:iam::123456789012:user/cli-user             2019-09-17T19:11:39+00:00
                            /          AIDA3333333333EXAMPLE   cli-user
```

La quarta colonna è il campo `PasswordLastUsed` ed è vuota per le ultime due voci perché gli utenti non accedono mai alla Console di gestione AWS.

Important

Se specifichi l'output *`text`*, ti consigliamo di utilizzare sempre l'opzione *`--query`* per garantire un comportamento coerente.
Questo perché il formato di testo ordina alfabeticamente le colonne di output in base al nome della chiave dell'oggetto JSON sottostante restituito dal servizioAWS e risorse simili potrebbero non avere gli stessi nomi di chiavi. Ad esempio, una rappresentazione JSON di un'istanza Amazon EC2 basata su Linux potrebbe contenere elementi non presenti nella rappresentazione JSON di un'istanza basata su Windows o viceversa. Inoltre, negli aggiornamenti futuri delle risorse potrebbero essere aggiunti o eliminati elementi chiave-valore che modificano l'ordinamento della colonna. In questa situazione `--query` potenzia la funzionalità dell'output di testo `text` per consentire il controllo completo sul formato dell'output.
Nell'esempio seguente, il comando specifica gli elementi da visualizzare e definisce l'ordinamento delle colonne con la notazione elenco `[key1, key2, ...]`. In questo modo gli utenti hanno la certezza che i valori chiave siano visualizzati sempre nella colonna prevista. Vediamo infine in che modo l'AWS CLI restituisce `None` come valore per le chiavi non esistenti.

```
$ aws iam list-users --output text --query 'Users[*].
[UserName,Arn,CreateDate,PasswordLastUsed,UserId]'
```

```
Admin         arn:aws:iam::123456789012:user/Admin
 2014-10-16T16:03:09+00:00   2016-06-03T18:37:29+00:00   AIDA1111111111EXAMPLE
backup-user   arn:aws:iam::123456789012:user/backup-user
 2019-09-17T19:30:40+00:00   None                        AIDA2222222222EXAMPLE
```

```
cli-user      arn:aws:iam::123456789012:user/cli-backup
 2019-09-17T19:11:39+00:00   None                          AIDA3333333333EXAMPLE
```

L'esempio seguente mostra in che modo utilizzare `grep` e `awk` con l'output `text` dal comando `aws ec2 describe-instances`. Il primo comando visualizza la zona di disponibilità, lo stato corrente e l'ID istanza di ogni istanza nell'output `text`. Il secondo comando elabora l'output per visualizzare solo gli ID istanza di tutte le istanze in esecuzione nella zona di disponibilità `us-west-2a`.

```
$ aws ec2 describe-instances --query 'Reservations[*].Instances[*].
[Placement.AvailabilityZone, State.Name, InstanceId]' --output text
```

```
us-west-2a      running i-4b41a37c
us-west-2a      stopped i-a071c394
us-west-2b      stopped i-97a217a0
us-west-2a      running i-3045b007
us-west-2a      running i-6fc67758
```

```
$ aws ec2 describe-instances --query 'Reservations[*].Instances[*].
[Placement.AvailabilityZone, State.Name, InstanceId]' --output text | grep us-west-2a |
 grep running | awk '{print $3}'
```

```
i-4b41a37c
i-3045b007
i-6fc67758
```

L'esempio seguente fa un passo in più e mostra non solo come filtrare l'output, ma anche come utilizzarlo per automatizzare la modifica dei tipi di istanza per ogni istanza arrestata.

```
$ aws ec2 describe-instances --query 'Reservations[*].Instances[*].[State.Name,
 InstanceId]' --output text |
> grep stopped |
> awk '{print $2}' |
> while read line;
> do aws ec2 modify-instance-attribute --instance-id $line --instance-type '{"Value":
 "m1.medium"}';
> done
```

L'output `text` può essere utile anche in PowerShell. Poiché le colonne nell'output `text` sono delimitate da tabulazioni, è possibile suddividere facilmente l'output in un array utilizzando il delimitatore `` `t `` di PowerShell. Il comando seguente mostra il valore della terza colonna (`InstanceId`) se la prima colonna (`AvailabilityZone`) corrisponde alla stringa `us-west-2a`.

```
PS C:\>aws ec2 describe-instances --query 'Reservations[*].Instances[*].
[Placement.AvailabilityZone, State.Name, InstanceId]' --output text |
%{if ($_.split("`t")[0] -match "us-west-2a") { $_.split("`t")[2]; } }
```

```
-4b41a37c
i-a071c394
i-3045b007
i-6fc67758
```

Nota che anche se l'esempio precedente mostra come utilizzare il parametro `--query` per analizzare gli oggetti JSON sottostanti ed estrarre la colonna desiderata, PowerShell ha la propria capacità di gestire JSON se la compatibilità multipiattaforma non costituisce un problema. Invece di gestire l'output come testo, come richiesto dalla maggior parte delle shell di comando, PowerShell consente di utilizzare il cmdlet `ConvertFrom-JSON` per produrre un oggetto strutturato gerarchicamente. È quindi possibile accedere direttamente al membro desiderato da tale oggetto.

```
(aws ec2 describe-instances --output json | ConvertFrom-
Json).Reservations.Instances.InstanceId
```

Tip

Se l'output è di testo e si filtra l'output in un unico campo utilizzando il parametro `--query`, l'output è una singola riga di valori separati da tabulazione. Per ottenere ogni valore su una riga separata, puoi inserire il campo di output tra parentesi, come mostrato negli esempi seguenti:

Output a riga singola, separato da tabulazione:

```
$ aws iam list-groups-for-user --user-name susan  --output text --query
 "Groups[].GroupName"
```

```
HRDepartment    Developers      SpreadsheetUsers  LocalAdmins
```

Ogni valore nella propria riga inserendo `[GroupName]` tra parentesi:

```
$ aws iam list-groups-for-user --user-name susan  --output text --query
 "Groups[].[GroupName]"
```

```
HRDepartment
Developers
SpreadsheetUsers
LocalAdmins
```

Formato di output di tabella

Il formato `table` produce rappresentazioni leggibili degli output complessi di AWS CLI in formato tabulare.

```
$ aws iam list-users --output table
```

```
-------------------------------------------------------------------------------------------------
|
 ListUsers                                                                      |
+-----------------------------------------------------------------------------------------------
+
||
 Users                                                                        ||
|+-----------------------------------------------------+--------------------------
+--------------------------+----------+----------------------+-------------+|
||                         Arn                          |       CreateDate          |
 PasswordLastUsed       |   Path    |        UserId         |   UserName  ||
|+-----------------------------------------------------+--------------------------
+--------------------------+----------+----------------------+-------------+|
||  arn:aws:iam::123456789012:user/Admin              | 2014-10-16T16:03:09+00:00 |
 2016-06-03T18:37:29+00:00 | /        | AIDA1111111111EXAMPLE | Admin       ||
||  arn:aws:iam::123456789012:user/backup/backup-user | 2019-09-17T19:30:40+00:00 |
                       | /backup/ | AIDA2222222222EXAMPLE | backup-user ||
||  arn:aws:iam::123456789012:user/cli-user           | 2019-09-17T19:11:39+00:00 |
                       | /        | AIDA3333333333EXAMPLE | cli-user    ||
+-----------------------------------------------------------------------------------------------
+
```

Puoi combinare l'opzione `--query` con il formato `table` per visualizzare un set di elementi preselezionati dall'output non elaborato. Osserva le differenze di output nelle notazioni dizionario ed elenco: nel primo esempio, i nomi delle colonne sono in ordine alfabetico, mentre nel secondo esempio le colonne senza nome sono ordinate in base a quanto definito dall'utente. Per ulteriori informazioni sull'opzione `--query`, consulta Uscita di filtraggio nel AWS CLI.

```
$ aws ec2 describe-volumes --query 'Volumes[*].
{ID:VolumeId,InstanceId:Attachments[0].InstanceId,AZ:AvailabilityZone,Size:Size}' --
output table
```

```
----------------------------------------------------
|                  DescribeVolumes                 |
+------------+---------------+-------------+-------+
|     AZ     |      ID       | InstanceId  | Size  |
+------------+---------------+-------------+-------+
|  us-west-2a|  vol-e11a5288 |  i-a071c394 |  30   |
|  us-west-2a|  vol-2e410a47 |  i-4b41a37c |  8    |
+------------+---------------+-------------+-------+
```

```
$ aws ec2 describe-volumes --query 'Volumes[*].
[VolumeId,Attachments[0].InstanceId,AvailabilityZone,Size]' --output table
```

```
-----------------------------------------------------
|                 DescribeVolumes                   |
+--------------+-------------+-------------+-----+
|  vol-e11a5288|  i-a071c394 |  us-west-2a |  30 |
|  vol-2e410a47|  i-4b41a37c |  us-west-2a |  8  |
+--------------+-------------+-------------+-----+
```

Utilizzo delle opzioni di paginazione della AWS CLI

In questo argomento vengono descritti i diversi modi per paginare l'output dall'AWS Command Line Interface (AWS CLI).

Esistono principalmente due modi per controllare la paginazione dalla AWS CLI.

- Utilizzo dei parametri di paginazione lato server.
- Utilizzo del programma di paging lato client di output predefinito.

I parametri di paginazione lato server elaborano e qualsiasi output viene inviato alla paginazione lato client.

Paginazione lato server

Per i comandi che possono restituire un elenco di voci di grandi dimensioni, AWS CLI aggiunge diverse opzioni lato server opzioni che consentono di controllare il numero di voci incluse nell'output quando la AWS CLI chiama un'API del servizio per compilare l'elenco. La paginazione lato server in AWS CLI è abilitata dall'API del servizio AWS, pertanto queste opzioni funzionano solo se l'API del servizio le abilita.

Le opzioni per la maggior parte dei comandi AWS CLI includono le seguenti opzioni:

- Come utilizzare il parametro --no-paginate
- Come utilizzare il parametro --page-size
- Come utilizzare il parametro --max-items
- Come utilizzare il parametro --starting-token

Per impostazione predefinita, AWS CLI utilizza una dimensione di pagina determinata dal singolo servizio e recupera tutti gli elementi disponibili. Ad esempio, Amazon S3 ha una dimensione di pagina predefinita di 1000. Se esegui `aws s3api list-objects` in un bucket che contiene 3500 oggetti, la AWS CLI effettua automaticamente quattro chiamate a Amazon S3, gestendo la logica di paginazione specifica del servizio in background e restituendo tutti i 3500 oggetti nell'output finale.

Per informazioni sulla paginazione lato server di uno specifico comando, consulta la AWS CLI version 2 reference guide.

Come utilizzare il parametro --no-paginate

L'opzione `--no-paginate` disabilita i token di paginazione seguenti sul lato client. Quando si utilizza un comando, per impostazione predefinita la AWS CLI effettua automaticamente più chiamate per restituire tutti i risultati possibili per creare la paginazione. Una chiamata per ogni pagina. Disabilitando la paginazione, la AWS CLI effettua una sola chiamata per la prima pagina dei risultati del comando.

Ad esempio, se esegui `aws s3api list-objects` su un bucket Amazon S3 che contiene 3.500 oggetti, la AWS CLI effettua solo la prima chiamata ad Amazon S3, restituendo solo i primi 1.000 oggetti nell'output finale.

```
$ aws s3api list-objects \
    --bucket amzn-s3-demo-bucket \
    --no-paginate
{
    "Contents": [
...
```

Come utilizzare il parametro --page-size

Se vengono rilevati errori quando esegui i comandi dell'elenco per un numero elevato di risorse, le dimensioni di pagina predefinite potrebbero essere eccessive. Questo può causare il superamento del tempo massimo consentito per le chiamate ai servizi AWS e la generazione di un errore di tipo "timeout". Puoi utilizzare l'opzione `--page-size` per specificare che AWS CLI richieda un numero minore di elementi da ciascuna chiamata al servizio AWS. La AWS CLI recupera comunque l'elenco completo, ma esegue un maggior numero di chiamate API in background e recupera un numero di elementi inferiore per ogni chiamata. In questo modo, le chiamate individuali hanno una migliore possibilità di completamento prima della scadenza. La modifica delle dimensioni della pagina non pregiudica l'output, ma solo il numero di chiamate API che deve essere effettuato per generare l'output.

```
$ aws s3api list-objects \
    --bucket amzn-s3-demo-bucket \
    --page-size 100
{
    "Contents": [
...
```

Come utilizzare il parametro --max-items

Per includere un numero minore di elementi alla volta nell'output di AWS CLI, utilizza l'opzione `--max-items`. L'AWS CLI gestisce ancora la paginazione con il servizio come descritto in precedenza, ma stampa solo il numero di voci al momento specificato.

```
$ aws s3api list-objects \
    --bucket amzn-s3-demo-bucket \
    --max-items 100
{
    "NextToken": "eyJNYXJrZXIiOiBudWxsLCAiYm90b190cnVuY2F0ZV9hbW91bnQiOiAxfQ==",
    "Contents": [
...
```

Come utilizzare il parametro --starting-token

Se l'output del numero di elementi (`--max-items`) è inferiore rispetto al numero totale di elementi restituito dalle chiamate API sottostanti, l'output include `NextToken`, che può essere trasferito a un comando successivo per recuperare il set di elementi successivo. L'esempio seguente mostra come usare il valore `NextToken` restituito dall'esempio precedente e consente di recuperare il secondo centinaio di elementi.

> **Note**
>
> Il parametro `--starting-token` non può essere null o vuoto. Se il comando precedente non restituisce un valore `NextToken`, non esistono altre voci da restituire e non dovrai richiamare di nuovo il comando.

```
$ aws s3api list-objects \
    --bucket amzn-s3-demo-bucket \
    --max-items 100 \
    --starting-token eyJNYXJrZXIiOiBudWxsLCAiYm90b190cnVuY2F0ZV9hbW91bnQiOiAxfQ==
{
    "Contents": [
...
```

Il servizio AWS specificato potrebbe non restituire gli elementi nello stesso ordine ogni volta che effettui la chiamata. Se si specificano valori diversi per `--page-size` e `--max-items`, puoi ottenere risultati imprevisti con elementi mancanti o duplicati. Per evitarlo, utilizza lo stesso numero per `--page-size` e `--max-items` per sincronizzare la paginazione della AWS CLI con quella del servizio sottostante. Puoi anche recuperare l'intero elenco ed eseguire tutte le operazioni di paginazione necessarie in locale.

Pager lato client

La AWS CLI versione 2 fornisce l'uso di un programma di pager lato client per l'output. Per impostazione predefinita, questa caratteristica restituisce tutto l'output attraverso il programma pager predefinito del sistema operativo.

In ordine di precedenza, puoi specificare il pager di output nei seguenti modi:

- Utilizzando l'impostazione `cli_pager` nel file `config` nel profilo di `default` o denominato.

- Utilizzando la variabile di ambiente `AWS_PAGER`.
- Utilizzando la variabile di ambiente `PAGER`.

In ordine di precedenza, è possibile disabilitare qualsiasi utilizzo di un programma di paging esterno nei seguenti modi:

- Utilizza l'opzione della riga di comando `--no-cli-pager` per disabilitare il pager per l'utilizzo di un singolo comando.
- Imposta la configurazione `cli_pager` o la variabile `AWS_PAGER` su una stringa vuota.

Argomenti di pager lato client:

- Come utilizzare l'impostazione cli_pager
- Impostazione della variabile di ambiente AWS_PAGER
- Come utilizzare l'opzione --no-cli-pager
- Come utilizzare i flag del pager

Come utilizzare l'impostazione cli_pager

Puoi salvare le impostazioni di configurazione e le credenziali utilizzate di frequente nei file gestiti da AWS CLI. Le impostazioni nel profilo del nome hanno la precedenza sulle impostazioni del profilo di `default`. Per ulteriori informazioni sulle impostazioni di configurazione, consulta Impostazioni dei file di configurazione e delle credenziali nella AWS CLI.

L'esempio seguente imposta il pager di output predefinito sul programma `less`.

```
[default]
cli_pager=less
```

L'esempio seguente configura l'impostazione predefinita per disabilitare l'uso di un pager.

```
[default]
cli_pager=
```

Impostazione della variabile di ambiente AWS_PAGER

L'esempio seguente imposta il pager di output predefinito sul programma `less`. Per ulteriori informazioni sulle variabili di ambiente, consulta Configurazione delle variabili d'ambiente per AWS CLI.

Linux and macOS

```
$ export AWS_PAGER="less"
```

Windows

```
C:\> setx AWS_PAGER "less"
```

Come utilizzare l'opzione --no-cli-pager

Per disabilitare l'uso di un pager su un singolo comando, utilizza l'opzione `--no-cli-pager`. Per ulteriori informazioni, sulle opzioni della riga di comando, consulta Opzioni della riga di comando in AWS CLI.

```
$ aws s3api list-objects \
    --bucket amzn-s3-demo-bucket \
    --no-cli-pager
{
    "Contents": [
...
```

Come utilizzare i flag del pager

È possibile specificare i flag da utilizzare automaticamente con il programma di paging. I flag dipendono dal programma di paging utilizzato. Gli esempi seguenti riguardano le impostazioni predefinite tipiche di `less` e `more`.

Linux and macOS

Se non si specifica diversamente, il pager utilizzato per impostazione predefinita dalla AWS CLI versione 2 è `less`. Se la variabile di ambiente `LESS` non è impostata, la AWS CLI versione 2 utilizza i flag `FRX`. È possibile combinare i flag specificandoli durante l'impostazione del pager AWS CLI.

Nell'esempio seguente viene utilizzato il flag `S`. Questo flag si combina quindi con i flag `FRX` predefiniti per creare un flag `FRXS`.

```
$ export AWS_PAGER="less -S"
```

Se non desideri nessuno di questi flag `FRX`, puoi annullarli. L'esempio seguente nega il flag `F` per creare un flag `RX` finale.

```
$ export AWS_PAGER="less -+F"
```

Per ulteriori informazioni sui flag *`less`*, consulta less su manpages.org.

Windows

Se non si specifica diversamente, il pager AWS CLI versione 2 utilizzato per impostazione predefinita è `more` senza flag aggiuntivi.

Nell'esempio seguente viene utilizzato il parametro `/c`.

```
C:\> setx AWS_PAGER "more /c"
```

Per ulteriori informazioni sui flag `more`, consulta `more` su Microsoft Docs.

Uscita di filtraggio nel AWS CLI

La AWS Command Line Interface (AWS CLI) dispone di filtri sia lato server che lato client utilizzabili singolarmente o insieme per filtrare l'output della AWS CLI. Il filtraggio lato server viene elaborato per primo e restituisce l'output per il filtraggio lato client.

- Il filtraggio lato server è supportato dall'API e di solito lo si implementa con un parametro `--filter`. Il servizio restituisce solo risultati corrispondenti che possono velocizzare i tempi di risposta HTTP per set di dati di grandi dimensioni.
- Il filtraggio lato client è supportato dal client della con AWS CLI il parametro `--query`. Questo parametro ha funzionalità che il filtraggio lato server potrebbe non avere.

Argomenti

- Filtro lato server
- Filtro lato client

- Combinazione di filtri lato server e lato client
- Risorse aggiuntive

Filtro lato server

Il filtraggio lato server nella AWS CLI è fornito dall'API del servizio AWS. Il servizio AWS restituisce solo i record nella risposta HTTP che corrispondono al filtro, il che può velocizzare i tempi di risposta HTTP per set di dati di grandi dimensioni. Poiché il filtraggio lato server è definito dall'API del servizio, i nomi e le funzioni dei parametri variano tra i servizi. Alcuni nomi di parametri comuni utilizzati per il filtraggio sono:

- `--filter` come ses e ce.
- `--filters` come ec2, autoscaling e rds.
- Nomi che iniziano con la parola `filter`, ad esempio `--filter-expression` per il comando `aws dynamodb scan`.

Per informazioni sulla disponibilità di filtraggio lato server e sulle regole di filtraggio di uno specifico comando, consulta la Guida di riferimento della versione 2 della AWS CLI.

Filtro lato client

La AWS CLI fornisce funzionalità di filtro dell'output basate su JSON predefinite con il parametro `--query`. Il parametro `--query` è uno strumento potente che puoi utilizzare per personalizzare il contenuto e lo stile del tuo output. Il parametro `--query` prende la risposta HTTP che arriva dal server e filtra i risultati prima di visualizzarli. Poiché l'intera risposta HTTP viene inviata al client prima del filtraggio, il filtraggio lato client può essere più lento del filtraggio lato server per set di dati di grandi dimensioni.

L'interrogazione utilizza la sintassi JMESPath per creare espressioni per filtrare l'output. Per imparare la sintassi JMESPath, consulta il Tutorial sul sito web di JMESPath.

> **Important**
>
> Il tipo di output specificato influenza il funzionamento dell'opzione `--query`:
>
> - Se si specifica `--output text`, l'output viene paginato prima dell'applicazione del filtro `--query` e la AWS CLI esegue la query una volta in ogni pagina dell'output. Per questo

motivo, la query include il primo elemento corrispondente in ogni pagina, eventualmente generando un output aggiuntivo inaspettato. È inoltre possibile utilizzare altri strumenti a riga di comando, ad esempio `head` o `tail` per filtrare ulteriormente l'output.

- Se si specifica `--output json`, `--output yaml` o `--output yaml-stream`, l'output viene completamente elaborato come una struttura singola nativa prima dell'applicazione del filtro `--query`. La AWS CLI esegue la query solo una volta sull'intera struttura, producendo un risultato filtrato che viene quindi emesso.

Argomenti di filtraggio lato client

- Prima di iniziare
- Identificatori
- Selezione da un elenco
- Filtrare i dati nidificati
- Appiattimento dei risultati
- Filtro per valori specifici
- Espressioni con reindirizzamento
- Filtraggio in base a molteplici identificatori
- Aggiunta di etichette ai valori degli identificatori
- Funzioni
- Esempi avanzati di --query

Prima di iniziare

> **Note**
>
> Questi esempi di espressioni di filtro sono scritti per shell di base in stile Linux. Quando usi questi esempi, assicurati di utilizzare le regole di quotazione corrette per la shell del terminale. Il modo in cui il terminale interpreta i tuoi input può modificare notevolmente ciò che viene inviato alla AWS CLI. Il modo in cui il terminale legge le virgolette singole `'`, le virgolette doppie `"` o le virgolette inverse `` ` `` può cambiare il modo in cui viene letto il contenuto.
> Per ulteriori informazioni, consulta the section called "Virgolette con stringhe".

Il seguente output JSON mostra un esempio di ciò che può produrre il parametro `--query`. L'output illustra tre volumi Amazon EBS collegati a istanze Amazon EC2 separate.

Output di esempio

```
$ aws ec2 describe-volumes
{
  "Volumes": [
    {
      "AvailabilityZone": "us-west-2a",
      "Attachments": [
        {
          "AttachTime": "2013-09-17T00:55:03.000Z",
          "InstanceId": "i-a071c394",
          "VolumeId": "vol-e11a5288",
          "State": "attached",
          "DeleteOnTermination": true,
          "Device": "/dev/sda1"
        }
      ],
      "VolumeType": "standard",
      "VolumeId": "vol-e11a5288",
      "State": "in-use",
      "SnapshotId": "snap-f23ec1c8",
      "CreateTime": "2013-09-17T00:55:03.000Z",
      "Size": 30
    },
    {
      "AvailabilityZone": "us-west-2a",
      "Attachments": [
        {
          "AttachTime": "2013-09-18T20:26:16.000Z",
          "InstanceId": "i-4b41a37c",
          "VolumeId": "vol-2e410a47",
          "State": "attached",
          "DeleteOnTermination": true,
          "Device": "/dev/sda1"
        }
      ],
      "VolumeType": "standard",
      "VolumeId": "vol-2e410a47",
      "State": "in-use",
      "SnapshotId": "snap-708e8348",
      "CreateTime": "2013-09-18T20:26:15.000Z",
```

```
      "Size": 8
    },
    {
      "AvailabilityZone": "us-west-2a",
      "Attachments": [
        {
          "AttachTime": "2020-11-20T19:54:06.000Z",
          "InstanceId": "i-1jd73kv8",
          "VolumeId": "vol-a1b3c7nd",
          "State": "attached",
          "DeleteOnTermination": true,
          "Device": "/dev/sda1"
        }
      ],
      "VolumeType": "standard",
      "VolumeId": "vol-a1b3c7nd",
      "State": "in-use",
      "SnapshotId": "snap-234087fb",
      "CreateTime": "2020-11-20T19:54:05.000Z",
      "Size": 15
    }
  ]
}
```

Identificatori

Gli identificatori sono le etichette per i valori di output. Quando crei dei filtri, utilizzi gli identificatori per restringere i risultati delle query. Nel seguente esempio di output, tutti gli identificatori come `Volumes`, `AvailabilityZone` e `AttachTime` sono evidenziati.

```
$ aws ec2 describe-volumes
{
  "Volumes": [
    {
      "AvailabilityZone": "us-west-2a",
      "Attachments": [
        {
          "AttachTime": "2013-09-17T00:55:03.000Z",
          "InstanceId": "i-a071c394",
          "VolumeId": "vol-e11a5288",
          "State": "attached",
          "DeleteOnTermination": true,
          "Device": "/dev/sda1"
```

```
        }
      ],
      "VolumeType": "standard",
      "VolumeId": "vol-e11a5288",
      "State": "in-use",
      "SnapshotId": "snap-f23ec1c8",
      "CreateTime": "2013-09-17T00:55:03.000Z",
      "Size": 30
    },
    {
      "AvailabilityZone": "us-west-2a",
      "Attachments": [
        {
          "AttachTime": "2013-09-18T20:26:16.000Z",
          "InstanceId": "i-4b41a37c",
          "VolumeId": "vol-2e410a47",
          "State": "attached",
          "DeleteOnTermination": true,
          "Device": "/dev/sda1"
        }
      ],
      "VolumeType": "standard",
      "VolumeId": "vol-2e410a47",
      "State": "in-use",
      "SnapshotId": "snap-708e8348",
      "CreateTime": "2013-09-18T20:26:15.000Z",
      "Size": 8
    },
    {
      "AvailabilityZone": "us-west-2a",
      "Attachments": [
        {
          "AttachTime": "2020-11-20T19:54:06.000Z",
          "InstanceId": "i-1jd73kv8",
          "VolumeId": "vol-a1b3c7nd",
          "State": "attached",
          "DeleteOnTermination": true,
          "Device": "/dev/sda1"
        }
      ],
      "VolumeType": "standard",
      "VolumeId": "vol-a1b3c7nd",
      "State": "in-use",
      "SnapshotId": "snap-234087fb",
```

```
      "CreateTime": "2020-11-20T19:54:05.000Z",
      "Size": 15
    }
  ]
}
```

Per ulteriori informazioni, consulta Identificatori sul sito web JMESPath.

Selezione da un elenco

Un elenco o una matrice è un identificatore seguito da una parentesi quadra "`[`" come `Volumes` e `Attachments` in the section called "Prima di iniziare".

Sintassi

```
<listName>[ ]
```

Per filtrare tutto l'output di un array, puoi utilizzare la notazione con caratteri jolly. Le espressioni con caratteri jolly sono espressioni utilizzate per restituire elementi utilizzando la notazione `*`.

L'esempio seguente esegue una query su tutti i contenuti di `Volumes`.

```
$ aws ec2 describe-volumes \
    --query 'Volumes[*]'
[
  {
    "AvailabilityZone": "us-west-2a",
    "Attachments": [
      {
        "AttachTime": "2013-09-17T00:55:03.000Z",
        "InstanceId": "i-a071c394",
        "VolumeId": "vol-e11a5288",
        "State": "attached",
        "DeleteOnTermination": true,
        "Device": "/dev/sda1"
      }
    ],
    "VolumeType": "standard",
    "VolumeId": "vol-e11a5288",
    "State": "in-use",
    "SnapshotId": "snap-f23ec1c8",
    "CreateTime": "2013-09-17T00:55:03.000Z",
    "Size": 30
```

```
  },
  {
    "AvailabilityZone": "us-west-2a",
    "Attachments": [
      {
        "AttachTime": "2020-11-20T19:54:06.000Z",
        "InstanceId": "i-1jd73kv8",
        "VolumeId": "vol-a1b3c7nd",
        "State": "attached",
        "DeleteOnTermination": true,
        "Device": "/dev/sda1"
      }
    ],
    "VolumeType": "standard",
    "VolumeId": "vol-a1b3c7nd",
    "State": "in-use",
    "SnapshotId": "snap-234087fb",
    "CreateTime": "2020-11-20T19:54:05.000Z",
    "Size": 15
  }
]
```

Per visualizzare un volume specifico nell'array per indice, chiami l'indice dell'array. Ad esempio, il primo elemento dell'array `Volumes` presenta un indice pari a 0, che appare nella query `Volumes[0]`. Per ulteriori informazioni, consulta Espressioni indice sul sito web JMESPath.

```
$ aws ec2 describe-volumes \
    --query 'Volumes[0]'
{
  "AvailabilityZone": "us-west-2a",
  "Attachments": [
    {
      "AttachTime": "2013-09-17T00:55:03.000Z",
      "InstanceId": "i-a071c394",
      "VolumeId": "vol-e11a5288",
      "State": "attached",
      "DeleteOnTermination": true,
      "Device": "/dev/sda1"
    }
  ],
  "VolumeType": "standard",
  "VolumeId": "vol-e11a5288",
  "State": "in-use",
```

```
  "SnapshotId": "snap-f23ec1c8",
  "CreateTime": "2013-09-17T00:55:03.000Z",
  "Size": 30
}
```

Per visualizzare un intervallo specifico di volumi per indice, utilizza `slice` con la seguente sintassi, dove start è l'indice iniziale dell'array, stop è l'indice in cui il filtro interrompe l'elaborazione e step è l'intervallo di salto.

Sintassi

```
<arrayName>[<start>:<stop>:<step>]
```

Se uno di questi valori viene omesso dall'espressione slice, vengono utilizzati i seguenti valori predefiniti:

- Start - Il primo indice dell'elenco, 0.
- Stop - L'ultimo indice dell'elenco.
- Step - Non viene saltato nessun elemento, il valore è quindi 1.

Per restituire solo i primi due volumi, utilizza un valore di Start pari a 0, un valore di Stop pari a 2 e un valore di Step pari a 1, come illustrato nell'esempio seguente.

```
$ aws ec2 describe-volumes \
    --query 'Volumes[0:2:1]'
[
  {
    "AvailabilityZone": "us-west-2a",
    "Attachments": [
      {
        "AttachTime": "2013-09-17T00:55:03.000Z",
        "InstanceId": "i-a071c394",
        "VolumeId": "vol-e11a5288",
        "State": "attached",
        "DeleteOnTermination": true,
        "Device": "/dev/sda1"
      }
    ],
    "VolumeType": "standard",
    "VolumeId": "vol-e11a5288",
    "State": "in-use",
```

```
    "SnapshotId": "snap-f23ec1c8",
    "CreateTime": "2013-09-17T00:55:03.000Z",
    "Size": 30
  },
  {
    "AvailabilityZone": "us-west-2a",
    "Attachments": [
      {
        "AttachTime": "2013-09-18T20:26:16.000Z",
        "InstanceId": "i-4b41a37c",
        "VolumeId": "vol-2e410a47",
        "State": "attached",
        "DeleteOnTermination": true,
        "Device": "/dev/sda1"
      }
    ],
    "VolumeType": "standard",
    "VolumeId": "vol-2e410a47",
    "State": "in-use",
    "SnapshotId": "snap-708e8348",
    "CreateTime": "2013-09-18T20:26:15.000Z",
    "Size": 8
  }
]
```

Poiché questo esempio contiene valori predefiniti, puoi abbreviare il comando slice da `Volumes[0:2:1]` a `Volumes[:2]`.

L'esempio seguente omette i valori predefiniti e restituisce un volume ogni due dell'intero array.

```
$ aws ec2 describe-volumes \
    --query 'Volumes[::2]'
[
  {
    "AvailabilityZone": "us-west-2a",
    "Attachments": [
      {
        "AttachTime": "2013-09-17T00:55:03.000Z",
        "InstanceId": "i-a071c394",
        "VolumeId": "vol-e11a5288",
        "State": "attached",
        "DeleteOnTermination": true,
        "Device": "/dev/sda1"
```

```
      }
    ],
    "VolumeType": "standard",
    "VolumeId": "vol-e11a5288",
    "State": "in-use",
    "SnapshotId": "snap-f23ec1c8",
    "CreateTime": "2013-09-17T00:55:03.000Z",
    "Size": 30
  },
  {
    "AvailabilityZone": "us-west-2a",
    "Attachments": [
      {
        "AttachTime": "2020-11-20T19:54:06.000Z",
        "InstanceId": "i-1jd73kv8",
        "VolumeId": "vol-a1b3c7nd",
        "State": "attached",
        "DeleteOnTermination": true,
        "Device": "/dev/sda1"
      }
    ],
    "VolumeType": "standard",
    "VolumeId": "vol-a1b3c7nd",
    "State": "in-use",
    "SnapshotId": "snap-234087fb",
    "CreateTime": "2020-11-20T19:54:05.000Z",
    "Size": 15
  }
]
```

Inoltre, i passaggi possono utilizzare numeri negativi per filtrare in ordine inverso un array, come mostrato nell'esempio seguente.

```
$ aws ec2 describe-volumes \
    --query 'Volumes[::-2]'
[
  {
    "AvailabilityZone": "us-west-2a",
    "Attachments": [
      {
        "AttachTime": "2020-11-20T19:54:06.000Z",
        "InstanceId": "i-1jd73kv8",
        "VolumeId": "vol-a1b3c7nd",
```

```
        "State": "attached",
        "DeleteOnTermination": true,
        "Device": "/dev/sda1"
      }
    ],
    "VolumeType": "standard",
    "VolumeId": "vol-a1b3c7nd",
    "State": "in-use",
    "SnapshotId": "snap-234087fb",
    "CreateTime": "2020-11-20T19:54:05.000Z",
    "Size": 15
  },
  {
    "AvailabilityZone": "us-west-2a",
    "Attachments": [
      {
        "AttachTime": "2013-09-17T00:55:03.000Z",
        "InstanceId": "i-a071c394",
        "VolumeId": "vol-e11a5288",
        "State": "attached",
        "DeleteOnTermination": true,
        "Device": "/dev/sda1"
      }
    ],
    "VolumeType": "standard",
    "VolumeId": "vol-e11a5288",
    "State": "in-use",
    "SnapshotId": "snap-f23ec1c8",
    "CreateTime": "2013-09-17T00:55:03.000Z",
    "Size": 30
  }
]
```

Per ulteriori informazioni, consulta Partizioni sul sito web JMESPath.

Filtrare i dati nidificati

Per restringere il filtraggio dei `Volumes[*]` in caso di valori nidificati, utilizza le sottoespressioni aggiungendo un punto e i criteri di filtro.

Sintassi

```
<expression>.<expression>
```

L'esempio seguente visualizza tutte le informazioni `Attachments` per tutti i volumi.

```
$ aws ec2 describe-volumes \
    --query 'Volumes[*].Attachments'
[
  [
    {
      "AttachTime": "2013-09-17T00:55:03.000Z",
      "InstanceId": "i-a071c394",
      "VolumeId": "vol-e11a5288",
      "State": "attached",
      "DeleteOnTermination": true,
      "Device": "/dev/sda1"
    }
  ],
  [
    {
      "AttachTime": "2013-09-18T20:26:16.000Z",
      "InstanceId": "i-4b41a37c",
      "VolumeId": "vol-2e410a47",
      "State": "attached",
      "DeleteOnTermination": true,
      "Device": "/dev/sda1"
    }
  ],
  [
    {
      "AttachTime": "2020-11-20T19:54:06.000Z",
      "InstanceId": "i-1jd73kv8",
      "VolumeId": "vol-a1b3c7nd",
      "State": "attached",
      "DeleteOnTermination": true,
      "Device": "/dev/sda1"
    }
  ]
]
```

Per filtrare ulteriormente i valori nidificati, aggiungi l'espressione per ogni identificatore nidificato. L'esempio seguente elenca gli `State` per tutti i `Volumes`.

```
$ aws ec2 describe-volumes \
    --query 'Volumes[*].Attachments[*].State'
[
```

```
  [
    "attached"
  ],
  [
    "attached"
  ],
  [
    "attached"
  ]
]
```

Appiattimento dei risultati

Per ulteriori informazioni, consulta Sotto-espressioni sul sito web JMESPath.

È possibile appiattire i risultati del comando `Volumes[*].Attachments[*].State` rimuovendo la notazione con caratteri jolly risultante dalla query `Volumes[*].Attachments[].State`. L'appiattimento spesso è utile per migliorare la leggibilità dei risultati.

```
$ aws ec2 describe-volumes \
    --query 'Volumes[*].Attachments[].State'
[
  "attached",
  "attached",
  "attached"
]
```

Per ulteriori informazioni, consulta Appiattimento sul sito web JMESPath.

Filtro per valori specifici

Per filtrare valori specifici in un elenco, utilizza un'espressione di filtro come illustrato nella sintassi seguente.

Sintassi

```
? <expression> <comparator> <expression>]
```

I comparatori di espressioni includono ==, `!=`, <, <=, > e >=. L'esempio seguente filtra i `VolumeIds` per tutti i `Volumes` in un `AttachedState`.

```
$ aws ec2 describe-volumes \
```

```
    --query 'Volumes[*].Attachments[?State==`attached`].VolumeId'
[
  [
    "vol-e11a5288"
  ],
  [
    "vol-2e410a47"
  ],
  [
    "vol-a1b3c7nd"
  ]
]
```

Questo risultato può quindi essere appiattito, ottenendo l'esempio seguente.

```
$ aws ec2 describe-volumes \
    --query 'Volumes[*].Attachments[?State==`attached`].VolumeId[]'
[
  "vol-e11a5288",
  "vol-2e410a47",
  "vol-a1b3c7nd"
]
```

L'esempio seguente filtra i `VolumeIds` di tutti i `Volumes` che hanno una dimensione inferiore a 20.

```
$ aws ec2 describe-volumes \
    --query 'Volumes[?Size < `20`].VolumeId'
[
  "vol-2e410a47",
  "vol-a1b3c7nd"
]
```

Per ulteriori informazioni, consulta Espressioni filtro sul sito web JMESPath.

Espressioni con reindirizzamento

È possibile reindirizzare i risultati di un filtro a un nuovo elenco e quindi filtrare il risultato con un'altra espressione utilizzando la sintassi seguente:

Sintassi

```
<expression> | <expression>]
```

L'esempio seguente prende i risultati del filtro dell'espressione `Volumes[*].Attachments[].InstanceId` e restituisce il primo risultato nell'array.

```
$ aws ec2 describe-volumes \
    --query 'Volumes[*].Attachments[].InstanceId | [0]'
"i-a071c394"
```

Questo esempio esegue questa operazione creando innanzitutto l'array dalla seguente espressione.

```
$ aws ec2 describe-volumes \
    --query 'Volumes[*].Attachments[].InstanceId'
"i-a071c394",
  "i-4b41a37c",
  "i-1jd73kv8"
```

In seguito restituisce il primo elemento di tale array.

```
"i-a071c394"
```

Per ulteriori informazioni, consulta Espressioni reindirizzate su pipe sul sito web JMESPath.

Filtraggio in base a molteplici identificatori

Per filtrare più identificatori, utilizza un elenco a selezione multipla con la seguente sintassi:

Sintassi

```
<listName>[].[<expression>, <expression>]
```

Nell'esempio seguente, `VolumeId` e `VolumeType` vengono filtrati nell'elenco `Volumes` risultante dalla seguente espressione.

```
$ aws ec2 describe-volumes \
    --query 'Volumes[].[VolumeId, VolumeType]'
[
  [
    "vol-e11a5288",
    "standard"
  ],
  [
```

```
    "vol-2e410a47",
    "standard"
  ],
  [
    "vol-a1b3c7nd",
    "standard"
  ]
]
```

Per aggiungere dati annidati all'elenco, aggiungi un altro elenco a selezione multipla. L'esempio seguente amplia l'esempio precedente filtrando anche per `InstanceId` e `State` nell'elenco nidificato `Attachments`. I risultati sono illustrati di seguito.

```
$ aws ec2 describe-volumes \
    --query 'Volumes[].[VolumeId, VolumeType, Attachments[].[InstanceId, State]]'
[
  [
    "vol-e11a5288",
    "standard",
    [
      [
        "i-a071c394",
        "attached"
      ]
    ]
  ],
  [
    "vol-2e410a47",
    "standard",
    [
      [
        "i-4b41a37c",
        "attached"
      ]
    ]
  ],
  [
    "vol-a1b3c7nd",
    "standard",
    [
      [
        "i-1jd73kv8",
        "attached"
```

```
      ]
    ]
  ]
]
```

Per renderli più leggibili, appiattisci l'espressione come mostrato nell'esempio seguente.

```
$ aws ec2 describe-volumes \
    --query 'Volumes[].[VolumeId, VolumeType, Attachments[].[InstanceId, State][]][]'
[
  "vol-e11a5288",
  "standard",
  [
    "i-a071c394",
    "attached"
  ],
  "vol-2e410a47",
  "standard",
  [
    "i-4b41a37c",
    "attached"
  ],
  "vol-a1b3c7nd",
  "standard",
  [
    "i-1jd73kv8",
    "attached"
  ]
]
```

Per ulteriori informazioni, consulta Elenco multiselezione sul sito web JMESPath.

Aggiunta di etichette ai valori degli identificatori

Per facilitare la lettura di questo output, utilizza un hash a selezione multipla con la seguente sintassi.

Sintassi

```
<listName>[].{<label>: <expression>, <label>: <expression>}
```

L'etichetta identificativa non deve necessariamente corrispondere al nome dell'identificatore. L'esempio seguente utilizza l'etichetta `VolumeType` per i valori `VolumeType`.

```
$ aws ec2 describe-volumes \
    --query 'Volumes[].{VolumeType: VolumeType}'
[
  {
    "VolumeType": "standard",
  },
  {
    "VolumeType": "standard",
  },
  {
    "VolumeType": "standard",
  }
]
```

Per semplicità, l'esempio seguente mantiene i nomi degli identificatori per ogni etichetta e visualizza `VolumeId`, `VolumeType`, `InstanceId` e `State` per tutti i volumi:

```
$ aws ec2 describe-volumes \
    --query 'Volumes[].{VolumeId: VolumeId, VolumeType: VolumeType, InstanceId:
 Attachments[0].InstanceId, State: Attachments[0].State}'
[
  {
    "VolumeId": "vol-e11a5288",
    "VolumeType": "standard",
    "InstanceId": "i-a071c394",
    "State": "attached"
  },
  {
    "VolumeId": "vol-2e410a47",
    "VolumeType": "standard",
    "InstanceId": "i-4b41a37c",
    "State": "attached"
  },
  {
    "VolumeId": "vol-a1b3c7nd",
    "VolumeType": "standard",
    "InstanceId": "i-1jd73kv8",
    "State": "attached"
  }
]
```

Per ulteriori informazioni, consulta Hash a selezione multipla sul sito web JMESPath.

Funzioni

La sintassi JMESPath contiene molte funzioni che è possibile utilizzare per le query. Per informazioni sulle funzioni JMESPath, consulta la sezione Funzioni integrate nel sito web JMESPath.

Per dimostrare come incorporare una funzione nelle query, l'esempio seguente utilizza la funzione `sort_by`. La funzione `sort_by` ordina un array utilizzando un'espressione come chiave di ordinamento seguendo la sintassi seguente:

Sintassi

```
sort_by(<listName>, <sort expression>)[].<expression>
```

L'esempio seguente utilizza il precedente esempio di hash a selezione multipla e ordina l'output per `VolumeId`.

```
$ aws ec2 describe-volumes \
    --query 'sort_by(Volumes, &VolumeId)[].{VolumeId: VolumeId, VolumeType: VolumeType,
 InstanceId: Attachments[0].InstanceId, State: Attachments[0].State}'
[
  {
    "VolumeId": "vol-2e410a47",
    "VolumeType": "standard",
    "InstanceId": "i-4b41a37c",
    "State": "attached"
  },
  {
    "VolumeId": "vol-a1b3c7nd",
    "VolumeType": "standard",
    "InstanceId": "i-1jd73kv8",
    "State": "attached"
  },
  {
    "VolumeId": "vol-e11a5288",
    "VolumeType": "standard",
    "InstanceId": "i-a071c394",
    "State": "attached"
  }
]
```

Per ulteriori informazioni, consulta sort_by sul sito web JMESPath.

Esempi avanzati di **--query**

Come estrarre informazioni da un elemento specifico

L'esempio seguente utilizza il parametro `--query` per trovare un determinato elemento in un elenco e quindi estrae le informazioni da tale elemento. L'esempio seguente elenca tutte le `AvailabilityZones` associate all'endpoint del servizio specificato. Estrae l'elemento dall'elenco `ServiceDetails` che ha specificato `ServiceName`, quindi emette il campo `AvailabilityZones` da questo elemento selezionato.

```
$ aws --region us-east-1 ec2 describe-vpc-endpoint-services \
    --query 'ServiceDetails[?ServiceName==`com.amazonaws.us-
east-1.ecs`].AvailabilityZones'
[
    [
        "us-east-1a",
        "us-east-1b",
        "us-east-1c",
        "us-east-1d",
        "us-east-1e",
        "us-east-1f"
    ]
]
```

Come mostrare snapshot dopo la data di creazione specificata

L'esempio seguente visualizza come creare un elenco di tutte le snapshot che sono state create dopo una data specifica, includendo solo alcuni dei campi disponibili nell'output.

```
$ aws ec2 describe-snapshots --owner self \
    --output json \
    --query 'Snapshots[?StartTime>=`2018-02-07`].
{Id:SnapshotId,VId:VolumeId,Size:VolumeSize}'
[
    {
        "id": "snap-0effb42b7a1b2c3d4",
        "vid": "vol-0be9bb0bf12345678",
        "Size": 8
    }
]
```

Come mostrare le AMI più recenti

L'esempio seguente crea un elenco delle cinque Amazon Machine Images (AMI) più recenti che sono state create, ordinate dalla più recente alla meno recente.

```
$ aws ec2 describe-images \
    --owners self \
    --query 'reverse(sort_by(Images,&CreationDate))[:5].{id:ImageId,date:CreationDate}'
[
    {
        "id": "ami-0a1b2c3d4e5f60001",
        "date": "2018-11-28T17:16:38.000Z"
    },
    {
        "id": "ami-0a1b2c3d4e5f60002",
        "date": "2018-09-15T13:51:22.000Z"
    },
    {
        "id": "ami-0a1b2c3d4e5f60003",
        "date": "2018-08-19T10:22:45.000Z"
    },
    {
        "id": "ami-0a1b2c3d4e5f60004",
        "date": "2018-05-03T12:04:02.000Z"
    },
    {
        "id": "ami-0a1b2c3d4e5f60005",
        "date": "2017-12-13T17:16:38.000Z"
    }
]
```

Come mostrare istanze con dimensionamento automatico non integre

L'esempio seguente visualizza solo l'`InstanceId` per qualsiasi istanza non integra nel gruppo Auto Scaling specificato.

```
$ aws autoscaling describe-auto-scaling-groups \
    --auto-scaling-group-name My-AutoScaling-Group-Name \
    --output text \
    --query 'AutoScalingGroups[*].Instances[?HealthStatus==`Unhealthy`].InstanceId'
```

Come includere i volumi con il tag specificato

L'esempio seguente descrive tutte le istanze con un tag `test`. Finché è presente un altro tag oltre a `test` associato al volume, il volume viene comunque restituito nei risultati.

L'espressione seguente restituisce tutti i tag con il tag `test` in un array. Tutti i tag che non sono il tag `test` contengono un valore `null`.

```
$ aws ec2 describe-volumes \
    --query 'Volumes[*].Tags[?Value == `test`]'
```

Come escludere i volumi con il tag specificato

L'esempio seguente descrive tutte le istanze senza un tag `test`. L'uso di una semplice espressione `?Value != `test`` non funziona per escludere un volume poiché i volumi possono avere più tag. Finché è presente un altro tag oltre a `test` associato al volume, il volume viene comunque restituito nei risultati.

Per escludere tutti i volumi con il tag `test`, inizia con l'espressione seguente per restituire tutti i tag con il tag `test` in un array. Tutti i tag che non sono il tag `test` contengono un valore `null`.

```
$ aws ec2 describe-volumes \
    --query 'Volumes[*].Tags[?Value == `test`]'
```

Quindi filtra tutti i risultati positivi di `test` utilizzando la funzione `not_null`.

```
$ aws ec2 describe-volumes \
    --query 'Volumes[?!not_null(Tags[?Value == `test`].Value)]'
```

Reindirizza i risultati per appiattirli ottenendo la seguente query.

```
$ aws ec2 describe-volumes \
    --query 'Volumes[?!not_null(Tags[?Value == `test`].Value)] | []'
```

Combinazione di filtri lato server e lato client

Puoi utilizzare filtri lato server e lato client insieme. Per prima cosa viene completato il filtraggio lato server, che invia i dati al client che vengono quindi filtrati attraverso il parametro `--query`. Se utilizzi set di dati di grandi dimensioni, il filtro lato server permette innanzitutto di ridurre la quantità di dati inviati al client per ogni chiamata AWS CLI, pur mantenendo la potente personalizzazione fornita dal filtro lato client.

L'esempio seguente elenca i volumi Amazon EC2 che utilizzano filtri lato server e lato client. Il servizio filtra un elenco di tutti i volumi collegati nella zona di disponibilità `us-west-2a`. Il parametro `--query` limita ulteriormente l'output ai soli volumi con un valore `Size` maggiore di 50 e mostra solo i campi specificati con nomi definiti dall'utente.

```
$ aws ec2 describe-volumes \
    --filters "Name=availability-zone,Values=us-west-2a" "Name=status,Values=attached"
 \
    --query 'Volumes[?Size > `50`].{Id:VolumeId,Size:Size,Type:VolumeType}'
[
    {
        "Id": "vol-0be9bb0bf12345678",
        "Size": 80,
        "VolumeType": "gp2"
    }
]
```

L'esempio seguente consente di recuperare un elenco di immagini che soddisfano diversi criteri. Quindi utilizza il parametro `--query` per ordinare l'output in base a `CreationDate`, selezionando solo il più recente. Infine, visualizza l'`ImageId` di tale immagine.

```
$ aws ec2 describe-images \
    --owners amazon \
    --filters "Name=name,Values=amzn*gp2" "Name=virtualization-type,Values=hvm"
 "Name=root-device-type,Values=ebs" \
    --query "sort_by(Images, &CreationDate)[-1].ImageId" \
    --output text
ami-00ced3122871a4921
```

Nell'esempio seguente viene visualizzato il numero di volumi disponibili che sono più di 1000 IOPS utilizzando `length` per contarne il numero in un elenco.

```
$ aws ec2 describe-volumes \
    --filters "Name=status,Values=available" \
    --query 'length(Volumes[?Iops > `1000`])'
3
```

L'esempio seguente recupera i nomi dei gruppi Auto Scaling che utilizzano configurazioni di avvio nella Regione AWS specificata che usano stack CloudFormation.

```
$ aws autoscaling describe-auto-scaling-groups --region us-west-2 \
```

```
  --filters Name=tag-key,Values=aws:cloudformation:stack-name \
  --query 'AutoScalingGroups[?LaunchConfigurationName!=`null`].AutoScalingGroupName'
[
    "group-1",
    "group-2",
    "group-3"
]
```

Risorse aggiuntive

Prompt automatico della AWS CLI

Quando inizi a utilizzare le espressioni di filtro, puoi sfruttare la funzionalità di prompt automatico nella versione 2 della AWS CLI. La funzionalità di prompt automatico fornisce un'anteprima quando premi il tasto F5. Per ulteriori informazioni, consulta the section called "Prompt automatico".

Terminale JMESPath

Il Terminale JMESPath è un terminale per comandi interattivo per sperimentare le espressioni JMESPath utilizzate per il filtraggio lato client. Utilizzando il comando `jpterm`, il terminale mostra immediatamente i risultati delle query durante la digitazione. È possibile reindirizzare l'output della AWS CLI direttamente al terminale, abilitando la sperimentazione avanzata delle interrogazioni.

L'esempio seguente reindirizza l'output di `aws ec2 describe-volumes` direttamente al terminale JMESPath.

```
$ aws ec2 describe-volumes | jpterm
```

Per ulteriori informazioni sul Terminale JMESPath e le istruzioni di installazione, consulta Terminale JMESPath su GitHub.

Utility jq

L'utility `jq` offre un modo per trasformare l'output sul lato client nel formato di output desiderato. Per ulteriori informazioni su `jq` e le istruzioni di installazione, consulta jq su GitHub.

Codici restituiti dalla riga di comando in AWS CLI

Il codice restituito è in genere un codice nascosto inviato dopo l'esecuzione di un comando AWS Command Line Interface (AWS CLI) che descrive lo stato del comando. È possibile utilizzare il

comando `echo` per visualizzare il codice inviato dall'ultimo comando AWS CLI e utilizzare questi codici per determinare se un comando ha avuto esito positivo o negativo e perché un comando potrebbe contenere un errore. Oltre ai codici restituiti, è possibile visualizzare ulteriori dettagli su un errore eseguendo i comandi con lo switch `--debug`. Si ottiene un report dettagliato delle fasi utilizzate dalla AWS CLI per elaborare il comando e in cui è specificato il risultato di ogni fase.

Per determinare il codice restituito da un comando di AWS CLI, esegui uno di questi comandi subito dopo aver eseguito il comando CLI.

Linux and macOS

```
$ echo $?
0
```

Windows PowerShell

```
PS> echo $lastexitcode
0
```

Windows Command Prompt

```
C:\> echo %errorlevel%
0
```

Di seguito sono elencati i valori di codice che possono essere restituiti al termine dell'esecuzione di un comando di AWS Command Line Interface (AWS CLI).

Codic	Significato
0	Il servizio ha risposto con un codice di stato della risposta HTTP di 200, secondo il quale non sono stati generati errori da parte della AWS e del servizio AWS CLI e a cui è stata inviata la richiesta.
1	Una o più operazioni di trasferimento di Amazon S3 non riuscite. Limitato ai comandi S3.
2	Il significato di questo codice restituito dipende dal comando. • Applicabile a tutti i comandi AWS CLI: il comando immesso non può essere analizzat o. Gli errori di analisi possono essere causati, a titolo esemplificativo, dall'assenza di

Codic	Significato
	sottocomandi o argomenti richiesti oppure dall'utilizzo di comandi o argomenti sconosciuti. • Limitato ai comandi S3: uno o più file contrassegnati per il trasferimento sono stati ignorati durante il relativo processo. Tuttavia, tutti gli altri file contrassegnati per il trasferimento sono stati trasferiti. I file che vengono ignorati durante il processo di trasferimento includono: file non esistenti, file speciali per dispositivi a caratteri, file speciali per dispositivi a blocchi, code FIFO o socket e file per cui l'utente non dispone di autorizzazioni di lettura.
130	Il comando è stato interrotto da un SIGINT. Questo è il segnale inviato dall'utente per annullare un comando con `Ctrl+C`.
252	La sintassi del comando non era valida, era stato fornito un parametro sconosciuto o il valore di un parametro non era corretto e impediva l'esecuzione del comando.
253	L'ambiente o la configurazione del sistema non erano validi. Sebbene il comando fornito possa essere valido dal punto di vista sintattico, la configurazione o le credenziali mancanti ne hanno impedito l'esecuzione.
254	Il comando è stato analizzato correttamente e la richiesta è stata inoltrata al servizio specificato, ma il servizio ha restituito un errore. Ciò indica in genere un utilizzo errato dell'API o altri problemi specifici del servizio.
255	Il comando ha avuto esito negativo. Sono stati generati errori da AWS CLI o dal servizio AWS a cui è stata inviata la richiesta.

Utilizzo di procedure guidate personalizzate per eseguire comandi interattivi nella AWS CLI

AWS Command Line Interface (AWS CLI) offre la possibilità di utilizzare una procedura guidata per alcuni comandi. Per contribuire o visualizzare l'elenco completo delle procedure guidate della AWS CLI disponibili, consulta la cartella con le procedure guidate della AWS CLI su GitHub.

Come funziona

Analogamente alla console AWS, la AWS CLI dispone di un'interfaccia utente guidata ad uso dell'utente nella gestione delle risorse AWS. Per utilizzare la procedura guidata, è necessario chiamare il sottocomando `wizard` e il nome della procedura guidata dopo il nome del servizio in un comando. La struttura del comando è la seguente:

Sintassi:

```
$ aws <command> wizard <wizardName>
```

L'esempio seguente sta chiamando la procedura guidata per creare una nuova tabella `dynamodb`.

```
$ aws dynamodb wizard new-table
```

`aws configure` è l'unica procedura guidata che non ha un nome specifico. Durante l'esecuzione della procedura guidata, esegui il comando `aws configure wizard`, come illustrato nell'esempio seguente:

```
$ aws configure wizard
```

Dopo aver richiamato una procedura guidata, viene visualizzato un modulo nella shell. Per ogni parametro, viene fornito un elenco di opzioni tra cui scegliere o viene richiesto di inserirlo in una stringa. Per effettuare una selezione da un elenco, utilizza i tasti freccia su e giù e premi INVIO. Per visualizzare i dettagli di un'opzione, premi il tasto freccia destra. Dopo aver completato la compilazione di un parametro, premi INVIO.

```
$ aws configure wizard
What would you like to configure
> Static Credentials
  Assume Role
  Process Provider
  Additional CLI configuration
Enter the name of the profile:
Enter your Access Key Id:
Enter your Secret Access Key:
```

Per modificare le istruzioni precedenti, utilizza SHIFT + TAB. In alcune procedure guidate, dopo aver compilato tutte le istruzioni, è possibile visualizzare in anteprima un modello AWS CloudFormation o

il comando AWS CLI contenente le informazioni desiderate. Questa modalità di anteprima è utile per apprendere l'utilizzo della AWS CLI, le API di servizio e creare modelli per gli script.

Premi INVIO dopo l'anteprima o l'ultimo prompt per eseguire il comando finale.

```
$ aws configure wizard
What would you like to configure
Enter the name of the profile: testWizard
Enter your Access Key Id: AB1C2D3EF4GH5I678J90K
Enter your Secret Access Key: ab1c2def34gh5i67j8k90l1mnop2qr3s45tu678v90
<ENTER>
```

Creazione e utilizzo di alias in AWS CLI

Gli alias sono scorciatoie che è possibile creare in AWS Command Line Interface (AWS CLI) per abbreviare i comandi o gli script utilizzati di frequente. Gli alias vengono creati nel file `alias` che si trova nella cartella di configurazione.

Argomenti

- Prerequisiti
- Fase 1: creazione del file alias
- Fase 2: creazione di un alias
- Fase 3: chiamata di un alias
- Esempi di repository di alias
- Risorse

Prerequisiti

Per utilizzare i comandi alias, è necessario completare quanto segue:

- Istalla e configura la AWS CLI. Per ulteriori informazioni, consultare the section called "Installazione o aggiornamento" e Credenziali di autenticazione e accesso per la AWS CLI.
- Utilizza una versione minima della AWS CLI di 1.11.24 o 2.0.0.
- (Facoltativo) Per utilizzare gli script bash dell'alias della AWS CLI, è necessario utilizzare un terminale compatibile con bash.

Fase 1: creazione del file alias

Per creare il file `alias`, è possibile utilizzare la navigazione dei file e un editor di testo oppure il terminale preferito seguendo la procedura dettagliata. Per creare rapidamente il file alias, usa il seguente blocco di comandi.

Linux and macOS

```
$ mkdir -p ~/.aws/cli
$ echo '[toplevel]' > ~/.aws/cli/alias
```

Windows

```
C:\> md %USERPROFILE%\.aws\cli
C:\> echo [toplevel] > %USERPROFILE%/.aws/cli/alias
```

Per creare il file alias

1. Crea una cartella denominata `cli` nella cartella di configurazione AWS CLI. Per impostazione predefinita, la cartella di configurazione è `~/.aws/` su Linux o macOS e `%USERPROFILE%\.aws\` su Windows. Puoi crearlo tramite la navigazione dei file o utilizzando il comando seguente.

   Linux and macOS

   ```
   $ mkdir -p ~/.aws/cli
   ```

   Windows

   ```
   C:\> md %USERPROFILE%\.aws\cli
   ```

   Il percorso predefinito della cartella `cli` risultante è `~/.aws/cli/` su Linux o macOS e `%USERPROFILE%\.aws\cli` su Windows.
2. Nella cartella `cli`, crea un file di testo denominato `alias` senza estensione e aggiungi `[toplevel]` alla prima riga. È possibile creare questo file tramite l'editor di testo preferito o utilizzare il seguente comando.

Linux and macOS

```
$ echo '[toplevel]' > ~/.aws/cli/alias
```

Windows

```
C:\> echo [toplevel] > %USERPROFILE%/.aws/cli/alias
```

Fase 2: creazione di un alias

Puoi creare un alias usando comandi di base o script Bash.

Creazione di un alias di comando di base

Puoi creare il codice alias aggiungendo un comando con la sintassi seguente nel file `alias` creato nella fase precedente.

Sintassi

```
aliasname = command [--options]
```

`aliasname` è chiamato alias. *`command`* è il comando da chiamare, che può includere altri alias. È possibile includere opzioni o parametri nell'alias o aggiungerli quando si chiama l'alias.

L'esempio seguente crea un alias denominato `aws whoami` utilizzando il comando `aws sts get-caller-identity`. Poiché questo alias richiama un comando della AWS CLI esistente, è possibile scrivere il comando senza il prefisso `aws`.

```
whoami = sts get-caller-identity
```

L'esempio seguente riprende l'esempio `whoami` precedente e aggiunge le opzioni di filtro `Account` e testo `output`.

```
whoami2 = sts get-caller-identity --query Account --output text
```

Creazione di un alias di sottocomando

> **Note**
>
> La funzionalità degli alias dei sottocomandi richiede una versione minima della AWS CLI 1.11.24 o 2.0.0

È possibile creare un alias per i sottocomandi aggiungendo un comando utilizzando la seguente sintassi nel file `alias` creato nella fase precedente.

Sintassi

```
[command commandGroup]
aliasname = command [--options]
```

commandGroup è il namespace dei comandi, ad esempio il comando `aws ec2 describe-regions` si trova nel gruppo di comandi `ec2`. *aliasname* è chiamato alias. *command* è il comando da chiamare, che può includere altri alias. È possibile includere opzioni o parametri nell'alias o aggiungerli quando si chiama l'alias.

L'esempio seguente crea un alias denominato `aws ec2 regions` utilizzando il comando `aws ec2 describe-regions`. Poiché questo alias richiama un comando AWS CLI esistente sotto il namespace del comando `ec2`, è possibile scrivere il comando senza il prefisso `aws ec2`.

```
[command ec2]
regions = describe-regions --query Regions[].RegionName
```

Per creare alias da comandi esterni al namespace dei comandi, aggiungi un punto esclamativo al comando completo. L'esempio seguente crea un alias denominato `aws ec2 instance-profiles` utilizzando il comando `aws iam list-instance-profiles`.

```
[command ec2]
instance-profiles = !aws iam list-instance-profiles
```

> **Note**
>
> Gli alias utilizzano solo gli spazi dei namespace esistenti e non è possibile crearne di nuovi. Ad esempio, non è possibile creare un alias con la sezione `[command johnsmith]` poiché il namespace del comando `johnsmith` non esiste già.

Creare un alias di scripting Bash

> **Warning**
>
> Per utilizzare gli script Bash dell'alias della AWS CLI, è necessario utilizzare un terminale compatibile con Bash

È possibile creare un alias utilizzando gli script Bash per processi più avanzati utilizzando la seguente sintassi.

Sintassi

```
aliasname =
    !f() {
        script content
}; f
```

L'*`aliasname`* è ciò che chiami alias e il *`contenuto dello script`* è lo script che desideri eseguire quando chiami l'alias.

L'esempio seguente utilizza `opendns` per visualizzare l'indirizzo IP corrente. Poiché è possibile utilizzare alias in altri alias, il seguente alias `myip` è utile per consentire o revocare l'accesso all'indirizzo IP dall'interno di altri alias.

```
myip =
  !f() {
    dig +short myip.opendns.com @resolver1.opendns.com
  }; f
```

Il seguente esempio di script chiama l'alias `aws myip` precedente per autorizzare il tuo indirizzo IP all'ingresso di un gruppo di sicurezza Amazon EC2.

```
authorize-my-ip =
  !f() {
    ip=$(aws myip)
    aws ec2 authorize-security-group-ingress --group-id ${1} --cidr $ip/32 --protocol
 tcp --port 22
  }; f
```

Quando chiami degli alias che utilizzano lo scripting Bash, le variabili vengono sempre passate nell'ordine in cui le hai inserite. Nello scripting Bash, i nomi delle variabili non vengono presi in considerazione, ma solo l'ordine in cui appaiono. Nel seguente esempio di alias `textalert`, la variabile per l'opzione `--message` è la prima e l'opzione `--phone-number` è la seconda.

```
textalert =
  !f() {
    aws sns publish --message "${1}" --phone-number ${2}
  }; f
```

Fase 3: chiamata di un alias

Per eseguire l'alias creato nel file `alias`, utilizza la seguente sintassi. Puoi aggiungere opzioni aggiuntive quando chiami il tuo alias.

Sintassi

```
$ aws aliasname
```

Il seguente esempio utilizza il comando alias `aws whoami`.

```
$ aws
 whoami
{
    "UserId": "A12BCD34E5FGHI6JKLM",
    "Account": "1234567890987",
    "Arn": "arn:aws:iam::1234567890987:user/userName"
}
```

L'esempio seguente utilizza l'alias `aws whoami` con opzioni aggiuntive per restituire solo il numero di `Account` nell'output `text`.

```
$ aws whoami --query Account --output
 text
1234567890987
```

L'esempio seguente utilizza l'alias del sottocomando `aws ec2 regions`.

```
$ aws ec2
 regions
[
    "ap-south-1",
    "eu-north-1",
    "eu-west-3",
    "eu-west-2",
...
```

Chiamare un alias usando variabili di scripting Bash

Quando chiami degli alias che utilizzano lo scripting Bash, le variabili vengono passate nell'ordine in sono state inserite. Nello scripting Bash, i nomi delle variabili non vengono presi in considerazione, ma solo l'ordine in cui appaiono. Ad esempio, nel seguente alias `textalert`, la variabile per l'opzione `--message` è la prima e `--phone-number` è la seconda.

```
textalert =
  !f() {
    aws sns publish --message "${1}" --phone-number ${2}
  }; f
```

Quando chiami l'alias `textalert`, è necessario passare le variabili nello stesso ordine in cui vengono eseguite nell'alias. Nell'esempio seguente utilizziamo le variabili `$message` e `$phone`. La variabile `$message` viene passata come `${1}` per l'opzione `--message` e la variabile `$phone` viene passata come `${2}` per l'opzione `--phone-number`. Ciò comporta la corretta chiamata all'alias `textalert` per inviare un messaggio.

```
$ aws textalert $message
 $phone
{
    "MessageId": "1ab2cd3e4-fg56-7h89-i01j-2klmn34567"
}
```

Nell'esempio seguente, l'ordine viene cambiato quando chiami l'alias in `$phone` e`$message`. La variabile `$phone` viene passata come `${1}` per l'opzione `--message` e la variabile `$message` viene passata come `${2}` per l'opzione `--phone-number`. Poiché le variabili non sono ordinate, l'alias le passa in modo errato. Ciò causa un errore perché il contenuto di `$message` non corrisponde ai requisiti di formattazione del numero di telefono per l'opzione `--phone-number`.

```
$ aws textalert $phone
 $message
usage: aws [options] <command> <subcommand> [<subcommand> ...] [parameters]
To see help text, you can run:

  aws help
  aws <command> help
  aws <command> <subcommand> help

Unknown options: text
```

Esempi di repository di alias

Il repository di alias della AWS CLI su GitHub contiene esempi di alias della AWS CLI creati dal team di sviluppatori della AWS CLI e dalla community. Puoi utilizzare l'intero esempio di file `alias` o utilizzare singoli alias per uso personale.

> **Warning**
>
> L'esecuzione dei comandi in questa sezione elimina il file esistente `alias`. Per evitare di sovrascrivere il file alias esistente, modifica la posizione di download.

Per utilizzare gli alias del repository

1. Installa Git. Per le istruzioni di installazione, consulta Getting Started - Installing Git nella documentazione di Git.
2. Installa il comando `jp`. Il comando `jp` viene utilizzato nell'alias `tostring`. Per le istruzioni di installazione, consulta il file JMESPath (jp) README.md su GitHub.
3. Installa il comando `jq`. Il comando `jq` viene utilizzato nell'alias `tostring-with-jq`. Per le istruzioni di installazione, consulta JSON processor (jq) su GitHub.
4. Scarica il file `alias` in uno dei seguenti modi:

- Esegui i seguenti comandi che vengono scaricati dal repository e copia il file `alias` nella cartella di configurazione.

  Linux and macOS

  ```
  $ git clone https://github.com/awslabs/awscli-aliases.git
  $ mkdir -p ~/.aws/cli
  $ cp awscli-aliases/alias ~/.aws/cli/alias
  ```

  Windows

  ```
  C:\> git clone https://github.com/awslabs/awscli-aliases.git
  C:\> md %USERPROFILE%\.aws\cli
  C:\> copy awscli-aliases\alias %USERPROFILE%\.aws\cli
  ```

- Scaricalo direttamente dal repository e salvalo nella cartella `cli` della cartella di configurazione AWS CLI. Per impostazione predefinita, la cartella di configurazione è `~/.aws/` su Linux o macOS e `%USERPROFILE%\.aws\` su Windows.

5. Per verificare che gli alias funzionino, esegui il seguente alias.

   ```
   $ aws whoami
   ```

   Viene visualizzata la stessa risposta del comando `aws sts get-caller-identity`:

   ```
   {
       "Account": "012345678901",
       "UserId": "AIUAINBADX2VEG2TC6HD6",
       "Arn": "arn:aws:iam::012345678901:user/myuser"
   }
   ```

Risorse

- Il repository di alias della AWS CLI su GitHub contiene esempi di alias della AWS CLI creati dal team di sviluppatori della AWS CLI e dal contributo della community della AWS CLI.
- L'annuncio della funzionalità alias da AWS re:Invent 2016: The Effective AWS CLI User su YouTube.
- `aws sts get-caller-identity`

- `aws ec2 describe-instances`
- `aws sns publish`

Risoluzione degli errori per AWS CLI

Questa sezione tratta gli errori comuni e le procedure di risoluzione dei problemi da seguire per risolvere un problema. Ti consigliamo di iniziare con i suggerimenti generali per la risoluzione dei problemi.

Indice

- Risoluzione generale dei problemi come primo tentativo
  - Controlla la formattazione dei comandi AWS CLI
  - Verifica che AWS CLI comando Regione AWS stia usando il tuo comando
  - Verifica di disporre di una versione recente della AWS CLI
  - Utilizza l'opzione --debug
  - Abilita e rivedi i registri della cronologia dei AWS CLI comandi
  - Conferma che il tuo AWS CLI è configurato
- Errori di comando non trovato
- Il comando aws --version restituisce una versione diversa da quella installata
- Il comando "aws --version" restituisce una versione dopo la disinstallazione di AWS CLI
- Hanno AWS CLI elaborato un comando con un nome di parametro incompleto
- Errori di accesso negato
- Credenziali non valide ed errori delle chiavi
- Errori di firma non corrispondente
- Errori di certificato SSL
- Errori JSON non validi
- Risorse aggiuntive

Risoluzione generale dei problemi come primo tentativo

Se ricevi un errore o riscontri un problema con il AWS CLI, ti suggeriamo i seguenti suggerimenti generali per aiutarti a risolverlo.

Torna all'inizio

Controlla la formattazione dei comandi AWS CLI

Se ricevi un errore in cui viene segnalato che un comando non esiste o che non viene riconosciuto un parametro (`Parameter validation failed`) indicato nella documentazione come disponibile, è possibile che il comando sia formattato in modo errato. Ti suggeriamo di controllare quanto segue:

- Controlla la presenza di errori di ortografia e formattazione nel comando.
- Verifica che tutte le virgolette e tutti i caratteri di escape appropriati per il terminale siano riportati correttamente nel comando.
- Genera uno scheletro della AWS CLI per verificare la struttura del comando.
- Per JSON, consulta la sezione aggiuntiva sulla risoluzione dei problemi relativi ai valori JSON. Se riscontri problemi con l'elaborazione della formattazione JSON nel terminale, ti suggeriamo di ignorare le regole di utilizzo delle virgolette del terminale utilizzando Blob per passare i dati JSON direttamente alla AWS CLI.

Per ulteriori informazioni su come deve essere strutturato un comando specifico, consulta la guida di di riferimento.

Torna all'inizio

Verifica che AWS CLI comando Regione AWS stia usando il tuo comando

> **Note**
>
> È necessario specificare un valore Regione AWS quando si utilizza la AWS CLI, in modo esplicito o impostando una regione predefinita. Per un elenco di tutto ciò Regioni AWS che è possibile specificare, consulta AWS Regioni ed endpoint in. Riferimenti generali di Amazon Web Services I Regione AWS designatori utilizzati da AWS CLI sono gli stessi nomi che vedi negli Console di gestione AWS URLs endpoint del servizio.

Potrebbero verificarsi errori o risultati imprevisti se uno non Servizio AWS è disponibile per i dati specificati Regione AWS o se le risorse si trovano in un altro. Regione AWS In ordine di precedenza, Regione AWS viene impostato nei seguenti modi:

- Opzione della riga di comando `--region`

- Variabile di ambiente `AWS_REGION` compatibile con l'SDK
- Variabile di ambiente `AWS_DEFAULT_REGION`
- Impostazione del profilo `region`

Conferma di utilizzare quello corretto Regione AWS per le tue risorse.

Torna all'inizio

Verifica di disporre di una versione recente della AWS CLI

Se ricevi un errore che indica che un comando non esiste o che non riconosce un parametro che la guida di di riferimento indica disponibile, verifica innanzitutto che il comando sia formattato correttamente. Se la formattazione è corretta, ti consigliamo di eseguire l'aggiornamento alla versione più recente della AWS CLI. Le versioni aggiornate di AWS CLI vengono rilasciate quasi ogni giorno lavorativo. AWS Nuovi servizi, funzionalità e parametri vengono introdotti in queste nuove versioni di AWS CLI. L'unico modo per ottenere l'accesso a questi nuovi servizi, caratteristiche o parametri è eseguire l'aggiornamento a una versione rilasciata dopo che l'elemento è stato introdotto per la prima volta.

La modalità di aggiornamento della versione di AWS CLI dipende da come è stata originariamente installata, come descritto in the section called "Installazione o aggiornamento".

Se hai utilizzato uno dei programmi di installazione in bundle, potresti dover rimuovere l'installazione esistente e quindi scaricare e installare la versione più recente per il sistema operativo in uso.

Torna all'inizio

Utilizza l'opzione `--debug`

Quando AWS CLI riporta un errore che non comprendi immediatamente o produce risultati inaspettati, puoi ottenere maggiori dettagli sull'errore eseguendo nuovamente il comando con l'`--debug`opzione. Con questa opzione, la AWS CLI restituisce i dettagli di ogni fase di elaborazione del comando. I dettagli riportati nell'output possono aiutarti a determinare quando si è verificato l'errore e fornire spunti sul contesto in cui ha avuto origine.

Puoi trasferire l'output in un file di testo per esaminarlo in un secondo momento o per inviarlo al Supporto AWS quando richiesto.

Quando utilizzi l'opzione `--debug`, tra i dettagli sono inclusi:

- Ricerca delle credenziali
- Analisi dei parametri specificati
- Costruzione della richiesta inviata ai server AWS
- Il contenuto della richiesta inviata a AWS
- Il contenuto della risposta non elaborata
- L'output formattato

Ecco un esempio di un comando eseguito con e senza l'opzione `--debug`.

```
$ aws iam list-groups --profile MyTestProfile
{
    "Groups": [
        {
            "Path": "/",
            "GroupName": "MyTestGroup",
            "GroupId": "AGPA0123456789EXAMPLE",
            "Arn": "arn:aws:iam::123456789012:group/MyTestGroup",
            "CreateDate": "2019-08-12T19:34:04Z"
        }
    ]
}
```

```
$ aws iam list-groups --profile MyTestProfile --debug
2019-08-12 12:36:18,305 - MainThread - awscli.clidriver - DEBUG - CLI version: aws-
cli/1.16.215 Python/3.7.3 Linux/4.14.133-113.105.amzn2.x86_64 botocore/1.12.205
2019-08-12 12:36:18,305 - MainThread - awscli.clidriver - DEBUG - Arguments entered to
 CLI: ['iam', 'list-groups', '--debug']
2019-08-12 12:36:18,305 - MainThread - botocore.hooks - DEBUG - Event session-
initialized: calling handler <function add_scalar_parsers at 0x7fdf173161e0>
2019-08-12 12:36:18,305 - MainThread - botocore.hooks - DEBUG - Event session-
initialized: calling handler <function register_uri_param_handler at 0x7fdf17dec400>
2019-08-12 12:36:18,305 - MainThread - botocore.hooks - DEBUG - Event session-
initialized: calling handler <function inject_assume_role_provider_cache at
 0x7fdf17da9378>
2019-08-12 12:36:18,307 - MainThread - botocore.credentials - DEBUG - Skipping
 environment variable credential check because profile name was explicitly set.
2019-08-12 12:36:18,307 - MainThread - botocore.hooks - DEBUG - Event session-
initialized: calling handler <function attach_history_handler at 0x7fdf173ed9d8>
```

```
2019-08-12 12:36:18,308 - MainThread - botocore.loaders - DEBUG - Loading JSON
 file: /home/ec2-user/venv/lib/python3.7/site-packages/botocore/data/iam/2010-05-08/
service-2.json
2019-08-12 12:36:18,317 - MainThread - botocore.hooks - DEBUG - Event building-command-
table.iam: calling handler <function add_waiters at 0x7fdf1731a840>
2019-08-12 12:36:18,320 - MainThread - botocore.loaders - DEBUG - Loading JSON
 file: /home/ec2-user/venv/lib/python3.7/site-packages/botocore/data/iam/2010-05-08/
waiters-2.json
2019-08-12 12:36:18,321 - MainThread - awscli.clidriver - DEBUG - OrderedDict([('path-
prefix', <awscli.arguments.CLIArgument object at 0x7fdf171ac780>), ('marker',
 <awscli.arguments.CLIArgument object at 0x7fdf171b09e8>), ('max-items',
 <awscli.arguments.CLIArgument object at 0x7fdf171b09b0>)])
2019-08-12 12:36:18,322 - MainThread - botocore.hooks - DEBUG - Event building-
argument-table.iam.list-groups: calling handler <function add_streaming_output_arg at
 0x7fdf17316510>
2019-08-12 12:36:18,322 - MainThread - botocore.hooks - DEBUG - Event building-
argument-table.iam.list-groups: calling handler <function add_cli_input_json at
 0x7fdf17da9d90>
2019-08-12 12:36:18,322 - MainThread - botocore.hooks - DEBUG - Event building-
argument-table.iam.list-groups: calling handler <function unify_paging_params at
 0x7fdf17328048>
2019-08-12 12:36:18,326 - MainThread - botocore.loaders - DEBUG - Loading JSON
 file: /home/ec2-user/venv/lib/python3.7/site-packages/botocore/data/iam/2010-05-08/
paginators-1.json
2019-08-12 12:36:18,326 - MainThread - awscli.customizations.paginate - DEBUG -
 Modifying paging parameters for operation: ListGroups
2019-08-12 12:36:18,326 - MainThread - botocore.hooks - DEBUG - Event building-
argument-table.iam.list-groups: calling handler <function add_generate_skeleton at
 0x7fdf1737eae8>
2019-08-12 12:36:18,326 - MainThread - botocore.hooks - DEBUG - Event
 before-building-argument-table-parser.iam.list-groups: calling handler
 <bound method OverrideRequiredArgsArgument.override_required_args of
 <awscli.customizations.cliinputjson.CliInputJSONArgument object at 0x7fdf171b0a58>>
2019-08-12 12:36:18,327 - MainThread - botocore.hooks - DEBUG - Event
 before-building-argument-table-parser.iam.list-groups: calling handler
 <bound method GenerateCliSkeletonArgument.override_required_args of
 <awscli.customizations.generatecliskeleton.GenerateCliSkeletonArgument object at
 0x7fdf171c5978>>
2019-08-12 12:36:18,327 - MainThread - botocore.hooks - DEBUG - Event operation-
args-parsed.iam.list-groups: calling handler functools.partial(<function
 check_should_enable_pagination at 0x7fdf17328158>, ['marker', 'max-items'], {'max-
items': <awscli.arguments.CLIArgument object at 0x7fdf171b09b0>}, OrderedDict([('path-
prefix', <awscli.arguments.CLIArgument object at 0x7fdf171ac780>), ('marker',
 <awscli.arguments.CLIArgument object at 0x7fdf171b09e8>), ('max-items',
```

```
 <awscli.customizations.paginate.PageArgument object at 0x7fdf171c58d0>), ('cli-
input-json', <awscli.customizations.cliinputjson.CliInputJSONArgument object at
 0x7fdf171b0a58>), ('starting-token', <awscli.customizations.paginate.PageArgument
 object at 0x7fdf171b0a20>), ('page-size', <awscli.customizations.paginate.PageArgument
 object at 0x7fdf171c5828>), ('generate-cli-skeleton',
 <awscli.customizations.generatecliskeleton.GenerateCliSkeletonArgument object at
 0x7fdf171c5978>)]))
2019-08-12 12:36:18,328 - MainThread - botocore.hooks - DEBUG - Event load-cli-
arg.iam.list-groups.path-prefix: calling handler <awscli.paramfile.URIArgumentHandler
 object at 0x7fdf1725c978>
2019-08-12 12:36:18,328 - MainThread - botocore.hooks - DEBUG - Event load-cli-
arg.iam.list-groups.marker: calling handler <awscli.paramfile.URIArgumentHandler object
 at 0x7fdf1725c978>
2019-08-12 12:36:18,328 - MainThread - botocore.hooks - DEBUG - Event load-cli-
arg.iam.list-groups.max-items: calling handler <awscli.paramfile.URIArgumentHandler
 object at 0x7fdf1725c978>
2019-08-12 12:36:18,328 - MainThread - botocore.hooks - DEBUG -
 Event load-cli-arg.iam.list-groups.cli-input-json: calling handler
 <awscli.paramfile.URIArgumentHandler object at 0x7fdf1725c978>
2019-08-12 12:36:18,328 - MainThread - botocore.hooks - DEBUG -
 Event load-cli-arg.iam.list-groups.starting-token: calling handler
 <awscli.paramfile.URIArgumentHandler object at 0x7fdf1725c978>
2019-08-12 12:36:18,328 - MainThread - botocore.hooks - DEBUG - Event load-cli-
arg.iam.list-groups.page-size: calling handler <awscli.paramfile.URIArgumentHandler
 object at 0x7fdf1725c978>
2019-08-12 12:36:18,328 - MainThread - botocore.hooks - DEBUG - Event
 load-cli-arg.iam.list-groups.generate-cli-skeleton: calling handler
 <awscli.paramfile.URIArgumentHandler object at 0x7fdf1725c978>
2019-08-12 12:36:18,329 - MainThread - botocore.hooks - DEBUG
 - Event calling-command.iam.list-groups: calling handler
 <bound method CliInputJSONArgument.add_to_call_parameters of
 <awscli.customizations.cliinputjson.CliInputJSONArgument object at 0x7fdf171b0a58>>
2019-08-12 12:36:18,329 - MainThread - botocore.hooks - DEBUG -
 Event calling-command.iam.list-groups: calling handler <bound
 method GenerateCliSkeletonArgument.generate_json_skeleton of
 <awscli.customizations.generatecliskeleton.GenerateCliSkeletonArgument object at
 0x7fdf171c5978>>
2019-08-12 12:36:18,329 - MainThread - botocore.credentials - DEBUG - Looking for
 credentials via: assume-role
2019-08-12 12:36:18,329 - MainThread - botocore.credentials - DEBUG - Looking for
 credentials via: assume-role-with-web-identity
2019-08-12 12:36:18,329 - MainThread - botocore.credentials - DEBUG - Looking for
 credentials via: shared-credentials-file
```

```
2019-08-12 12:36:18,329 - MainThread - botocore.credentials - INFO - Found credentials
 in shared credentials file: ~/.aws/credentials
2019-08-12 12:36:18,330 - MainThread - botocore.loaders - DEBUG - Loading JSON file: /
home/ec2-user/venv/lib/python3.7/site-packages/botocore/data/endpoints.json
2019-08-12 12:36:18,334 - MainThread - botocore.hooks - DEBUG - Event choose-service-
name: calling handler <function handle_service_name_alias at 0x7fdf1898eb70>
2019-08-12 12:36:18,337 - MainThread - botocore.hooks - DEBUG - Event creating-client-
class.iam: calling handler <function add_generate_presigned_url at 0x7fdf18a028c8>
2019-08-12 12:36:18,337 - MainThread - botocore.regions - DEBUG - Using partition
 endpoint for iam, us-west-2: aws-global
2019-08-12 12:36:18,337 - MainThread - botocore.args - DEBUG - The s3 config key is not
 a dictionary type, ignoring its value of: None
2019-08-12 12:36:18,340 - MainThread - botocore.endpoint - DEBUG - Setting iam timeout
 as (60, 60)
2019-08-12 12:36:18,341 - MainThread - botocore.loaders - DEBUG - Loading JSON file: /
home/ec2-user/venv/lib/python3.7/site-packages/botocore/data/_retry.json
2019-08-12 12:36:18,341 - MainThread - botocore.client - DEBUG - Registering retry
 handlers for service: iam
2019-08-12 12:36:18,342 - MainThread - botocore.hooks - DEBUG - Event before-
parameter-build.iam.ListGroups: calling handler <function generate_idempotent_uuid at
 0x7fdf189b10d0>
2019-08-12 12:36:18,342 - MainThread - botocore.hooks - DEBUG - Event before-
call.iam.ListGroups: calling handler <function inject_api_version_header_if_needed at
 0x7fdf189b2a60>
2019-08-12 12:36:18,343 - MainThread - botocore.endpoint - DEBUG - Making
 request for OperationModel(name=ListGroups) with params: {'url_path': '/',
 'query_string': '', 'method': 'POST', 'headers': {'Content-Type': 'application/x-
www-form-urlencoded; charset=utf-8', 'User-Agent': 'aws-cli/1.16.215 Python/3.7.3
 Linux/4.14.133-113.105.amzn2.x86_64 botocore/1.12.205'}, 'body': {'Action':
 'ListGroups', 'Version': '2010-05-08'}, 'url': 'https://iam.amazonaws.com/',
 'context': {'client_region': 'aws-global', 'client_config': <botocore.config.Config
 object at 0x7fdf16e9a4a8>, 'has_streaming_input': False, 'auth_type': None}}
2019-08-12 12:36:18,343 - MainThread - botocore.hooks - DEBUG - Event request-
created.iam.ListGroups: calling handler <bound method RequestSigner.handler of
 <botocore.signers.RequestSigner object at 0x7fdf16e9a470>>
2019-08-12 12:36:18,343 - MainThread - botocore.hooks - DEBUG - Event choose-
signer.iam.ListGroups: calling handler <function set_operation_specific_signer at
 0x7fdf18996f28>
2019-08-12 12:36:18,343 - MainThread - botocore.auth - DEBUG - Calculating signature
 using v4 auth.
2019-08-12 12:36:18,343 - MainThread - botocore.auth - DEBUG - CanonicalRequest:
POST
/
```

```
content-type:application/x-www-form-urlencoded; charset=utf-8
host:iam.amazonaws.com
x-amz-date:20190812T193618Z

content-type;host;x-amz-date
5f776d91EXAMPLE9b8cb5eb5d6d4a787a33ae41c8cd6eEXAMPLEca69080e1e1f
2019-08-12 12:36:18,344 - MainThread - botocore.auth - DEBUG - StringToSign:
AWS4-HMAC-SHA256
20190812T193618Z
20190812/us-east-1/iam/aws4_request
ab7e367eEXAMPLE2769f178ea509978cf8bfa054874b3EXAMPLE8d043fab6cc9
2019-08-12 12:36:18,344 - MainThread - botocore.auth - DEBUG - Signature:
d85a0EXAMPLEb40164f2f539cdc76d4f294fe822EXAMPLE18ad1ddf58a1a3ce7
2019-08-12 12:36:18,344 - MainThread - botocore.endpoint - DEBUG - Sending
 http request: <AWSPreparedRequest stream_output=False, method=POST,
 url=https://iam.amazonaws.com/, headers={'Content-Type': b'application/
x-www-form-urlencoded; charset=utf-8', 'User-Agent': b'aws-cli/1.16.215
 Python/3.7.3 Linux/4.14.133-113.105.amzn2.x86_64 botocore/1.12.205',
 'X-Amz-Date': b'20190812T193618Z', 'Authorization': b'AWS4-HMAC-SHA256
 Credential=AKIA01234567890EXAMPLE-east-1/iam/aws4_request, SignedHeaders=content-
type;host;x-amz-date, Signature=d85a07692aceb401EXAMPLEa1b18ad1ddf58a1a3ce7EXAMPLE',
 'Content-Length': '36'}>
2019-08-12 12:36:18,344 - MainThread - urllib3.util.retry - DEBUG - Converted retries
 value: False -> Retry(total=False, connect=None, read=None, redirect=0, status=None)
2019-08-12 12:36:18,344 - MainThread - urllib3.connectionpool - DEBUG - Starting new
 HTTPS connection (1): iam.amazonaws.com:443
2019-08-12 12:36:18,664 - MainThread - urllib3.connectionpool - DEBUG - https://
iam.amazonaws.com:443 "POST / HTTP/1.1" 200 570
2019-08-12 12:36:18,664 - MainThread - botocore.parsers - DEBUG - Response headers:
 {'x-amzn-RequestId': '74c11606-bd38-11e9-9c82-559da0adb349', 'Content-Type': 'text/
xml', 'Content-Length': '570', 'Date': 'Mon, 12 Aug 2019 19:36:18 GMT'}
2019-08-12 12:36:18,664 - MainThread - botocore.parsers - DEBUG - Response body:
b'<ListGroupsResponse xmlns="https://iam.amazonaws.com/doc/2010-05-08/">\n
 <ListGroupsResult>\n    <IsTruncated>false</IsTruncated>\n    <Groups>\n
  <member>\n        <Path>/</Path>\n        <GroupName>MyTestGroup</GroupName>
\n        <Arn>arn:aws:iam::123456789012:group/MyTestGroup</Arn>\n
 <GroupId>AGPA1234567890EXAMPLE</GroupId>\n        <CreateDate>2019-08-12T19:34:04Z</
CreateDate>\n      </member>\n    </Groups>\n  </ListGroupsResult>\n
 <ResponseMetadata>\n    <RequestId>74c11606-bd38-11e9-9c82-559da0adb349</RequestId>\n
 </ResponseMetadata>\n</ListGroupsResponse>\n'
2019-08-12 12:36:18,665 - MainThread - botocore.hooks - DEBUG - Event needs-
retry.iam.ListGroups: calling handler <botocore.retryhandler.RetryHandler object at
 0x7fdf16e9a780>
2019-08-12 12:36:18,665 - MainThread - botocore.retryhandler - DEBUG - No retry needed.
```

```
2019-08-12 12:36:18,665 - MainThread - botocore.hooks - DEBUG - Event after-
call.iam.ListGroups: calling handler <function json_decode_policies at 0x7fdf189b1d90>
{
    "Groups": [
        {
            "Path": "/",
            "GroupName": "MyTestGroup",
            "GroupId": "AGPA123456789012EXAMPLE",
            "Arn": "arn:aws:iam::123456789012:group/MyTestGroup",
            "CreateDate": "2019-08-12T19:34:04Z"
        }
    ]
}
```

Torna all'inizio

Abilita e rivedi i registri della cronologia dei AWS CLI comandi

È possibile abilitare i registri della cronologia dei AWS CLI comandi utilizzando l'impostazione del `cli_history` file. Dopo aver abilitato questa impostazione, AWS CLI registra la cronologia dei `aws` comandi.

Puoi generare l'elenco della cronologia utilizzando il comando `aws history list` e utilizzare il valore di command_ids risultante nel comando `aws history show` per i dettagli. Per ulteriori informazioni, consulta `aws history` nella Guida di riferimento della AWS CLI .

Quando utilizzi l'opzione `--debug`, tra i dettagli sono inclusi:

- Chiamate API effettuate a botocore
- Codici di stato
- Risposte HTTP
- Headers
- Codici restituiti

Puoi utilizzare queste informazioni per verificare che i dati dei parametri e le chiamate API si comportino nel modo previsto e quindi dedurre in quale fase del processo viene generato l'errore del comando.

Torna all'inizio

Conferma che il tuo AWS CLI è configurato

Se il tuo utente o ruolo IAM oppure i tuoi file `credentials` e `config` non sono configurati correttamente, possono verificarsi diversi errori. Per ulteriori informazioni sulla risoluzione degli errori relativi ai file `credentials` e `config` oppure all'utente o ai ruoli IAM, consulta the section called "Errori di accesso negato" e the section called "Credenziali non valide ed errori delle chiavi".

Torna all'inizio

Errori di comando non trovato

Questo errore indica che il sistema operativo non riesce a trovare il AWS CLI comando. L'installazione potrebbe essere incompleta o deve essere aggiornata.

Possibile causa: stai cercando di utilizzare una AWS CLI funzionalità più recente della versione installata o hai una formattazione errata

Esempio di testo dell'errore:

```
$ aws s3 copy
aws: [ERROR]: argument operation: Found invalid choice 'copy'

usage: aws [options] <command> <subcommand> [<subcommand> ...] [parameters]
To see help text, you can run:

  aws help
  aws <command> help
  aws <command> <subcommand> help
```

Se il comando è formattato in modo errato o se utilizzi una versione precedente al rilascio della funzionalità, possono verificarsi diversi errori. Per ulteriori informazioni sulla risoluzione degli errori relativi a questi due problemi, consulta the section called "Controlla la formattazione dei comandi AWS CLI" e the section called "Verifica di disporre di una versione recente della AWS CLI".

Torna all'inizio

Possibile causa: il terminale deve essere riavviato dopo l'installazione

Esempio di testo dell'errore:

```
$ aws --version
```

```
command not found: aws
```

Se il `aws` comando non viene trovato dopo la prima installazione o l'aggiornamento di AWS CLI, potrebbe essere necessario riavviare il terminale per far sì che riconosca eventuali `PATH` aggiornamenti.

Torna all'inizio

Possibile causa: l'installazione AWS CLI non è stata completata

Esempio di testo dell'errore:

```
$ aws --version
command not found: aws
```

Se il `aws` comando non viene trovato dopo la prima installazione o l'aggiornamento di AWS CLI, è possibile che non sia stato installato completamente. Prova a reinstallarla seguendo le fasi specifiche per la piattaforma in uso riportate in the section called "Installazione o aggiornamento".

Torna all'inizio

Possibile causa: AWS CLI non dispone di autorizzazioni (Linux)

Se il `aws` comando non viene trovato dopo la prima installazione o l'aggiornamento AWS CLI su Linux, è possibile che non disponga `execute` delle autorizzazioni per la cartella in cui è installato. Esegui il seguente comando con la `PATH` tua AWS CLI installazione, per fornire `chmod` le autorizzazioni a: AWS CLI

```
$ sudo chmod -R 755 /usr/local/aws-cli/
```

Torna all'inizio

Possibile causa: la variabile **`PATH`** del sistema operativo non è stata aggiornata durante l'installazione

Esempio di testo dell'errore:

```
$ aws --version
command not found: aws
```

Potrebbe essere necessario aggiungere l'eseguibile `aws` alla variabile di ambiente `PATH` del sistema operativo in uso. Per aggiungerlo AWS CLI al tuo`PATH`, usa le seguenti istruzioni per il tuo sistema operativo.

Linux and macOS

1. Individua lo script del profilo della shell nella directory utente. Se non hai la certezza di quale sia la tua shell, esegui `echo $SHELL`.

```
$ ls -a ~
.  ..  .bash_logout  .bash_profile  .bashrc  Desktop  Documents  Downloads
```

   - Bash - `.bash_profile`, `.profile` o `.bash_login`
   - Zsh – `.zshrc`
   - Tcsh - `.tcshrc`, `.cshrc` o `.login`

2. Aggiungi un comando di esportazione allo script del tuo profilo. Il comando seguente aggiunge il contenitore locale alla variabile PATH corrente.

```
export PATH=/usr/local/bin:$PATH
```

3. Ricarica il profilo aggiornato nella sessione corrente.

```
$ source ~/.bash_profile
```

Windows

1. In un prompt dei comandi di Windows, utilizza il comando `where` con il parametro `/R` *`path`* per trovare la posizione del file `aws`. I risultati restituiti includono tutte le cartelle contenenti `aws`.

```
C:\> where /R c:\ aws
c:\Program Files\Amazon\AWSCLIV2\aws.exe
...
```

   Per impostazione predefinita, la AWS CLI versione 2 si trova in:

```
c:\Program Files\Amazon\AWSCLIV2\aws.exe
```

2. Premere il tasto Windows e immettere **`environment variables`**.
3. Dall'elenco di suggerimenti, seleziona Modifica variabili di ambiente per l'account.
4. Seleziona PATH, quindi scegli Modifica.

5. Nel campo Valore variabile, aggiungi il percorso trovato nel primo passaggio, ad esempio ***`C:\Program Files\Amazon\AWSCLIV2\aws.exe`***.
6. Fai doppio clic su OK per applicare le nuove impostazioni.
7. Chiudi tutte le richieste di comando in esecuzione e riapri la finestra del prompt dei comandi.

Torna all'inizio

Il comando **`aws --version`** restituisce una versione diversa da quella installata

Il tuo terminale potrebbe restituire un `PATH` messaggio diverso AWS CLI da quello previsto.

Possibile causa: il terminale deve essere riavviato dopo l'installazione

Se il comando `aws` mostra la versione errata, potrebbe essere necessario riavviare il terminale in modo che riconosca gli aggiornamenti della variabile `PATH`. Tutti i terminali aperti devono essere chiusi, non solo quello attivo.

Torna all'inizio

Possibile causa: il sistema deve essere riavviato dopo l'installazione

Se il comando `aws` mostra la versione errata e il riavvio del terminale non ha funzionato, potrebbe essere necessario riavviare il sistema per far sì che riconosca gli aggiornamenti della variabile `PATH`.

Torna all'inizio

Possibile causa: sono disponibili più versioni di AWS CLI

Se hai aggiornato AWS CLI e utilizzato un metodo di installazione diverso da quello dell'installazione preesistente, è possibile che vengano installate più versioni. Ad esempio, se su Linux o macOS hai utilizzato `pip` per l'installazione corrente, ma hai provato a eseguire l'aggiornamento utilizzando il file di installazione `.pkg`, è possibile che si verifichino alcuni conflitti, soprattutto con la variabile `PATH` che punta alla versione precedente.

Per risolvere il problema, disinstalla tutte le versioni della AWS CLI ed esegui un'installazione pulita.

Dopo aver disinstallato tutte le versioni, segui le istruzioni appropriate per il sistema operativo in uso in modo da installare la versione desiderata della AWS CLI versione 1 o della AWS CLI versione 2.

> **Note**
>
> Se ciò accade dopo aver installato la AWS CLI versione 2 con un'installazione preesistente della AWS CLI versione 1, segui le istruzioni di migrazione contenute nelle Istruzioni di versione 1. the section called "Istruzioni di migrazione"

Torna all'inizio

Il comando "`aws --version`" restituisce una versione dopo la disinstallazione di AWS CLI

Ciò si verifica spesso quando ne è ancora AWS CLI installato uno da qualche parte sul sistema.

Possibile causa: il terminale deve essere riavviato dopo la disinstallazione

Se il comando `aws --version` funziona ancora, potrebbe essere necessario riavviare il terminale per far sì che riconosca eventuali aggiornamenti.

Torna all'inizio

Possibile causa: AWS CLI sul sistema sono presenti più versioni di o non è stato utilizzato lo stesso metodo di disinstallazione utilizzato originariamente per l'installazione di AWS CLI

AWS CLI Potrebbe non essere disinstallato correttamente se lo hai disinstallato AWS CLI utilizzando un metodo diverso da quello usato per installarlo o se hai installato più versioni. Ad esempio, se hai utilizzato `pip` per l'installazione corrente, devi utilizzare `pip` anche per la disinstallazione. Per risolvere il problema, esegui la disinstallazione AWS CLI utilizzando lo stesso metodo utilizzato per l'installazione.

1. Segui le istruzioni appropriate per il sistema operativo in uso e per il metodo di installazione utilizzato originariamente per disinstallare la AWS CLI versione 1 e la AWS CLI versione 2.
2. Chiudi tutti i terminali aperti.
3. Apri il tuo terminale preferito, inserisci il seguente comando e verifica che non venga restituita alcuna versione.

```
$ aws --version
command not found: aws
```

Se nell'output è ancora presente una versione, è probabile che sia AWS CLI stata installata utilizzando un metodo diverso o che esistano più versioni. Se non sai quale metodo hai installato AWS CLI, segui le istruzioni per ogni metodo di disinstallazione per la AWS CLI versione 1 e la AWS CLI versione 2 appropriati al tuo sistema operativo fino a quando non viene ricevuta alcuna versione in uscita.

> **Note**
>
> Se per installare la AWS CLI hai utilizzato un gestore di pacchetti (`pip`, `apt`, `brew` e così via), devi utilizzare lo stesso gestore di pacchetti per disinstallarla. Assicurati di seguire le istruzioni fornite dal gestore di pacchetti su come disinstallare tutte le relative versioni.

Torna all'inizio

Hanno AWS CLI elaborato un comando con un nome di parametro incompleto

Possibile causa: è stata utilizzata un'abbreviazione riconosciuta del parametro AWS CLI

Poiché AWS CLI è stato creato utilizzando Python, AWS CLI utilizza la `argparse` libreria Python, incluso l'argomento. `allow_abbrev` Le abbreviazioni dei parametri vengono riconosciute da ed elaborate AWS CLI .

Il seguente esempio di comando modifica il nome dello CloudFormation stack. Il parametro `--change-set-n` viene riconosciuto come abbreviazione di `--change-set-name` ed AWS CLI elabora il comando.

```
$ aws cloudformation create-change-set --stack-name my-stack --change-set-n my-change-set
```

Quando l'abbreviazione può essere utilizzata per più comandi, il parametro non viene riconosciuto come abbreviazione.

Il seguente esempio di comando modifica il nome dello CloudFormation stack. Il parametro `--change-set-` non è riconosciuto come abbreviazione poiché potrebbe essere un'abbreviazione di più parametri, ad esempio `--change-set-name` e `--change-set-type`. Pertanto AWS CLI non elabora il comando.

```
$ aws cloudformation create-change-set --stack-name my-stack --change-set- my-change-set
```

> **Warning**
>
> Non utilizzare intenzionalmente le abbreviazioni dei parametri. Non sono affidabili e non sono retrocompatibili. Se ai comandi vengono aggiunti nuovi parametri che rendono dubbie le abbreviazioni, i comandi vengono interrotti.
> Inoltre, se il parametro è un argomento a valore singolo, può causare un comportamento imprevisto dei comandi. Se vengono passate più istanze di un argomento a valore singolo, viene eseguita solo l'ultima istanza. Nell'esempio seguente, il parametro `--filters` è un argomento a valore singolo. Sono specificati entrambi i parametri `--filters` e `--filter`. Il parametro `--filter` è l'abbreviazione di `--filters`. Ciò comporterebbe l'applicazione di due istanze di `--filters`, ma in realtà viene applicato solo l'ultimo argomento `--filter`.
>
> ```
> $ aws ec2 describe-vpc-peering-connections \
>     --filters Name=tag:TagName,Values=VpcPeeringConnection \
>     --filter Name=status-code,Values=active
> ```
>
> Prima di eseguire un comando, verifica di utilizzare parametri validi per evitare comportamenti imprevisti.

Torna all'inizio

Errori di accesso negato

Possibile causa: il file di AWS CLI programma non dispone del permesso di «esecuzione»

Su Linux o macOS, assicurati che il programma `aws` disponga delle autorizzazioni di esecuzione per l'utente che effettua la chiamata. In genere, le autorizzazioni sono impostate su `755`.

Per aggiungere l'autorizzazione di esecuzione per l'utente, esegui il comando seguente, sostituendolo *~/.local/bin/aws* con il percorso del programma sul tuo computer.

```
$ chmod +x ~/.local/bin/aws
```

Torna all'inizio

Possibile causa: l'identità IAM non dispone dell'autorizzazione per eseguire l'operazione

Esempio di testo dell'errore:

```
$ aws s3 ls
An error occurred (AcessDenied) when calling the ListBuckets operation: Access
 denied.
```

Quando esegui un AWS CLI comando, AWS le operazioni vengono eseguite per tuo conto, utilizzando credenziali che ti associano a un account o a un ruolo IAM. Le policy associate devono concedere l'autorizzazione per chiamare le operazioni API corrispondenti ai comandi eseguiti con la AWS CLI.

La maggior parte dei comandi chiamano una singola operazione con un nome corrispondente al nome del comando. Tuttavia, comandi personalizzati come le `aws s3 sync` chiamate multiple APIs. È possibile vedere quale APIs comando chiama utilizzando l'`--debug`opzione.

Se sei sicuro che l'utente o il ruolo disponga delle autorizzazioni appropriate assegnate dalla policy, assicurati che il AWS CLI comando utilizzi le credenziali previste. Consultate la sezione successiva sulle credenziali per verificare che le credenziali utilizzate AWS CLI siano quelle previste.

Per informazioni sull'assegnazione delle autorizzazioni IAM, consulta Panoramica della gestione degli accessi: autorizzazioni e policy nella Guida per l'utente di IAM.

Torna all'inizio

Credenziali non valide ed errori delle chiavi

Esempio di testo dell'errore:

```
$ aws s3 ls
An error occurred (InvalidAccessKeyId) when calling the ListBuckets operation: The AWS
 Access Key Id
```

```
you provided does not exist in our records.
```

```
$ aws s3 ls
An error occurred (InvalidClientTokenId) when calling the ListBuckets operation: The
 security token
included in the request is invalid.
```

Possibile causa: AWS CLI sta leggendo credenziali errate o da una posizione inaspettata

È AWS CLI possibile che stia leggendo le credenziali da una posizione diversa da quella prevista o che le informazioni sulla coppia di chiavi non siano corrette. Puoi eseguire `aws configure list` per verificare le credenziali utilizzate.

L'esempio seguente mostra come controllare le credenziali utilizzate per il profilo predefinito.

```
$ aws configure list
NAME       : VALUE                : TYPE                    : LOCATION
profile    : <not set>            : None                    : None
access_key : ****************ABCD : shared-credentials-file :
secret_key : ****************ABCD : shared-credentials-file :
region     : us-west-2            : env                     : AWS_DEFAULT_REGION
```

L'esempio seguente mostra come controllare le credenziali di un profilo denominato.

```
$ aws configure list --profile dev01
NAME       : VALUE                : TYPE                    : LOCATION
profile    : dev01                : None                    : --profile
access_key : ****************ABCD : shared-credentials-file :
secret_key : ****************ABCD : shared-credentials-file :
region     : us-west-2            : config-file             : ~/.aws/config
```

Per verificare i dettagli della tua coppia di chiavi, controlla i file `credentials` e `config`. Per ulteriori informazioni sui file `config` e `credentials`, consulta the section called "Impostazioni di configurazione". Per ulteriori informazioni sulle credenziali e sull'autenticazione, inclusa la precedenza delle credenziali, consulta Credenziali di autenticazione e accesso.

Torna all'inizio

Possibile causa: l'orologio del computer non è sincronizzato

Se utilizzi credenziali valide, l'orologio potrebbe non essere sincronizzato. Su Linux o macOS, esegui `date` per controllare l'ora.

```
$ date
```

Se l'orologio di sistema non viene corretto nel giro di pochi minuti, utilizza `ntpd` per sincronizzarlo.

```
$ sudo service ntpd stop
$ sudo ntpdate time.nist.gov
$ sudo service ntpd start
$ ntpstat
```

In Windows utilizza le opzioni di data e ora nel pannello di controllo per configurare l'orologio di sistema.

Torna all'inizio

Errori di firma non corrispondente

Esempio di testo dell'errore:

```
$ aws s3 ls
An error occurred (SignatureDoesNotMatch) when calling the ListBuckets operation: The
 request signature we
calculated does not match the signature you provided. Check your key and signing
 method.
```

Quando AWS CLI esegue un comando, invia una richiesta crittografata ai AWS server per eseguire le operazioni di AWS servizio appropriate. Le credenziali (la chiave di accesso e la chiave segreta) sono coinvolte nella crittografia e AWS consentono l'autenticazione della persona che effettua la richiesta. Ci sono diversi aspetti che possono interferire con il corretto funzionamento di questo processo, come illustrato di seguito:

Possibile causa: l'orologio non è sincronizzato con i server AWS

Per proteggersi dagli attacchi replay, è possibile utilizzare l'ora corrente durante il encryption/decryption processo. Se l'ora del client e quella del server non corrispondono per un intervallo superiore a quello consentito, il processo può avere esito negativo e la richiesta viene rifiutata. Ciò si può verificare anche quando esegui un comando in una macchina virtuale il cui orologio non è sincronizzato con l'orologio del computer host. Una possibile causa è legata al caso in cui la macchina virtuale entra in stato di ibernazione e l'intervallo richiesto per la sua riattivazione e la successiva sincronizzazione dell'orologio con il computer host risulta eccessivo.

Su Linux o macOS, esegui `date` per controllare l'ora.

```
$ date
```

Se l'orologio di sistema non viene corretto nel giro di pochi minuti, utilizza `ntpd` per sincronizzarlo.

```
$ sudo service ntpd stop
$ sudo ntpdate time.nist.gov
$ sudo service ntpd start
$ ntpstat
```

In Windows utilizza le opzioni di data e ora nel pannello di controllo per configurare l'orologio di sistema.

Torna all'inizio

Possibile causa: il sistema operativo sta gestendo male AWS le chiavi che contengono determinati caratteri speciali

Se le AWS chiavi includono determinati caratteri speciali, ad esempio`-`,, o `+ /%`, alcune varianti del sistema operativo elaborano la stringa in modo errato e causano un'interpretazione errata della stringa chiave.

Se elaborate le chiavi utilizzando altri strumenti o script, ad esempio strumenti che creano il file delle credenziali su una nuova istanza durante la sua creazione, tali strumenti e script potrebbero avere una propria gestione dei caratteri speciali che li trasforma in qualcosa che non è più riconoscibile. AWS

Ti suggeriamo di rigenerare la chiave segreta in modo da ottenerne una in cui non è presente il carattere speciale che causa problemi.

Torna all'inizio

Errori di certificato SSL

Possibile causa: AWS CLI non si fida del certificato del proxy

Esempio di testo dell'errore:

```
$ aws s3 ls
```

```
[SSL: CERTIFICATE_VERIFY_FAILED] certificate verify failed
```

Quando si utilizza un AWS CLI comando, viene visualizzato un messaggio `[SSL: CERTIFICATE_VERIFY_FAILED] certificate verify failed` di errore. Ciò è dovuto al fatto che il certificato del proxy AWS CLI non è considerato affidabile a causa di fattori quali l'autofirma del certificato del proxy e l'autorità di certificazione (CA) della società. Ciò impedisce alla AWS CLI di trovare il certificato principale CA della società nel registro CA locale.

Per risolvere questo problema, indica AWS CLI dove trovare il file aziendale utilizzando l'impostazione `.pem` del file di `ca_bundle` configurazione, l'opzione della riga di `--ca-bundle` comando o la `AWS_CA_BUNDLE` variabile di ambiente.

Torna all'inizio

Possibile causa: la configurazione non punta alla posizione corretta del certificato root della CA

Esempio di testo dell'errore:

```
$ aws s3 ls
SSL validation failed for regionname [Errno 2] No such file or directory
```

Ciò è dovuto al fatto che la posizione del file del bundle della CA non è configurata correttamente nella AWS CLI. Per risolvere questo problema, indica alla `.pem` la posizione in cui trovare il file AWS CLI della società utilizzando l'impostazione del file di configurazione `ca_bundle`, l'opzione della riga di comando `--ca-bundle` o la variabile di ambiente `AWS_CA_BUNDLE`.

Torna all'inizio

Possibile causa: la tua configurazione non utilizza il codice corretto Regione AWS

Esempio di testo dell'errore:

```
$ aws s3 ls
[SSL: CERTIFICATE_ VERIFY_FAILED] certificate verify failed
```

Potrebbero verificarsi errori o risultati imprevisti se uno non Servizio AWS è disponibile per i dati specificati Regione AWS o se le risorse si trovano in un altro Regione AWS. Per la risoluzione dei problemi, consulta the section called "Verifica che AWS CLI comando Regione AWS stia usando il tuo comando".

Torna all'inizio

Possibile causa: la versione di TLS deve essere aggiornata

Esempio di testo dell'errore:

```
$ aws s3 ls
[SSL: UNSAFE_LEGACY_RENEGOTIATION_DISABLED] unsafe legacy renegotiation disabled
```

Servizio AWS Sta utilizzando una versione di TLS incompatibile con la versione TLS del dispositivo. Per risolvere il problema, esegui l'aggiornamento a una versione di TLS supportata. Per ulteriori informazioni, consulta the section called "Applicazione di una versione minima di TLS".

Torna all'inizio

Errori JSON non validi

Esempio di testo dell'errore:

```
$ aws dynamodb update-table \
    --provisioned-throughput '{"ReadCapacityUnits":15,WriteCapacityUnits":10}' \
    --table-name MyDDBTable
Error parsing parameter '--provisioned-throughput': Invalid JSON: Expecting property
 name enclosed in
double quotes: line 1 column 25 (char 24)
JSON received: {"ReadCapacityUnits":15,WriteCapacityUnits":10}
```

Quando si utilizza un AWS CLI comando, viene visualizzato un messaggio di errore `Invalid JSON` "». Di solito si tratta di un errore visualizzato quando si immette un comando con un formato JSON previsto e AWS CLI non è possibile leggere correttamente il codice JSON.

Possibile causa: non hai inserito un codice JSON valido da utilizzare AWS CLI

Verifica di aver inserito codice JSON valido per il comando. Se riscontri problemi di formattazione del codice JSON, ti suggeriamo di utilizzare un validatore JSON.

Per un utilizzo più avanzato del codice JSON su una riga di comando, prendi in considerazione l'utilizzo di un processore JSON a riga di comando, ad esempio `jq`, per creare stringhe JSON. Per ulteriori informazioni su`jq`, consulta il repository jq su. GitHub

Torna all'inizio

Possibile causa: le regole di quotazione del terminale impediscono l'invio di un codice JSON valido a AWS CLI

Prima di AWS CLI ricevere qualcosa da un comando, il terminale elabora il comando utilizzando le proprie regole di citazione ed escape. A causa delle regole di formattazione di un terminale, parte del contenuto JSON potrebbe essere rimosso prima che il comando venga passato alla AWS CLI. Quando formuli i comandi, assicurati di applicare le regole di utilizzo delle virgolette del terminale.

Per risolvere il problema, utilizza il comando `echo` per vedere in che modo i parametri vengono gestiti dalla shell:

```
$ echo {"ReadCapacityUnits":15,"WriteCapacityUnits":10}
ReadCapacityUnits:15 WriteCapacityUnits:10
```

```
$ echo '{"ReadCapacityUnits":15,"WriteCapacityUnits":10}'
{"ReadCapacityUnits":15,"WriteCapacityUnits":10}
```

Modifica il comando finché non viene restituito codice JSON valido.

Per una risoluzione dei problemi più approfondita, utilizza il parametro `--debug` per visualizzare i log di debug che mostreranno i dati esatti passati alla AWS CLI:

```
$ aws dynamodb update-table \
    --provisioned-throughput '{"ReadCapacityUnits":15,WriteCapacityUnits":10}' \
    --table-name MyDDBTable \
    --debug
2022-07-19 22:25:07,741 - MainThread - awscli.clidriver - DEBUG - CLI version: aws-
cli/1.18.147
Python/2.7.18 Linux/5.4.196-119.356.amzn2int.x86_64 botocore/1.18.6
2022-07-19 22:25:07,741 - MainThread - awscli.clidriver - DEBUG - Arguments entered
 to CLI:
['dynamodb', 'update-table', '--provisioned-throughput',
 '{"ReadCapacityUnits":15,WriteCapacityUnits":10}',
 '--table-name', 'MyDDBTable', '--debug']
```

Applica le regole di utilizzo delle virgolette del terminale per risolvere eventuali problemi riscontrati dall'input JSON quando viene inviato alla AWS CLI. Per ulteriori informazioni sulle regole di utilizzo delle virgolette, consulta the section called "Virgolette con stringhe".

> **Note**
>
> Se hai problemi a inserire un codice JSON valido su AWS CLI, ti consigliamo di aggirare le regole di quotazione di un terminale per l'immissione di dati JSON utilizzando Blobs per passare i dati JSON direttamente a. AWS CLI Per ulteriori informazioni sui Blob, consulta Blob.

Torna all'inizio

Risorse aggiuntive

Per ulteriore assistenza con i tuoi AWS CLI problemi, visita la community su o la community.AWS CLIGitHubAWS re:Post

Torna all'inizio

Esempio per la AWS CLI

Questo capitolo fornisce una raccolta di esempi che mostrano come utilizzare AWS Command Line Interface (AWS CLI) con Servizi AWS.

In questa guida sono riportati i seguenti tipi di esempi per la AWS CLI:

- Esempi di comandi guidati - Esempi di comandi guidati per la Guida per l'utente della AWS CLI su come utilizzare la AWS CLI con alcuni Servizi AWS. Questi sono spesso esempi più dettagliati rispetto agli esempi della AWS CLI version 2 reference guide.
- Esempi di comandi della AWS CLI - Esempi di comandi open source disponibili anche nella AWS CLI version 2 reference guide. Gli esempi dei comandi sono ospitati nel repository della AWS CLI su GitHub.
- AWS CLI con esempi di codice di script Bash - Esempi di script Bash open source. Gli esempi di script Bash sono ospitati nel repository degli esempi di codice AWS su GitHub.

> **Feedback di esempio**
>
> Non riesci a trovare quello che ti serve? Richiedi un esempio di comando utilizzando il link Fornisci feedback nella parte inferiore di questa pagina o nella pagina del comando pertinente nella AWS CLI version 2 reference guide.
> Vuoi contribuire? Contribuisci con esempi di comandi della AWS CLI nel Repository di esempi di codice AWS su GitHub. Per ulteriori informazioni su come contribuire, consulta i passaggi rapidi relativi ai contributi di esempio del codice della AWS CLI nelle pagine GitHub.

Esempi di comandi guidati per la AWS CLI

AWS Command Line Interface (AWS CLI) è uno strumento open source che consente di interagire con un'ampia gamma di Servizi AWS utilizzando i comandi nella shell a riga di comando. Questa sezione fornisce esempi guidati che mostrano come sfruttare la AWS CLI per accedere ad alcuni dei Servizi AWS. Ciò include alcuni comandi della AWS CLI personalizzati, come i comandi `aws s3` di alto livello. Questi esempi di comandi illustrano le azioni comuni utilizzate per alcuni Servizi AWS e forniscono risorse aggiuntive per ulteriori informazioni.

Per utenti AWS esperti o alle prime armi con la AWS CLI, questi esempi guidati rappresentano una risorsa per semplificare le operazioni AWS.

Per un riferimento completo a tutti i comandi disponibili per ciascun Servizio AWS, consulta la AWS CLI version 2 reference guide. Inoltre, puoi utilizzare la guida integrata nella riga di comando per esplorare l'array di Servizi AWS, i comandi, le opzioni e le funzionalità della AWS CLI.

Per altri esempi di comandi non disponibili in questa sezione, consulta la sezione AWS CLI esempi di comandi. Questi sono esempi di comandi open source disponibili anche nella AWS CLI version 2 reference guide. Gli esempi dei comandi sono ospitati nel repository della AWS CLI su GitHub.

Per gli esempi di script Bash open source, consulta the section called "Esempi di script Bash". Gli esempi di script Bash sono ospitati nel repository degli esempi di codice AWS su GitHub.

Servizi

- Utilizzo di Amazon DynamoDB nella AWS CLI
- Utilizzo di Amazon EC2 nella AWS CLI
- Utilizzo di Amazon Glacier in AWS CLI
- Utilizzo di IAM nella AWS CLI
- Utilizzo di Amazon S3 nella AWS CLI
- Accedere ad Amazon SNS in AWS CLI

Utilizzo di Amazon DynamoDB nella AWS CLI

AWS Command Line Interface (AWS CLI) fornisce supporto per tutti i servizi di database AWS, tra cui Amazon DynamoDB. Puoi utilizzare la AWS CLI per operazioni ad hoc, quali la creazione di tabelle. È possibile utilizzarla anche per incorporare operazioni DynamoDB all'interno di script di utilità.

Per ulteriori informazioni sull'utilizzo della AWS CLI con DynamoDB, consulta `dynamodb` in AWS CLI Command Reference.

Per elencare i comandi della AWS CLI per DynamoDB, utilizza il comando seguente.

```
$ aws dynamodb help
```

Argomenti

- Prerequisiti
- Creazione e utilizzo di tabelle DynamoDB
- Utilizzo di DynamoDB locale
- Risorse

Prerequisiti

Per eseguire i comandi `dynamodb`, è necessario:

- Istalla e configura la AWS CLI. Per ulteriori informazioni, consultare the section called "Installazione o aggiornamento" e Credenziali di autenticazione e accesso per la AWS CLI.

Creazione e utilizzo di tabelle DynamoDB

Il formato a riga di comando consiste in un nome di comando DynamoDB seguito dai parametri per tale comando. L'AWS CLI supporta la sintassi abbreviata della CLI per i valori dei parametri, oltre a JSON completo.

L'esempio seguente crea una tabella denominata `MusicCollection`.

```
$ aws dynamodb create-table \
    --table-name MusicCollection \
    --attribute-definitions AttributeName=Artist,AttributeType=S
 AttributeName=SongTitle,AttributeType=S \
    --key-schema AttributeName=Artist,KeyType=HASH
 AttributeName=SongTitle,KeyType=RANGE \
    --provisioned-throughput ReadCapacityUnits=1,WriteCapacityUnits=1
```

Puoi aggiungere nuove righe alla tabella con comandi simili a quelli mostrati nell'esempio seguente. In questi esempi viene utilizzata una combinazione di sintassi abbreviata e JSON.

```
$ aws dynamodb put-item \
    --table-name MusicCollection \
    --item '{
        "Artist": {"S": "No One You Know"},
        "SongTitle": {"S": "Call Me Today"} ,
        "AlbumTitle": {"S": "Somewhat Famous"}
      }' \
    --return-consumed-capacity TOTAL
```

```
{
    "ConsumedCapacity": {
        "CapacityUnits": 1.0,
        "TableName": "MusicCollection"
    }
}
```

```
$ aws dynamodb put-item \
    --table-name MusicCollection \
    --item '{
        "Artist": {"S": "Acme Band"},
        "SongTitle": {"S": "Happy Day"} ,
        "AlbumTitle": {"S": "Songs About Life"}
      }' \
    --return-consumed-capacity TOTAL
{
    "ConsumedCapacity": {
        "CapacityUnits": 1.0,
        "TableName": "MusicCollection"
    }
}
```

Può essere difficile comporre un JSON valido in un comando su una sola riga. Per semplificare l'operazione, l'AWS CLI può leggere i file JSON. Considera ad esempio il seguente frammento JSON, che viene memorizzato in un file denominato `expression-attributes.json`.

```
{
  ":v1": {"S": "No One You Know"},
  ":v2": {"S": "Call Me Today"}
}
```

Puoi utilizzare questo file per inviare una richiesta `query` con l'AWS CLI. Nell'esempio seguente, il contenuto del file `expression-attributes.json` viene utilizzato come valore del parametro `--expression-attribute-values`.

```
$ aws dynamodb query --table-name MusicCollection \
    --key-condition-expression "Artist = :v1 AND SongTitle = :v2" \
    --expression-attribute-values file://expression-attributes.json
{
    "Count": 1,
    "Items": [
```

```
        {
            "AlbumTitle": {
                "S": "Somewhat Famous"
            },
            "SongTitle": {
                "S": "Call Me Today"
            },
            "Artist": {
                "S": "No One You Know"
            }
        }
    ],
    "ScannedCount": 1,
    "ConsumedCapacity": null
}
```

Utilizzo di DynamoDB locale

Oltre che con DynamoDB, puoi utilizzare la AWS CLI con DynamoDB locale. DynamoDB locale è un piccolo database e server lato client che imita il servizio DynamoDB. DynamoDB locale consente di scrivere applicazioni che utilizzano l'API DynamoDB senza effettivamente manipolare tabelle o dati nel servizio web DynamoDB. Tutte le operazioni API vengono invece reinstradate a un database locale. In questo modo è possibile salvare il throughput assegnato, lo storage dei dati e le tariffe per il trasferimento dei dati.

Per ulteriori informazioni su DynamoDB locale e su come utilizzarlo con la AWS CLI, consulta le seguenti sezioni della Guida per sviluppatori di Amazon DynamoDB:

- DynamoDB Local
- Utilizzo della AWS CLI con DynamoDB locale

Risorse

Riferimento alla AWS CLI:

- `aws dynamodb`
- `aws dynamodb create-table`
- `aws dynamodb put-item`
- `aws dynamodb query`

Riferimento al servizio:

- DynamoDB locale nella Guida per sviluppatori di Amazon DynamoDB
- Utilizzo della AWS CLI con DynamoDB locale nella Guida per sviluppatori di Amazon DynamoDB

Utilizzo di Amazon EC2 nella AWS CLI

Un'introduzione ad Amazon Elastic Compute Cloud
Introduzione a Amazon EC2 - Server Elastic Cloud e hosting con AWS

Amazon Elastic Compute Cloud (Amazon EC2) fornisce ambienti di elaborazione virtuale altamente scalabili e flessibili. Amazon EC2 consente di effettuare il provisioning e di gestire server virtuali, noti come istanze Amazon EC2, per soddisfare un'ampia gamma di esigenze di elaborazione.

Le istanze di Amazon EC2 sono macchine virtuali che possono essere personalizzate con varie configurazioni di CPU, memoria, archiviazione e capacità di rete. Puoi scegliere tra una vasta gamma di tipi di istanze, che spaziano da opzioni leggere ed economiche a istanze potenti e ad alte prestazioni, a seconda dei requisiti dell'applicazione. Questa flessibilità consente di soddisfare le esigenze di elaborazione per ottimizzare le prestazioni e l'efficienza dei costi.

Inoltre, Amazon EC2 offre una suite di funzionalità che consente di gestire le risorse di elaborazione in modo efficace. Queste risorse includono la possibilità di avviare rapidamente nuove istanze, creare immagini di macchine (AMI) personalizzate per una rapida implementazione e aumentare o ridurre la capacità di calcolo in base alle esigenze.

Puoi accedere alle funzionalità di Amazon EC2 utilizzando AWS Command Line Interface (AWS CLI). Per elencare i comandi della AWS CLI per Amazon EC2, usa il comando seguente.

```
aws ec2 help
```

Prima di eseguire qualsiasi comando, imposta le tue credenziali di default. Per ulteriori informazioni, consulta Configurazione delle impostazioni della AWS CLI..

Questo argomento mostra esempi di comandi della AWS CLI che eseguono operazioni comuni per Amazon EC2.

Per esempi di comandi della AWS CLI in formato lungo, consulta il repository degli esempi di codice della AWS CLI su GitHub.

Argomenti

- Creazione, visualizzazione ed eliminazione di coppie di chiavi Amazon EC2 nella AWS CLI
- Creazione, configurazione ed eliminazione di gruppi di sicurezza Amazon EC2 nella AWS CLI
- Avvio, elenco ed eliminazione di EC2 istanze Amazon nel AWS CLI
- Modificare un tipo di istanza Amazon EC2 con uno script Bash nella AWS CLI

Creazione, visualizzazione ed eliminazione di coppie di chiavi Amazon EC2 nella AWS CLI

Puoi utilizzare AWS Command Line Interface (AWS CLI) per creare, visualizzare ed eliminare le coppie di chiavi per Amazon Elastic Compute Cloud (Amazon EC2). Puoi utilizzare le coppie di chiavi per connetterti a un'istanza Amazon EC2. Puoi fornire la coppia di chiavi a EC2 quando crei l'istanza, quindi devi utilizzare questa coppia di chiavi per autenticarti quando ti connetti all'istanza.

> **Note**
>
> Per ulteriori esempi di comandi, consulta la AWS CLI reference guide.

Argomenti

- Prerequisiti
- Creazione di una coppia di chiavi
- Visualizzazione della coppia di chiavi
- Eliminazione della coppia di chiavi
- Riferimenti

Prerequisiti

Per eseguire i comandi `ec2`, è necessario:

- Istalla e configura la AWS CLI. Per ulteriori informazioni, consultare the section called "Installazione o aggiornamento" e Credenziali di autenticazione e accesso per la AWS CLI.

- Impostare le autorizzazioni IAM per consentire l'accesso ad Amazon EC2. Per ulteriori informazioni sulle autorizzazioni IAM per Amazon EC2, consulta Policy IAM per Amazon EC2 nella Guida per l'utente di Amazon EC2.

Creazione di una coppia di chiavi

Per creare una coppia di chiavi, utilizza il comando `aws ec2 create-key-pair` con l'opzione `--query` e l'opzione `--output text` per indirizzare la chiave privata direttamente in un file.

```
$ aws ec2 create-key-pair --key-name MyKeyPair --query 'KeyMaterial' --output text
 > MyKeyPair.pem
```

Per PowerShell, l'impostazione di default del reindirizzamento di `> file` è la codifica UTF-8, che non può essere usata con alcuni client SSH. Pertanto, devi convertire l'output reindirizzandolo al comando `out-file` e impostando esplicitamente la codifica su `ascii`.

```
PS C:\>aws ec2 create-key-pair --key-name MyKeyPair --query 'KeyMaterial' --output text
 | out-file -encoding ascii -filepath MyKeyPair.pem
```

L'aspetto del file `MyKeyPair.pem` risultante è simile a quello riportato di seguito.

```
-----BEGIN RSA PRIVATE KEY-----
EXAMPLEKEYKCAQEAy7WZhaDsrA1W3mRlQtvhwyORRX8gnxgDAfRt/gx42kWXsT4rXE/b5CpSgie/
vBoU7jLxx92pNHoFnByP+Dc21eyyz6CvjTmWA0JwfWiW5/akH7iO5dSrvC7dQkW2duV5QuUdE0QW
Z/aNxMniGQE6XAgfwlnXVBwrerrQo+ZWQeqiUwwMkuEbLeJFLhMCvYURpUMSC1oehm449ilx9X1F
G50TCFeOzfl8dqqCP6GzbPaIjiU19xX/azOR9V+tpUOzEL+wmXnZt3/nHPQ5xvD2OJH67km6SuPW
oPzev/D8V+x4+bHthfSjR9Y7DvQFjfBVwHXigBdtZcU2/wei8D/HYwIDAQABAoIBAGZ1kaEvnrqu
/uler7vgIn5m7lN5LKw4hJLAIW6tUT/fzvtcHK0SkbQCQXuriHmQ2MQyJX/0kn2NfjLV/ufGxbL1
mb5qwMGUnEpJaZD6QSSs3kICLwWUYUiGfc0uiSbmJoap/GTLU0W5Mfcv36PaBUNy5p53V6G7hXb2
bahyWyJNfjLe4M86yd2YK3V2CmK+X/BOsShnJ36+hjrXPPWmV3N9zEmCdJjA+K15DYmhm/tJWSD9
81oGk9TopEp7CkIfatEATyyZiVqoRq6k64iuM9JkA3OzdXzMQexXVJ1TLZVEH0E7bhlY9d8O1ozR
oQs/FiZNAx2iijCWyv0lpjE73+kCgYEA9mZtyhkHkFDpwrSM1APaL8oNAbbjwEy7Z5Mqfql+lIp1
YkriL0DbLXlvRAH+yHPRit2hHOjtUNZh4Axv+cpg09qbUI3+43eEy24B7G/Uh+GTfbjsXsOxQx/x
p9otyVwc7hsQ5TA5PZb+mvkJ5OBEKzet9XcKwONBYELGhnEPe7cCgYEA06Vgov6YHleHui9kHuws
ayav0elc5zkxjF9nfHFJRry21R1trw2Vdpn+9g481URrpzWVOEihvm+xTtmaZlSp//lkq75XDwnU
WA8gkn6O3QE3fq2yN98BURsAKdJfJ5RL1HvGQvTe10HLYYXpJnEkHv+Unl2ajLivWUt5pbBrKbUC
gYBjbO+OZk0sCcpZ29sbzjYjpIddErySIyRX5gV2uNQwAjLdp9PfN295yQ+BxMBXiIycWVQiw0bH
oMo7yykABY7Ozd5wQewBQ4AdSlWSX4nGDtsiFxWiI5sKuAAeOCbTosy1s8w8fxoJ5Tz1sdoxNeGs
Arq6Wv/G16zQuAE9zK9vvwKBgF+09VI/1wJBirsDGz9whVWfFPrTkJNvJZzYt69qezxlsjgFKshy
WBhd4xHZtmCqpBPlAymEjr/TOlbxyARmXMnIOWIAnNXMGB4KGSyl1mzSVAoQ+fqR+cJ3d0dyPl1j
```

```
jjb0Ed/NY8frlNDxAVHE8BSkdsx2f6ELEyBKJSRr9snRAoGAMrTwYneXzvTskF/S5Fyu0iOegLDa
NWUH38v/nDCgEpIXD5Hn3qAEcju1IjmbwlvtW+nY2jVhv7UGd8MjwUTNGItdb6nsYqM2asrnF3qS
VRkAKKKYeGjkpUfVTrW0YFjXkfcrR/V+QFL5OndHAKJXjW7a4ejJLncTzmZSpYzwApc=
-----END RSA PRIVATE KEY-----
```

La chiave privata non è archiviata in AWS e può essere recuperata solo quando viene creata. Non puoi recuperarla successivamente. Se perdi la chiave privata, invece, devi creare una nuova coppia di chiavi.

Se ti connetti all'istanza da un computer Linux, consigliamo di utilizzare il comando seguente per impostare le autorizzazioni del file della chiave privata in modo da essere l'unico a poterlo leggere.

```
$ chmod 400 MyKeyPair.pem
```

Visualizzazione della coppia di chiavi

Dalla coppia di chiavi viene generata un'impronta che può essere utilizzata per verificare che la chiave privata che si trova sul tuo computer locale corrisponda alla chiave pubblica archiviata in AWS.

L'impronta è un hash SHA1 preso da una copia codificata DER della chiave privata. Questo valore viene acquisito quando la coppia di chiavi viene creata e archiviata in AWS con la chiave pubblica. Puoi visualizzare l'impronta nella console Amazon EC2 o eseguendo il comando `aws ec2 describe-key-pairs` della AWS CLI.

L'esempio seguente mostra l'impronta per `MyKeyPair`.

```
$ aws ec2 describe-key-pairs --key-name MyKeyPair
{
    "KeyPairs": [
        {
            "KeyName": "MyKeyPair",
            "KeyFingerprint":
 "1f:51:ae:28:bf:89:e9:d8:1f:25:5d:37:2d:7d:b8:ca:9f:f5:f1:6f"
        }
    ]
}
```

Per ulteriori informazioni sulle chiavi e sulle impronte, consulta Coppie di chiavi Amazon EC2 nella Guida per l'utente di Amazon EC2.

Eliminazione della coppia di chiavi

Per eliminare una coppia di chiavi, esegui il comando `aws ec2 delete-key-pair`, sostituendo *MyKeyPair* con il nome della coppia da eliminare.

```
$ aws ec2 delete-key-pair --key-name MyKeyPair
```

Riferimenti

Riferimento alla AWS CLI:

- `aws ec2`
- `aws ec2 create-key-pair`
- `aws ec2 delete-key-pair`
- `aws ec2 describe-key-pairs`

Altri riferimenti:

- Amazon Elastic Compute Cloud Documentation
- Per visualizzare AWS SDK e contribuire, e per gli esempi di codice della AWS CLI, consulta AWS Code Examples Repository su GitHub.

Creazione, configurazione ed eliminazione di gruppi di sicurezza Amazon EC2 nella AWS CLI

Puoi creare un gruppo di sicurezza per le istanze Amazon Elastic Compute Cloud (Amazon EC2) che essenzialmente funziona come un firewall, con regole che determinano il traffico di rete a cui è consentito l'ingresso e l'uscita.

Utilizza AWS Command Line Interface (AWS CLI) per creare un nuovo gruppo di sicurezza, aggiungere regole per i gruppi di sicurezza esistenti ed eliminare i gruppi di sicurezza.

> **Note**
>
> Per ulteriori esempi di comandi, consulta la AWS CLI reference guide.

Argomenti

- Prerequisiti
- Creazione di un gruppo di sicurezza
- Aggiunta di regole al gruppo di sicurezza
- Eliminazione del gruppo di sicurezza:
- Riferimenti

Prerequisiti

Per eseguire i comandi `ec2`, è necessario:

- Istalla e configura la AWS CLI. Per ulteriori informazioni, consultare the section called "Installazione o aggiornamento" e Credenziali di autenticazione e accesso per la AWS CLI.
- Impostare le autorizzazioni IAM per consentire l'accesso ad Amazon EC2. Per ulteriori informazioni sulle autorizzazioni IAM per Amazon EC2, consulta Policy IAM per Amazon EC2 nella Guida per l'utente di Amazon EC2.

Creazione di un gruppo di sicurezza

Puoi creare gruppi di sicurezza associati a cloud privati virtuali (VPC).

Nell'esempio seguente `aws ec2 create-security-group` mostra come creare un gruppo di sicurezza per un VPC specificato.

```
$ aws ec2 create-security-group --group-name my-sg --description "My security group" --
vpc-id vpc-1a2b3c4d
{
    "GroupId": "sg-903004f8"
}
```

Per visualizzare le informazioni iniziali per un gruppo di sicurezza, esegui il comando `aws ec2 describe-security-groups`. Puoi fare riferimento a un gruppo di sicurezza EC2-VPC solo utilizzando il relativo `vpc-id`, non il nome.

```
$ aws ec2 describe-security-groups --group-ids sg-903004f8
{
    "SecurityGroups": [
        {
            "IpPermissionsEgress": [
```

```
                {
                    "IpProtocol": "-1",
                    "IpRanges": [
                        {
                            "CidrIp": "0.0.0.0/0"
                        }
                    ],
                    "UserIdGroupPairs": []
                }
            ],
            "Description": "My security group"
            "IpPermissions": [],
            "GroupName": "my-sg",
            "VpcId": "vpc-1a2b3c4d",
            "OwnerId": "123456789012",
            "GroupId": "sg-903004f8"
        }
    ]
}
```

Aggiunta di regole al gruppo di sicurezza

Quando esegui un'istanza Amazon EC2, devi abilitare le regole nel gruppo di sicurezza per consentire il traffico di rete in entrata per il tuo strumento di connessione all'immagine.

Ad esempio, se avvii un'istanza di Windows, in genere devi aggiungere una regola per consentire il traffico in entrata sulla porta TCP 3389 per supportare Remote Desktop Protocol (RDP). Se avvii un'istanza di Linux, in genere devi aggiungere una regola per consentire il traffico in entrata sulla porta TCP 22 per supportare le connessioni SSH.

Utilizza il comando `aws ec2 authorize-security-group-ingress` per aggiungere una regola al gruppo di sicurezza. Un parametro obbligatorio di questo comando è l'indirizzo IP pubblico del computer o la rete (sotto forma di intervallo di indirizzi) a cui è collegato il computer, in notazione CIDR.

> **Note**
>
> Forniamo il servizio seguente, https://checkip.global.api.aws/, per consentire di determinare l'indirizzo IP pubblico. Per trovare altri servizi che possono aiutarti a identificare il tuo indirizzo IP, utilizza il browser per cercare "qual è il indirizzo IP". Se ti connetti tramite un ISP o da dietro un firewall utilizzando un indirizzo IP dinamico (tramite un gateway NAT da una rete

privata), l'indirizzo può cambiare periodicamente. In questo caso, devi scoprire l'intervallo di indirizzi IP utilizzato dai computer client.

L'esempio seguente mostra come aggiungere una regola per il protocollo RDP (porta TCP 3389) a un gruppo di sicurezza EC2-VPC con ID `sg-903004f8` utilizzando il tuo indirizzo IP.

Per iniziare, trova il tuo indirizzo IP.

```
$ curl https://checkip.amazonaws.com
x.x.x.x
```

Puoi quindi aggiungere l'indirizzo IP al tuo gruppo di sicurezza eseguendo il comando [aws ec2 authorize-security-group-ingress](#).

```
$ aws ec2 authorize-security-group-ingress --group-id sg-903004f8 --protocol tcp --port
 3389 --cidr x.x.x.x/x
```

Il comando seguente aggiunge un'altra regola per abilitare l'autenticazione SSH nelle istanze dello stesso gruppo di sicurezza.

```
$ aws ec2 authorize-security-group-ingress --group-id sg-903004f8 --protocol tcp --port
 22 --cidr x.x.x.x/x
```

Per visualizzare le modifiche al gruppo di sicurezza, esegui il comando [aws ec2 describe-security-groups](#).

```
$ aws ec2 describe-security-groups --group-ids sg-903004f8
{
    "SecurityGroups": [
        {
            "IpPermissionsEgress": [
                {
                    "IpProtocol": "-1",
                    "IpRanges": [
                        {
                            "CidrIp": "0.0.0.0/0"
                        }
                    ],
                    "UserIdGroupPairs": []
```

```
                }
            ],
            "Description": "My security group"
            "IpPermissions": [
                {
                    "ToPort": 22,
                    "IpProtocol": "tcp",
                    "IpRanges": [
                        {
                            "CidrIp": "x.x.x.x/x"
                        }
                    ]
                    "UserIdGroupPairs": [],
                    "FromPort": 22
                }
            ],
            "GroupName": "my-sg",
            "OwnerId": "123456789012",
            "GroupId": "sg-903004f8"
        }
    ]
}
```

Eliminazione del gruppo di sicurezza:

Per eliminare un gruppo di sicurezza, utilizza il comando `aws ec2 delete-security-group`.

> **Note**
>
> Non puoi eliminare un gruppo di sicurezza se è attualmente collegato a un ambiente.

Il comando di esempio seguente elimina un gruppo di sicurezza EC2-VPC.

```
$ aws ec2 delete-security-group --group-id sg-903004f8
```

Riferimenti

Riferimento alla AWS CLI:

- `aws ec2`
- `aws ec2 authorize-security-group-ingress`

- `aws ec2 create-security-group`
- `aws ec2 delete-security-group`
- `aws ec2 describe-security-groups`

Altri riferimenti:

- Amazon Elastic Compute Cloud Documentation
- Per visualizzare AWS SDK e contribuire, e per gli esempi di codice della AWS CLI, consulta AWS Code Examples Repository su GitHub.

Avvio, elenco ed eliminazione di EC2 istanze Amazon nel AWS CLI

Puoi utilizzare AWS Command Line Interface (AWS CLI) per avviare, elencare ed eliminare istanze di Amazon Elastic Compute Cloud (Amazon EC2). Se avvii un'istanza che non rientra nel piano AWS gratuito, ti verrà fatturata una volta avviata l'istanza e ti verrà addebitato il tempo in cui l'istanza è in esecuzione, anche se rimane inattiva.

> **Note**
>
> Per ulteriori esempi di comandi, consulta la riferimento.

Argomenti

- Prerequisiti
- Lancio dell'istanza
- Aggiunta di un dispositivo a blocchi all'istanza
- Aggiunta di un tag all'istanza
- Connessione all'istanza
- Elencazione delle istanze
- Eliminare l'istanza
- Riferimenti

Prerequisiti

Per eseguire i comandi della `ec2` descritti in questo argomento, è necessario:

- Istalla e configura la AWS CLI. Per ulteriori informazioni, consultare the section called "Installazione o aggiornamento" e Credenziali di autenticazione e accesso per la AWS CLI.
- Imposta le tue autorizzazioni IAM per consentire l' EC2 accesso ad Amazon. Per ulteriori informazioni sulle autorizzazioni IAM per Amazon EC2, consulta le politiche IAM per Amazon EC2 nella Amazon EC2 User Guide.
- Creare una coppia di chiavi e un gruppo di sicurezza.
- Seleziona un'Amazon Machine Image (AMI) e prendi nota dell'ID AMI. Per ulteriori informazioni, consulta la sezione Ricerca di un'AMI adatta nella Amazon EC2 User Guide.

Lancio dell'istanza

Per avviare un' EC2 istanza Amazon utilizzando l'AMI selezionata, usa il `aws ec2 run-instances` comando. Puoi avviare l'istanza in un cloud privato virtuale (VPC).

Inizialmente l'istanza viene visualizzata in stato `pending`, ma dopo pochi minuti passa allo stato `running`.

L'esempio seguente mostra come avviare un'istanza `t2.micro` nella sottorete specificata di un VPC. Sostituisci i valori del parametro *`italicized`* con i tuoi valori.

```
$ aws ec2 run-instances --image-id ami-xxxxxxxx --count 1 --instance-type t2.micro --
key-name MyKeyPair --security-group-ids sg-903004f8 --subnet-id subnet-6e7f829e
{
    "OwnerId": "123456789012",
    "ReservationId": "r-5875ca20",
    "Groups": [
        {
            "GroupName": "my-sg",
            "GroupId": "sg-903004f8"
        }
    ],
    "Instances": [
        {
            "Monitoring": {
                "State": "disabled"
            },
            "PublicDnsName": null,
            "Platform": "windows",
            "State": {
                "Code": 0,
```

```
            "Name": "pending"
        },
        "EbsOptimized": false,
        "LaunchTime": "2013-07-19T02:42:39.000Z",
        "PrivateIpAddress": "10.0.1.114",
        "ProductCodes": [],
        "VpcId": "vpc-1a2b3c4d",
        "InstanceId": "i-5203422c",
        "ImageId": "ami-173d747e",
        "PrivateDnsName": "ip-10-0-1-114.ec2.internal",
        "KeyName": "MyKeyPair",
        "SecurityGroups": [
            {
                "GroupName": "my-sg",
                "GroupId": "sg-903004f8"
            }
        ],
        "ClientToken": null,
        "SubnetId": "subnet-6e7f829e",
        "InstanceType": "t2.micro",
        "NetworkInterfaces": [
            {
                "Status": "in-use",
                "SourceDestCheck": true,
                "VpcId": "vpc-1a2b3c4d",
                "Description": "Primary network interface",
                "NetworkInterfaceId": "eni-a7edb1c9",
                "PrivateIpAddresses": [
                    {
                        "PrivateDnsName": "ip-10-0-1-114.ec2.internal",
                        "Primary": true,
                        "PrivateIpAddress": "10.0.1.114"
                    }
                ],
                "PrivateDnsName": "ip-10-0-1-114.ec2.internal",
                "Attachment": {
                    "Status": "attached",
                    "DeviceIndex": 0,
                    "DeleteOnTermination": true,
                    "AttachmentId": "eni-attach-52193138",
                    "AttachTime": "2013-07-19T02:42:39.000Z"
                },
                "Groups": [
                    {
```

```
                        "GroupName": "my-sg",
                        "GroupId": "sg-903004f8"
                    }
                ],
                "SubnetId": "subnet-6e7f829e",
                "OwnerId": "123456789012",
                "PrivateIpAddress": "10.0.1.114"
            }
        ],
        "SourceDestCheck": true,
        "Placement": {
            "Tenancy": "default",
            "GroupName": null,
            "AvailabilityZone": "us-west-2b"
        },
        "Hypervisor": "xen",
        "BlockDeviceMappings": [
            {
                "DeviceName": "/dev/sda1",
                "Ebs": {
                    "Status": "attached",
                    "DeleteOnTermination": true,
                    "VolumeId": "vol-877166c8",
                    "AttachTime": "2013-07-19T02:42:39.000Z"
                }
            }
        ],
        "Architecture": "x86_64",
        "StateReason": {
            "Message": "pending",
            "Code": "pending"
        },
        "RootDeviceName": "/dev/sda1",
        "VirtualizationType": "hvm",
        "RootDeviceType": "ebs",
        "Tags": [
            {
                "Value": "MyInstance",
                "Key": "Name"
            }
        ],
        "AmiLaunchIndex": 0
    }
]
```

```
}
```

Aggiunta di un dispositivo a blocchi all'istanza

A ogni istanza avviata è associato un volume dispositivo root. Puoi utilizzare una mappatura dei dispositivi a blocchi per specificare volumi di Amazon Elastic Block Store (Amazon EBS) aggiuntivi o i volumi di archivio dell'istanza da collegare a un'istanza quando viene avviata.

Per aggiungere un dispositivo a blocchi all'istanza, specifica l'opzione `--block-device-mappings` quando utilizzi `run-instances`.

Il parametro di esempio seguente esegue il provisioning di un volume Amazon EBS standard di 20 GB e lo mappa all'istanza utilizzando l'identificatore `/dev/sdf`.

```
--block-device-mappings "[{\"DeviceName\":\"/dev/sdf\",\"Ebs\":{\"VolumeSize\":20,
\"DeleteOnTermination\":false}}]"
```

L'esempio seguente aggiunge un volume Amazon EBS, mappato a `/dev/sdf`, basato su uno snapshot esistente. Una snapshot rappresenta un'immagine caricata sul volume per te. Quando specifichi una snapshot, non devi specificare una dimensione del volume perché sarà sufficiente a contenere la tua immagine. Tuttavia, se specifichi la dimensione, questa deve essere maggiore o uguale a quella della snapshot.

```
--block-device-mappings "[{\"DeviceName\":\"/dev/sdf\",\"Ebs\":{\"SnapshotId\":\"snap-
a1b2c3d4\"}}]"
```

L'esempio seguente aggiunge due volumi all'istanza. Il numero di volumi disponibili per l'istanza dipende dal tipo di istanza.

```
--block-device-mappings "[{\"DeviceName\":\"/dev/sdf\",\"VirtualName\":\"ephemeral0\"},
{\"DeviceName\":\"/dev/sdg\",\"VirtualName\":\"ephemeral1\"}]"
```

L'esempio seguente crea la mappatura (`/dev/sdj`), ma non esegue il provisioning di un volume per l'istanza.

```
--block-device-mappings "[{\"DeviceName\":\"/dev/sdj\",\"NoDevice\":\"\"}]"
```

Per ulteriori informazioni, consulta Block Device Mapping nella Amazon EC2 User Guide.

Aggiunta di un tag all'istanza

Un tag è un'etichetta che assegni a una AWS risorsa. Questo consente di aggiungere metadati alle risorse, che possono essere utilizzati per vari scopi. Per ulteriori informazioni, consulta Tagging Your Resources nella Amazon EC2 User Guide.

L'esempio seguente mostra come aggiungere un tag con il nome della chiave "`Name`" e il valore "`MyInstance`" all'istanza specificata, utilizzando il comando `aws ec2 create-tags`.

```
$ aws ec2 create-tags --resources i-5203422c --tags Key=Name,Value=MyInstance
```

Connessione all'istanza

Mentre l'istanza è in esecuzione, puoi connetterti e utilizzarla esattamente come se si trovasse nel tuo computer. Per ulteriori informazioni, consulta Connect to Your Amazon EC2 Instance nella Amazon EC2 User Guide.

Elencazione delle istanze

Puoi usare il AWS CLI per elencare le tue istanze e visualizzare le informazioni su di esse. Puoi creare un elenco di tutte le istanze oppure filtrare i risultati in base alle istanze che ti interessano.

Negli esempi seguenti viene illustrato come utilizzare il comando `aws ec2 describe-instances`.

Il comando seguente elenca tutte le tue istanze.

```
$ aws ec2 describe-instances
```

Il comando seguente filtra l'elenco visualizzando solo le istanze `t2.micro` e restituisce solo i valori `InstanceId` per ogni corrispondenza.

```
$ aws ec2 describe-instances --filters "Name=instance-type,Values=t2.micro" --query
 "Reservations[].Instances[].InstanceId"
[
    "i-05e998023d9c69f9a"
]
```

Il comando seguente elenca qualsiasi istanza con il tag `Name=MyInstance`.

```
$ aws ec2 describe-instances --filters "Name=tag:Name,Values=MyInstance"
```

Il comando seguente elenca le istanze che sono state avviate utilizzando uno dei seguenti comandi AMIs:`ami-x0123456`,`ami-y0123456`, e. `ami-z0123456`

```
$ aws ec2 describe-instances --filters "Name=image-id,Values=ami-x0123456,ami-
y0123456,ami-z0123456"
```

Eliminare l'istanza

Puoi usare il AWS CLI per terminare (eliminare) un' EC2 istanza Amazon quando non ne hai più bisogno.

> **Important**
>
> La terminazione di un'istanza è permanente e irreversibile.
> Dopo aver terminato un'istanza, non puoi più connetterti a essa e non puoi ripristinarla. Inoltre, tutti i volumi Amazon EBS collegati e configurati per essere eliminati in caso di terminazione vengono eliminati in modo definitivo e non possono essere recuperati. Tutti i dati archiviati nei volumi di archivio dell'istanza vengono persi in modo definitivo. Per ulteriori informazioni, consulta Come funziona la chiusura dell'istanza.
> Prima di terminare un'istanza, assicurati di aver eseguito il backup di tutti i dati che devi mantenere dopo la terminazione sull'archivio persistente.

Appena lo stato dell'istanza cambia in `shutting-down` o `terminated`, vengono bloccati i rispettivi addebiti. Per riconnetterti a un'istanza in un secondo momento, utilizza stop-instances invece di `terminate-instances`. Per ulteriori informazioni, consulta Terminate Your Instance nella Amazon EC2 User Guide.

L'esempio seguente mostra come eliminare un'istanza utilizzando il `aws ec2 terminate-instances` comando.

```
$ aws ec2 terminate-instances --instance-ids i-5203422c
{
    "TerminatingInstances": [
        {
            "InstanceId": "i-5203422c",
            "CurrentState": {
                "Code": 32,
                "Name": "shutting-down"
```

```
            },
            "PreviousState": {
                "Code": 16,
                "Name": "running"
            }
        }
    ]
}
```

Riferimenti

AWS CLI riferimento:

- `aws ec2`
- `aws ec2 create-tags`
- `aws ec2 describe-instances`
- `aws ec2 run-instances`
- `aws ec2 terminate-instances`

Altri riferimenti:

- Amazon Elastic Compute Cloud Documentation
- Per visualizzare e contribuire all' AWS SDK e agli esempi di AWS CLI codice, consulta il AWS Code Examples Repository su. GitHub

Modificare un tipo di istanza Amazon EC2 con uno script Bash nella AWS CLI

Questo esempio di script Bash per Amazon EC2 modifica il tipo di istanza per un'istanza Amazon EC2 utilizzando AWS Command Line Interface (AWS CLI). Arresta l'istanza se è in esecuzione, cambia il tipo di istanza e quindi, se richiesto, riavvia l'istanza. Gli script di shell sono programmi progettati per l'esecuzione in un'interfaccia a riga di comando.

> **Note**
>
> Per ulteriori esempi di comandi, consulta la AWS CLI reference guide.

Argomenti

- Prima di iniziare
- Informazioni sull'esempio
- Parametri
- File
- Riferimenti

Prima di iniziare

Prima di poter eseguire uno qualsiasi degli esempi seguenti, è necessario completare le seguenti operazioni.

- Istalla e configura la AWS CLI. Per ulteriori informazioni, consultare the section called "Installazione o aggiornamento" e Credenziali di autenticazione e accesso per la AWS CLI.
- Il profilo utilizzato deve disporre delle autorizzazioni che consentano alla AWS di eseguire le operazioni degli esempi.
- Un'istanza Amazon EC2 in esecuzione nell'account per il quale l'utente è autorizzato a interrompere e modificare. Se esegui lo script di test, avvia automaticamente un'istanza, verifica la modifica del tipo e quindi termina l'istanza.
- Come best practice di AWS, concedi a questo codice il privilegio minimo o solo le autorizzazioni richieste per eseguire un processo. Per ulteriori informazioni sui ruoli IAM, consulta Assegnare il privilegio minimo nella Guida per l'utente di AWS Identity and Access Management.
- Questo codice non è stato testato in tutte le Regioni AWS. Alcuni servizi AWS sono disponibili solo in Regioni specifiche. Per ulteriori informazioni, consulta Service Endpoints and Quotas nella AWS General Reference Guide.
- L'esecuzione di questo codice potrebbe comportare l'addebito di costi sul tuo account AWS. È tua responsabilità assicurarti che tutte le risorse create da questo script vengano rimosse quando hai finito di utilizzarle.

Informazioni sull'esempio

Questo esempio è scritto come una funzione nel file di script della shell `change_ec2_instance_type.sh` che è possibile `source` da un altro script o dalla riga di comando. Ogni file di script contiene commenti che descrivono ciascuna delle funzioni. Una volta che la funzione è in memoria, è possibile invocarla dalla riga di comando. Ad esempio, i seguenti comandi modificano il tipo dell'istanza specificata in `t2.nano`:

```
$ source ./change_ec2_instance_type.sh
$ ./change_ec2_instance_type -i *instance-id* -t new-type
```

Per l'esempio completo e i file di script scaricabili, consulta Change Amazon EC2 Instance Type in AWS Code Examples Repository su GitHub.

Parametri

-i - (stringa) Specifica l'ID dell'istanza da modificare.

-t - (stringa) Specifica il tipo di istanza Amazon EC2 a cui passare.

-r - (switch) Per impostazione predefinita, non è impostato. Se `-r` è impostato, riavvia l'istanza dopo il cambio di tipo.

-f - (switch) Per impostazione predefinita, lo script richiede all'utente di confermare la chiusura dell'istanza prima di effettuare il passaggio. Se `-f` è impostata, la funzione non richiede all'utente di cambiare tipo prima di chiudere l'istanza

-v - (switch) Per impostazione predefinita, lo script funziona silenziosamente e visualizza l'output solo in caso di errore. Se `-v` è impostata, la funzione visualizza lo stato durante tutto il suo funzionamento.

File

`change_ec2_instance_type.sh`

Il file di script principale contiene la funzione `change_ec2_instance_type()` che effettua le attività seguenti:

- Verifica l'esistenza dell'istanza Amazon EC2 specificata.
- A meno che `-f` non sia selezionata, avvisa l'utente prima di interrompere l'istanza.
- Modifica il tipo di istanza
- Se `-r` è impostata, riavvia l'istanza e conferma che l'istanza è in esecuzione

Visualizza il codice per `change_ec2_instance_type.sh` su GitHub.

`test_change_ec2_instance_type.sh`

Lo script del file `test_change_ec2_instance_type.sh` verifica i vari percorsi di codice per la funzione `change_ec2_instance_type`. Se tutte le fasi dello script di test sono state eseguite correttamente, lo script di test rimuove tutte le risorse che ha creato.

Puoi eseguire lo script di test con i parametri seguenti:

- -v - (switch) Ogni test mostra lo stato di superamento/non superamento durante l'esecuzione. Per impostazione predefinita, i test vengono eseguiti silenziosamente e l'output include solo lo stato finale complessivo di superamento/non superamento.
- -i - (switch) Lo script viene messo in pausa dopo ogni test per consentire all'utente di sfogliare i risultati intermedi di ogni fase. Consente di esaminare lo stato corrente dell'istanza tramite la console Amazon EC2. Lo script procede alla fase successiva dopo aver premuto INVIO al prompt.

Visualizza il codice per `test_change_ec2_instance_type.sh` su GitHub.

`awsdocs_general.sh`

Il file di script `awsdocs_general.sh` contiene funzioni generiche utilizzate negli esempi avanzati per la AWS CLI.

Visualizza il codice per `awsdocs_general.sh` su GitHub.

Riferimenti

Riferimento alla AWS CLI:

- `aws ec2`
- `aws ec2 describe-instances`
- `aws ec2 modify-instance-attribute`
- `aws ec2 start-instances`
- `aws ec2 stop-instances`
- `aws ec2 wait instance-running`
- `aws ec2 wait instance-stopped`

Altri riferimenti:

- Amazon Elastic Compute Cloud Documentation
- Per visualizzare AWS SDK e contribuire, e per gli esempi di codice della AWS CLI, consulta AWS Code Examples Repository su GitHub.

Utilizzo di Amazon Glacier in AWS CLI

Un'introduzione ad Amazon Glacier
Introduzione ad Amazon Glacier

Questo argomento mostra esempi di AWS CLI comandi che eseguono attività comuni per Amazon Glacier. Gli esempi mostrano come utilizzare AWS CLI per caricare un file di grandi dimensioni su Amazon Glacier suddividendolo in parti più piccole e caricandole dalla riga di comando.

Puoi accedere alle funzionalità di Amazon Glacier utilizzando AWS Command Line Interface ().AWS CLI Per elencare i AWS CLI comandi per Amazon Glacier, usa il seguente comando.

```
aws glacier help
```

> **Note**
>
> Per i riferimenti e ulteriori esempi sui comandi, consulta `aws glacier` in AWS CLI Command Reference.

Argomenti

- Prerequisiti
- Crea un vault Amazon Glacier
- Preparazione di un file per il caricamento
- Avvio di un caricamento in più parti e caricamento dei file
- Completamento del caricamento
- Resources

Prerequisiti

Per eseguire i comandi `glacier`, è necessario:

- Istalla e configura la AWS CLI. Per ulteriori informazioni, consultare the section called "Installazione o aggiornamento" e Credenziali di autenticazione e accesso per la AWS CLI.

- Questo tutorial utilizza diversi strumenti a riga di comando che in genere vengono preinstallati su sistemi operativi di tipo Unix, tra cui Linux e macOS. Gli utenti Windows possono utilizzare gli stessi strumenti installando Cygwin ed eseguendo i comandi dal terminale Cygwin. Le utilità e i comandi nativi di Windows che eseguono le stesse funzioni sono indicati, laddove disponibili.

Crea un vault Amazon Glacier

Creazione di un insieme di credenziali con il comando `create-vault`.

```
$ aws glacier create-vault --account-id - --vault-name myvault
{
    "location": "/123456789012/vaults/myvault"
}
```

> **Note**
>
> Tutti i comandi Amazon Glacier richiedono un parametro ID account. Utilizza il trattino (`--account-id -`) per utilizzare l'account corrente.

Preparazione di un file per il caricamento

Creazione di un file per il caricamento di prova. I seguenti comandi creano un file denominato *largefile* che contiene esattamente 3 MiB di dati casuali.

Linux o macOS

```
$ dd if=/dev/urandom of=largefile bs=3145728 count=1
1+0 records in
1+0 records out
3145728 bytes (3.1 MB) copied, 0.205813 s, 15.3 MB/s
```

dd è un'utilità che consente di copiare una serie di byte da un file di input a un file di output. L'esempio precedente usa il file del dispositivo di sistema `/dev/urandom` come origine dei dati casuali. `fsutil` esegue una funzione simile in Windows.

Windows

```
C:\> fsutil file createnew largefile 3145728
```

```
File C:\temp\largefile is created
```

Quindi, suddividi il file in blocchi da 1 MiB (1048576 byte) con un file splitter.

```
$ split -b 1048576 --verbose largefile chunk
creating file `chunkaa'
creating file `chunkab'
creating file `chunkac'
```

Avvio di un caricamento in più parti e caricamento dei file

Crea un caricamento multiparte in Amazon Glacier utilizzando il comando. initiate-multipart-upload

```
$ aws glacier initiate-multipart-upload --account-id - --archive-description "multipart
 upload test" --part-size 1048576 --vault-name myvault
{
    "uploadId": "19gaRezEXAMPLES6Ry5YYdqthHOC_kGRCT03L9yetr220UmPtBYKk-
OssZtLqyFu7sY1_lR7vgFuJV6NtcV5zpsJ",
    "location": "/123456789012/vaults/myvault/multipart-
uploads/19gaRezEXAMPLES6Ry5YYdqthHOC_kGRCT03L9yetr220UmPtBYKk-
OssZtLqyFu7sY1_lR7vgFuJV6NtcV5zpsJ"
}
```

Amazon Glacier richiede la dimensione di ogni parte in byte (1 MiB in questo esempio), il nome del vault e un ID account per configurare il caricamento multiparte. Al termine dell' AWS CLI operazione, emette un ID di caricamento. Salva l'ID in una variabile shell da utilizzare in futuro.

Linux o macOS

```
$ UPLOADID="19gaRezEXAMPLES6Ry5YYdqthHOC_kGRCT03L9yetr220UmPtBYKk-
OssZtLqyFu7sY1_lR7vgFuJV6NtcV5zpsJ"
```

Windows

```
C:\> set UPLOADID="19gaRezEXAMPLES6Ry5YYdqthHOC_kGRCT03L9yetr220UmPtBYKk-
OssZtLqyFu7sY1_lR7vgFuJV6NtcV5zpsJ"
```

Quindi, utilizza il comando upload-multipart-part per caricare singolarmente le tre parti.

```
$ aws glacier upload-multipart-part --upload-id $UPLOADID --body chunkaa --range 'bytes
 0-1048575/*' --account-id - --vault-name myvault
{
    "checksum": "e1f2a7cd6e047fa606fe2f0280350f69b9f8cfa602097a9a026360a7edc1f553"
}
$ aws glacier upload-multipart-part --upload-id $UPLOADID --body chunkab --range 'bytes
 1048576-2097151/*' --account-id - --vault-name myvault
{
    "checksum": "e1f2a7cd6e047fa606fe2f0280350f69b9f8cfa602097a9a026360a7edc1f553"
}
$ aws glacier upload-multipart-part --upload-id $UPLOADID --body chunkac --range 'bytes
 2097152-3145727/*' --account-id - --vault-name myvault
{
    "checksum": "e1f2a7cd6e047fa606fe2f0280350f69b9f8cfa602097a9a026360a7edc1f553"
}
```

> **Note**
>
> L'esempio precedente utilizza il simbolo del dollaro ($) per fare riferimento ai contenuti della variabile della shell UPLOADID in Linux. Nella riga di comando di Windows, utilizza un segno di percentuale (%) su entrambe le estremità del nome di variabile (ad esempio, %UPLOADID%).

È necessario specificare l'intervallo di byte di ogni parte al momento del caricamento in modo che Amazon Glacier possa riassemblarla nell'ordine corretto. Ogni parte è 1048576 byte, pertanto la prima parte occupa i byte 0-1048575, la seconda i byte 1048576-2097151 e la terza i byte 2097152-3145727.

Completamento del caricamento

Amazon Glacier richiede un tree hash del file originale per confermare che tutti i pezzi caricati siano rimasti intatti. AWS

Per calcolare una struttura hash, devi suddividere il file in parti da 1 MiB e calcolare un hash SHA-256 binario di ogni parte. Quindi si suddivide l'elenco di hash in coppie, si combinano i due hash binari in ciascuna coppia e si utilizzano gli hash risultanti. Ripeti il processo fino a esaurire tutti gli hash. Se il numero di hash è dispari a qualsiasi livello, promuovilo al livello successivo senza modificarlo.

La chiave per calcolare correttamente una struttura hash quando si usano le utility a riga di comando consiste nel memorizzare ogni hash in formato binario ed effettuare la conversione in esadecimale solo nell'ultima fase. Combinando ed eseguendo l'hashing della versione esadecimale di qualsiasi hash nella struttura si otterrà un risultato non corretto.

> **Note**
>
> Gli utenti Windows possono utilizzare il comando `type` anziché `cat`. OpenSSL è disponibile per Windows all'indirizzo OpenSSL.org.

Per calcolare una struttura hash

1. Se non lo hai già fatto, suddividi il file originale in parti da 1 MiB.

```
$ split --bytes=1048576 --verbose largefile chunk
creating file `chunkaa'
creating file `chunkab'
creating file `chunkac'
```

2. Calcola e memorizza l'hash binario SHA-256 di ogni blocco.

```
$ openssl dgst -sha256 -binary chunkaa > hash1
$ openssl dgst -sha256 -binary chunkab > hash2
$ openssl dgst -sha256 -binary chunkac > hash3
```

3. Combina i primi due hash e utilizza l'hash binario del risultato.

```
$ cat hash1 hash2 > hash12
$ openssl dgst -sha256 -binary hash12 > hash12hash
```

4. Combina l'hash padre dei blocchi `aa` e `ab` con l'hash del blocco `ac` ed esegui l'hashing del risultato, questa volta con un output esadecimale. Memorizza il risultato in una variabile della shell.

```
$ cat hash12hash hash3 > hash123
$ openssl dgst -sha256 hash123
SHA256(hash123)= 9628195fcdbcbbe76cdde932d4646fa7de5f219fb39823836d81f0cc0e18aa67
$ TREEHASH=9628195fcdbcbbe76cdde932d4646fa7de5f219fb39823836d81f0cc0e18aa67
```

Infine, completa il caricamento con il comando `complete-multipart-upload`. Questo comando utilizza le dimensioni del file originale in byte, il valore hash nella struttura finale esadecimale, il tuo ID account e il nome dell'insieme di credenziali.

```
$ aws glacier complete-multipart-upload --checksum $TREEHASH --archive-size 3145728 --
upload-id $UPLOADID --account-id - --vault-name myvault
{
    "archiveId": "d3AbWhE0YE1m6f_fI1jPG82F8xzbMEEZmrAlLGAAONJAzo5QdP-
N83MKqd96Unspoa5H5lItWX-sK8-QS0ZhwsyGiu9-R-
kwWUyS1dSBlmgPPWkEbeFfqDSav053rU7FvVLHfRc6hg",
    "checksum": "9628195fcdbcbbe76cdde932d4646fa7de5f219fb39823836d81f0cc0e18aa67",
    "location": "/123456789012/vaults/myvault/archives/
d3AbWhE0YE1m6f_fI1jPG82F8xzbMEEZmrAlLGAAONJAzo5QdP-N83MKqd96Unspoa5H5lItWX-sK8-
QS0ZhwsyGiu9-R-kwWUyS1dSBlmgPPWkEbeFfqDSav053rU7FvVLHfRc6hg"
}
```

Puoi anche controllare lo stato del vault utilizzando il comando `describe-vault`.

```
$ aws glacier describe-vault --account-id - --vault-name myvault
{
    "SizeInBytes": 3178496,
    "VaultARN": "arn:aws:glacier:us-west-2:123456789012:vaults/myvault",
    "LastInventoryDate": "2018-12-07T00:26:19.028Z",
    "NumberOfArchives": 1,
    "CreationDate": "2018-12-06T21:23:45.708Z",
    "VaultName": "myvault"
}
```

> **Note**
>
> Lo stato dell'insieme di credenziali viene aggiornato circa una volta al giorno. Per ulteriori informazioni, consulta l'argomento relativo all'utilizzo di vault.

Ora puoi rimuovere i file di blocco e hash creati in modo sicuro.

```
$ rm chunk* hash*
```

Per ulteriori informazioni sui caricamenti in più parti, consulta Uploading Large Archives in Parts and Computing Checksums nella Amazon Glacier Developer Guide.

Resources

AWS CLI riferimento:

- `aws glacier`
- `aws glacier complete-multipart-upload`
- `aws glacier create-vault`
- `aws glacier describe-vault`
- `aws glacier initiate-multipart-upload`

Riferimento al servizio:

- Guida per sviluppatori Amazon Glacier
- Caricamento di archivi di grandi dimensioni in parti nella Amazon Glacier Developer Guide
- Elaborazione dei checksum nella Guida per gli sviluppatori di Amazon Glacier
- Utilizzo di Vaults nella Amazon Glacier Developer Guide

Utilizzo di IAM nella AWS CLI

Introduzione ad AWS Identity and Access Management.
Introduction to AWS Identity and Access Management

Puoi accedere alle funzionalità di AWS Identity and Access Management (AWS Command Line Interface) utilizzando (AWS CLI). Per elencare i comandi della AWS CLI per IAM, utilizza il comando seguente.

```
aws iam help
```

Questo argomento mostra esempi di comandi della AWS CLI che eseguono operazioni comuni per IAM.

Prima di eseguire qualsiasi comando, imposta le tue credenziali di default. Per ulteriori informazioni, consulta Configurazione delle impostazioni della AWS CLI..

Per ulteriori informazioni sul servizio IAM, consulta la Guida per l'utente AWS Identity and Access Management.

Argomenti

- Creazione di gruppi e utenti IAM
- Collegamento di una policy IAM gestita a un utente
- Impostazione di una password iniziale per un utente IAM
- Creare una chiave di accesso per un utente IAM

Creazione di gruppi e utenti IAM

Per creare un gruppo e aggiungervi un nuovo utente

1. Utilizza il comando `create-group` per creare il gruppo.

```
$ aws iam create-group --group-name MyIamGroup
{
    "Group": {
        "GroupName": "MyIamGroup",
        "CreateDate": "2018-12-14T03:03:52.834Z",
        "GroupId": "AGPAJNUJ2W4IJVEXAMPLE",
        "Arn": "arn:aws:iam::123456789012:group/MyIamGroup",
        "Path": "/"
    }
}
```

2. Utilizza il comando `create-user` per creare l'utente.

```
$ aws iam create-user --user-name MyUser
{
    "User": {
        "UserName": "MyUser",
        "Path": "/",
        "CreateDate": "2018-12-14T03:13:02.581Z",
        "UserId": "AIDAJY2PE5XUZ4EXAMPLE",
        "Arn": "arn:aws:iam::123456789012:user/MyUser"
    }
}
```

3. Utilizza il comando `add-user-to-group` per aggiungere l'utente al gruppo.

```
$ aws iam add-user-to-group --user-name MyUser --group-name MyIamGroup
```

4. Per verificare che il gruppo `MyIamGroup` contenga `MyUser`, utilizza il comando `get-group`.

```
$ aws iam get-group --group-name MyIamGroup
{
    "Group": {
        "GroupName": "MyIamGroup",
        "CreateDate": "2018-12-14T03:03:52Z",
        "GroupId": "AGPAJNUJ2W4IJVEXAMPLE",
        "Arn": "arn:aws:iam::123456789012:group/MyIamGroup",
        "Path": "/"
    },
    "Users": [
        {
            "UserName": "MyUser",
            "Path": "/",
            "CreateDate": "2018-12-14T03:13:02Z",
            "UserId": "AIDAJY2PE5XUZ4EXAMPLE",
            "Arn": "arn:aws:iam::123456789012:user/MyUser"
        }
    ],
    "IsTruncated": "false"
}
```

Collegamento di una policy IAM gestita a un utente

La policy nell'esempio fornisce all'utente "Power User Access" (Accesso utenti avanzati).

Per collegare una policy gestita IAM a un utente

1. Determina l'Amazon Resource Name (ARN) della policy da allegare. Il comando seguente utilizza `list-policies` per trovare l'ARN del criterio con il nome `PowerUserAccess`. Quindi, archivia l'ARN in una variabile di ambiente.

```
$ export POLICYARN=$(aws iam list-policies --query 'Policies[?
PolicyName==`PowerUserAccess`].{ARN:Arn}' --output text)       ~
$ echo $POLICYARN
arn:aws:iam::aws:policy/PowerUserAccess
```

2. Per collegare la policy, utilizza il comando `attach-user-policy` e fai riferimento alla variabile di ambiente che contiene l'ARN della policy.

```
$ aws iam attach-user-policy --user-name MyUser --policy-arn $POLICYARN
```

3. Verifica che la policy sia collegata all'utente eseguendo il comando `list-attached-user-policies`.

```
$ aws iam list-attached-user-policies --user-name MyUser
{
    "AttachedPolicies": [
        {
            "PolicyName": "PowerUserAccess",
            "PolicyArn": "arn:aws:iam::aws:policy/PowerUserAccess"
        }
    ]
}
```

Per ulteriori informazioni, consulta la sezione relativa all'accesso alle risorse di gestione. In questo argomento sono disponibili link a una panoramica di autorizzazioni e policy e a esempi di policy per l'accesso ad Amazon S3, Amazon EC2 e ad altri servizi.

Impostazione di una password iniziale per un utente IAM

Il comando seguente utilizza `create-login-profile` per impostare una password iniziale per l'utente specificato. Quando l'utente accede per la prima volta, deve modificare la password in qualcosa che solo l'utente conosce.

```
$ aws iam create-login-profile --user-name MyUser --password My!User1Login8P@ssword --
password-reset-required
{
    "LoginProfile": {
        "UserName": "MyUser",
        "CreateDate": "2018-12-14T17:27:18Z",
        "PasswordResetRequired": true
    }
}
```

Puoi utilizzare il comando `update-login-profile` per cambiare la password di un utente.

```
$ aws iam update-login-profile --user-name MyUser --password My!User1ADifferentP@ssword
```

Creare una chiave di accesso per un utente IAM

Puoi utilizzare il comando `create-access-key` per creare una chiave di accesso per un utente. Una chiave di accesso è un set di credenziali di sicurezza costituito da un ID chiave di accesso e una chiave segreta.

Un utente può creare solo due chiavi di accesso alla volta. Se provi a creare un terzo set, il comando restituisce un errore `LimitExceeded`.

```
$ aws iam create-access-key --user-name MyUser
{
    "AccessKey": {
        "UserName": "MyUser",
        "AccessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "Status": "Active",
        "SecretAccessKey": "wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY",
        "CreateDate": "2018-12-14T17:34:16Z"
    }
}
```

Utilizza il comando `delete-access-key` per eliminare una chiave di accesso per un utente. Specifica la chiave di accesso da eliminare utilizzando l'ID chiave di accesso.

```
$ aws iam delete-access-key --user-name MyUser --access-key-id AKIAIOSFODNN7EXAMPLE
```

Utilizzo di Amazon S3 nella AWS CLI

Un'introduzione a Amazon Simple Storage Service (Amazon S3)
Introduzione ad Amazon Simple Storage Service (Amazon S3) - Archiviazione nel cloud su AWS

Puoi accedere alle funzionalità di Amazon Simple Storage Service (Amazon S3) con AWS Command Line Interface (AWS CLI). Amazon S3 è un servizio di archiviazione di oggetti altamente scalabile e durevole. Amazon S3 è progettato per fornire una capacità di storage praticamente illimitata, una caratteristica che lo rende una soluzione ideale per un'ampia gamma di esigenze di archiviazione e gestione dei dati.

Amazon S3 consente di memorizzare e recuperare qualsiasi quantità di dati, da file di piccole dimensioni a set di dati di grandi dimensioni, sotto forma di oggetti. Ogni oggetto è archiviato in un container chiamato bucket, a cui è possibile accedere e che può essere gestito tramite la Console di gestione AWS o, in modo programmatico, tramite gli SDK AWS, gli strumenti e la AWS CLI.

Oltre allo storage di base, Amazon S3 offre anche una gamma di funzionalità tra cui gestione del ciclo di vita, controllo delle versioni, scalabilità e sicurezza. Queste funzionalità si integrano con altri Servizi AWS e ti consentono di creare soluzioni basate sul cloud che si adattano alle tue esigenze.

La AWS CLI offre due livelli di comandi per l'accesso ad Amazon S3:

- s3: comandi di livello alto realizzati specificatamente per la AWS CLI che semplificano le operazioni di uso frequente, ad esempio la creazione, la modifica, l'eliminazione e la sincronizzazione di oggetti e bucket.
- s3api: offre accesso diretto a tutte le operazioni API di Amazon S3, consentendo di eseguire operazioni avanzate.

Argomenti in questa guida:

- Utilizzo di comandi di alto livello (s3) in AWS CLI
- Utilizzo dei comandi a livello di API (s3api) con la AWS CLI
- Esempio di scripting per il ciclo di vita del bucket Amazon S3 nel AWS CLI

Utilizzo di comandi di alto livello (s3) in AWS CLI

Questo argomento descrive alcuni dei comandi che è possibile utilizzare per gestire i bucket e gli oggetti Amazon S3 utilizzando i comandi `aws s3` presenti nella AWS CLI. Per i comandi non trattati in questo argomento e per altri esempi di comandi, consulta i comandi `aws s3` nella Guida di riferimento della AWS CLI .

I comandi `aws s3` di alto livello semplificano la gestione degli oggetti Amazon S3. Questi comandi consentono di gestire i contenuti di Amazon S3 al suo interno e utilizzando le directory locali.

Argomenti

- Prerequisiti
- Prima di iniziare
- Creazione di un bucket
- Elenca i bucket e gli oggetti

- Eliminazione di bucket
- Eliminazione di oggetti
- Sposta oggetti
- Copia oggetti
- Oggetti sincronizzati
- Opzioni utilizzate di frequente per i comandi S3
- Resources

Prerequisiti

Per eseguire i comandi `s3`, è necessario:

- Istalla e configura la AWS CLI. Per ulteriori informazioni, consultare the section called "Installazione o aggiornamento" e Credenziali di autenticazione e accesso per la AWS CLI.
- Il profilo utilizzato deve disporre di autorizzazioni che consentano le AWS operazioni eseguite dagli esempi.
- Glossario dei termini di Amazon S3:
  - Bucket: cartella Amazon S3 di primo livello.
  - Prefisso: una cartella Amazon S3 in un bucket.
  - Oggetto: qualsiasi elemento ospitato in un bucket Amazon S3.

Prima di iniziare

Questa sezione descrive alcune cose da tenere a mente prima di utilizzare i comandi `aws s3`.

Caricamenti di oggetti di grandi dimensioni

Quando usi i comandi `aws s3` per caricare oggetti di grandi dimensioni in un bucket Amazon S3, AWS CLI esegue automaticamente un caricamento in più parti. Quando si utilizzano questi comandi `aws s3`, non è possibile riprendere un caricamento non riuscito.

Se il caricamento in più parti non riesce a causa di un timeout o se l'hai annullato manualmente in AWS CLI, AWS CLI interrompe il caricamento e pulisce tutti i file che sono stati creati. Questo processo può richiedere alcuni minuti.

Se il processo di caricamento in più parti o di pulizia viene annullato da un comando kill o da un errore di sistema, i file creati rimangono nel bucket Amazon S3.

Proprietà e tag dei file nelle copie in più parti

Quando utilizzi la AWS CLI versione 1 dei comandi nello spazio dei `aws s3` nomi per copiare un file da una posizione del bucket Amazon S3 a un'altra posizione del bucket Amazon S3 e tale operazione utilizza la copia multiparte, nessuna proprietà del file dall'oggetto di origine viene copiata nell'oggetto di destinazione.

Per impostazione predefinita, i comandi della AWS CLI versione 2 nello spazio dei `s3` nomi che eseguono copie in più parti trasferiscono tutti i tag e il seguente set di proprietà dalla copia di origine a quella di destinazione: `content-type` `content-languagecontent-encoding`,`content-disposition`,,`cache-control`,`expires`, `emetadata`.

Ciò può comportare chiamate AWS API aggiuntive all'endpoint Amazon S3 che non sarebbero state effettuate se avessi utilizzato AWS CLI la versione 1. Questi possono includere: `HeadObject`, `GetObjectTagging` e `PutObjectTagging`.

Se devi modificare questo comportamento predefinito nei comandi della AWS CLI versione 2, usa il `--copy-props` parametro per specificare una delle seguenti opzioni:

- default - Il valore predefinito. Specifica che la copia include tutti i tag collegati all'oggetto di origine e le proprietà racchiuse nel parametro `--metadata-directive` utilizzato per le copie non multipart: `content-type`, `content-language`, `content-encoding`, `content-disposition`, `cache-control`, `expires` e `metadata`.
- metadata-directive - Specifica che la copia include solo le proprietà racchiuse nel parametro `--metadata-directive` utilizzato per le copie non multipart. Non copia alcun tag.
- none - Specifica che la copia non include nessuna delle proprietà dell'oggetto di origine.

Creazione di un bucket

Utilizza il comando `s3 mb` per creare un bucket. I nomi dei bucket devono essere univoci globalmente (univoci in tutto Amazon S3) e devono essere compatibili con DNS.

Possono includere lettere minuscole, numeri, trattini e punti. Possono iniziare e terminare solo con una lettera o un numero e non possono contenere un punto accanto a un trattino o a un altro periodo.

Sintassi

```
$ aws s3 mb <target> [--options]
```

Esempi di s3 mb

L'esempio seguente crea il bucket `s3://amzn-s3-demo-bucket`.

```
$ aws s3 mb s3://amzn-s3-demo-bucket
```

Elenca i bucket e gli oggetti

Per elencare i bucket, le cartelle o gli oggetti, utilizza il comando `s3 ls`. L'utilizzo del comando senza destinazione o opzioni elenca tutti i bucket.

Sintassi

```
$ aws s3 ls <target> [--options]
```

Per alcune opzioni comuni da utilizzare con questo comando ed esempi, consulta Opzioni utilizzate di frequente per i comandi S3. Per un elenco completo delle opzioni disponibili, vedi `s3 ls` in AWS CLI Command Reference.

Esempi di s3 ls

L'esempio seguente elenca tutti i tuoi bucket Amazon S3.

```
$ aws s3 ls
2018-12-11 17:08:50 amzn-s3-demo-bucket1
2018-12-14 14:55:44 amzn-s3-demo-bucket2
```

Il comando seguente elenca tutti gli oggetti e i prefissi inclusi in un bucket. In questo output di esempio, il prefisso `example/` ha un file denominato `MyFile1.txt`.

```
$ aws s3 ls s3://amzn-s3-demo-bucket
                           PRE example/
2018-12-04 19:05:48          3 MyFile1.txt
```

Puoi filtrare l'output per un determinato prefisso includendolo nel comando. Il comando seguente elenca gli oggetti in *bucket-name/example/* (ovvero gli oggetti *bucket-name* filtrati in base al prefisso*example/*).

```
$ aws s3 ls s3://amzn-s3-demo-bucket/example/
2018-12-06 18:59:32          3 MyFile1.txt
```

Per visualizzare solo i bucket e gli oggetti in una Regione specifica, utilizza le opzioni `--region`

```
$ aws s3 ls --region us-east-2
2018-12-06 18:59:32          3 MyFile1.txt
```

Se disponi di un ampio elenco di bucket e oggetti, puoi paginare i risultati utilizzando le opzioni `--max-items` o `--page-size`. L'opzione `--max-items` limita il numero totale di bucket e oggetti restituiti in una chiamata e l'opzione `--page-size` limita il numero di bucket e oggetti elencati in una pagina.

```
$ aws s3 ls --max-items 100 --page-size 10
```

Per ulteriori informazioni sulla paginazione, consulta the section called "--page-size" e the section called "--max-items".

Eliminazione di bucket

Per eliminare un bucket, utilizza il comando `s3 rb`.

Sintassi

```
$ aws s3 rb <target> [--options]
```

Esempi di s3 rb

L'esempio seguente rimuove il bucket `s3://amzn-s3-demo-bucket`.

```
$ aws s3 rb s3://amzn-s3-demo-bucket
```

Per default, perché l'operazione riesca il bucket deve essere vuoto. Per rimuovere un bucket non vuoto, è necessario includere l'opzione `--force`. Se utilizzi un bucket con versioni multiple che contiene oggetti precedentemente eliminati, ma mantenuti, questo comando non consentirà di rimuovere il bucket. Devi prima rimuovere tutti i contenuti.

Il seguente comando di esempio elimina tutti gli oggetti nel bucket ed elimina il bucket stesso.

```
$ aws s3 rb s3://amzn-s3-demo-bucket --force
```

Eliminazione di oggetti

Per eliminare oggetti in un bucket o nella directory locale, usa il comando `s3 rm`.

Sintassi

```
$ aws s3 rm  <target> [--options]
```

Per alcune opzioni comuni da utilizzare con questo comando ed esempi, consulta Opzioni utilizzate di frequente per i comandi S3. Per una lista completa di opzioni, consulta `s3 rm` in AWS CLI Command Reference.

Esempi di s3 rm

L'esempio seguente elimina `filename.txt` da `s3://amzn-s3-demo-bucket/example`.

```
$ aws s3 rm s3://amzn-s3-demo-bucket/example/filename.txt
```

L'esempio seguente elimina tutti gli oggetti da `s3://amzn-s3-demo-bucket/example` utilizzando l'opzione `--recursive`.

```
$ aws s3 rm s3://amzn-s3-demo-bucket/example --recursive
```

Sposta oggetti

Usa il comando `s3 mv` per spostare oggetti da un bucket o da una directory locale. Il comando `s3 mv` copia l'oggetto o file di origine nella destinazione specificata, quindi elimina l'oggetto o il file di origine.

Sintassi

```
$ aws s3 mv <source> <target> [--options]
```

Per alcune opzioni comuni da utilizzare con questo comando ed esempi, consulta Opzioni utilizzate di frequente per i comandi S3. Per un elenco completo delle opzioni disponibili, vedi `s3 mv` in AWS CLI Command Reference.

> **Warning**
>
> Se utilizzi qualsiasi tipo di punto di accesso ARNs o alias del punto di accesso nella tua origine o URIs destinazione Amazon S3, devi prestare particolare attenzione che Amazon URIs S3 di origine e destinazione si risolvano in diversi bucket sottostanti. Se i bucket di origine e di destinazione sono gli stessi, il file o l'oggetto di origine può essere spostato

su sé stesso, il che può comportare l'eliminazione accidentale del file o dell'oggetto di origine. Per verificare che i bucket di origine e di destinazione non siano gli stessi, utilizza il parametro `--validate-same-s3-paths` o imposta la variabile di ambiente `AWS_CLI_S3_MV_VALIDATE_SAME_S3_PATHS` su `true`.

Esempi di s3 mv

L'esempio seguente sposta tutti gli oggetti da `s3://amzn-s3-demo-bucket/example` a `s3://amzn-s3-demo-bucket/`.

```
$ aws s3 mv s3://amzn-s3-demo-bucket/example s3://amzn-s3-demo-bucket/
```

L'esempio seguente sposta un file locale dalla directory di lavoro corrente nel bucket Amazon S3 con il comando `s3 mv`.

```
$ aws s3 mv filename.txt s3://amzn-s3-demo-bucket
```

L'esempio seguente sposta un file dal bucket Amazon S3 alla directory di lavoro corrente, dove `./` specifica la directory di lavoro corrente.

```
$ aws s3 mv s3://amzn-s3-demo-bucket/filename.txt ./
```

Copia oggetti

Usa il comando `s3 cp` per copiare oggetti da un bucket o da una directory locale.

Sintassi

```
$ aws s3 cp <source> <target> [--options]
```

È possibile utilizzare il parametro dash per l'invio in streaming di file su standard input (`stdin`) o standard output (`stdout`).

Warning

Se lo stai utilizzando PowerShell, la shell potrebbe alterare la codifica di un CRLF o aggiungere un CRLF all'input o all'output reindirizzato tramite piping o all'output reindirizzato.

Il comando `s3 cp` utilizza la seguente sintassi per caricare uno streaming di file da `stdin` a un bucket specificato.

Sintassi

```
$ aws s3 cp - <target> [--options]
```

Il comando `s3 cp` utilizza la seguente sintassi per scaricare uno streaming di file Amazon S3 su `stdout`.

Sintassi

```
$ aws s3 cp <target> [--options] -
```

Per alcune opzioni comuni da utilizzare con questo comando ed esempi, consulta Opzioni utilizzate di frequente per i comandi S3. Per la lista completa di opzioni, consulta `s3 cp` in AWS CLI Command Reference.

Esempi di **`s3 cp`**

L'esempio seguente copia tutti gli oggetti da `s3://amzn-s3-demo-bucket/example` a `s3://amzn-s3-demo-bucket/`.

```
$ aws s3 cp s3://amzn-s3-demo-bucket/example s3://amzn-s3-demo-bucket/
```

L'esempio seguente copia un file locale dalla directory di lavoro corrente nel bucket Amazon S3 con il comando `s3 cp`.

```
$ aws s3 cp filename.txt s3://amzn-s3-demo-bucket
```

L'esempio seguente copia un file dal bucket Amazon S3 nella directory di lavoro corrente, dove `./` specifica la directory di lavoro corrente.

```
$ aws s3 cp s3://amzn-s3-demo-bucket/filename.txt ./
```

L'esempio seguente utilizza echo per trasmettere il testo "hello world" al file `s3://bucket-name/filename.txt`.

```
$ echo "hello world" | aws s3 cp - s3://amzn-s3-demo-bucket/filename.txt
```

L'esempio seguente trasmette il file `s3://amzn-s3-demo-bucket/filename.txt` a `stdout` e ne stampa il contenuto sulla console.

```
$ aws s3 cp s3://amzn-s3-demo-bucket/filename.txt -
hello world
```

L'esempio seguente trasmette il contenuto di `s3://bucket-name/pre` a `stdout`, utilizza il comando `bzip2` per comprimere i file e carica il nuovo file compresso denominato `key.bz2` su `s3://bucket-name`.

```
$ aws s3 cp s3://amzn-s3-demo-bucket/pre - | bzip2 --best | aws s3 cp - s3://amzn-s3-demo-bucket/key.bz2
```

Oggetti sincronizzati

Il comando `s3 sync` sincronizza i contenuti di un bucket e una directory o i contenuti di due bucket. In genere, `s3 sync` copia i file o gli oggetti mancanti od obsoleti tra l'origine e la destinazione. Tuttavia, puoi anche utilizzare l'opzione `--delete` per rimuovere dalla destinazione i file o gli oggetti non presenti nell'origine.

Sintassi

```
$ aws s3 sync <source> <target> [--options]
```

Per alcune opzioni comuni da utilizzare con questo comando ed esempi, consulta Opzioni utilizzate di frequente per i comandi S3. Per una lista completa di opzioni, consulta `s3 sync` in AWS CLI Command Reference.

Esempi di s3 sync

L'esempio seguente sincronizza il contenuto di un prefisso Amazon S3 denominato path nel bucket denominato amzn-s3-demo-bucket con la directory di lavoro corrente.

`s3 sync` aggiorna tutti i file che hanno dimensioni diverse o un orario modificato rispetto ai file con lo stesso nome nella destinazione. L'output mostra le operazioni specifiche eseguite durante la sincronizzazione. Nota che l'operazione sincronizza ricorsivamente la sottodirectory `MySubdirectory` e il suo contenuto con `s3://amzn-s3-demo-bucket/path/MySubdirectory`.

```
$ aws s3 sync . s3://amzn-s3-demo-bucket/path
```

```
upload: MySubdirectory\MyFile3.txt to s3://amzn-s3-demo-bucket/path/MySubdirectory/
MyFile3.txt
upload: MyFile2.txt to s3://amzn-s3-demo-bucket/path/MyFile2.txt
upload: MyFile1.txt to s3://amzn-s3-demo-bucket/path/MyFile1.txt
```

L'esempio seguente, che amplia quello precedente, mostra come utilizzare l'opzione `--delete`.

```
// Delete local file
$ rm ./MyFile1.txt

// Attempt sync without --delete option - nothing happens
$ aws s3 sync . s3://amzn-s3-demo-bucket/path

// Sync with deletion - object is deleted from bucket
$ aws s3 sync . s3://amzn-s3-demo-bucket/path --delete
delete: s3://amzn-s3-demo-bucket/path/MyFile1.txt

// Delete object from bucket
$ aws s3 rm s3://amzn-s3-demo-bucket/path/MySubdirectory/MyFile3.txt
delete: s3://amzn-s3-demo-bucket/path/MySubdirectory/MyFile3.txt

// Sync with deletion - local file is deleted
$ aws s3 sync s3://amzn-s3-demo-bucket/path . --delete
delete: MySubdirectory\MyFile3.txt

// Sync with Infrequent Access storage class
$ aws s3 sync . s3://amzn-s3-demo-bucket/path --storage-class STANDARD_IA
```

Quando si utilizza l'opzione `--delete`, le opzioni `--exclude` e `--include` possono filtrare file o oggetti da eliminare durante un'operazione `s3 sync`. In questo caso, la stringa di parametro deve specificare i file da escludere o includere per l'eliminazione nel contesto della directory o del bucket di destinazione. Di seguito viene riportato un esempio.

```
Assume local directory and s3://amzn-s3-demo-bucket/path currently in sync and each
 contains 3 files:
MyFile1.txt
MyFile2.rtf
MyFile88.txt
'''

// Sync with delete, excluding files that match a pattern. MyFile88.txt is deleted,
 while remote MyFile1.txt is not.
```

```
$ aws s3 sync . s3://amzn-s3-demo-bucket/path --delete --exclude "path/MyFile?.txt"
delete: s3://amzn-s3-demo-bucket/path/MyFile88.txt
'''

// Sync with delete, excluding MyFile2.rtf - local file is NOT deleted
$ aws s3 sync s3://amzn-s3-demo-bucket/path . --delete --exclude "./MyFile2.rtf"
download: s3://amzn-s3-demo-bucket/path/MyFile1.txt to MyFile1.txt
'''

// Sync with delete, local copy of MyFile2.rtf is deleted
$ aws s3 sync s3://amzn-s3-demo-bucket/path . --delete
delete: MyFile2.rtf
```

Opzioni utilizzate di frequente per i comandi S3

Le seguenti opzioni vengono utilizzate di frequente per i comandi descritti in questo argomento. Per un elenco completo delle opzioni che puoi usare su un comando, consulta il comando specifico nella guida di riferimento alla versione 2 della .

acl

`s3 sync` e `s3 cp` possono utilizzare l'opzione `--acl`. Ciò consente di impostare le autorizzazioni di accesso per i file copiati su Amazon S3. L'opzione `--acl` accetta i valori `private`, `public-read` e `public-read-write`. Per maggiori informazioni, consulta Canned ACL nella Guida per l'utente di Amazon S3.

```
$ aws s3 sync . s3://amzn-s3-demo-bucket/path --acl public-read
```

exclude

Quando usi il comando `s3 cp`, `s3 mv`, `s3 sync` o `s3 rm`, puoi filtrare i risultati utilizzando l'opzione `--exclude` o `--include`. L'opzione `--exclude` imposta le regole per escludere solo gli oggetti dal comando e le opzioni si applicano nell'ordine specificato. Questo viene mostrato nell'esempio seguente.

```
Local directory contains 3 files:
MyFile1.txt
MyFile2.rtf
MyFile88.txt

// Exclude all .txt files, resulting in only MyFile2.rtf being copied
$ aws s3 cp . s3://amzn-s3-demo-bucket/path --exclude "*.txt"
```

```
// Exclude all .txt files but include all files with the "MyFile*.txt" format,
 resulting in, MyFile1.txt, MyFile2.rtf, MyFile88.txt being copied
$ aws s3 cp . s3://amzn-s3-demo-bucket/path --exclude "*.txt" --include
 "MyFile*.txt"

// Exclude all .txt files, but include all files with the "MyFile*.txt" format,
 but exclude all files with the "MyFile?.txt" format resulting in, MyFile2.rtf and
 MyFile88.txt being copied
$ aws s3 cp . s3://amzn-s3-demo-bucket/path --exclude "*.txt" --include
 "MyFile*.txt" --exclude "MyFile?.txt"
```

include

Quando usi il comando `s3 cp`, `s3 mv`, `s3 sync` o `s3 rm`, puoi filtrare i risultati utilizzando l'opzione `--exclude` o `--include`. L'opzione `--include` imposta le regole per includere solo gli oggetti specificati nel comando e le opzioni si applicano nell'ordine specificato. Questo viene mostrato nell'esempio seguente.

```
Local directory contains 3 files:
MyFile1.txt
MyFile2.rtf
MyFile88.txt

// Include all .txt files, resulting in MyFile1.txt and MyFile88.txt being copied
$ aws s3 cp . s3://amzn-s3-demo-bucket/path --include "*.txt"

// Include all .txt files but exclude all files with the "MyFile*.txt" format,
 resulting in no files being copied
$ aws s3 cp . s3://amzn-s3-demo-bucket/path --include "*.txt" --exclude
 "MyFile*.txt"

// Include all .txt files, but exclude all files with the "MyFile*.txt" format, but
 include all files with the "MyFile?.txt" format resulting in MyFile1.txt being
 copied

$ aws s3 cp . s3://amzn-s3-demo-bucket/path --include "*.txt" --exclude
 "MyFile*.txt" --include "MyFile?.txt"
```

grant

I comandi `s3 cp`, `s3 mv` e `s3 sync` includono un'opzione `--grants` che può essere utilizzata per concedere autorizzazioni sull'oggetto a utenti o gruppi specificati. Imposta l'opzione `--`

`grants` per un elenco di autorizzazioni utilizzando la sintassi seguente. Sostituisci `Permission`, `Grantee_Type` e `Grantee_ID` con il tuo valore.

Sintassi

```
--grants Permission=Grantee_Type=Grantee_ID
         [Permission=Grantee_Type=Grantee_ID ...]
```

Ogni valore contiene i seguenti elementi:

- *`Permission`*— Speciifica le autorizzazioni concesse. Può essere impostato su `read`, `readacl`, `writeacl` o `full`.
- *`Grantee_Type`*— Specifica come identificare il beneficiario. Può essere impostato su `uri`, `emailaddress` o `id`.
- *`Grantee_ID`*— Speciifica il beneficiario in base a. *`Grantee_Type`*
  - `uri` - L'URI del gruppo. Per ulteriori informazioni, consulta Chi è l'assegnatario?
  - `emailaddress` - L'indirizzo e-mail dell'account.
  - `id` - L'ID canonico dell'account.

Per ulteriori informazioni sul controllo degli accessi di Amazon S3, consulta Controllo degli accessi.

L'esempio seguente copia un oggetto in un bucket. Concede le autorizzazioni `read` per l'oggetto a tutti e le autorizzazioni `full` (`read`, `readacl` e `writeacl`) all'account associato a `user@example.com`.

```
$ aws s3 cp file.txt s3://amzn-s3-demo-bucket/ --grants read=uri=http://
acs.amazonaws.com/groups/global/AllUsers full=emailaddress=user@example.com
```

Puoi anche specificare una classe di archiviazione non predefinita (`REDUCED_REDUNDANCY` o `STANDARD_IA`) per gli oggetti caricati su Amazon S3. Per farlo, utilizza l'opzione `--storage-class`.

```
$ aws s3 cp file.txt s3://amzn-s3-demo-bucket/ --storage-class REDUCED_REDUNDANCY
```

nessuna sovrascrittura

I `s3 sync` comandi `s3 cp` `s3 mv`, e includono un'`--no-overwrite`opzione che è possibile utilizzare per impedire la sovrascrittura di oggetti già esistenti nella destinazione.

L'esempio seguente copia un oggetto da un bucket nella directory locale solo se non esiste già nella directory locale.

```
$ aws s3 cp --no-overwrite s3://amzn-s3-demo-bucket/file.txt file.txt
```

L'esempio seguente copia in modo ricorsivo i file da una directory locale a un bucket. Copierà solo i file che non esistono già nel bucket.

```
$ aws s3 cp --recursive --no-overwrite /path/to/demo-files/ s3://amzn-s3-demo-bucket/demo-files/
```

L'esempio seguente sposta un oggetto da una directory locale a un bucket solo se non esiste già nella posizione di destinazione del bucket.

```
$ aws s3 mv --no-overwrite file.txt s3://amzn-s3-demo-bucket/file.txt
```

L'esempio seguente sincronizza i file da una directory locale a un bucket. Sincronizzerà solo i file che non esistono già nel bucket di destinazione.

```
$ aws s3 sync --no-overwrite /path/to/demo-files/ s3://amzn-s3-demo-bucket/demo-files/
```

recursive

Quando utilizzi questa opzione, il comando viene eseguito su tutti i file o gli oggetti nella directory o nel prefisso specificati. L'esempio seguente elimina `s3://amzn-s3-demo-bucket/path` e tutti i suoi contenuti.

```
$ aws s3 rm s3://amzn-s3-demo-bucket/path --recursive
```

Resources

AWS CLI riferimento:

- `aws s3`
- `aws s3 cp`
- `aws s3 mb`
- `aws s3 mv`

- `aws s3 ls`
- `aws s3 rb`
- `aws s3 rm`
- `aws s3 sync`

Riferimento al servizio:

- Utilizzo di bucket Amazon S3 nella Guida per l'utente di Amazon S3
- Utilizzo di oggetti Amazon S3 nella Guida per l'utente di Amazon S3
- Elenco gerarchico delle chiavi utilizzando un prefisso e un delimitatore nella Guida per l'utente di Amazon S3
- Interrompi i caricamenti in più parti su un bucket S3 utilizzando la AWS SDK per .NET (di basso livello) nella Amazon S3 User Guide

Utilizzo dei comandi a livello di API (s3api) con la AWS CLI

I comandi a livello di API (contenuti nel set di `s3api` comandi) forniscono l'accesso diretto ad Amazon Simple Storage Service (Amazon APIs S3) e abilitano alcune operazioni che non sono esposte nei comandi di alto livello. `s3` Questi comandi sono equivalenti agli altri servizi AWS che forniscono l'accesso a livello di API alle funzionalità dei servizi. Per ulteriori informazioni sui comandi `s3`, consulta Utilizzo di comandi di alto livello (s3) in AWS CLI

Questo argomento fornisce esempi che dimostrano come utilizzare i comandi di livello inferiore mappati ad Amazon S3. APIs Inoltre, puoi trovare esempi per ogni comando dell'API S3 nella `s3api` sezione della guida di riferimento, versione 2.

Argomenti

- Prerequisiti
- Applicazione di una lista di controllo accessi (ACL) personalizzata
- Configurazione di una policy di registrazione
- Resources

Prerequisiti

Per eseguire i comandi `s3api`, è necessario:

- Istalla e configura la AWS CLI. Per ulteriori informazioni, consultare the section called "Installazione o aggiornamento" e Credenziali di autenticazione e accesso per la AWS CLI.
- Il profilo che utilizzi deve disporre di autorizzazioni che consentano le AWS operazioni eseguite dagli esempi.
- Glossario dei termini di Amazon S3:
  - Bucket: cartella Amazon S3 di primo livello.
  - Prefisso: una cartella Amazon S3 in un bucket.
  - Oggetto: qualsiasi elemento ospitato in un bucket Amazon S3.

Applicazione di una lista di controllo accessi (ACL) personalizzata

Con i comandi di alto livello, puoi utilizzare l'`--acl`opzione per applicare elenchi di controllo degli accessi predefiniti (ACLs) agli oggetti Amazon S3. Ma non puoi usare quel comando per impostare bucket-wide. ACLs Tuttavia, puoi farlo utilizzando il comando `put-bucket-acl` a livello di API.

L'esempio seguente mostra come concedere il controllo completo a due AWS utenti (user1@example.com e user2@example.com) e il permesso di lettura a tutti. L'identificatore per "tutti" proviene da un URI speciale passato come parametro.

```
$ aws s3api put-bucket-acl --bucket amzn-s3-demo-bucket --grant-full-control
 'emailaddress="user1@example.com",emailaddress="user2@example.com"' --grant-read
 'uri="http://acs.amazonaws.com/groups/global/AllUsers"'
```

Per dettagli su come costruire il ACLs, consulta PUT Bucket acl nel riferimento all'API di Amazon Simple Storage Service. I comandi ACL `s3api` nell'interfaccia a riga di comando, come `put-bucket-acl`, utilizzano la stessa notazione di argomento abbreviata.

Configurazione di una policy di registrazione

Il comando API `put-bucket-logging` configura una policy di registrazione di log dei bucket.

Nell'esempio seguente, all' AWS utente user@example.com viene concesso il pieno controllo sui file di registro e tutti gli utenti hanno accesso in lettura ad essi. Nota che il comando `put-bucket-acl` è richiesto anche per concedere al sistema di distribuzione dei log di Amazon S3 (specificato da un URI) le autorizzazioni necessarie per leggere e scrivere i log nel bucket.

```
$ aws s3api put-bucket-acl --bucket amzn-s3-demo-bucket --grant-read-acp 'URI="http://
acs.amazonaws.com/groups/s3/LogDelivery"' --grant-write 'URI="http://acs.amazonaws.com/
groups/s3/LogDelivery"'
```

```
$ aws s3api put-bucket-logging --bucket amzn-s3-demo-bucket --bucket-logging-status
 file://logging.json
```

Il file `logging.json` nel comando precedente ha il seguente contenuto.

```
{
  "LoggingEnabled": {
    "TargetBucket": "amzn-s3-demo-bucket",
    "TargetPrefix": "amzn-s3-demo-bucketLogs/",
    "TargetGrants": [
      {
        "Grantee": {
          "Type": "AmazonCustomerByEmail",
          "EmailAddress": "user@example.com"
        },
        "Permission": "FULL_CONTROL"
      },
      {
        "Grantee": {
          "Type": "Group",
          "URI": "http://acs.amazonaws.com/groups/global/AllUsers"
        },
        "Permission": "READ"
      }
    ]
  }
}
```

Resources

AWS CLI riferimento:

- aws s3api
- aws s3api put-bucket-acl
- aws s3api put-bucket-logging

Riferimento al servizio:

- Utilizzo di bucket Amazon S3 nella Guida per l'utente di Amazon S3
- Utilizzo di oggetti Amazon S3 nella Guida per l'utente di Amazon S3

- Elenco gerarchico delle chiavi utilizzando un prefisso e un delimitatore nella Guida per l'utente di Amazon S3
- Interrompi i caricamenti in più parti su un bucket S3 utilizzando la AWS SDK per .NET (di basso livello) nella Amazon S3 User Guide

Esempio di scripting per il ciclo di vita del bucket Amazon S3 nel AWS CLI

Questo argomento utilizza un esempio di script Bash per le operazioni del ciclo di vita dei bucket Amazon S3 utilizzando AWS Command Line Interface (AWS CLI). Questo esempio di script utilizza il set di comandi `aws s3api`. Gli script di shell sono programmi progettati per l'esecuzione in un'interfaccia a riga di comando.

Argomenti

- Prima di iniziare
- Informazioni sull'esempio
- File
- Riferimenti

Prima di iniziare

Prima di poter eseguire uno qualsiasi degli esempi seguenti, è necessario completare le seguenti operazioni.

- Istalla e configura la AWS CLI. Per ulteriori informazioni, consultare the section called "Installazione o aggiornamento" e Credenziali di autenticazione e accesso per la AWS CLI.
- Il profilo che usi deve avere permessi che consentano le AWS operazioni eseguite dagli esempi.

- Come procedura AWS consigliata, concedi a questo codice il privilegio minimo o solo le autorizzazioni necessarie per eseguire un'operazione. Per ulteriori informazioni, consulta Assegnare il privilegio minimo nella Guida per l'utente di IAM.
- Questo codice non è stato testato in tutte le AWS regioni. Alcuni AWS servizi sono disponibili solo in regioni specifiche. Per ulteriori informazioni, consulta Service Endpoints and Quotas nella AWS General Reference Guide.

- L'esecuzione di questo codice può comportare addebiti sul tuo AWS account. È tua responsabilità assicurarti che tutte le risorse create da questo script vengano rimosse quando hai finito di utilizzarle.

Il servizio Amazon S3 utilizza i termini seguenti:

- Bucket: una cartella Amazon S3 di primo livello.
- Prefisso: una cartella Amazon S3 in un bucket.
- Oggetto: qualsiasi elemento ospitato in un bucket Amazon S3.

Informazioni sull'esempio

Questo esempio dimostra come interagire con alcune delle operazioni di base di Amazon S3 utilizzando un set di funzioni nei file di script di shell. Le funzioni si trovano nel file di script di shell denominato `bucket-operations.sh`. È possibile richiamare queste funzioni in un altro file. Ogni file di script contiene commenti che descrivono ciascuna delle funzioni.

Per visualizzare i risultati intermedi di ogni fase, esegui lo script con un parametro `-i`. È possibile visualizzare lo stato attuale del bucket o il suo contenuto utilizzando la console Amazon S3. Lo script passa alla fase successiva solo quando premi invio al prompt.

Per l'esempio completo e i file di script scaricabili, consulta Amazon S3 Bucket Lifecycle Operations nel Code Examples Repository su.AWS GitHub

File

L'esempio contiene i file seguenti:

bucket-operations.sh

Questo file di script principale può provenire da un altro file. Include funzioni in grado di eseguire le seguenti attività:

- Creare un bucket e verificarne l'esistenza
- Copiare un file dal computer locale in un bucket
- Copiare un file da una posizione del bucket a un'altra posizione del bucket
- Elenco dei contenuti di un bucket
- Eliminazione di un file da un bucket

- Eliminazione di un bucket

Visualizza il codice per on. `bucket-operations.sh` GitHub

test-bucket-operations.sh

Il file di script della shell `test-bucket-operations.sh` dimostra come chiamare le funzioni recuperando il file `bucket-operations.sh` e chiamando ciascuna delle funzioni. Dopo aver chiamato le funzioni, lo script di test rimuove tutte le risorse che ha creato.

Visualizza il codice per `test-bucket-operations.sh` on GitHub.

awsdocs-general.sh

Il file di script `awsdocs-general.sh` contiene funzioni generiche utilizzate negli esempi di codice avanzati per la AWS CLI.

Visualizza il codice per `awsdocs-general.sh` on GitHub.

Riferimenti

AWS CLI riferimento:

- `aws s3api`
- `aws s3api create-bucket`
- `aws s3api copy-object`
- `aws s3api delete-bucket`
- `aws s3api delete-object`
- `aws s3api head-bucket`
- `aws s3api list-objects`
- `aws s3api put-object`

Altri riferimenti:

- Utilizzo di bucket Amazon S3 nella Guida per l'utente di Amazon S3
- Utilizzo di oggetti Amazon S3 nella Guida per l'utente di Amazon S3

- Per visualizzare e contribuire all' AWS SDK e agli esempi di AWS CLI codice, consulta il AWS Code Examples Repository su. GitHub

Accedere ad Amazon SNS in AWS CLI

Puoi accedere alle funzionalità di Amazon Simple Notification Service (Amazon SNS) tramite AWS Command Line Interface (AWS CLI). Per elencare i comandi della AWS CLI per Amazon SNS, utilizza il comando seguente.

```
aws sns help
```

Prima di eseguire qualsiasi comando, imposta le tue credenziali di default. Per ulteriori informazioni, consulta Configurazione delle impostazioni della AWS CLI..

Questo argomento mostra esempi di comandi della AWS CLI che eseguono operazioni comuni per Amazon SNS.

Argomenti

- Creazione di un argomento
- Sottoscrizione a un argomento
- Pubblicazione in un argomento
- Annullamento della sottoscrizione a un argomento
- Eliminazione di un argomento

Creazione di un argomento

Per creare un argomento, utilizza il comando `sns create-topic` e specifica il nome da assegnare all'argomento.

```
$ aws sns create-topic --name my-topic
{
    "TopicArn": "arn:aws:sns:us-west-2:123456789012:my-topic"
}
```

Prendi nota del valore `TopicArn` della risposta, che utilizzerai in seguito per pubblicare un messaggio.

Sottoscrizione a un argomento

Per sottoscrivere un argomento, utilizza il comando `sns subscribe`.

L'esempio seguente specifica il protocollo `email` e un indirizzo e-mail per `notification-endpoint`.

```
$ aws sns subscribe --topic-arn arn:aws:sns:us-west-2:123456789012:my-topic --
protocol email --notification-endpoint saanvi@example.com
{
    "SubscriptionArn": "pending confirmation"
}
```

AWS invia subito un'e-mail di conferma all'indirizzo specificato nel comando `subscribe`. Il testo dell'e-mail è il seguente.

```
You have chosen to subscribe to the topic:
arn:aws:sns:us-west-2:123456789012:my-topic
To confirm this subscription, click or visit the following link (If this was in error
 no action is necessary):
Confirm subscription
```

Quando il destinatario fa clic sul collegamento Confirm subscription (Conferma sottoscrizione), il browser del destinatario visualizza un messaggio di notifica con informazioni simili alle seguenti.

```
Subscription confirmed!

You have subscribed saanvi@example.com to the topic:my-topic.

Your subscription's id is:
arn:aws:sns:us-west-2:123456789012:my-topic:1328f057-de93-4c15-512e-8bb22EXAMPLE

If it was not your intention to subscribe, click here to unsubscribe.
```

Pubblicazione in un argomento

Per inviare un messaggio a tutti gli abbonati a un argomento, utilizza il comando `sns publish`.

L'esempio seguente invia il messaggio "Hello World!" a tutti gli abbonati dell'argomento specificato.

```
$ aws sns publish --topic-arn arn:aws:sns:us-west-2:123456789012:my-topic --
message "Hello World!"
```

```
{
    "MessageId": "4e41661d-5eec-5ddf-8dab-2c867EXAMPLE"
}
```

In questo esempio, la AWS invia un'e-mail con il testo "Hello World!" alla `saanvi@example.com`.

Annullamento della sottoscrizione a un argomento

Per annullare l'abbonamento a un argomento e interrompere la ricezione dei messaggi pubblicati nell'argomento, utilizza il comando `sns unsubscribe` e specifica l'ARN dell'argomento di cui si desidera annullare l'abbonamento.

```
$ aws sns unsubscribe --subscription-arn arn:aws:sns:us-west-2:123456789012:my-topic:1328f057-de93-4c15-512e-8bb22EXAMPLE
```

Per verificare il corretto annullamento della sottoscrizione, utilizza il comando `sns list-subscriptions` per confermare che l'ARN non appaia più nell'elenco.

```
$ aws sns list-subscriptions
```

Eliminazione di un argomento

Per eliminare un argomento, è necessario eseguire il comando `sns delete-topic`.

```
$ aws sns delete-topic --topic-arn arn:aws:sns:us-west-2:123456789012:my-topic
```

Per verificare che AWS abbia eliminato correttamente l'argomento, utilizza il comando `sns list-topics` per confermare che l'argomento non appaia più nell'elenco.

```
$ aws sns list-topics
```

AWS CLI esempi di comandi

Gli esempi di codice riportati in questo argomento mostrano come utilizzare AWS Command Line Interface with AWS.

Nozioni di base: esempi di codice che mostrano come eseguire le operazioni essenziali all'interno di un servizio.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Scenari: esempi di codice che mostrano come eseguire un'attività specifica chiamando più funzioni all'interno dello stesso servizio o combinate con altri Servizi AWS.

Alcuni servizi contengono ulteriori categorie di esempi che mostrano come sfruttare le librerie o le funzioni specifiche del servizio.

Servizi

- Esempi di utilizzo di ACM AWS CLI
- Esempi di API Gateway utilizzando AWS CLI
- API Gateway HTTP e esempi di WebSocket API che utilizzano AWS CLI
- Esempi di API Gateway Management che utilizzano AWS CLI
- Esempi di App Mesh utilizzando AWS CLI
- Esempi di App Runner utilizzando AWS CLI
- AWS AppConfig esempi utilizzando AWS CLI
- Esempi di Application Auto Scaling utilizzando AWS CLI
- Esempi di Application Discovery Service utilizzando AWS CLI
- Esempi di segnali applicativi utilizzando AWS CLI
- AppRegistry esempi utilizzando AWS CLI
- Esempi di Athena che utilizzano AWS CLI
- Esempi di Auto Scaling utilizzando AWS CLI
- Esempi di Piani di Auto Scaling che utilizzano AWS CLI
- AWS Backup esempi utilizzando AWS CLI
- AWS Batch esempi utilizzando AWS CLI
- Budget AWS esempi utilizzando AWS CLI
- Esempi di utilizzo di Amazon Chime AWS CLI
- Esempi di API Cloud Control che utilizzano AWS CLI
- AWS Cloud Map esempi utilizzando AWS CLI
- AWS Cloud9 esempi utilizzando AWS CLI

- CloudFormation esempi utilizzando AWS CLI
- CloudFront esempi utilizzando AWS CLI
- CloudSearch Esempi di utilizzo di Amazon AWS CLI
- CloudTrail esempi utilizzando AWS CLI
- CloudWatch esempi utilizzando AWS CLI
- CloudWatch Esempi di log utilizzando AWS CLI
- CloudWatch Esempi di monitoraggio della rete utilizzando AWS CLI
- CloudWatch Esempi di Observability Access Monitor utilizzando AWS CLI
- CloudWatch Esempi di Observability Admin utilizzando AWS CLI
- CloudWatch Esempi di Synthetics utilizzando AWS CLI
- CodeArtifact esempi utilizzando AWS CLI
- CodeBuild esempi utilizzando AWS CLI
- CodeCommit esempi utilizzando AWS CLI
- CodeDeploy esempi utilizzando AWS CLI
- CodeGuru Esempi di revisori che utilizzano AWS CLI
- CodePipeline esempi utilizzando AWS CLI
- AWS CodeStar Esempi di notifiche utilizzando AWS CLI
- CodeConnections esempi utilizzando AWS CLI
- Esempi di identità di Amazon Cognito utilizzando AWS CLI
- Esempi di Amazon Cognito Identity Provider utilizzando AWS CLI
- Esempi di Amazon Comprehend con AWS CLI
- Esempi di utilizzo di Amazon Comprehend Medical AWS CLI
- AWS Config esempi utilizzando AWS CLI
- Esempi di utilizzo di Amazon Connect AWS CLI
- AWS Cost and Usage Report esempi utilizzando AWS CLI
- Esempi di utilizzo del servizio Cost Explorer AWS CLI
- Esempi di utilizzo di Firehose AWS CLI
- Esempi di utilizzo di Amazon Data Lifecycle Manager AWS CLI
- AWS Data Pipeline esempi utilizzando AWS CLI

- DataSync esempi utilizzando AWS CLI
- Esempi di utilizzo di DAX AWS CLI
- Esempi di Detective che utilizzano AWS CLI
- Esempi di Device Farm utilizzando AWS CLI
- Direct Connect esempi utilizzando AWS CLI
- Directory Service esempi utilizzando AWS CLI
- Directory Service Esempi di dati utilizzando AWS CLI
- AWS DMS esempi utilizzando AWS CLI
- Esempi di utilizzo di Amazon DocumentDB AWS CLI
- Esempi di utilizzo di DynamoDB AWS CLI
- Esempi di DynamoDB Streams utilizzando AWS CLI
- EC2 Esempi di utilizzo di Amazon AWS CLI
- Esempi di Amazon EC2 Instance Connect utilizzando AWS CLI
- Esempi di utilizzo di Amazon ECR AWS CLI
- Amazon ECR Public: esempi di utilizzo AWS CLI
- Esempi di utilizzo di Amazon ECS AWS CLI
- Esempi di utilizzo di Amazon EFS AWS CLI
- Esempi di utilizzo di Amazon EKS AWS CLI
- Esempi di utilizzo di Elastic Beanstalk AWS CLI
- Elastic Load Balancing - Esempi di utilizzo della versione 1 AWS CLI
- Elastic Load Balancing - Esempi di utilizzo della versione 2 AWS CLI
- ElastiCache esempi utilizzando AWS CLI
- MediaStore esempi utilizzando AWS CLI
- Esempi di utilizzo di Amazon EMR AWS CLI
- Esempi di utilizzo di Amazon EMR su EKS AWS CLI
- EventBridge esempi utilizzando AWS CLI
- EventBridge Esempi di tubi che utilizzano AWS CLI
- Esempi di Firewall Manager che utilizzano AWS CLI
- AWS FIS esempi utilizzando AWS CLI

- Esempi di Amazon GameLift Servers che utilizzano AWS CLI
- Esempi di utilizzo di Amazon Glacier AWS CLI
- Esempi di Global Accelerator utilizzando AWS CLI
- AWS Glue esempi utilizzando AWS CLI
- GuardDuty esempi utilizzando AWS CLI
- AWS Health esempi utilizzando AWS CLI
- HealthImaging esempi utilizzando AWS CLI
- HealthLake esempi utilizzando AWS CLI
- HealthOmics esempi utilizzando AWS CLI
- Esempi IAM che utilizzano AWS CLI
- Esempi di IAM Access Analyzer con AWS CLI
- Esempi di Image Builder che utilizzano AWS CLI
- Esempi di Incident Manager che utilizzano AWS CLI
- Esempi di contatti di Incident Manager utilizzando AWS CLI
- Esempi di utilizzo di Amazon Inspector AWS CLI
- AWS IoT esempi utilizzando AWS CLI
- AWS IoT Analytics esempi utilizzando AWS CLI
- Esempi di Device Advisor che utilizzano AWS CLI
- AWS IoT data esempi utilizzando AWS CLI
- AWS IoT Events esempi utilizzando AWS CLI
- AWS IoT Events-Data esempi utilizzando AWS CLI
- AWS IoT Greengrass esempi utilizzando AWS CLI
- AWS IoT Greengrass V2 esempi utilizzando AWS CLI
- AWS IoT Processi SDK rilascio esempi utilizzando AWS CLI
- AWS IoT SiteWise esempi utilizzando AWS CLI
- AWS IoT Things Graph esempi utilizzando AWS CLI
- Wireless AWS IoT esempi utilizzando AWS CLI
- Esempi di utilizzo di Amazon IVS AWS CLI
- Esempi di Amazon IVS Chat con AWS CLI

- Esempi di streaming in tempo reale di Amazon IVS utilizzando AWS CLI
- Esempi di utilizzo di Amazon Kendra AWS CLI
- Esempi di Kinesis che utilizzano AWS CLI
- AWS KMS esempi utilizzando AWS CLI
- Esempi di Lake Formation utilizzando AWS CLI
- Esempi di utilizzo di Lambda AWS CLI
- Esempi di License Manager che utilizzano AWS CLI
- Esempi di utilizzo di Lightsail AWS CLI
- Esempi di Macie utilizzando AWS CLI
- Esempi di Amazon Managed Grafana utilizzando AWS CLI
- MediaConnect esempi utilizzando AWS CLI
- MediaConvert esempi utilizzando AWS CLI
- MediaLive esempi utilizzando AWS CLI
- MediaPackage esempi utilizzando AWS CLI
- MediaPackage Esempi di VOD che utilizzano AWS CLI
- MediaStore Esempi di piani dati che utilizzano AWS CLI
- MediaTailor esempi utilizzando AWS CLI
- Esempi di MemoryDB che utilizzano AWS CLI
- Esempi di utilizzo di Amazon MSK AWS CLI
- Esempi di Network Flow Monitor utilizzando AWS CLI
- Esempi di Network Manager che utilizzano AWS CLI
- OpenSearch Esempi di servizi che utilizzano AWS CLI
- Organizations: esempi che utilizzano AWS CLI
- AWS Outposts esempi utilizzando AWS CLI
- AWS Payment Cryptography esempi utilizzando AWS CLI
- AWS Payment Cryptography Esempi di piani dati che utilizzano AWS CLI
- Esempi di utilizzo di Amazon Pinpoint AWS CLI
- Esempi di utilizzo di Amazon Polly AWS CLI
- Listino prezzi AWS esempi utilizzando AWS CLI

- AWS Private CA esempi utilizzando AWS CLI
- AWS Proton esempi utilizzando AWS CLI
- Esempi di utilizzo di Amazon RDS AWS CLI
- Esempi di Amazon RDS Data Service utilizzando AWS CLI
- Esempi di utilizzo di Amazon RDS Performance Insights AWS CLI
- Esempi di utilizzo di Amazon Redshift AWS CLI
- Esempi di utilizzo di Amazon Rekognition AWS CLI
- AWS RAM esempi utilizzando AWS CLI
- Esempi di Resource Explorer utilizzando AWS CLI
- Esempi di Resource Groups utilizzando AWS CLI
- Esempi di API di etichettatura dei Resource Groups utilizzando AWS CLI
- Esempi di Route 53 utilizzando AWS CLI
- Esempi di registrazione del dominio Route 53 utilizzando AWS CLI
- Esempi di profili Route 53 utilizzando AWS CLI
- Esempi di Route 53 Resolver che utilizzano AWS CLI
- Esempi di utilizzo di Amazon S3 AWS CLI
- Esempi di Amazon S3 Control con AWS CLI
- Esempi di Secrets Manager che utilizzano AWS CLI
- Esempi di CSPM di Security Hub utilizzando AWS CLI
- Esempi di utilizzo di Security Lake AWS CLI
- AWS Serverless Application Repository esempi utilizzando AWS CLI
- Esempi di Service Catalog utilizzando AWS CLI
- Esempi di Service Quotas utilizzando AWS CLI
- Esempi di utilizzo di Amazon SES AWS CLI
- Esempi di utilizzo di Shield AWS CLI
- Esempi di firmatari che utilizzano AWS CLI
- Esempi di utilizzo di Snowball Edge AWS CLI
- Esempi di utilizzo di Amazon SNS AWS CLI
- Esempi di utilizzo di Amazon SQS AWS CLI
- Esempi di Storage Gateway utilizzando AWS CLI

- AWS STS esempi utilizzando AWS CLI
- Supporto esempi utilizzando AWS CLI
- Esempi di Amazon SWF con AWS CLI
- Esempi di Systems Manager che utilizzano AWS CLI
- Esempi di utilizzo di Amazon Textract AWS CLI
- Esempi di Amazon Transcribe utilizzando AWS CLI
- Esempi di Amazon Translate con AWS CLI
- Trusted Advisor esempi utilizzando AWS CLI
- Esempi di autorizzazioni verificate utilizzando AWS CLI
- Esempi di VPC Lattice utilizzando AWS CLI
- AWS WAF Classic esempi utilizzando AWS CLI
- AWS WAF Classic regionale esempi utilizzando AWS CLI
- AWS WAFV2 esempi utilizzando AWS CLI
- WorkDocs esempi utilizzando AWS CLI
- WorkMail Esempi di utilizzo di Amazon AWS CLI
- Esempi di Amazon WorkMail Message Flow con AWS CLI
- WorkSpaces esempi utilizzando AWS CLI
- Esempi di raggi X utilizzando AWS CLI

Esempi di utilizzo di ACM AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with ACM.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`add-tags-to-certificate`

Il seguente esempio di codice mostra come utilizzare`add-tags-to-certificate`.

AWS CLI

Come aggiungere tag a un certificato ACM esistente

Il comando `add-tags-to-certificate` seguente aggiunge due tag al certificato specificato. Usare uno spazio per separare più tag:

```
aws acm add-tags-to-certificate --certificate-
arn arn:aws:acm:region:account:certificate/12345678-1234-1234-1234-123456789012 --
tags Key=Admin,Value=Alice Key=Purpose,Value=Website
```

- Per i dettagli sull'API, consulta AddTagsToCertificate AWS CLICommand Reference.

`delete-certificate`

Il seguente esempio di codice mostra come utilizzare`delete-certificate`.

AWS CLI

Come eliminare un certificato ACM da un account

Il comando `delete-certificate` seguente elimina il certificato con il nome della risorsa Amazon (ARN) specificato:

```
aws acm delete-certificate --certificate-
arn arn:aws:acm:region:account:certificate/12345678-1234-1234-1234-123456789012
```

- Per i dettagli sull'API, consulta DeleteCertificate AWS CLICommand Reference.

`describe-certificate`

Il seguente esempio di codice mostra come utilizzare`describe-certificate`.

AWS CLI

Come recuperare i campi contenuti in un certificato ACM

Il comando `describe-certificate` seguente recupera tutti i campi del certificato con l'ARN specificato:

```
aws acm describe-certificate --certificate-
arn arn:aws:acm:region:account:certificate/12345678-1234-1234-1234-123456789012
```

Viene visualizzato un output simile a quello riportato di seguito:

```
{
  "Certificate": {
    "CertificateArn":
 "arn:aws:acm:region:account:certificate/12345678-1234-1234-1234-123456789012",
    "CreatedAt": 1446835267.0,
    "DomainName": "www.example.com",
    "DomainValidationOptions": [
      {
        "DomainName": "www.example.com",
        "ValidationDomain": "www.example.com",
        "ValidationEmails": [
          "hostmaster@example.com",
          "admin@example.com",
          "owner@example.com.whoisprivacyservice.org",
          "tech@example.com.whoisprivacyservice.org",
          "admin@example.com.whoisprivacyservice.org",
          "postmaster@example.com",
          "webmaster@example.com",
          "administrator@example.com"
        ]
      },
      {
        "DomainName": "www.example.net",
        "ValidationDomain": "www.example.net",
        "ValidationEmails": [
          "postmaster@example.net",
          "admin@example.net",
          "owner@example.net.whoisprivacyservice.org",
          "tech@example.net.whoisprivacyservice.org",
          "admin@example.net.whoisprivacyservice.org",
          "hostmaster@example.net",
          "administrator@example.net",
          "webmaster@example.net"
        ]
      }
```

```
    ],
    "InUseBy": [],
    "IssuedAt": 1446835815.0,
    "Issuer": "Amazon",
    "KeyAlgorithm": "RSA-2048",
    "NotAfter": 1478433600.0,
    "NotBefore": 1446768000.0,
    "Serial": "0f:ac:b0:a3:8d:ea:65:52:2d:7d:01:3a:39:36:db:d6",
    "SignatureAlgorithm": "SHA256WITHRSA",
    "Status": "ISSUED",
    "Subject": "CN=www.example.com",
    "SubjectAlternativeNames": [
      "www.example.com",
      "www.example.net"
    ]
  }
}
```

- Per i dettagli sull'API, consulta DescribeCertificate AWS CLICommand Reference.

`export-certificate`

Il seguente esempio di codice mostra come utilizzare`export-certificate`.

AWS CLI

Come esportare un certificato privato emesso un'autorità di certificazione privata.

Il comando `export-certificate` seguente esporta un certificato privato, una catena di certificati e una chiave privata sullo schermo:

```
aws acm export-certificate --certificate-
arn arn:aws:acm:region:account:certificate/12345678-1234-1234-1234-123456789012 --
passphrase file://path-to-passphrase-file
```

Per esportare il certificato, la catena e la chiave privata in un file locale, utilizza il seguente comando:

```
aws acm export-certificate --certificate-
arn arn:aws:acm:region:sccount:certificate/12345678-1234-1234-1234-123456789012 --
passphrase file://path-to-passphrase-file > c:\temp\export.txt
```

- Per i dettagli sull'API, consulta ExportCertificate AWS CLICommand Reference.

`get-certificate`

Il seguente esempio di codice mostra come utilizzare`get-certificate`.

AWS CLI

Come recuperare un certificato ACM

Il comando `get-certificate` seguente recupera il certificato per l'ARN e la catena di certificati specificati:

```
aws acm get-certificate --certificate-
arn arn:aws:acm:region:account:certificate/12345678-1234-1234-1234-123456789012
```

Viene visualizzato un output simile a quello riportato di seguito:

```
{
  "Certificate": "-----BEGIN CERTIFICATE-----
MIICiTCCAfICCQD6m7oRw0uXOjANBgkqhkiG9w0BAQUFADCBiDELMAkGA1UEBhMC
VVMxCzAJBgNVBAgTAldBMRAwDgYDVQQHEwdTZWF0dGxlMQ8wDQYDVQQKEwZBbWF6
b24xFDASBgNVBAsTC0lBTSBDb25zb2xlMRIwEAYDVQQDEwlUZXN0Q2lsYWMxHzAd
BgkqhkiG9w0BCQEWEG5vb25lQGFtYXpvbi5jb20wHhcNMTEwNDI1MjA0NTIxWhcN
MTIwNDI0MjA0NTIxWjCBiDELMAkGA1UEBhMCVVMxCzAJBgNVBAgTAldBMRAwDgYD
VQQHEwdTZWF0dGxlMQ8wDQYDVQQKEwZBbWF6b24xFDASBgNVBAsTC0lBTSBDb25z
b2xlMRIwEAYDVQQDEwlUZXN0Q2lsYWMxHzAdBgkqhkiG9w0BCQEWEG5vb25lQGFt
YXpvbi5jb20wgZ8wDQYJKoZIhvcNAQEBBQADgY0AMIGJAoGBAMaK0dn+a4GmWIWJ
21uUSfwfEvySWtC2XADZ4nB+BLYgVIk60CpiwsZ3G93vUEIO3IyNoH/f0wYK8m9T
rDHudUZg3qX4waLG5M43q7Wgc/MbQITxOUSQv7c7ugFFDzQGBzZswY6786m86gpE
Ibb3OhjZnzcvQAaRHhdlQWIMm2nrAgMBAAEwDQYJKoZIhvcNAQEFBQADgYEAtCu4
nUhVVxYUntneD9+h8Mg9q6q+auNKyExzyLwaxlAoo7TJHidbtS4J5iNmZgXL0Fkb
FFBjvSfpJIlJ00zbhNYS5f6GuoEDmFJl0ZxBHjJnyp378OD8uTs7fLvjx79LjSTb
NYiytVbZPQUQ5Yaxu2jXnimvw3rrszlaEXAMPLE=
-----END CERTIFICATE-----",

  "CertificateChain": "-----BEGIN CERTIFICATE-----
MIICiTCCAfICCQD6m7oRw0uXOjANBgkqhkiG9w0BAQUFADCBiDELMAkGA1UEBhMC
VVMxCzAJBgNVBAgTAldBMRAwDgYDVQQHEwdTZWF0dGxlMQ8wDQYDVQQKEwZBbWF6
b24xFDASBgNVBAsTC0lBTSBDb25zb2xlMRIwEAYDVQQDEwlUZXN0Q2lsYWMxHzAd
BgkqhkiG9w0BCQEWEG5vb25lQGFtYXpvbi5jb20wHhcNMTEwNDI1MjA0NTIxWhcN
MTIwNDI0MjA0NTIxWjCBiDELMAkGA1UEBhMCVVMxCzAJBgNVBAgTAldBMRAwDgYD
VQQHEwdTZWF0dGxlMQ8wDQYDVQQKEwZBbWF6b24xFDASBgNVBAsTC0lBTSBDb25z
```

```
b2xlMRIwEAYDVQQDEwlUZXN0Q2lsYWMxHzAdBgkqhkiG9w0BCQEWEG5vb25lQGFt
YXpvbi5jb20wgZ8wDQYJKoZIhvcNAQEBBQADgY0AMIGJAoGBAMaK0dn+a4GmWIWJ
21uUSfwfEvySWtC2XADZ4nB+BLYgVIk60CpiwsZ3G93vUEIO3IyNoH/f0wYK8m9T
rDHudUZg3qX4waLG5M43q7Wgc/MbQITxOUSQv7c7ugFFDzQGBzZswY6786m86gpE
Ibb3OhjZnzcvQAaRHhdlQWIMm2nrAgMBAAEwDQYJKoZIhvcNAQEFBQADgYEAtCu4
nUhVVxYUntneD9+h8Mg9q6q+auNKyExzyLwaxlAoo7TJHidbtS4J5iNmZgXL0Fkb
FFBjvSfpJIlJ00zbhNYS5f6GuoEDmFJl0ZxBHjJnyp378OD8uTs7fLvjx79LjSTb
NYiytVbZPQUQ5Yaxu2jXnimvw3rrszlaEXAMPLE=
-----END CERTIFICATE-----",
"-----BEGIN CERTIFICATE-----
MIICiTCCAfICCQD6m7oRw0uXOjANBgkqhkiG9w0BAQUFADCBiDELMAkGA1UEBhMC
VVMxCzAJBgNVBAgTAldBMRAwDgYDVQQHEwdTZWF0dGxlMQ8wDQYDVQQKEwZBbWF6
b24xFDASBgNVBAsTC0lBTSBDb25zb2xlMRIwEAYDVQQDEwlUZXN0Q2lsYWMxHzAd
BgkqhkiG9w0BCQEWEG5vb25lQGFtYXpvbi5jb20wHhcNMTEwNDI1MjA0NTIxWhcN
MTIwNDI0MjA0NTIxWjCBiDELMAkGA1UEBhMCVVMxCzAJBgNVBAgTAldBMRAwDgYD
VQQHEwdTZWF0dGxlMQ8wDQYDVQQKEwZBbWF6b24xFDASBgNVBAsTC0lBTSBDb25z
b2xlMRIwEAYDVQQDEwlUZXN0Q2lsYWMxHzAdBgkqhkiG9w0BCQEWEG5vb25lQGFt
YXpvbi5jb20wgZ8wDQYJKoZIhvcNAQEBBQADgY0AMIGJAoGBAMaK0dn+a4GmWIWJ
21uUSfwfEvySWtC2XADZ4nB+BLYgVIk60CpiwsZ3G93vUEIO3IyNoH/f0wYK8m9T
rDHudUZg3qX4waLG5M43q7Wgc/MbQITxOUSQv7c7ugFFDzQGBzZswY6786m86gpE
Ibb3OhjZnzcvQAaRHhdlQWIMm2nrAgMBAAEwDQYJKoZIhvcNAQEFBQADgYEAtCu4
nUhVVxYUntneD9+h8Mg9q6q+auNKyExzyLwaxlAoo7TJHidbtS4J5iNmZgXL0Fkb
FFBjvSfpJIlJ00zbhNYS5f6GuoEDmFJl0ZxBHjJnyp378OD8uTs7fLvjx79LjSTb
NYiytVbZPQUQ5Yaxu2jXnimvw3rrszlaEXAMPLE=
-----END CERTIFICATE-----",
"-----BEGIN CERTIFICATE-----
MIICiTCCAfICCQD6m7oRw0uXOjANBgkqhkiG9w0BAQUFADCBiDELMAkGA1UEBhMC
VVMxCzAJBgNVBAgTAldBMRAwDgYDVQQHEwdTZWF0dGxlMQ8wDQYDVQQKEwZBbWF6
b24xFDASBgNVBAsTC0lBTSBDb25zb2xlMRIwEAYDVQQDEwlUZXN0Q2lsYWMxHzAd
BgkqhkiG9w0BCQEWEG5vb25lQGFtYXpvbi5jb20wHhcNMTEwNDI1MjA0NTIxWhcN
MTIwNDI0MjA0NTIxWjCBiDELMAkGA1UEBhMCVVMxCzAJBgNVBAgTAldBMRAwDgYD
VQQHEwdTZWF0dGxlMQ8wDQYDVQQKEwZBbWF6b24xFDASBgNVBAsTC0lBTSBDb25z
b2xlMRIwEAYDVQQDEwlUZXN0Q2lsYWMxHzAdBgkqhkiG9w0BCQEWEG5vb25lQGFt
YXpvbi5jb20wgZ8wDQYJKoZIhvcNAQEBBQADgY0AMIGJAoGBAMaK0dn+a4GmWIWJ
21uUSfwfEvySWtC2XADZ4nB+BLYgVIk60CpiwsZ3G93vUEIO3IyNoH/f0wYK8m9T
rDHudUZg3qX4waLG5M43q7Wgc/MbQITxOUSQv7c7ugFFDzQGBzZswY6786m86gpE
Ibb3OhjZnzcvQAaRHhdlQWIMm2nrAgMBAAEwDQYJKoZIhvcNAQEFBQADgYEAtCu4
nUhVVxYUntneD9+h8Mg9q6q+auNKyExzyLwaxlAoo7TJHidbtS4J5iNmZgXL0Fkb
FFBjvSfpJIlJ00zbhNYS5f6GuoEDmFJl0ZxBHjJnyp378OD8uTs7fLvjx79LjSTb
NYiytVbZPQUQ5Yaxu2jXnimvw3rrszlaEXAMPLE=
-----END CERTIFICATE-----"
}
```

- Per i dettagli sull'API, consulta [GetCertificate AWS CLI](Command Reference).

`import-certificate`

Il seguente esempio di codice mostra come utilizzare`import-certificate`.

AWS CLI

Come importare un certificato in ACM.

Il comando `import-certificate` seguente importa un certificato in ACM. Sostituire i nomi di file con i nomi desiderati.

```
aws acm import-certificate --certificate file://Certificate.pem --certificate-
chain file://CertificateChain.pem --private-key file://PrivateKey.pem
```

- Per i dettagli sull'API, consulta ImportCertificate AWS CLICommand Reference.

`list-certificates`

Il seguente esempio di codice mostra come utilizzare`list-certificates`.

AWS CLI

Per elencare i certificati ACM per un AWS account

Il `list-certificates` comando seguente elenca ARNs i certificati presenti nel tuo account:

```
aws acm list-certificates
```

Il comando precedente genera un output simile al seguente:

```
{
    "CertificateSummaryList": [
        {
            "CertificateArn":
 "arn:aws:acm:region:account:certificate/12345678-1234-1234-1234-123456789012",
            "DomainName": "www.example.com"
        },
        {
            "CertificateArn": "arn:aws:acm:region:account:certificate/aaaaaaaa-bbbb-
cccc-dddd-eeeeeeeeeeee",
            "DomainName": "www.example.net"
```

```
        }
    ]
}
```

È possibile decidere quanti certificati visualizzare ogni volta che viene chiamato il comando `list-certificates`. Ad esempio, se sono presenti quattro certificati ma si desidera visualizzarne non più di due alla volta, imposta l'argomento `max-items` su 2 come nell'esempio seguente:

```
aws acm list-certificates --max-items 2
```

Verranno visualizzati due certificati ARNs e un `NextToken` valore:

```
"CertificateSummaryList": [
  {
    "CertificateArn": "arn:aws:acm:region:account: \
            certificate/12345678-1234-1234-1234-123456789012",
    "DomainName": "www.example.com"
  },
  {
    "CertificateArn": "arn:aws:acm:region:account: \
             certificate/aaaaaaaa-bbbb-cccc-dddd-eeeeeeeeeeee",
    "DomainName": "www.example.net"
  }
  ],
    "NextToken": "9f4d9f69-275a-41fe-b58e-2b837bd9ba48"
```

Per visualizzare i due certificati successivi nell'account, imposta il valore `NextToken` nella prossima chiamata:

```
aws acm list-certificates --max-items 2 --next-token 9f4d9f69-275a-41fe-
b58e-2b837bd9ba48
```

È possibile filtrare l'output utilizzando l'argomento `certificate-statuses`. Il comando seguente visualizza i certificati il cui stato è PENDING_VALIDATION:

```
aws acm list-certificates --certificate-statuses PENDING_VALIDATION
```

È inoltre possibile filtrare l'output utilizzando l'argomento `includes`. Il comando seguente visualizza i certificati filtrati in base alle seguenti proprietà. I certificati da visualizzare:

```
- Specify that the RSA algorithm and a 2048 bit key are used to generate key pairs.
- Contain a Key Usage extension that specifies that the certificates can be used to
 create digital signatures.
- Contain an Extended Key Usage extension that specifies that the certificates can
 be used for code signing.

aws acm list-certificates --max-items 10 --includes
 extendedKeyUsage=CODE_SIGNING,keyUsage=DIGITAL_SIGNATURE,keyTypes=RSA_2048
```

- Per i dettagli sull'API, consulta ListCertificates AWS CLICommand Reference.

`list-tags-for-certificate`

Il seguente esempio di codice mostra come utilizzare`list-tags-for-certificate`.

AWS CLI

Come elencare i tag applicati a un certificato ACM

Il comando `list-tags-for-certificate` seguente elenca i tag applicati a un certificato presente nell'account:

```
aws acm list-tags-for-certificate --certificate-
arn arn:aws:acm:region:account:certificate/12345678-1234-1234-1234-123456789012
```

Il comando precedente genera un output simile al seguente:

```
{
  "Tags": [
      {
          "Value": "Website",
          "Key": "Purpose"
      },
      {
          "Value": "Alice",
          "Key": "Admin"
      }
  ]
}
```

- Per i dettagli sull'API, consulta ListTagsForCertificate AWS CLICommand Reference.

`remove-tags-from-certificate`

Il seguente esempio di codice mostra come utilizzare`remove-tags-from-certificate`.

AWS CLI

Come rimuovere un tag da un certificato ACM

Il comando `remove-tags-from-certificate` seguente rimuove due tag dal certificato specificato. Usare uno spazio per separare più tag:

```
aws acm remove-tags-from-certificate --certificate-
arn arn:aws:acm:region:account:certificate/12345678-1234-1234-1234-123456789012 --
tags Key=Admin,Value=Alice Key=Purpose,Value=Website
```

- Per i dettagli sull'API, consulta RemoveTagsFromCertificate AWS CLICommand Reference.

`request-certificate`

Il seguente esempio di codice mostra come utilizzare`request-certificate`.

AWS CLI

Come richiedere un nuovo certificato ACM

Il comando `request-certificate` seguente richiede un nuovo certificato per il dominio www.example.com utilizzando la convalida DNS:

```
aws acm request-certificate --domain-name www.example.com --validation-method DNS
```

Nel comando `request-certificate` è possibile includere un token di idempotenza per distinguere le chiamate:

```
aws acm request-certificate --domain-name www.example.com --validation-method DNS --
idempotency-token 91adc45q
```

È possibile inserire uno o più nomi di soggetto alternativi per richiedere un certificato che protegga più di un dominio apex:

```
aws acm request-certificate --domain-name example.com --validation-method DNS --
idempotency-token 91adc45q --subject-alternative-names www.example.net
```

È possibile inserire un nome alternativo che può essere utilizzato anche per raggiungere un sito web:

```
aws acm request-certificate --domain-name example.com --validation-method DNS --
idempotency-token 91adc45q --subject-alternative-names www.example.com
```

È possibile utilizzare un asterisco (*) come carattere jolly per creare un certificato per diversi sottodomini dello stesso dominio:

```
aws acm request-certificate --domain-name example.com --validation-method DNS --
idempotency-token 91adc45q --subject-alternative-names *.example.com
```

È anche possibile inserire più nomi alternativi:

```
aws acm request-certificate --domain-name example.com --validation-method DNS --
subject-alternative-names b.example.com c.example.com d.example.com
```

Se per la convalida viene utilizzata l'e-mail, è possibile inserire le opzioni di convalida del dominio per specificare il dominio a cui verrà inviata l'e-mail di convalida:

```
aws acm request-certificate --domain-name example.com --validation-
method EMAIL --subject-alternative-names www.example.com --domain-validation-
options DomainName=example.com,ValidationDomain=example.com
```

Il comando seguente disattiva la registrazione di log della trasparenza dei certificati se si richiede un nuovo certificato:

```
aws acm request-certificate --domain-name www.example.com --validation-method DNS --
options CertificateTransparencyLoggingPreference=DISABLED --idempotency-token 184627
```

- Per i dettagli sull'API, consulta RequestCertificate AWS CLICommand Reference.

`resend-validation-email`

Il seguente esempio di codice mostra come utilizzare`resend-validation-email`.

AWS CLI

Come inviare nuovamente l'e-mail di convalida per la richiesta del certificato ACM

Il comando `resend-validation-email` seguente indica all'autorità di certificazione Amazon di inviare l'e-mail di convalida agli indirizzi appropriati:

```
aws acm resend-validation-email --certificate-
arn arn:aws:acm:region:account:certificate/12345678-1234-1234-1234-123456789012 --
domain www.example.com --validation-domain example.com
```

- Per i dettagli sull'API, consulta ResendValidationEmail AWS CLICommand Reference.

`update-certificate-options`

Il seguente esempio di codice mostra come utilizzare`update-certificate-options`.

AWS CLI

Come aggiornare le opzioni del certificato

Il comando `update-certificate-options` seguente disattiva la registrazione dei log della trasparenza dei certificati:

```
aws acm update-certificate-options --certificate-
arn arn:aws:acm:region:account:certificate/12345678-1234-1234-1234-123456789012 --
options CertificateTransparencyLoggingPreference=DISABLED
```

- Per i dettagli sull'API, consulta UpdateCertificateOptions AWS CLICommand Reference.

Esempi di API Gateway utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with API Gateway.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`create-api-key`

Il seguente esempio di codice mostra come utilizzare`create-api-key`.

AWS CLI

Come creare una chiave API abilitata per un'API e una fase esistenti

Comando:

```
aws apigateway create-api-key --name 'Dev API Key' --description 'Used for
 development' --enabled --stage-keys restApiId='a1b2c3d4e5',stageName='dev'
```

- Per i dettagli sull'API, consulta CreateApiKey AWS CLICommand Reference.

`create-authorizer`

Il seguente esempio di codice mostra come utilizzare`create-authorizer`.

AWS CLI

Esempio 1: come creare un sistema di autorizzazione personalizzato di Gateway API basato su token per l'API

L'esempio `create-authorizer` seguente crea un sistema di autorizzazione basato su token.

```
aws apigateway create-authorizer \
    --rest-api-id 1234123412 \
    --name 'First_Token_Custom_Authorizer' \
    --type TOKEN \
    --authorizer-uri 'arn:aws:apigateway:us-west-2:lambda:path/2015-03-31/functions/
arn:aws:lambda:us-west-2:123412341234:function:customAuthFunction/invocations' \
    --identity-source 'method.request.header.Authorization' \
    --authorizer-result-ttl-in-seconds 300
```

Output:

```
{
    "authType": "custom",
    "name": "First_Token_Custom_Authorizer",
```

```
    "authorizerUri": "arn:aws:apigateway:us-west-2:lambda:path/2015-03-31/functions/
arn:aws:lambda:us-west-2:123412341234:function:customAuthFunction/invocations",
    "authorizerResultTtlInSeconds": 300,
    "identitySource": "method.request.header.Authorization",
    "type": "TOKEN",
    "id": "z40xj0"
}
```

Esempio 2: come creare un sistema di autorizzazione personalizzato di Gateway API basato su pool di utenti Cognito per l'API

L'esempio `create-authorizer` seguente crea un sistema di autorizzazione personalizzato per Gateway API basato su pool di utenti Cognito.

```
aws apigateway create-authorizer \
    --rest-api-id 1234123412 \
    --name 'First_Cognito_Custom_Authorizer' \
    --type COGNITO_USER_POOLS \
    --provider-arns 'arn:aws:cognito-idp:us-east-1:123412341234:userpool/us-
east-1_aWcZeQbuD' \
    --identity-source 'method.request.header.Authorization'
```

Output:

```
{
    "authType": "cognito_user_pools",
    "identitySource": "method.request.header.Authorization",
    "name": "First_Cognito_Custom_Authorizer",
    "providerARNs": [
        "arn:aws:cognito-idp:us-east-1:342398297714:userpool/us-east-1_qWbZzQhzE"
    ],
    "type": "COGNITO_USER_POOLS",
    "id": "5yid1t"
}
```

Esempio 3: come creare un sistema di autorizzazione personalizzato di Gateway API basato su richiesta per l'API

L'esempio `create-authorizer` seguente crea un sistema di autorizzazione basato su richiesta.

```
aws apigateway create-authorizer \
```

```
    --rest-api-id 1234123412 \
    --name 'First_Request_Custom_Authorizer' \
    --type REQUEST \
    --authorizer-uri 'arn:aws:apigateway:us-west-2:lambda:path/2015-03-31/functions/
arn:aws:lambda:us-west-2:123412341234:function:customAuthFunction/invocations' \
    --identity-source 'method.request.header.Authorization,context.accountId' \
    --authorizer-result-ttl-in-seconds 300
```

Output:

```
{
    "id": "z40xj0",
    "name": "First_Request_Custom_Authorizer",
    "type": "REQUEST",
    "authType": "custom",
    "authorizerUri": "arn:aws:apigateway:us-west-2:lambda:path/2015-03-31/functions/
arn:aws:lambda:us-west-2:123412341234:function:customAuthFunction/invocations",
    "identitySource": "method.request.header.Authorization,context.accountId",
    "authorizerResultTtlInSeconds": 300
}
```

- Per i dettagli sull'API, consulta CreateAuthorizer AWS CLICommand Reference.

`create-base-path-mapping`

Il seguente esempio di codice mostra come utilizzare`create-base-path-mapping`.

AWS CLI

Come creare la mappatura del percorso di base per un nome di dominio personalizzato

Comando:

```
aws apigateway create-base-path-mapping --domain-name subdomain.domain.tld --rest-
api-id 1234123412 --stage prod --base-path v1
```

- Per i dettagli sull'API, consulta CreateBasePathMapping AWS CLICommand Reference.

`create-deployment`

Il seguente esempio di codice mostra come utilizzare`create-deployment`.

AWS CLI

Come implementare le risorse configurate per un'API in una nuova fase

Comando:

```
aws apigateway create-deployment --rest-api-id 1234123412 --stage-name dev --stage-
description 'Development Stage' --description 'First deployment to the dev stage'
```

Come implementare le risorse configurate per un'API in una fase esistente

Comando:

```
aws apigateway create-deployment --rest-api-id 1234123412 --stage-name dev --
description 'Second deployment to the dev stage'
```

Come implementare le risorse configurate per un'API in una fase esistente con variabili di fase

aws apigateway create-deployment -- rest-api-id 1234123412 --stage-name dev --description 'Terza implementazione nella fase di sviluppo' --variables key='value', otherKey='otherValue'

- Per i dettagli sull'API CreateDeployment, vedi AWS CLI in Command Reference.

`create-domain-name-access-association`

Il seguente esempio di codice mostra come utilizzare`create-domain-name-access-association`.

AWS CLI

Come creare un'associazione di accesso al nome di dominio

L'esempio `create-domain-name-access-association` seguente crea un'associazione di accesso al nome di dominio tra un nome di dominio privato personalizzato e l'endpoint VPC.

```
aws apigateway create-domain-name-access-association \
    --domain-name-arn arn:aws:apigateway:us-west-2:111122223333:/domainnames/
my.private.domain.tld+abcd1234 \
    --access-association-source vpce-abcd1234efg \
    --access-association-source-type VPCE
```

Output:

```
{
    "domainNameAccessAssociationArn": "arn:aws:apigateway:us-west-2:012345678910:/
domainnameaccessassociations/domainname/my.private.domain.tld/vpcesource/vpce-
abcd1234efg
    "accessAssociationSource": "vpce-abcd1234efg",
    "accessAssociationSourceType": "VPCE",
    "domainNameArn" : "arn:aws:apigateway:us-west-2:111122223333:/domainnames/
private.example.com+abcd1234"
}
```

Per ulteriori informazioni, consulta Nomi di dominio personalizzati per uso privato APIs in API Gateway nella Amazon API Gateway Developer Guide.

- Per i dettagli sull'API, consulta CreateDomainNameAccessAssociation AWS CLICommand Reference.

`create-domain-name`

Il seguente esempio di codice mostra come utilizzare`create-domain-name`.

AWS CLI

Esempio 1: come creare un nome di dominio pubblico personalizzato

L'esempio `create-domain-name` seguente crea un nome di dominio pubblico personalizzato.

```
aws apigateway create-domain-name \
    --domain-name 'my.domain.tld' \
    --certificate-name 'my.domain.tld cert'\
    --certificate-arn 'arn:aws:acm:us-east-1:012345678910:certificate/fb1b9770-
a305-495d-aefb-27e5e101ff3'
```

Output:

```
{
    "domainName": "my.domain.tld",
    "certificateName": "my.domain.tld cert",
    "certificateArn": "arn:aws:acm:us-east-1:012345678910:certificate/fb1b9770-
a305-495d-aefb-27e5e101ff3",
```

```
    "certificateUploadDate": "2024-10-08T11:29:49-07:00",
    "distributionDomainName": "abcd1234.cloudfront.net",
    "distributionHostedZoneId": "Z2FDTNDATAQYW2",
    "endpointConfiguration": {
        "types": [
            "EDGE"
        ]
    },
    "domainNameStatus": "AVAILABLE",
    "securityPolicy": "TLS_1_2"
}
```

Per ulteriori informazioni, consulta la sezione Nome di dominio personalizzato per REST pubblico APIs in API Gateway nella Amazon API Gateway Developer Guide.

Esempio 2: come creare un nome di dominio privato personalizzato

L'esempio `create-domain-name` seguente crea un nome di dominio privato personalizzato.

```
aws apigateway create-domain-name \
    --domain-name 'my.private.domain.tld' \
    --certificate-name 'my.domain.tld cert' \
    --certificate-arn 'arn:aws:acm:us-east-1:012345678910:certificate/fb1b9770-
a305-495d-aefb-27e5e101ff3' \
    --endpoint-configuration '{"types": ["PRIVATE"]}' \
    --security-policy 'TLS_1_2' \
    --policy file://policy.json
```

Contenuto di `policy.json`:

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": "*",
            "Action": "execute-api:Invoke",
            "Resource": [
                "execute-api:/*"
            ]
        },
        {
```

```
            "Effect": "Deny",
            "Principal": "*",
            "Action": "execute-api:Invoke",
            "Resource": [
                "execute-api:/*"
            ],
            "Condition" : {
                "StringNotEquals": {
                    "aws:SourceVpce": "vpce-abcd1234efg"
                }
            }
        }
    ]
}
```

Output:

```
{
    "domainName": "my.private.domain.tld",
    "domainNameId": "abcd1234",
    "domainNameArn": "arn:aws:apigateway:us-east-1:012345678910:/domainnames/
my.private.domain.tld+abcd1234",
    "certificateArn": "arn:aws:acm:us-east-1:012345678910:certificate/fb1b9770-
a305-495d-aefb-27e5e101ff3",
    "certificateUploadDate": "2024-09-10T10:31:20-07:00",
    "endpointConfiguration": {
        "types": [
            "PRIVATE"
        ]
    },
    "domainNameStatus": "AVAILABLE",
    "securityPolicy": "TLS_1_2",
    "policy": "{\"Version\":\"2012-10-17\",\"Statement\":[{\"Effect\":
\"Allow\",\"Principal\":\"*\",\"Action\":\"execute-api:Invoke\",\"Resource\":
\"arn:aws:execute-api:us-east-1:012345678910:/domainnames/my.private.domain.tld
+abcd1234\"},{\"Effect\":\"Deny\",\"Principal\":\"*\",\"Action\":\"execute-
api:Invoke\",\"Resource\":\"arn:aws:execute-api:us-east-1:012345678910:/domainnames/
my.private.domain.tld+abcd1234\",\"Condition\":{\"StringNotEquals\":{\"aws:SourceVpc
\":\"vpc-1a2b3c4d\"}}}]}"
}
```

Per ulteriori informazioni, consulta la sezione Nome di dominio personalizzato per REST pubblico APIs in API Gateway nella Amazon API Gateway Developer Guide.

- Per i dettagli sull'API, consulta CreateDomainName AWS CLICommand Reference.

`create-model`

Il seguente esempio di codice mostra come utilizzare`create-model`.

AWS CLI

Come creare un modello per un'API

Comando:

```
aws apigateway create-model --rest-api-id 1234123412 --name 'firstModel' --
description 'The First Model' --content-type 'application/json'  --schema
 '{ "$schema": "http://json-schema.org/draft-04/schema#", "title": "firstModel",
 "type": "object", "properties": { "firstProperty" : { "type": "object",
 "properties": { "key": { "type": "string" } } } } }'
```

Output:

```
{
    "contentType": "application/json",
    "description": "The First Model",
    "name": "firstModel",
    "id": "2rzg0l",
    "schema": "{ \"$schema\": \"http://json-schema.org/draft-04/schema#\", \"title
\": \"firstModel\", \"type\": \"object\", \"properties\": { \"firstProperty
\" : { \"type\": \"object\", \"properties\": { \"key\": { \"type\": \"string
\" } } } } }"
}
```

- Per i dettagli sull'API, consulta CreateModel AWS CLICommand Reference.

`create-resource`

Il seguente esempio di codice mostra come utilizzare`create-resource`.

AWS CLI

Come creare una risorsa in un'API

Comando:

```
aws apigateway create-resource --rest-api-id 1234123412 --parent-id a1b2c3 --path-
part 'new-resource'
```

- Per i dettagli sull'API, consulta CreateResource AWS CLICommand Reference.

`create-rest-api`

Il seguente esempio di codice mostra come utilizzare`create-rest-api`.

AWS CLI

Come creare un'API

Comando:

```
aws apigateway create-rest-api --name 'My First API' --description 'This is my first
 API'
```

Come creare un'API duplicata da un'API esistente

Comando:

```
aws apigateway create-rest-api --name 'Copy of My First API' --description 'This is
 a copy of my first API' --clone-from 1234123412
```

- Per i dettagli sull'API, consulta CreateRestApi AWS CLICommand Reference.

`create-stage`

Il seguente esempio di codice mostra come utilizzare`create-stage`.

AWS CLI

Come creare una fase in un'API che conterrà un'implementazione esistente

Comando:

```
aws apigateway create-stage --rest-api-id 1234123412 --stage-name 'dev' --
description 'Development stage' --deployment-id a1b2c3
```

Come creare una fase in un'API che conterrà un'implementazione esistente e variabili di fase personalizzate

Comando:

```
aws apigateway create-stage --rest-api-id 1234123412 --stage-name 'dev'
 --description 'Development stage' --deployment-id a1b2c3 --variables
 key='value',otherKey='otherValue'
```

- Per i dettagli sull'API, consulta CreateStage AWS CLICommand Reference.

`create-usage-plan-key`

Il seguente esempio di codice mostra come utilizzare`create-usage-plan-key`.

AWS CLI

Come associare una chiave API esistente a un piano di utilizzo

Comando:

```
aws apigateway create-usage-plan-key --usage-plan-id a1b2c3 --key-type "API_KEY" --
key-id 4vq3yryqm5
```

- Per i dettagli sull'API, consulta CreateUsagePlanKey AWS CLICommand Reference.

`create-usage-plan`

Il seguente esempio di codice mostra come utilizzare`create-usage-plan`.

AWS CLI

Come creare un piano di utilizzo con limiti di quota e di velocità di elaborazione delle richieste che viene reimpostato a inizio mese

Comando:

```
aws apigateway create-usage-plan --name "New Usage Plan" --description "A new usage
 plan" --throttle burstLimit=10,rateLimit=5 --quota limit=500,offset=0,period=MONTH
```

- Per i dettagli sull'API, consulta CreateUsagePlan AWS CLICommand Reference.

`delete-api-key`

Il seguente esempio di codice mostra come utilizzare`delete-api-key`.

AWS CLI

Come eliminare una chiave API

Comando:

```
aws apigateway delete-api-key --api-key 8bklk8bl1k3sB38D9B3l0enyWT8c09B30lkq0blk
```

- Per i dettagli sull'API, consulta DeleteApiKey AWS CLICommand Reference.

`delete-authorizer`

Il seguente esempio di codice mostra come utilizzare`delete-authorizer`.

AWS CLI

Come eliminare un sistema di autorizzazione personalizzato in un'API

Comando:

```
aws apigateway delete-authorizer --rest-api-id 1234123412 --authorizer-id 7gkfbo
```

- Per i dettagli sull'API, consulta DeleteAuthorizer AWS CLICommand Reference.

`delete-base-path-mapping`

Il seguente esempio di codice mostra come utilizzare`delete-base-path-mapping`.

AWS CLI

Come eliminare una mappatura del percorso di base per un nome di dominio personalizzato

Comando:

```
aws apigateway delete-base-path-mapping --domain-name 'api.domain.tld' --base-path
 'dev'
```

- Per i dettagli sull'API, consulta DeleteBasePathMapping AWS CLICommand Reference.

`delete-client-certificate`

Il seguente esempio di codice mostra come utilizzare`delete-client-certificate`.

AWS CLI

Come eliminare un certificato client

Comando:

```
aws apigateway delete-client-certificate --client-certificate-id a1b2c3
```

- Per i dettagli sull'API, consulta DeleteClientCertificate AWS CLICommand Reference.

`delete-deployment`

Il seguente esempio di codice mostra come utilizzare`delete-deployment`.

AWS CLI

Come eliminare un'implementazione in un'API

Comando:

```
aws apigateway delete-deployment --rest-api-id 1234123412 --deployment-id a1b2c3
```

- Per i dettagli sull'API, consulta DeleteDeployment AWS CLICommand Reference.

`delete-domain-name-access-association`

Il seguente esempio di codice mostra come utilizzare`delete-domain-name-access-association`.

AWS CLI

Come eliminare un'associazione di accesso al nome di dominio

L'esempio `delete-domain-name-access-association` seguente elimina un'associazione di accesso al nome di dominio tra un nome di dominio privato personalizzato e l'endpoint VPC.

```
aws apigateway delete-domain-name-access-association \
```

```
    --domain-name-access-association-arn arn:aws:apigateway:us-west-2:012345678910:/
domainnameaccessassociations/domainname/my.private.domain.tld/vpcesource/vpce-
abcd1234efg
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Nomi di dominio personalizzati per uso privato APIs in API Gateway nella Amazon API Gateway Developer Guide.

- Per i dettagli sull'API, consulta DeleteDomainNameAccessAssociation AWS CLICommand Reference.

`delete-domain-name`

Il seguente esempio di codice mostra come utilizzare`delete-domain-name`.

AWS CLI

Come eliminare un nome di dominio personalizzato

Comando:

```
aws apigateway delete-domain-name --domain-name 'api.domain.tld'
```

- Per i dettagli sull'API, consulta DeleteDomainName AWS CLICommand Reference.

`delete-integration-response`

Il seguente esempio di codice mostra come utilizzare`delete-integration-response`.

AWS CLI

Come eliminare una risposta di integrazione per una risorsa, un metodo e un codice di stato specificati in un'API

Comando:

```
aws apigateway delete-integration-response --rest-api-id 1234123412 --resource-
id a1b2c3 --http-method GET --status-code 200
```

- Per i dettagli sull'API, consulta DeleteIntegrationResponse AWS CLICommand Reference.

`delete-integration`

Il seguente esempio di codice mostra come utilizzare`delete-integration`.

AWS CLI

Come eliminare un'integrazione per una risorsa e un metodo specificati in un'API

Comando:

```
aws apigateway delete-integration --rest-api-id 1234123412 --resource-id a1b2c3 --
http-method GET
```

- Per i dettagli sull'API, consulta DeleteIntegration AWS CLICommand Reference.

`delete-method-response`

Il seguente esempio di codice mostra come utilizzare`delete-method-response`.

AWS CLI

Come eliminare una risposta del metodo per la risorsa, il metodo e il codice di stato specificati in un'API

Comando:

```
aws apigateway delete-method-response --rest-api-id 1234123412 --resource-id a1b2c3
 --http-method GET --status-code 200
```

- Per i dettagli sull'API, consulta DeleteMethodResponse AWS CLICommand Reference.

`delete-method`

Il seguente esempio di codice mostra come utilizzare`delete-method`.

AWS CLI

Come eliminare un metodo per la risorsa specificata in un'API

Comando:

```
aws apigateway delete-method --rest-api-id 1234123412 --resource-id a1b2c3 --http-
method GET
```

- Per i dettagli sull'API, consulta DeleteMethod AWS CLICommand Reference.

`delete-model`

Il seguente esempio di codice mostra come utilizzare`delete-model`.

AWS CLI

Come eliminare un modello nell'API specificata

Comando:

```
aws apigateway delete-model --rest-api-id 1234123412 --model-name 'customModel'
```

- Per i dettagli sull'API, consulta DeleteModel AWS CLICommand Reference.

`delete-resource`

Il seguente esempio di codice mostra come utilizzare`delete-resource`.

AWS CLI

Come eliminare una risorsa in un'API

Comando:

```
aws apigateway delete-resource --rest-api-id 1234123412 --resource-id a1b2c3
```

- Per i dettagli sull'API, consulta DeleteResource AWS CLICommand Reference.

`delete-rest-api`

Il seguente esempio di codice mostra come utilizzare`delete-rest-api`.

AWS CLI

Come eliminare un'API

Comando:

```
aws apigateway delete-rest-api --rest-api-id 1234123412
```

- Per i dettagli sull'API, consulta DeleteRestApi AWS CLICommand Reference.

`delete-stage`

Il seguente esempio di codice mostra come utilizzare`delete-stage`.

AWS CLI

Come eliminare una fase in un'API

Comando:

```
aws apigateway delete-stage --rest-api-id 1234123412 --stage-name 'dev'
```

- Per i dettagli sull'API, consulta DeleteStage AWS CLICommand Reference.

`delete-usage-plan-key`

Il seguente esempio di codice mostra come utilizzare`delete-usage-plan-key`.

AWS CLI

Come rimuovere una chiave API da un piano di utilizzo

Comando:

```
aws apigateway delete-usage-plan-key --usage-plan-id a1b2c3 --key-
id 1NbjQzMReAkeEQPNAW8r3dXsU2rDD7fc7f2Sipnu
```

- Per i dettagli sull'API, consulta DeleteUsagePlanKey AWS CLICommand Reference.

`delete-usage-plan`

Il seguente esempio di codice mostra come utilizzare`delete-usage-plan`.

AWS CLI

Come eliminare un piano di utilizzo

Comando:

```
aws apigateway delete-usage-plan --usage-plan-id a1b2c3
```

- Per i dettagli sull'API, consulta DeleteUsagePlan AWS CLICommand Reference.

`flush-stage-authorizers-cache`

Il seguente esempio di codice mostra come utilizzare`flush-stage-authorizers-cache`.

AWS CLI

Come scaricare l'intera cache del sistema di autorizzazione in una fase

Comando:

```
aws apigateway flush-stage-authorizers-cache --rest-api-id 1234123412 --stage-
name dev
```

- Per i dettagli sull'API, consulta FlushStageAuthorizersCache AWS CLICommand Reference.

`flush-stage-cache`

Il seguente esempio di codice mostra come utilizzare`flush-stage-cache`.

AWS CLI

Come scaricare la cache per una fase dell'API

L'esempio `flush-stage-cache` seguente scarica la cache di una fase.

```
aws apigateway flush-stage-cache \
    --rest-api-id 1234123412 \
    --stage-name dev
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Scaricare la cache della fase API in Gateway API nella Guida per gli sviluppatori di Gateway Amazon API.

- Per i dettagli sull'API, consulta FlushStageCache AWS CLICommand Reference.

`generate-client-certificate`

Il seguente esempio di codice mostra come utilizzare`generate-client-certificate`.

AWS CLI

Come creare un certificato SSL lato client

Comando:

```
aws apigateway generate-client-certificate --description 'My First Client
 Certificate'
```

- Per i dettagli sull'API, consulta GenerateClientCertificate AWS CLICommand Reference.

`get-account`

Il seguente esempio di codice mostra come utilizzare`get-account`.

AWS CLI

Come ottenere le impostazioni dell'account Gateway API

Comando:

```
aws apigateway get-account
```

Output:

```
{
    "cloudwatchRoleArn": "arn:aws:iam::123412341234:role/
APIGatewayToCloudWatchLogsRole",
    "throttleSettings": {
        "rateLimit": 500.0,
        "burstLimit": 1000
    }
```

```
}
```

- Per i dettagli sull'API, consulta GetAccount AWS CLICommand Reference.

`get-api-key`

Il seguente esempio di codice mostra come utilizzare`get-api-key`.

AWS CLI

Come recuperare le informazioni su una chiave API specifica

Comando:

```
aws apigateway get-api-key --api-key 8bklk8bl1k3sB38D9B3l0enyWT8c09B30lkq0blk
```

Output:

```
{
    "description": "My first key",
    "enabled": true,
    "stageKeys": [
        "a1b2c3d4e5/dev",
        "e5d4c3b2a1/dev"
    ],
    "lastUpdatedDate": 1456184515,
    "createdDate": 1456184452,
    "id": "8bklk8bl1k3sB38D9B3l0enyWT8c09B30lkq0blk",
    "name": "My key"
}
```

- Per i dettagli sull'API, consulta GetApiKey AWS CLICommand Reference.

`get-api-keys`

Il seguente esempio di codice mostra come utilizzare`get-api-keys`.

AWS CLI

Come ottenere l'elenco delle chiavi API

Comando:

```
aws apigateway get-api-keys
```

Output:

```
{
    "items": [
        {
            "description": "My first key",
            "enabled": true,
            "stageKeys": [
                "a1b2c3d4e5/dev",
                "e5d4c3b2a1/dev"
            ],
            "lastUpdatedDate": 1456184515,
            "createdDate": 1456184452,
            "id": "8bklk8bl1k3sB38D9B3l0enyWT8c09B30lkq0blk",
            "name": "My key"
        }
    ]
}
```

- Per i dettagli sull'API, consulta GetApiKeys AWS CLICommand Reference.

`get-authorizer`

Il seguente esempio di codice mostra come utilizzare`get-authorizer`.

AWS CLI

Come ottenere le impostazioni del sistema di autorizzazione per API di Gateway API

Comando:

```
aws apigateway get-authorizer --rest-api-id 1234123412 --authorizer-id gfi4n3
```

Output:

```
{
```

```
    "authorizerResultTtlInSeconds": 300,
    "name": "MyAuthorizer",
    "type": "TOKEN",
    "identitySource": "method.request.header.Authorization",
    "authorizerUri": "arn:aws:apigateway:us-west-2:lambda:path/2015-03-31/functions/
arn:aws:lambda:us-west-2:123412341234:function:authorizer_function/invocations",
    "id": "gfi4n3"
}
```

- Per i dettagli sull'API, consulta GetAuthorizer AWS CLICommand Reference.

`get-authorizers`

Il seguente esempio di codice mostra come utilizzare`get-authorizers`.

AWS CLI

Come ottenere l'elenco dei sistemi di autorizzazione per una REST API

Comando:

```
aws apigateway get-authorizers --rest-api-id 1234123412
```

Output:

```
{
    "items": [
        {
            "name": "MyAuthorizer",
            "authorizerUri": "arn:aws:apigateway:us-west-2:lambda:path/2015-03-31/
functions/arn:aws:lambda:us-west-2:123412341234:function:My_Authorizer_Function/
invocations",
            "authorizerResultTtlInSeconds": 300,
            "identitySource": "method.request.header.Authorization",
            "type": "TOKEN",
            "id": "gfi4n3"
        }
    ]
}
```

- Per i dettagli sull'API, consulta GetAuthorizers AWS CLICommand Reference.

`get-base-path-mapping`

Il seguente esempio di codice mostra come utilizzare`get-base-path-mapping`.

AWS CLI

Come ottenere la mappatura del percorso di base per un nome di dominio personalizzato

Comando:

```
aws apigateway get-base-path-mapping --domain-name subdomain.domain.tld --base-
path v1
```

Output:

```
{
    "basePath": "v1",
    "restApiId": "1234w4321e",
    "stage": "api"
}
```

- Per i dettagli sull'API, consulta GetBasePathMapping AWS CLICommand Reference.

`get-base-path-mappings`

Il seguente esempio di codice mostra come utilizzare`get-base-path-mappings`.

AWS CLI

Come ottenere le mappature del percorso di base per un nome di dominio personalizzato

Comando:

```
aws apigateway get-base-path-mappings --domain-name subdomain.domain.tld
```

Output:

```
{
    "items": [
        {
```

```
            "basePath": "(none)",
            "restApiId": "1234w4321e",
            "stage": "dev"
        },
        {
            "basePath": "v1",
            "restApiId": "1234w4321e",
            "stage": "api"
        }
    ]
}
```

- Per i dettagli sull'API, consulta GetBasePathMappings AWS CLICommand Reference.

`get-client-certificate`

Il seguente esempio di codice mostra come utilizzare`get-client-certificate`.

AWS CLI

Come ottenere un certificato client

Comando:

```
aws apigateway get-client-certificate --client-certificate-id a1b2c3
```

- Per i dettagli sull'API, consulta GetClientCertificate AWS CLICommand Reference.

`get-client-certificates`

Il seguente esempio di codice mostra come utilizzare`get-client-certificates`.

AWS CLI

Come ottenere un elenco di certificati client

Comando:

```
aws apigateway get-client-certificates
```

Output:

```
{
    "items": [
        {
            "pemEncodedCertificate": "-----BEGIN CERTIFICATE----- <certificate
 content> -----END CERTIFICATE-----",
            "clientCertificateId": "a1b2c3",
            "expirationDate": 1483556561,
            "description": "My Client Certificate",
            "createdDate": 1452020561
        }
    ]
}
```

- Per i dettagli sull'API, consulta GetClientCertificates AWS CLICommand Reference.

`get-deployment`

Il seguente esempio di codice mostra come utilizzare`get-deployment`.

AWS CLI

Come ottenere informazioni su un'implementazione

Comando:

```
aws apigateway get-deployment --rest-api-id 1234123412 --deployment-id ztt4m2
```

Output:

```
{
    "description": "myDeployment",
    "id": "ztt4m2",
    "createdDate": 1455218022
}
```

- Per i dettagli sull'API, consulta GetDeployment AWS CLICommand Reference.

`get-deployments`

Il seguente esempio di codice mostra come utilizzare`get-deployments`.

AWS CLI

Come ottenere un elenco di implementazioni per una REST API

Comando:

```
aws apigateway get-deployments --rest-api-id 1234123412
```

Output:

```
{
    "items": [
        {
            "createdDate": 1453797217,
            "id": "0a2b4c",
            "description": "Deployed my API for the first time"
        }
    ]
}
```

- Per i dettagli sull'API, consulta GetDeployments AWS CLICommand Reference.

`get-domain-name-access-associations`

Il seguente esempio di codice mostra come utilizzare`get-domain-name-access-associations`.

AWS CLI

Esempio 1: come elencare tutte le associazioni di accesso ai nomi di dominio

L'esempio `get-domain-name-access-associations` seguente mostra come elencare tutte le associazioni di accesso ai nomi di dominio.

```
aws apigateway get-domain-name-access-associations
```

Output:

```
{
    "items": [
        {
```

```
        "domainNameAccessAssociationArn": "arn:aws:apigateway:us-
west-2:012345678910:/domainnameaccessassociations/domainname/my.private.domain.tld/
vpcesource/vpce-abcd1234efg
        "accessAssociationSource": "vpce-abcd1234efg",
        "accessAssociationSourceType": "VPCE",
        "domainNameArn" : "arn:aws:apigateway:us-west-2:111122223333:/domainnames/
private.example.com+abcd1234"
        }
    ]
}
```

Per ulteriori informazioni, consulta Nomi di dominio personalizzati per uso privato APIs in API Gateway nella Amazon API Gateway Developer Guide.

Esempio 2: per elencare tutte le associazioni di accesso ai nomi di dominio di proprietà di questo AWS account

L'`get-domain-name-access-associations`esempio seguente elenca tutte le associazioni di accesso ai nomi di dominio di proprietà dell' AWS account corrente.

```
aws apigateway get-domain-name-access-associations \
    --resource-owner SELF
```

Output:

```
{
    "items": [
        {
        "domainNameAccessAssociationArn": "arn:aws:apigateway:us-
west-2:012345678910:/domainnameaccessassociations/domainname/my.private.domain.tld/
vpcesource/vpce-abcd1234efg
        "accessAssociationSource": "vpce-abcd1234efg",
        "accessAssociationSourceType": "VPCE",
        "domainNameArn" : "arn:aws:apigateway:us-west-2:111122223333:/domainnames/
private.example.com+abcd1234"
        }
    ]
}
```

Per ulteriori informazioni, consulta Nomi di dominio personalizzati per uso privato APIs in API Gateway nella Amazon API Gateway Developer Guide.

- Per i dettagli sull'API, consulta GetDomainNameAccessAssociations AWS CLICommand Reference.

`get-domain-name`

Il seguente esempio di codice mostra come utilizzare`get-domain-name`.

AWS CLI

Esempio 1: come ottenere informazioni su un nome di dominio pubblico personalizzato

L'esempio `get-domain-name` seguente ottiene informazioni su un nome di dominio pubblico personalizzato.

```
aws apigateway get-domain-name \
    --domain-name api.domain.tld
```

Output:

```
{
    "domainName": "api.domain.tld",
    "distributionDomainName": "d1a2f3a4c5o6d.cloudfront.net",
    "certificateName": "uploadedCertificate",
    "certificateUploadDate": 1462565487
}
```

Per ulteriori informazioni, consulta la sezione Nome di dominio personalizzato per REST pubblico APIs in API Gateway nella Amazon API Gateway Developer Guide.

Esempio 2: come ottenere informazioni su un nome di dominio privato personalizzato

L'esempio `get-domain-name` seguente ottiene informazioni su un nome di dominio privato personalizzato.

```
aws apigateway get-domain-name \
    --domain-name api.private.domain.tld \
    --domain-name-id abcd1234
```

Output:

```
{
```

```
    "domainName": "my.private.domain.tld",
    "domainNameId": "abcd1234",
    "domainNameArn": "arn:aws:apigateway:us-east-1:012345678910:/domainnames/
my.private.domain.tld+abcd1234",
    "certificateArn": "arn:aws:acm:us-east-1:012345678910:certificate/fb1b9770-
a305-495d-aefb-27e5e101ff3",
    "certificateUploadDate": "2024-09-10T10:31:20-07:00",
    "endpointConfiguration": {
        "types": [
            "PRIVATE"
        ]
    },
    "domainNameStatus": "AVAILABLE",
    "securityPolicy": "TLS_1_2",
    "policy": "{\"Version\":\"2012-10-17\",\"Statement\":[{\"Effect\":
\"Allow\",\"Principal\":\"*\",\"Action\":\"execute-api:Invoke\",\"Resource\":
\"arn:aws:execute-api:us-east-1:012345678910:/domainnames/my.private.domain.tld
+abcd1234\"},{\"Effect\":\"Deny\",\"Principal\":\"*\",\"Action\":\"execute-
api:Invoke\",\"Resource\":\"arn:aws:execute-api:us-east-1:012345678910:/domainnames/
my.private.domain.tld+abcd1234\",\"Condition\":{\"StringNotEquals\":{\"aws:SourceVpc
\":\"vpc-1a2b3c4d\"}}}]}"
}
```

Per ulteriori informazioni, consulta la sezione Nome di dominio personalizzato per REST pubblico APIs in API Gateway nella Amazon API Gateway Developer Guide.

- Per i dettagli sull'API, consulta GetDomainName AWS CLICommand Reference.

`get-domain-names`

Il seguente esempio di codice mostra come utilizzare`get-domain-names`.

AWS CLI

Esempio 1: come ottenere un elenco di nomi di dominio personalizzati

Il comando `get-domain-names` seguente ottiene un elenco di nomi di dominio.

```
aws apigateway get-domain-names
```

Output:

```
{
```

```
    "items": [
        {
            "distributionDomainName": "d9511k3l09bkd.cloudfront.net",
            "certificateUploadDate": 1452812505,
            "certificateName": "my_custom_domain-certificate",
            "domainName": "subdomain.domain.tld"
        }
    ]
}
```

Per ulteriori informazioni, consulta Nomi di dominio personalizzati per uso privato APIs in API Gateway nella Amazon API Gateway Developer Guide.

Esempio 2: per ottenere un elenco di nomi di dominio personalizzati di proprietà di questo AWS account

Il `get-domain-names` comando seguente ottiene un elenco di nomi di dominio di proprietà di questo AWS account.

```
aws apigateway get-domain-names \
    --resource-owner SELF
```

Output:

```
{
    "items": [
        {
            "domainName": "my.domain.tld",
            "domainNameArn": "arn:aws:apigateway:us-east-1::/domainnames/
my.private.domain.tld",
            "certificateUploadDate": "2024-08-15T17:02:55-07:00",
            "regionalDomainName": "d-abcd1234.execute-api.us-east-1.amazonaws.com",
            "regionalHostedZoneId": "Z1UJRXOUMOOFQ8",
            "regionalCertificateArn": "arn:aws:acm:us-
east-1:012345678910:certificate/fb1b9770-a305-495d-aefb-27e5e101ff3",
            "endpointConfiguration": {
                "types": [
                    "REGIONAL"
                ]
            },
            "domainNameStatus": "AVAILABLE",
            "securityPolicy": "TLS_1_2"
```

```
        },
        {
            "domainName": "my.private.domain.tld",
            "domainNameId": "abcd1234",
            "domainNameArn": "arn:aws:apigateway:us-east-1:012345678910:/
domainnames/my.private.domain.tld+abcd1234",
            "certificateArn": "arn:aws:acm:us-east-1:012345678910:certificate/
fb1b9770-a305-495d-aefb-27e5e101ff3",
            "certificateUploadDate": "2024-11-26T11:44:40-08:00",
            "endpointConfiguration": {
                "types": [
                    "PRIVATE"
                ]
            },
            "domainNameStatus": "AVAILABLE",
            "securityPolicy": "TLS_1_2"
        }
    ]
}
```

Per ulteriori informazioni, consulta Nomi di dominio personalizzati per uso privato APIs in API Gateway nella Amazon API Gateway Developer Guide.

Esempio 3: per ottenere un elenco di nomi di dominio personalizzati di proprietà di altri AWS account con cui puoi creare un'associazione di accesso ai nomi di dominio.

Il `get-domain-names` comando seguente ottiene un elenco di nomi di dominio di proprietà di altri AWS account a cui si ha accesso per creare un'associazione di accesso ai nomi di dominio.

```
aws apigateway get-domain-names \
    --resource-owner OTHER_ACCOUNTS
```

Output:

```
{
    "items": [
        {
            "domainName": "my.private.domain.tld",
            "domainNameId": "abcd1234",
            "domainNameArn": "arn:aws:apigateway:us-east-1:012345678910:/
domainnames/my.private.domain.tld+abcd1234"
        }
```

```
    ]
}
```

Per ulteriori informazioni, consulta Nomi di dominio personalizzati per uso privato APIs in API Gateway nella Amazon API Gateway Developer Guide.

- Per i dettagli sull'API, consulta GetDomainNames AWS CLICommand Reference.

`get-export`

Il seguente esempio di codice mostra come utilizzare`get-export`.

AWS CLI

Come ottenere il modello Swagger JSON per una fase

Comando:

```
aws apigateway get-export --rest-api-id a1b2c3d4e5 --stage-name dev --export-
type swagger /path/to/filename.json
```

Come ottenere il modello Swagger JSON e le estensioni di Gateway API per una fase

Comando:

```
aws apigateway get-export --parameters extensions='integrations' --rest-api-
id a1b2c3d4e5 --stage-name dev --export-type swagger /path/to/filename.json
```

Come ottenere il modello Swagger JSON e le estensioni di Postman per una fase

Comando:

```
aws apigateway get-export --parameters extensions='postman' --rest-api-id a1b2c3d4e5
 --stage-name dev --export-type swagger /path/to/filename.json
```

- Per i dettagli sull'API, consulta GetExport AWS CLICommand Reference.

`get-integration-response`

Il seguente esempio di codice mostra come utilizzare`get-integration-response`.

AWS CLI

Come ottenere la configurazione della risposta di integrazione per un metodo HTTP definito nella risorsa di una REST API

Comando:

```
aws apigateway get-integration-response --rest-api-id 1234123412 --resource-
id y9h6rt --http-method GET --status-code 200
```

Output:

```
{
    "statusCode": "200",
    "responseTemplates": {
        "application/json": null
    }
}
```

- Per i dettagli sull'API, consulta GetIntegrationResponse AWS CLICommand Reference.

`get-integration`

Il seguente esempio di codice mostra come utilizzare`get-integration`.

AWS CLI

Come ottenere la configurazione dell'integrazione per un metodo HTTP definito nella risorsa di una REST API

Comando:

```
aws apigateway get-integration --rest-api-id 1234123412 --resource-id y9h6rt --http-
method GET
```

Output:

```
{
    "httpMethod": "POST",
```

```
    "integrationResponses": {
        "200": {
            "responseTemplates": {
                "application/json": null
            },
            "statusCode": "200"
        }
    },
    "cacheKeyParameters": [],
    "type": "AWS",
    "uri": "arn:aws:apigateway:us-west-2:lambda:path/2015-03-31/functions/
arn:aws:lambda:us-west-2:123412341234:function:My_Function/invocations",
    "cacheNamespace": "y9h6rt"
}
```

- Per i dettagli sull'API, consulta GetIntegration AWS CLICommand Reference.

`get-method-response`

Il seguente esempio di codice mostra come utilizzare`get-method-response`.

AWS CLI

Come ottenere la configurazione della risorsa di risposta del metodo per un metodo HTTP definito nella risorsa di una REST API

Comando:

```
aws apigateway get-method-response --rest-api-id 1234123412 --resource-id y9h6rt --
http-method GET --status-code 200
```

Output:

```
{
    "responseModels": {
        "application/json": "Empty"
    },
    "statusCode": "200"
}
```

- Per i dettagli sull'API, consulta GetMethodResponse AWS CLICommand Reference.

`get-method`

Il seguente esempio di codice mostra come utilizzare`get-method`.

AWS CLI

Come ottenere la configurazione della risorsa del metodo per un metodo HTTP definito nella risorsa di una REST API

Comando:

```
aws apigateway get-method --rest-api-id 1234123412 --resource-id y9h6rt --http-
method GET
```

Output:

```
{
    "apiKeyRequired": false,
    "httpMethod": "GET",
    "methodIntegration": {
        "integrationResponses": {
            "200": {
                "responseTemplates": {
                    "application/json": null
                },
                "statusCode": "200"
            }
        },
        "cacheKeyParameters": [],
        "uri": "arn:aws:apigateway:us-west-2:lambda:path/2015-03-31/functions/
arn:aws:lambda:us-west-2:123412341234:function:My_Function/invocations",
        "httpMethod": "POST",
        "cacheNamespace": "y9h6rt",
        "type": "AWS"
    },
    "requestParameters": {},
    "methodResponses": {
        "200": {
            "responseModels": {
                "application/json": "Empty"
            },
            "statusCode": "200"
        }
```

```
    },
    "authorizationType": "NONE"
}
```

- Per i dettagli sull'API, consulta GetMethod AWS CLICommand Reference.

`get-model-template`

Il seguente esempio di codice mostra come utilizzare`get-model-template`.

AWS CLI

Come ottenere il modello di mappatura per un modello definito in una REST API

Comando:

```
aws apigateway get-model-template --rest-api-id 1234123412 --model-name Empty
```

Output:

```
{
    "value": "#set($inputRoot = $input.path('$'))\n{ }"
}
```

- Per i dettagli sull'API, consulta GetModelTemplate AWS CLICommand Reference.

`get-model`

Il seguente esempio di codice mostra come utilizzare`get-model`.

AWS CLI

Come ottenere la configurazione per un modello definito in una REST API

Comando:

```
aws apigateway get-model --rest-api-id 1234123412 --model-name Empty
```

Output:

```
{
    "contentType": "application/json",
    "description": "This is a default empty schema model",
    "name": "Empty",
    "id": "etd5w5",
    "schema": "{\n  \"$schema\": \"http://json-schema.org/draft-04/schema#\",\n
 \"title\" : \"Empty Schema\",\n  \"type\" : \"object\"\n}"
}
```

- Per i dettagli sull'API, consulta GetModel AWS CLICommand Reference.

`get-models`

Il seguente esempio di codice mostra come utilizzare`get-models`.

AWS CLI

Come ottenere un elenco di modelli per una REST API

Comando:

```
aws apigateway get-models --rest-api-id 1234123412
```

Output:

```
{
    "items": [
        {
            "description": "This is a default error schema model",
            "schema": "{\n  \"$schema\" : \"http://json-schema.org/draft-04/schema#
\",\n  \"title\" : \"Error Schema\",\n  \"type\" : \"object\",\n  \"properties\" :
 {\n    \"message\" : { \"type\" : \"string\" }\n  }\n}",
            "contentType": "application/json",
            "id": "7tpbze",
            "name": "Error"
        },
        {
            "description": "This is a default empty schema model",
            "schema": "{\n  \"$schema\": \"http://json-schema.org/draft-04/schema#
\",\n  \"title\" : \"Empty Schema\",\n  \"type\" : \"object\"\n}",
            "contentType": "application/json",
```

```
            "id": "etd5w5",
            "name": "Empty"
        }
    ]
}
```

- Per i dettagli sull'API, consulta GetModels AWS CLICommand Reference.

`get-resource`

Il seguente esempio di codice mostra come utilizzare`get-resource`.

AWS CLI

Come ottenere informazioni su una risorsa

Comando:

```
aws apigateway get-resource --rest-api-id 1234123412 --resource-id zwo0y3
```

Output:

```
{
    "path": "/path",
    "pathPart": "path",
    "id": "zwo0y3",
    "parentId": "uyokt6ij2g"
}
```

- Per i dettagli sull'API, consulta GetResource AWS CLICommand Reference.

`get-resources`

Il seguente esempio di codice mostra come utilizzare`get-resources`.

AWS CLI

Come ottenere un elenco di risorse per una REST API

Comando:

```
aws apigateway get-resources --rest-api-id 1234123412
```

Output:

```
{
    "items": [
        {
            "path": "/resource/subresource",
            "resourceMethods": {
                "POST": {}
            },
            "id": "024ace",
            "pathPart": "subresource",
            "parentId": "ai5b02"
        }
    ]
}
```

- Per i dettagli sull'API, consulta GetResources AWS CLICommand Reference.

`get-rest-api`

Il seguente esempio di codice mostra come utilizzare`get-rest-api`.

AWS CLI

Come ottenere informazioni su un'API

Comando:

```
aws apigateway get-rest-api --rest-api-id 1234123412
```

Output:

```
{
    "name": "myAPI",
    "id": "o1y243m4f5",
    "createdDate": 1453416433
}
```

- Per i dettagli sull'API, consulta GetRestApi AWS CLICommand Reference.

`get-rest-apis`

Il seguente esempio di codice mostra come utilizzare`get-rest-apis`.

AWS CLI

Per ottenere un elenco di REST APIs

Comando:

```
aws apigateway get-rest-apis
```

Output:

```
{
    "items": [
        {
            "createdDate": 1438884790,
            "id": "12s44z21rb",
            "name": "My First API"
        }
    ]
}
```

- Per i dettagli sull'API, consulta GetRestApis AWS CLICommand Reference.

`get-sdk`

Il seguente esempio di codice mostra come utilizzare`get-sdk`.

AWS CLI

Come ottenere l'SDK Android per una fase della REST API

Comando:

```
aws apigateway get-sdk --rest-api-id 1234123412 --stage-name dev --sdk-type android
 --parameters
 groupId='com.mycompany',invokerPackage='com.mycompany.clientsdk',artifactId='Mycompany-
client',artifactVersion='1.0.0' /path/to/android_sdk.zip
```

Output:

```
{
    "contentType": "application/octet-stream",
    "contentDisposition": "attachment; filename=\"android_2016-02-22_23-52Z.zip\""
}
```

Come ottenere l'SDK iOS per una fase della REST API

Comando:

```
aws apigateway get-sdk --rest-api-id 1234123412 --stage-name dev --sdk-
type objectivec --parameters classPrefix='myprefix' /path/to/iOS_sdk.zip
```

Output:

```
{
    "contentType": "application/octet-stream",
    "contentDisposition": "attachment; filename=\"objectivec_2016-02-22_23-52Z.zip
\""
}
```

Come ottenere l'SDK Javascript per una fase della REST API

Comando:

```
aws apigateway get-sdk --rest-api-id 1234123412 --stage-name dev --sdk-
type javascript /path/to/javascript_sdk.zip
```

Output:

```
{
    "contentType": "application/octet-stream",
    "contentDisposition": "attachment; filename=\"javascript_2016-02-22_23-52Z.zip
\""
}
```

- Per i dettagli sull'API, consulta GetSdk AWS CLICommand Reference.

`get-stage`

Il seguente esempio di codice mostra come utilizzare`get-stage`.

AWS CLI

Come ottenere informazioni su una fase di un'API

Comando:

```
aws apigateway get-stage --rest-api-id 1234123412 --stage-name dev
```

Output:

```
{
    "stageName": "dev",
    "cacheClusterSize": "0.5",
    "cacheClusterEnabled": false,
    "cacheClusterStatus": "NOT_AVAILABLE",
    "deploymentId": "rbh1fj",
    "lastUpdatedDate": 1466802961,
    "createdDate": 1460682074,
    "methodSettings": {
        "*/*": {
            "cacheTtlInSeconds": 300,
            "loggingLevel": "INFO",
            "dataTraceEnabled": false,
            "metricsEnabled": true,
            "unauthorizedCacheControlHeaderStrategy":
 "SUCCEED_WITH_RESPONSE_HEADER",
            "throttlingRateLimit": 500.0,
            "cacheDataEncrypted": false,
            "cachingEnabled": false,
            "throttlingBurstLimit": 1000,
            "requireAuthorizationForCacheControl": true
        },
        "~1resource/GET": {
            "cacheTtlInSeconds": 300,
            "loggingLevel": "INFO",
            "dataTraceEnabled": false,
            "metricsEnabled": true,
            "unauthorizedCacheControlHeaderStrategy":
 "SUCCEED_WITH_RESPONSE_HEADER",
            "throttlingRateLimit": 500.0,
            "cacheDataEncrypted": false,
            "cachingEnabled": false,
            "throttlingBurstLimit": 1000,
```

```
            "requireAuthorizationForCacheControl": true
        }
    }
}
```

- Per i dettagli sull'API, consulta GetStage AWS CLICommand Reference.

`get-stages`

Il seguente esempio di codice mostra come utilizzare`get-stages`.

AWS CLI

Come ottenere un elenco di fasi per una REST API

Comando:

```
aws apigateway get-stages --rest-api-id 1234123412
```

Output:

```
{
    "item": [
        {
            "stageName": "dev",
            "cacheClusterSize": "0.5",
            "cacheClusterEnabled": true,
            "cacheClusterStatus": "AVAILABLE",
            "deploymentId": "123h64",
            "lastUpdatedDate": 1456185138,
            "createdDate": 1453589092,
            "methodSettings": {
                "~1resource~1subresource/POST": {
                    "cacheTtlInSeconds": 300,
                    "loggingLevel": "INFO",
                    "dataTraceEnabled": true,
                    "metricsEnabled": true,
                    "throttlingRateLimit": 500.0,
                    "cacheDataEncrypted": false,
                    "cachingEnabled": false,
                    "throttlingBurstLimit": 1000
                }
```

```
            }
        }
    ]
}
```

- Per i dettagli sull'API, consulta GetStages AWS CLICommand Reference.

`get-usage-plan-key`

Il seguente esempio di codice mostra come utilizzare`get-usage-plan-key`.

AWS CLI

Come ottenere i dettagli di una chiave API associata a un piano di utilizzo

Comando:

```
aws apigateway get-usage-plan-key --usage-plan-id a1b2c3 --key-
id 1NbjQzMReAkeEQPNAW8r3dXsU2rDD7fc7f2Sipnu
```

- Per i dettagli sull'API, consulta GetUsagePlanKey AWS CLICommand Reference.

`get-usage-plan-keys`

Il seguente esempio di codice mostra come utilizzare`get-usage-plan-keys`.

AWS CLI

Come ottenere l'elenco delle chiavi API associate a un piano di utilizzo

Comando:

```
aws apigateway get-usage-plan-keys --usage-plan-id a1b2c3
```

- Per i dettagli sull'API, consulta GetUsagePlanKeys AWS CLICommand Reference.

`get-usage-plan`

Il seguente esempio di codice mostra come utilizzare`get-usage-plan`.

AWS CLI

Come ottenere i dettagli di un piano di utilizzo

Comando:

```
aws apigateway get-usage-plan --usage-plan-id a1b2c3
```

- Per i dettagli sull'API, consulta GetUsagePlan AWS CLICommand Reference.

`get-usage-plans`

Il seguente esempio di codice mostra come utilizzare`get-usage-plans`.

AWS CLI

Come ottenere i dettagli di tutti i piani di utilizzo

Comando:

```
aws apigateway get-usage-plans
```

- Per i dettagli sull'API, consulta GetUsagePlans AWS CLICommand Reference.

`get-usage`

Il seguente esempio di codice mostra come utilizzare`get-usage`.

AWS CLI

Come ottenere i dettagli sull'utilizzo per un piano di utilizzo

Comando:

```
aws apigateway get-usage --usage-plan-id a1b2c3 --start-date "2016-08-16" --end-
date "2016-08-17"
```

- Per i dettagli sull'API, consulta GetUsage AWS CLICommand Reference.

`import-rest-api`

Il seguente esempio di codice mostra come utilizzare`import-rest-api`.

AWS CLI

Come importare un modello Swagger e creare un'API

Comando:

```
aws apigateway import-rest-api --body 'file:///path/to/API_Swagger_template.json'
```

- Per i dettagli sull'API, consulta ImportRestApi AWS CLICommand Reference.

`put-integration-response`

Il seguente esempio di codice mostra come utilizzare`put-integration-response`.

AWS CLI

Come creare una risposta di integrazione come risposta predefinita con un modello di mappatura definito

Comando:

```
aws apigateway put-integration-response --rest-api-id 1234123412 --resource-
id a1b2c3 --http-method GET --status-code 200 --selection-pattern "" --response-
templates '{"application/json": "{\"json\": \"template\"}"}'
```

Come creare una risposta di integrazione con un'espressione regolare impostata su 400 e un valore di intestazione definito staticamente

Comando:

```
aws apigateway put-integration-response --rest-api-id 1234123412 --resource-
id a1b2c3 --http-method GET --status-code 400 --selection-pattern 400 --response-
parameters '{"method.response.header.custom-header": "'"'"'custom-value'"'"'"}'
```

- Per i dettagli sull'API, consulta PutIntegrationResponse AWS CLICommand Reference.

`put-integration`

Il seguente esempio di codice mostra come utilizzare`put-integration`.

AWS CLI

Come creare una richiesta di integrazione MOCK

Comando:

```
aws apigateway put-integration --rest-api-id 1234123412 --resource-id a1b2c3 --http-
method GET --type MOCK --request-templates '{ "application/json": "{\"statusCode\":
 200}" }'
```

Come creare una richiesta di integrazione HTTP

Comando:

```
aws apigateway put-integration --rest-api-id 1234123412 --resource-id a1b2c3 --http-
method GET --type HTTP --integration-http-method GET --uri 'https://domain.tld/path'
```

Per creare una richiesta di AWS integrazione con un endpoint della funzione Lambda

Comando:

```
aws apigateway put-integration --rest-api-id 1234123412 --resource-
id a1b2c3 --http-method GET --type AWS --integration-http-method POST --uri
 'arn:aws:apigateway:us-west-2:lambda:path/2015-03-31/functions/arn:aws:lambda:us-
west-2:123412341234:function:function_name/invocations'
```

- Per i dettagli sull'API, consulta AWS CLI Command PutIntegrationReference.

`put-method-response`

Il seguente esempio di codice mostra come utilizzare`put-method-response`.

AWS CLI

Come creare una risposta del metodo con il codice di stato specificato con un'intestazione di risposta del metodo personalizzata

Comando:

```
aws apigateway put-method-response --rest-api-id 1234123412 --
resource-id a1b2c3 --http-method GET --status-code 400 --response-
parameters "method.response.header.custom-header=false"
```

- Per i dettagli sull'API, consulta PutMethodResponse AWS CLICommand Reference.

`put-method`

Il seguente esempio di codice mostra come utilizzare`put-method`.

AWS CLI

Come creare un metodo per una risorsa in un'API senza autorizzazione, senza chiave API e con un'intestazione di richiesta del metodo personalizzata

Comando:

```
aws apigateway put-method --rest-api-id 1234123412 --resource-id a1b2c3 --
http-method PUT --authorization-type "NONE" --no-api-key-required --request-
parameters "method.request.header.custom-header=false"
```

- Per i dettagli sull'API, consulta PutMethod AWS CLICommand Reference.

`put-rest-api`

Il seguente esempio di codice mostra come utilizzare`put-rest-api`.

AWS CLI

Come sovrascrivere un'API esistente utilizzando un modello Swagger

Comando:

```
aws apigateway put-rest-api --rest-api-id 1234123412 --mode overwrite --body
 'fileb:///path/to/API_Swagger_template.json'
```

Come unire un modello Swagger a un'API esistente

Comando:

```
aws apigateway put-rest-api --rest-api-id 1234123412 --mode merge --body 'fileb:///
path/to/API_Swagger_template.json'
```

- Per i dettagli sull'API, consulta PutRestApi AWS CLICommand Reference.

`reject-domain-name-access-association`

Il seguente esempio di codice mostra come utilizzare`reject-domain-name-access-association`.

AWS CLI

Come rifiutare un'associazione di accesso al nome di dominio

L'esempio `reject-domain-name-access-association` seguente rifiuta un'associazione di accesso al nome di dominio tra un nome di dominio privato personalizzato e l'endpoint VPC.

```
aws apigateway reject-domain-name-access-association \
    --domain-name-access-association-arn arn:aws:apigateway:us-west-2:012345678910:/domainnameaccessassociations/domainname/my.private.domain.tld/vpcesource/vpce-abcd1234efg \
    --domain-name-arn arn:aws:apigateway:us-east-1:012345678910:/domainnames/my.private.domain.tld+abcd1234
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Nomi di dominio personalizzati per uso privato APIs in API Gateway nella Amazon API Gateway Developer Guide.

- Per i dettagli sull'API, consulta RejectDomainNameAccessAssociation AWS CLICommand Reference.

`test-invoke-authorizer`

Il seguente esempio di codice mostra come utilizzare`test-invoke-authorizer`.

AWS CLI

Come testare l'invocazione di una richiesta a un sistema di autorizzazione personalizzato che includa l'intestazione e il valore richiesti

Comando:

```
aws apigateway test-invoke-authorizer --rest-api-id 1234123412 --authorizer-id 5yid1t --headers Authorization='Value'
```

- Per i dettagli sull'API, consulta TestInvokeAuthorizer AWS CLICommand Reference.

`test-invoke-method`

Il seguente esempio di codice mostra come utilizzare`test-invoke-method`.

AWS CLI

Come testare l'invocazione della risorsa root in un'API eseguendo una richiesta GET

Comando:

```
aws apigateway test-invoke-method --rest-api-id 1234123412 --resource-id avl5sg8fw8
 --http-method GET --path-with-query-string '/'
```

Come testare l'invocazione di una risorsa secondaria in un'API eseguendo una richiesta GET con un valore di parametro di percorso specificato

Comando:

```
aws apigateway test-invoke-method --rest-api-id 1234123412 --resource-id 3gapai --
http-method GET --path-with-query-string '/pets/1'
```

- Per i dettagli sull'API, consulta TestInvokeMethod AWS CLICommand Reference.

`update-account`

Il seguente esempio di codice mostra come utilizzareu`pdate-account`.

AWS CLI

Per modificare l'ARN del ruolo IAM per la registrazione nei registri CloudWatch

Comando:

```
aws apigateway update-account --patch-operations op='replace',path='/
cloudwatchRoleArn',value='arn:aws:iam::123412341234:role/APIGatewayToCloudWatchLogs'
```

Output:

```
{
    "cloudwatchRoleArn": "arn:aws:iam::123412341234:role/
APIGatewayToCloudWatchLogs",
```

```
    "throttleSettings": {
        "rateLimit": 1000.0,
        "burstLimit": 2000
    }
}
```

- Per i dettagli sull'API, consulta Command UpdateAccountReference AWS CLI .

`update-api-key`

Il seguente esempio di codice mostra come utilizzare`update-api-key`.

AWS CLI

Come modificare il nome di una chiave API

Comando:

```
aws apigateway update-api-key --api-key sNvjQDMReA1eEQPNAW8r37XsU2rDD7fc7m2SiMnu --
patch-operations op='replace',path='/name',value='newName'
```

Output:

```
{
    "description": "currentDescription",
    "enabled": true,
    "stageKeys": [
        "41t2j324r5/dev"
    ],
    "lastUpdatedDate": 1470086052,
    "createdDate": 1445460347,
    "id": "sNvjQDMReA1vEQPNzW8r3dXsU2rrD7fcjm2SiMnu",
    "name": "newName"
}
```

Come disabilitare la chiave API

Comando:

```
aws apigateway update-api-key --api-key sNvjQDMReA1eEQPNAW8r37XsU2rDD7fc7m2SiMnu --
patch-operations op='replace',path='/enabled',value='false'
```

Output:

```
{
    "description": "currentDescription",
    "enabled": false,
    "stageKeys": [
        "41t2j324r5/dev"
    ],
    "lastUpdatedDate": 1470086052,
    "createdDate": 1445460347,
    "id": "sNvjQDMReA1vEQPNzW8r3dXsU2rrD7fcjm2SiMnu",
    "name": "newName"
}
```

- Per i dettagli sull'API, consulta UpdateApiKey AWS CLICommand Reference.

`update-authorizer`

Il seguente esempio di codice mostra come utilizzareu`pdate-authorizer`.

AWS CLI

Come modificare il nome del sistema di autorizzazione personalizzato

Comando:

```
aws apigateway update-authorizer --rest-api-id 1234123412 --authorizer-id gfi4n3 --
patch-operations op='replace',path='/name',value='testAuthorizer'
```

Output:

```
{
    "authType": "custom",
    "name": "testAuthorizer",
    "authorizerUri": "arn:aws:apigateway:us-west-2:lambda:path/2015-03-31/functions/
arn:aws:lambda:us-west-2:123412341234:function:customAuthorizer/invocations",
    "authorizerResultTtlInSeconds": 300,
    "identitySource": "method.request.header.Authorization",
    "type": "TOKEN",
    "id": "gfi4n3"
}
```

Come modificare la funzione Lambda invocata dal sistema di autorizzazione personalizzato

Comando:

```
aws apigateway update-authorizer --rest-api-id 1234123412 --authorizer-id gfi4n3 --
patch-operations op='replace',path='/authorizerUri',value='arn:aws:apigateway:us-
west-2:lambda:path/2015-03-31/functions/arn:aws:lambda:us-
west-2:123412341234:function:newAuthorizer/invocations'
```

Output:

```
{
    "authType": "custom",
    "name": "testAuthorizer",
    "authorizerUri": "arn:aws:apigateway:us-west-2:lambda:path/2015-03-31/functions/
arn:aws:lambda:us-west-2:123412341234:function:newAuthorizer/invocations",
    "authorizerResultTtlInSeconds": 300,
    "identitySource": "method.request.header.Authorization",
    "type": "TOKEN",
    "id": "gfi4n3"
}
```

- Per i dettagli sull'API, consulta UpdateAuthorizer AWS CLICommand Reference.

update-base-path-mapping

Il seguente esempio di codice mostra come utilizzareupdate-base-path-mapping.

AWS CLI

Come modificare il percorso di base per un nome di dominio personalizzato

Comando:

```
aws apigateway update-base-path-mapping --domain-name api.domain.tld --base-
path prod --patch-operations op='replace',path='/basePath',value='v1'
```

Output:

```
{
    "basePath": "v1",
```

```
    "restApiId": "1234123412",
    "stage": "api"
}
```

- Per i dettagli sull'API, consulta UpdateBasePathMapping AWS CLICommand Reference.

`update-client-certificate`

Il seguente esempio di codice mostra come utilizzareu`pdate-client-certificate`.

AWS CLI

Come aggiornare la descrizione di un certificato client

Comando:

```
aws apigateway update-client-certificate --client-certificate-id a1b2c3 --patch-operations op='replace',path='/description',value='My new description'
```

- Per i dettagli sull'API, consulta UpdateClientCertificate AWS CLICommand Reference.

`update-deployment`

Il seguente esempio di codice mostra come utilizzareu`pdate-deployment`.

AWS CLI

Come modificare la descrizione di un'implementazione

Comando:

```
aws apigateway update-deployment --rest-api-id 1234123412 --deployment-id ztt4m2 --patch-operations op='replace',path='/description',value='newDescription'
```

Output:

```
{
    "description": "newDescription",
    "id": "ztt4m2",
    "createdDate": 1455218022
}
```

- Per i dettagli sull'API, consulta UpdateDeployment AWS CLICommand Reference.

`update-domain-name`

Il seguente esempio di codice mostra come utilizzare`update-domain-name`.

AWS CLI

Come modificare il nome del certificato per un nome di dominio personalizzato

L'esempio `update-domain-name` seguente modifica il nome del certificato per un dominio personalizzato.

```
aws apigateway update-domain-name \
    --domain-name api.domain.tld \
    --patch-operations op='replace',path='/certificateArn',value='arn:aws:acm:us-
west-2:111122223333:certificate/CERTEXAMPLE123EXAMPLE'
```

Output:

```
{
    "domainName": "api.domain.tld",
    "distributionDomainName": "d123456789012.cloudfront.net",
    "certificateArn": "arn:aws:acm:us-west-2:111122223333:certificate/
CERTEXAMPLE123EXAMPLE",
    "certificateUploadDate": 1462565487
}
```

Per ulteriori informazioni consulta Nomi di dominio personalizzati per REST API pubbliche in Gateway API nella Guida per gli sviluppatori di Gateway Amazon API.

- Per i dettagli sull'API, consulta UpdateDomainName AWS CLICommand Reference.

`update-integration-response`

Il seguente esempio di codice mostra come utilizzare`update-integration-response`.

AWS CLI

Come modificare l'intestazione di una risposta di integrazione in modo da avere una mappatura statica di "*"

Comando:

```
aws apigateway update-integration-response --rest-api-id 1234123412 --
resource-id 3gapai --http-method GET --status-code 200 --patch-operations
 op='replace',path='/responseParameters/method.response.header.Access-Control-Allow-
Origin',value='"'"'*'"'"'
```

Output:

```
{
    "statusCode": "200",
    "responseParameters": {
        "method.response.header.Access-Control-Allow-Origin": "'*'"
    }
}
```

Come rimuovere l'intestazione di una risposta di integrazione

Comando:

```
aws apigateway update-integration-response --rest-api-id 1234123412 --resource-
id 3gapai --http-method GET --status-code 200 --patch-operations op='remove',path='/
responseParameters/method.response.header.Access-Control-Allow-Origin'
```

- Per i dettagli sull'API, consulta UpdateIntegrationResponse AWS CLICommand Reference.

`update-integration`

Il seguente esempio di codice mostra come utilizzare`update-integration`.

AWS CLI

Come aggiungere il modello di mappatura "Content-Type: application/json" configurato con Passthrough di input

Comando:

```
aws apigateway update-integration \
    --rest-api-id a1b2c3d4e5 \
    --resource-id a1b2c3 \
    --http-method POST \
```

```
    --patch-operations "op='add',path='/requestTemplates/application~1json'"
```

Come aggiornare (sostituire) il modello di mappatura "Content-Type: application/json" configurato con un modello personalizzato

Comando:

```
aws apigateway update-integration \
    --rest-api-id a1b2c3d4e5 \
    --resource-id a1b2c3 \
    --http-method POST \
    --patch-operations "op='replace',path='/requestTemplates/
application~1json',value='{"example": "json"}'"
```

Come aggiornare (sostituire) un modello personalizzato associato a "Content-Type: application/json" con Passthrough di input

Comando:

```
aws apigateway update-integration \
    --rest-api-id a1b2c3d4e5 \
    --resource-id a1b2c3 \
    --http-method POST \
    --patch-operations "op='replace',path='requestTemplates/application~1json'"
```

Come rimuovere il modello di mappatura "Content-Type: application/json"

Comando:

```
aws apigateway update-integration \
    --rest-api-id a1b2c3d4e5 \
    --resource-id a1b2c3 \
    --http-method POST \
    --patch-operations "op='remove',path='/requestTemplates/application~1json'"
```

- Per i dettagli sull'API, consulta UpdateIntegration AWS CLICommand Reference.

`update-method-response`

Il seguente esempio di codice mostra come utilizzare`update-method-response`.

AWS CLI

Come creare una nuova intestazione di risposta del metodo per la risposta 200 in un metodo e definirla come non obbligatoria (per impostazione predefinita)

Comando:

```
aws apigateway update-method-response --rest-api-id 1234123412 --resource-
id a1b2c3 --http-method GET --status-code 200 --patch-operations op="add",path="/
responseParameters/method.response.header.custom-header",value="false"
```

Come eliminare un modello di risposta per la risposta 200 in un metodo

Comando:

```
aws apigateway update-method-response --rest-api-id 1234123412 --resource-id a1b2c3
 --http-method GET --status-code 200 --patch-operations op="remove",path="/
responseModels/application~1json"
```

- Per i dettagli sull'API, consulta UpdateMethodResponse AWS CLICommand Reference.

`update-method`

Il seguente esempio di codice mostra come utilizzare`update-method`.

AWS CLI

Esempio 1: come modificare un metodo per richiedere una chiave API

L'esempio `update-method` seguente modifica il metodo per richiedere una chiave API.

```
aws apigateway update-method \
    --rest-api-id 1234123412 \
    --resource-id a1b2c3 \
    --http-method GET \
    --patch-operations op="replace",path="/apiKeyRequired",value="true"
```

Output:

```
{
    "httpMethod": "GET",
```

```
    "authorizationType": "NONE",
    "apiKeyRequired": true,
    "methodResponses": {
        "200": {
            "statusCode": "200",
            "responseModels": {}
        }
    },
    "methodIntegration": {
        "type": "AWS",
        "httpMethod": "POST",
        "uri": "arn:aws:apigateway:us-east-1:lambda:path/2015-03-31/functions/
arn:aws:lambda:us-east-1:123456789111:function:hello-world/invocations",
        "passthroughBehavior": "WHEN_NO_MATCH",
        "contentHandling": "CONVERT_TO_TEXT",
        "timeoutInMillis": 29000,
        "cacheNamespace": "h7i8j9",
        "cacheKeyParameters": [],
        "integrationResponses": {
            "200": {
                "statusCode": "200",
                "responseTemplates": {}
            }
        }
    }
}
```

Esempio 2: come modificare un metodo per richiedere l'autorizzazione IAM

L'esempio `update-method` seguente modifica il metodo per richiedere l'autorizzazione IAM.

```
aws apigateway update-method \
    --rest-api-id 1234123412 \
    --resource-id a1b2c3 \
    --http-method GET \
    --patch-operations op="replace",path="/authorizationType",value="AWS_IAM"
```

Output:

```
 {
    "httpMethod": "GET",
    "authorizationType": "AWS_IAM",
    "apiKeyRequired": false,
```

```
    "methodResponses": {
        "200": {
            "statusCode": "200",
            "responseModels": {}
        }
    },
    "methodIntegration": {
        "type": "AWS",
        "httpMethod": "POST",
        "uri": "arn:aws:apigateway:us-east-1:lambda:path/2015-03-31/functions/
arn:aws:lambda:us-east-1:123456789111:function:hello-world/invocations",
        "passthroughBehavior": "WHEN_NO_MATCH",
        "contentHandling": "CONVERT_TO_TEXT",
        "timeoutInMillis": 29000,
        "cacheNamespace": "h7i8j9",
        "cacheKeyParameters": [],
        "integrationResponses": {
            "200": {
                "statusCode": "200",
                "responseTemplates": {}
            }
        }
    }
}
```

Esempio 3: come modificare un metodo per richiedere l'autorizzazione Lambda

L'esempio `update-method` seguente modifica il metodo per richiedere l'autorizzazione Lambda.

```
aws apigateway update-method --rest-api-id 1234123412 \
    --resource-id a1b2c3 \
    --http-method GET \
    --patch-operations op="replace",path="/authorizationType",value="CUSTOM"
 op="replace",path="/authorizerId",value="e4f5g6"
```

Output:

```
 {
    "httpMethod": "GET",
    "authorizationType": "CUSTOM",
    "authorizerId" : "e4f5g6",
    "apiKeyRequired": false,
    "methodResponses": {
```

```
        "200": {
            "statusCode": "200",
            "responseModels": {}
        }
    },
    "methodIntegration": {
        "type": "AWS",
        "httpMethod": "POST",
        "uri": "arn:aws:apigateway:us-east-1:lambda:path/2015-03-31/functions/
arn:aws:lambda:us-east-1:123456789111:function:hello-world/invocations",
        "passthroughBehavior": "WHEN_NO_MATCH",
        "contentHandling": "CONVERT_TO_TEXT",
        "timeoutInMillis": 29000,
        "cacheNamespace": "h7i8j9",
        "cacheKeyParameters": [],
        "integrationResponses": {
            "200": {
                "statusCode": "200",
                "responseTemplates": {}
            }
        }
    }
}
```

Per ulteriori informazioni, consulta Creazione, configurazione e test dei piani di utilizzo mediante la CLI e la REST API di API Gateway e Controllo e gestione degli accessi alle REST API in Gateway API nella Guida per gli sviluppatori di Gateway Amazon API.

- Per i dettagli sull'API, consulta UpdateMethod AWS CLICommand Reference.

`update-model`

Il seguente esempio di codice mostra come utilizzare`update-model`.

AWS CLI

Come modificare la descrizione di un modello in un'API

Comando:

```
aws apigateway update-model --rest-api-id 1234123412 --model-name 'Empty' --patch-
operations op=replace,path=/description,value='New Description'
```

Come modificare lo schema di un modello in un'API

Comando:

```
aws apigateway update-model --rest-api-id 1234123412 --model-name 'Empty' --patch-
operations op=replace,path=/schema,value='"{ \"$schema\": \"http://json-schema.org/
draft-04/schema#\", \"title\" : \"Empty Schema\", \"type\" : \"object\" }"'
```

- Per i dettagli sull'API, consulta UpdateModel AWS CLICommand Reference.

`update-resource`

Il seguente esempio di codice mostra come utilizzareu`pdate-resource`.

AWS CLI

Come spostare una risorsa e posizionarla in una risorsa principale diversa in un'API

Comando:

```
aws apigateway update-resource --rest-api-id 1234123412 --resource-id 1a2b3c --
patch-operations op=replace,path=/parentId,value='3c2b1a'
```

Output:

```
{
    "path": "/resource",
    "pathPart": "resource",
    "id": "1a2b3c",
    "parentId": "3c2b1a"
}
```

Come rinominare una risorsa (pathPart) in un'API

Comando:

```
aws apigateway update-resource --rest-api-id 1234123412 --resource-id 1a2b3c --
patch-operations op=replace,path=/pathPart,value=newresourcename
```

Output:

```
{
    "path": "/newresourcename",
    "pathPart": "newresourcename",
    "id": "1a2b3c",
    "parentId": "3c2b1a"
}
```

- Per i dettagli sull'API, consulta UpdateResource AWS CLICommand Reference.

`update-rest-api`

Il seguente esempio di codice mostra come utilizzare`update-rest-api`.

AWS CLI

Come modificare il nome di un'API

Comando:

```
aws apigateway update-rest-api --rest-api-id 1234123412 --patch-operations
 op=replace,path=/name,value='New Name'
```

Come modificare la descrizione di un'API

Comando:

```
aws apigateway update-rest-api --rest-api-id 1234123412 --patch-operations
 op=replace,path=/description,value='New Description'
```

- Per i dettagli sull'API, consulta UpdateRestApi AWS CLICommand Reference.

`update-stage`

Il seguente esempio di codice mostra come utilizzare`update-stage`.

AWS CLI

Esempio 1: come sostituire le impostazioni della fase per una risorsa e un metodo

L'`update-stage`esempio seguente sostituisce le impostazioni dello stage e disattiva la request/response registrazione completa per una risorsa e un metodo specifici.

```
aws apigateway update-stage \
    --rest-api-id 1234123412 \
    --stage-name 'dev' \
    --patch-operations op=replace,path=/~1resourceName/GET/logging/
dataTrace,value=false
```

Output:

```
{
    "deploymentId": "5ubd17",
    "stageName": "dev",
    "cacheClusterEnabled": false,
    "cacheClusterStatus": "NOT_AVAILABLE",
    "methodSettings": {
        "~1resourceName/GET": {
            "metricsEnabled": false,
            "dataTraceEnabled": false,
            "throttlingBurstLimit": 5000,
            "throttlingRateLimit": 10000.0,
            "cachingEnabled": false,
            "cacheTtlInSeconds": 300,
            "cacheDataEncrypted": false,
            "requireAuthorizationForCacheControl": true,
            "unauthorizedCacheControlHeaderStrategy": "SUCCEED_WITH_RESPONSE_HEADER"
        }
    },
    "tracingEnabled": false,
    "createdDate": "2022-07-18T10:11:18-07:00",
    "lastUpdatedDate": "2022-07-18T10:19:04-07:00"
}
```

Per ulteriori informazioni consulta Configurazione di una fase per una REST API in Gateway API nella Guida per gli sviluppatori di Gateway Amazon API.

Esempio 2: come aggiornare le impostazioni della fase per tutte le risorse e tutti i metodi di una fase API

L'`update-stage`esempio seguente attiva la request/response registrazione completa per tutte le risorse e i metodi di una fase API.

```
aws apigateway update-stage \
    --rest-api-id 1234123412 \
```

```
    --stage-name 'dev' \
    --patch-operations 'op=replace,path=/*/*/logging/dataTrace,value=true'
```

Output:

```
{
    "deploymentId": "5ubd17",
    "stageName": "dev",
    "cacheClusterEnabled": false,
    "cacheClusterStatus": "NOT_AVAILABLE",
    "methodSettings": {
        "*/*": {
            "metricsEnabled": false,
            "dataTraceEnabled": true,
            "throttlingBurstLimit": 5000,
            "throttlingRateLimit": 10000.0,
            "cachingEnabled": false,
            "cacheTtlInSeconds": 300,
            "cacheDataEncrypted": false,
            "requireAuthorizationForCacheControl": true,
            "unauthorizedCacheControlHeaderStrategy": "SUCCEED_WITH_RESPONSE_HEADER"
        }
    },
    "tracingEnabled": false,
    "createdDate": "2022-07-18T10:11:18-07:00",
    "lastUpdatedDate": "2022-07-18T10:31:04-07:00"
}
```

Per ulteriori informazioni consulta Configurazione di una fase per una REST API in Gateway API nella Guida per gli sviluppatori di Gateway Amazon API.

- Per i dettagli sull'API, consulta UpdateStage AWS CLICommand Reference.

`update-usage-plan`

Il seguente esempio di codice mostra come utilizzare`update-usage-plan`.

AWS CLI

Come modificare il periodo definito in un piano di utilizzo

Comando:

```
aws apigateway update-usage-plan --usage-plan-id a1b2c3 --patch-operations
 op="replace",path="/quota/period",value="MONTH"
```

Come modificare il limite di quota definito in un piano di utilizzo

Comando:

```
aws apigateway update-usage-plan --usage-plan-id a1b2c3 --patch-operations
 op="replace",path="/quota/limit",value="500"
```

Come modificare il limite di velocità di elaborazione delle richieste definito in un piano di utilizzo

Comando:

```
aws apigateway update-usage-plan --usage-plan-id a1b2c3 --patch-operations
 op="replace",path="/throttle/rateLimit",value="10"
```

Come modificare il limite di burst di elaborazione delle richieste definito in un piano di utilizzo

Comando:

```
aws apigateway update-usage-plan --usage-plan-id a1b2c3 --patch-operations
 op="replace",path="/throttle/burstLimit",value="20"
```

- Per i dettagli sull'API, consulta UpdateUsagePlan AWS CLICommand Reference.

`update-usage`

Il seguente esempio di codice mostra come utilizzare`update-usage`.

AWS CLI

Come modificare temporaneamente la quota su una chiave API per il periodo corrente definito nel piano di utilizzo

Comando:

```
aws apigateway update-usage --usage-plan-id a1b2c3 --key-
id 1NbjQzMReAkeEQPNAW8r3dXsU2rDD7fc7f2Sipnu --patch-operations op="replace",path="/
remaining",value="50"
```

- Per i dettagli sull'API, consulta UpdateUsage AWS CLICommand Reference.

`update-vpc-link`

Il seguente esempio di codice mostra come utilizzare`update-vpc-link`.

AWS CLI

Esempio 1: aggiornare il nome di un link VPC esistente

L'`update-vpc-link`esempio seguente aggiorna il nome del link VPC specificato.

```
aws apigateway update-vpc-link  \
    --vpc-link-id ab3de6 \
    --patch-operations op=replace,path=/name,value=my-vpc-link
```

Output:

```
{
    "id": "ab3de6",
    "name": "my-vpc-link",
    "targetArns": [
        "arn:aws:elasticloadbalancing:us-east-1:123456789012:loadbalancer/net/my-
lb/12a456s89aaa12345"
    ],
    "status": "AVAILABLE",
    "statusMessage": "Your vpc link is ready for use",
    "tags": {}
}
```

Per ulteriori informazioni, consulta Aggiornamento del collegamento VPC esistente nel riferimento ai comandi AWS CLI.

Esempio 2: aggiornare il nome e la descrizione di un link VPC esistente

L'`update-vpc-link`esempio seguente aggiorna il nome del link VPC specificato.

```
aws apigateway update-vpc-link  \
    --vpc-link-id ab3de6 \
    --patch-operations op=replace,path=/name,value=my-vpc-link op=replace,path=/
description,value="My custom description"
```

Output:

```
{
    "id": "ab3de6",
    "name": "my-vpc-link",
    "description": "My custom description",
    "targetArns": [
        "arn:aws:elasticloadbalancing:us-east-1:123456789012:loadbalancer/net/my-
lb/12a456s89aaa12345"
    ],
    "status": "AVAILABLE",
    "statusMessage": "Your vpc link is ready for use",
    "tags": {}
}
```

Per ulteriori informazioni, consulta Aggiornamento del collegamento VPC esistente nel riferimento ai comandi AWS CLI.

- Per i dettagli sull'API, consulta AWS CLI Command UpdateVpcLinkReference.

API Gateway HTTP e esempi di WebSocket API che utilizzano AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando HTTP e WebSocket API AWS Command Line Interface with API Gateway.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`create-api-mapping`

Il seguente esempio di codice mostra come utilizzare`create-api-mapping`.

AWS CLI

Come creare una mappatura API per un'API

L'esempio `create-api-mapping` seguente mappa la fase `test` di un'API al percorso `/myApi` del nome di dominio personalizzato `regional.example.com`.

```
aws apigatewayv2 create-api-mapping \
    --domain-name regional.example.com \
    --api-mapping-key myApi \
    --api-id a1b2c3d4 \
    --stage test
```

Output:

```
{
    "ApiId": "a1b2c3d4",
    "ApiMappingId": "0qzs2sy7bh",
    "ApiMappingKey": "myApi"
    "Stage": "test"
}
```

Per ulteriori informazioni, consulta Configurazione di un nome di dominio personalizzato regionale in Gateway API nella Guida per gli sviluppatori di Gateway Amazon API.

- Per i dettagli sull'API, consulta CreateApiMapping AWS CLICommand Reference.

`create-api`

Il seguente esempio di codice mostra come utilizzare`create-api`.

AWS CLI

Come creare un'API HTTP

L'esempio `create-api` seguente crea un'API HTTP utilizzando la funzione di creazione rapida. Puoi utilizzare la creazione rapida per creare un'API con un'integrazione AWS Lambda o HTTP, un percorso generico predefinito e una fase predefinita configurata per distribuire automaticamente le modifiche. Il comando seguente utilizza la funzione di creazione rapida per creare un'API che si integra con una funzione Lambda.

```
aws apigatewayv2 create-api \
```

```
    --name my-http-api \
    --protocol-type HTTP \
    --target arn:aws:lambda:us-west-2:123456789012:function:my-lambda-function
```

Output:

```
{
    "ApiEndpoint": "https://a1b2c3d4.execute-api.us-west-2.amazonaws.com",
    "ApiId": "a1b2c3d4",
    "ApiKeySelectionExpression": "$request.header.x-api-key",
    "CreatedDate": "2020-04-08T19:05:45+00:00",
    "Name": "my-http-api",
    "ProtocolType": "HTTP",
    "RouteSelectionExpression": "$request.method $request.path"
}
```

Per ulteriori informazioni, consulta Sviluppo di API HTTP in Gateway API nella Guida per gli sviluppatori di Gateway Amazon API.

Per creare un'API WebSocket

L'`create-api`esempio seguente crea un' WebSocket API con il nome specificato.

```
aws apigatewayv2 create-api \
    --name "myWebSocketApi" \
    --protocol-type WEBSOCKET \
    --route-selection-expression '$request.body.action'
```

Output:

```
{
    "ApiKeySelectionExpression": "$request.header.x-api-key",
    "Name": "myWebSocketApi",
    "CreatedDate": "2018-11-15T06:23:51Z",
    "ProtocolType": "WEBSOCKET",
    "RouteSelectionExpression": "'$request.body.action'",
    "ApiId": "aabbccddee"
}
```

Per ulteriori informazioni, consulta Creare un' WebSocket API in API Gateway nella Amazon API Gateway Developer Guide.

- Per i dettagli sull'API, consulta CreateApi AWS CLICommand Reference.

`create-authorizer`

Il seguente esempio di codice mostra come utilizzare`create-authorizer`.

AWS CLI

Come creare un sistema di autorizzazione JWT per un'API HTTP

L'esempio `create-authorizer` seguente crea un sistema di autorizzazione JWT che utilizza Amazon Cognito come gestore dell'identità.

```
aws apigatewayv2 create-authorizer \
    --name my-jwt-authorizer \
    --api-id a1b2c3d4 \
    --authorizer-type JWT \
    --identity-source '$request.header.Authorization' \
    --jwt-configuration Audience=123456abc,Issuer=https://cognito-idp.us-
west-2.amazonaws.com/us-west-2_abc123
```

Output:

```
{
    "AuthorizerId": "a1b2c3",
    "AuthorizerType": "JWT",
    "IdentitySource": [
        "$request.header.Authorization"
    ],
    "JwtConfiguration": {
        "Audience": [
            "123456abc"
        ],
        "Issuer": "https://cognito-idp.us-west-2.amazonaws.com/us-west-2_abc123"
    },
    "Name": "my-jwt-authorizer"
}
```

Per ulteriori informazioni, consulta Controllare l'accesso a HTTP APIs con gli autorizzatori JWT nella Amazon API Gateway Developer Guide.

- Per i dettagli sull'API, consulta AWS CLI Command CreateAuthorizerReference.

`create-deployment`

Il seguente esempio di codice mostra come utilizzare`create-deployment`.

AWS CLI

Come creare un'implementazione per un'API

L'esempio `create-deployment` seguente crea un'implementazione per un'API e la associa alla fase `dev` dell'API.

```
aws apigatewayv2 create-deployment \
    --api-id a1b2c3d4 \
    --stage-name dev
```

Output:

```
{
    "AutoDeployed": false,
    "CreatedDate": "2020-04-06T23:38:08Z",
    "DeploymentId": "53lz9l",
    "DeploymentStatus": "DEPLOYED"
}
```

Per ulteriori informazioni, consulta la definizione di Distribuzione API nella Guida per gli sviluppatori di Gateway Amazon API.

- Per i dettagli sull'API, consulta CreateDeployment AWS CLICommand Reference.

`create-domain-name`

Il seguente esempio di codice mostra come utilizzare`create-domain-name`.

AWS CLI

Come creare un nome di dominio personalizzato

L'esempio `create-domain-name` seguente crea un nome di dominio regionale personalizzato per un'API.

```
aws apigatewayv2 create-domain-name \
    --domain-name regional.example.com \
```

```
    --domain-name-configurations CertificateArn=arn:aws:acm:us-
west-2:123456789012:certificate/123456789012-1234-1234-1234-12345678
```

Output:

```
{
    "ApiMappingSelectionExpression": "$request.basepath",
    "DomainName": "regional.example.com",
    "DomainNameConfigurations": [
        {
            "ApiGatewayDomainName": "d-id.execute-api.us-west-2.amazonaws.com",
            "CertificateArn": "arn:aws:acm:us-
west-2:123456789012:certificate/123456789012-1234-1234-1234-12345678",
            "EndpointType": "REGIONAL",
            "HostedZoneId": "123456789111",
            "SecurityPolicy": "TLS_1_2",
            "DomainNameStatus": "AVAILABLE"
        }
    ]
}
```

Per ulteriori informazioni, consulta Configurazione di un nome di dominio personalizzato regionale in Gateway API nella Guida per gli sviluppatori di Gateway Amazon API.

- Per i dettagli sull'API, consulta CreateDomainName AWS CLICommand Reference.

`create-integration`

Il seguente esempio di codice mostra come utilizzare`create-integration`.

AWS CLI

Per creare un'integrazione WebSocket API

L'`create-integration`esempio seguente crea un'integrazione fittizia per un' WebSocket API.

```
aws apigatewayv2 create-integration \
    --api-id aabbccddee \
    --passthrough-behavior WHEN_NO_MATCH \
    --timeout-in-millis 29000 \
    --connection-type INTERNET \
    --integration-type MOCK
```

Output:

```
{
    "ConnectionType": "INTERNET",
    "IntegrationId": "0abcdef",
    "IntegrationResponseSelectionExpression": "${integration.response.statuscode}",
    "IntegrationType": "MOCK",
    "PassthroughBehavior": "WHEN_NO_MATCH",
    "PayloadFormatVersion": "1.0",
    "TimeoutInMillis": 29000
}
```

Per ulteriori informazioni, consulta Configurare una richiesta di integrazione WebSocket API in API Gateway nella Amazon API Gateway Developer Guide.

Come creare un'integrazione per un'API HTTP

L'`create-integration`esempio seguente crea un'integrazione AWS Lambda per un'API HTTP.

```
aws apigatewayv2 create-integration \
    --api-id a1b2c3d4 \
    --integration-type AWS_PROXY \
    --integration-uri arn:aws:lambda:us-west-2:123456789012:function:my-function \
    --payload-format-version 2.0
```

Output:

```
{
    "ConnectionType": "INTERNET",
    "IntegrationId": "0abcdef",
    "IntegrationMethod": "POST",
    "IntegrationType": "AWS_PROXY",
    "IntegrationUri": "arn:aws:lambda:us-west-2:123456789012:function:my-function",
    "PayloadFormatVersion": "2.0",
    "TimeoutInMillis": 30000
}
```

Per ulteriori informazioni, consulta Configurazione delle integrazioni per HTTP APIs nella Amazon API Gateway Developer Guide.

- Per i dettagli sull'API, consulta AWS CLI Command CreateIntegrationReference.

`create-route`

Il seguente esempio di codice mostra come utilizzare`create-route`.

AWS CLI

Per creare una route $default per un'API WebSocket o HTTP

L'`create-route`esempio seguente crea una `$default` route per un'API WebSocket o HTTP.

```
aws apigatewayv2 create-route \
    --api-id aabbccddee \
    --route-key '$default'
```

Output:

```
{
    "ApiKeyRequired": false,
    "AuthorizationType": "NONE",
    "RouteKey": "$default",
    "RouteId": "1122334"
}
```

Per ulteriori informazioni, consulta Working with routes for WebSocket APIs nella Amazon API Gateway Developer Guide

Come creare una route per un'API HTTP

L'esempio `create-route` seguente crea una route denominata `signup` che accetta richieste POST.

```
aws apigatewayv2 create-route \
    --api-id aabbccddee \
    --route-key 'POST /signup'
```

Output:

```
{
    "ApiKeyRequired": false,
    "AuthorizationType": "NONE",
    "RouteKey": "POST /signup",
    "RouteId": "1122334"
```

```
}
```

Per ulteriori informazioni, consulta Working with routes for HTTP APIs nella Amazon API Gateway Developer Guide

- Per i dettagli sull'API, consulta CreateRoute AWS CLICommand Reference.

`create-routing-rule`

Il seguente esempio di codice mostra come utilizzare`create-routing-rule`.

AWS CLI

Come creare una regola di routing

L'esempio `create-routing-rule` seguente crea una regola di routing con la priorità `50`.

```
aws apigatewayv2 create-routing-rule \
    --domain-name 'regional.example.com' \
    --priority 50 \
    --conditions '[ \
        { \
            "MatchBasePaths": { \
                "AnyOf": [ \
                    "PetStoreShopper" \
                ] \
            } \
        } \
    ]' \
    --actions '[ \
        { \
            "InvokeApi": { \
                "ApiId": "abcd1234", \
                "Stage": "prod" \
            } \
        } \
    ]'
```

Output:

```
{
    "Actions": [
```

```
        {
            "InvokeApi": {
                "ApiId": "abcd1234",
                "Stage": "prod",
                "StripBasePath": false
            }
        }
    ],
    "Conditions": [
        {
            "MatchBasePaths": {
                "AnyOf": [
                    "PetStoreShopper"
                ]
            }
        }
    ],
    "Priority": 50,
    "RoutingRuleArn": "arn:aws:apigateway:us-east-2:123456789012:/domainnames/
regional.example.com/routingrules/aaa111",
    "RoutingRuleId": "aaa111"
}
```

Per ulteriori informazioni, consulta Regole di routing per connettere le fasi dell'API a un nome di dominio personalizzato per REST APIs nella Amazon API Gateway Developer Guide.

- Per i dettagli sull'API, consulta CreateRoutingRule AWS CLICommand Reference.

`create-stage`

Il seguente esempio di codice mostra come utilizzare`create-stage`.

AWS CLI

Come creare una fase

L'esempio `create-stage` seguente crea una fase denominata dev per un'API.

```
aws apigatewayv2 create-stage \
    --api-id a1b2c3d4 \
    --stage-name dev
```

Output:

```
{
    "CreatedDate": "2020-04-06T23:23:46Z",
    "DefaultRouteSettings": {
        "DetailedMetricsEnabled": false
    },
    "LastUpdatedDate": "2020-04-06T23:23:46Z",
    "RouteSettings": {},
    "StageName": "dev",
    "StageVariables": {},
    "Tags": {}
}
```

Per ulteriori informazioni, consulta Working with stages for HTTP APIs nella Amazon API Gateway Developer Guide.

- Per i dettagli sull'API, consulta CreateStage AWS CLICommand Reference.

`create-vpc-link`

Il seguente esempio di codice mostra come utilizzare`create-vpc-link`.

AWS CLI

Come creare un link VPC per un'API HTTP

L'`create-vpc-link`esempio seguente crea un collegamento VPC per HTTP. APIs

```
aws apigatewayv2 create-vpc-link \
    --name MyVpcLink \
    --subnet-ids subnet-aaaa subnet-bbbb \
    --security-group-ids sg1234 sg5678
```

Output:

```
{
    "CreatedDate": "2020-04-07T00:11:46Z",
    "Name": "MyVpcLink",
    "SecurityGroupIds": [
        "sg1234",
        "sg5678"
    ],
    "SubnetIds": [
```

```
        "subnet-aaaa",
        "subnet-bbbb"
    ],
    "Tags": {},
    "VpcLinkId": "abcd123",
    "VpcLinkStatus": "PENDING",
    "VpcLinkStatusMessage": "VPC link is provisioning ENIs",
    "VpcLinkVersion": "V2"
}
```

Per ulteriori informazioni, consulta Working with VPC links for HTTP APIs nella Amazon API Gateway Developer Guide.

- Per i dettagli sull'API, consulta CreateVpcLink AWS CLICommand Reference.

`delete-access-log-settings`

Il seguente esempio di codice mostra come utilizzare`delete-access-log-settings`.

AWS CLI

Come disabilitare la registrazione degli accessi per un'API

L'esempio `delete-access-log-settings` seguente elimina le impostazioni del log degli accessi per la fase `$default` di un'API. Per disabilitare la registrazione degli accessi per una fase, elimina le impostazioni del relativo log degli accessi.

```
aws apigatewayv2 delete-access-log-settings \
    --api-id a1b2c3d4 \
    --stage-name '$default'
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Configurazione della registrazione dei log per le API HTTP in Gateway API nella Guida per gli sviluppatori di Gateway Amazon API.

- Per i dettagli sull'API, consulta DeleteAccessLogSettings AWS CLICommand Reference.

`delete-api-mapping`

Il seguente esempio di codice mostra come utilizzare`delete-api-mapping`.

AWS CLI

Come eliminare una mappatura API

L'esempio `delete-api-mapping` seguente elimina una mappatura API per il nome di dominio personalizzato `api.example.com`.

```
aws apigatewayv2 delete-api-mapping \
    --api-mapping-id a1b2c3 \
    --domain-name api.example.com
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Configurazione di un nome di dominio personalizzato regionale in Gateway API nella Guida per gli sviluppatori di Gateway Amazon API.

- Per i dettagli sull'API, consulta DeleteApiMapping AWS CLICommand Reference.

`delete-api`

Il seguente esempio di codice mostra come utilizzare`delete-api`.

AWS CLI

Come eliminare un'API

L'esempio `delete-api` seguente elimina un'API.

```
aws apigatewayv2 delete-api \
    --api-id a1b2c3d4
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Working with HTTP APIs e Working with WebSocket APIs nella Amazon API Gateway Developer Guide.

- Per i dettagli sull'API, consulta DeleteApi AWS CLICommand Reference.

`delete-authorizer`

Il seguente esempio di codice mostra come utilizzare`delete-authorizer`.

AWS CLI

Come eliminare un sistema di autorizzazione

L'esempio `delete-authorizer` seguente elimina un sistema di autorizzazione.

```
aws apigatewayv2 delete-authorizer \
    --api-id a1b2c3d4 \
    --authorizer-id a1b2c3
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Controllare l'accesso a HTTP APIs con gli autorizzatori JWT nella Amazon API Gateway Developer Guide.

- Per i dettagli sull'API, consulta AWS CLI Command DeleteAuthorizerReference.

`delete-cors-configuration`

Il seguente esempio di codice mostra come utilizzare`delete-cors-configuration`.

AWS CLI

Come eliminare la configurazione CORS per un'API HTTP

L'esempio `delete-cors-configuration` seguente disabilita CORS per un'API HTTP eliminando la relativa configurazione CORS.

```
aws apigatewayv2 delete-cors-configuration \
    --api-id a1b2c3d4
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Configurazione di CORS per le API HTTP in Gateway API nella Guida per gli sviluppatori di Gateway Amazon API.

- Per i dettagli sull'API, consulta DeleteCorsConfiguration AWS CLICommand Reference.

`delete-deployment`

Il seguente esempio di codice mostra come utilizzare`delete-deployment`.

AWS CLI

Come eliminare un'implementazione

L'esempio `delete-deployment` seguente elimina un'implementazione di un'API.

```
aws apigatewayv2 delete-deployment \
    --api-id a1b2c3d4 \
    --deployment-id a1b2c3
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta la definizione di Distribuzione API nella Guida per gli sviluppatori di Gateway Amazon API.

- Per i dettagli sull'API, consulta DeleteDeployment AWS CLICommand Reference.

`delete-domain-name`

Il seguente esempio di codice mostra come utilizzare`delete-domain-name`.

AWS CLI

Come eliminare un nome di dominio personalizzato

L'esempio `delete-domain-name` seguente elimina un nome di dominio personalizzato.

```
aws apigatewayv2 delete-domain-name \
    --domain-name api.example.com
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Configurazione di un nome di dominio personalizzato regionale in Gateway API nella Guida per gli sviluppatori di Gateway Amazon API.

- Per i dettagli sull'API, consulta DeleteDomainName AWS CLICommand Reference.

`delete-integration`

Il seguente esempio di codice mostra come utilizzare`delete-integration`.

AWS CLI

Come eliminare un'integrazione

L'esempio `delete-integration` seguente elimina un'integrazione API.

```
aws apigatewayv2 delete-integration \
    --api-id a1b2c3d4 \
    --integration-id a1b2c3
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Configurazione delle integrazioni per HTTP APIs e Configurazione delle integrazioni WebSocket API nella Amazon API Gateway Developer Guide.

- Per i dettagli sull'API, consulta Command DeleteIntegrationReference AWS CLI .

`delete-route-settings`

Il seguente esempio di codice mostra come utilizzare`delete-route-settings`.

AWS CLI

Come eliminare le impostazioni di una route

L'esempio `delete-route-settings` seguente elimina le impostazioni di routing per la route specificata.

```
aws apigatewayv2 delete-route-settings \
    --api-id a1b2c3d4 \
    --stage-name dev \
    --route-key 'GET /pets'
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Working with routes for HTTP APIs nella Amazon API Gateway Developer Guide.

- Per i dettagli sull'API, consulta DeleteRouteSettings AWS CLICommand Reference.

`delete-route`

Il seguente esempio di codice mostra come utilizzare`delete-route`.

AWS CLI

Come eliminare una route

L'esempio `delete-route` seguente elimina una route per l'API.

```
aws apigatewayv2 delete-route \
    --api-id a1b2c3d4 \
    --route-id a1b2c3
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Working with routes for HTTP APIs nella Amazon API Gateway Developer Guide.

- Per i dettagli sull'API, consulta DeleteRoute AWS CLICommand Reference.

`delete-routing-rule`

Il seguente esempio di codice mostra come utilizzare`delete-routing-rule`.

AWS CLI

Come eliminare una regola di routing

L'esempio `delete-routing-rule` seguente elimina una regola di routing per un nome di dominio personalizzato.

```
aws apigatewayv2 delete-routing-rule \
    --domain-name 'regional.example.com' \
    --routing-rule-id aaa111
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Regole di routing per connettere le fasi dell'API a un nome di dominio personalizzato per REST APIs nella Amazon API Gateway Developer Guide.

- Per i dettagli sull'API, consulta DeleteRoutingRule AWS CLICommand Reference.

`delete-stage`

Il seguente esempio di codice mostra come utilizzare`delete-stage`.

AWS CLI

Come eliminare una fase

L'esempio `delete-stage` seguente elimina la fase `test` di un'API.

```
aws apigatewayv2 delete-stage \
    --api-id a1b2c3d4 \
    --stage-name test
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Working with stages for HTTP APIs nella Amazon API Gateway Developer Guide.

- Per i dettagli sull'API, consulta DeleteStage AWS CLICommand Reference.

`delete-vpc-link`

Il seguente esempio di codice mostra come utilizzare`delete-vpc-link`.

AWS CLI

Come eliminare un link VPC per un'API HTTP

L'esempio `delete-vpc-link` seguente elimina un link VPC.

```
aws apigatewayv2 delete-vpc-link \
    --vpc-link-id abcd123
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Working with VPC links for HTTP APIs nella Amazon API Gateway Developer Guide.

- Per i dettagli sull'API, consulta DeleteVpcLink AWS CLICommand Reference.

`export-api`

Il seguente esempio di codice mostra come utilizzare`export-api`.

AWS CLI

Come esportare una definizione OpenAPI di un'API HTTP

L'esempio `export-api` seguente esporta una definizione OpenAPI 3.0 di una fase API denominata `prod` in un file YAML denominato `stage-definition.yaml`. Per impostazione predefinita il file di definizione esportato include le estensioni Gateway API.

```
aws apigatewayv2 export-api \
    --api-id a1b2c3d4 \
    --output-type YAML \
    --specification OAS30 \
    --stage-name prod \
    stage-definition.yaml
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Esportazione di API HTTP da Gateway API nella Guida per gli sviluppatori di Gateway Amazon API.

- Per i dettagli sull'API, consulta ExportApi AWS CLICommand Reference.

`get-api-mapping`

Il seguente esempio di codice mostra come utilizzare`get-api-mapping`.

AWS CLI

Come ottenere informazioni su una mappatura API per un nome di dominio personalizzato

L'esempio `get-api-mapping` seguente mostra le informazioni relative a una mappatura API per il nome di dominio personalizzato `api.example.com`.

```
aws apigatewayv2 get-api-mapping \
    --api-mapping-id a1b2c3 \
    --domain-name api.example.com
```

Output:

```
{
    "ApiId": "a1b2c3d4",
```

```
    "ApiMappingId": "a1b2c3d5",
    "ApiMappingKey": "myTestApi"
    "Stage": "test"
}
```

Per ulteriori informazioni, consulta Configurazione di un nome di dominio personalizzato regionale in Gateway API nella Guida per gli sviluppatori di Gateway Amazon API.

- Per i dettagli sull'API, consulta GetApiMapping AWS CLICommand Reference.

`get-api-mappings`

Il seguente esempio di codice mostra come utilizzare`get-api-mappings`.

AWS CLI

Come ottenere le mappature API per un nome di dominio personalizzato

L'esempio `get-api-mappings` seguente mostra l'elenco di tutte le mappature API per il nome di dominio personalizzato `api.example.com`.

```
aws apigatewayv2 get-api-mappings \
    --domain-name api.example.com
```

Output:

```
{
    "Items": [
        {
            "ApiId": "a1b2c3d4",
            "ApiMappingId": "a1b2c3d5",
            "ApiMappingKey": "myTestApi"
            "Stage": "test"
        },
        {
            "ApiId": "a5b6c7d8",
            "ApiMappingId": "a1b2c3d6",
            "ApiMappingKey": "myDevApi"
            "Stage": "dev"
        },
    ]
}
```

Per ulteriori informazioni, consulta Configurazione di un nome di dominio personalizzato regionale in Gateway API nella Guida per gli sviluppatori di Gateway Amazon API.

- Per i dettagli sull'API, consulta GetApiMappings AWS CLICommand Reference.

`get-api`

Il seguente esempio di codice mostra come utilizzare`get-api`.

AWS CLI

Come recuperare le informazioni relative a un'API

L'esempio `get-api` seguente mostra le informazioni relative a un'API.

```
aws apigatewayv2 get-api \
    --api-id a1b2c3d4
```

Output:

```
{
    "ApiEndpoint": "https://a1b2c3d4.execute-api.us-west-2.amazonaws.com",
    "ApiId": "a1b2c3d4",
    "ApiKeySelectionExpression": "$request.header.x-api-key",
    "CreatedDate": "2020-03-28T00:32:37Z",
    "Name": "my-api",
    "ProtocolType": "HTTP",
    "RouteSelectionExpression": "$request.method $request.path",
    "Tags": {
        "department": "finance"
    }
}
```

- Per i dettagli sull'API, consulta GetApi AWS CLICommand Reference.

`get-apis`

Il seguente esempio di codice mostra come utilizzare`get-apis`.

AWS CLI

Per recuperare un elenco di APIs

L'`get-apis`esempio seguente elenca tutti i dati APIs relativi all'utente corrente.

```
aws apigatewayv2 get-apis
```

Output:

```
{
    "Items": [
        {
            "ApiEndpoint": "wss://a1b2c3d4.execute-api.us-west-2.amazonaws.com",
            "ApiId": "a1b2c3d4",
            "ApiKeySelectionExpression": "$request.header.x-api-key",
            "CreatedDate": "2020-04-07T20:21:59Z",
            "Name": "my-websocket-api",
            "ProtocolType": "WEBSOCKET",
            "RouteSelectionExpression": "$request.body.message",
            "Tags": {}
        },
        {
            "ApiEndpoint": "https://a1b2c3d5.execute-api.us-west-2.amazonaws.com",
            "ApiId": "a1b2c3d5",
            "ApiKeySelectionExpression": "$request.header.x-api-key",
            "CreatedDate": "2020-04-07T20:23:50Z",
            "Name": "my-http-api",
            "ProtocolType": "HTTP",
            "RouteSelectionExpression": "$request.method $request.path",
            "Tags": {}
        }
    ]
}
```

Per ulteriori informazioni, consulta Working with HTTP APIs e Working with WebSocket APIs nella Amazon API Gateway Developer Guide.

- Per i dettagli sull'API, consulta GetApis AWS CLICommand Reference.

`get-authorizer`

Il seguente esempio di codice mostra come utilizzare`get-authorizer`.

AWS CLI

Come recuperare le informazioni relative a un sistema di autorizzazione

L'esempio `get-authorizer` seguente mostra le informazioni relative a un sistema di autorizzazione.

```
aws apigatewayv2 get-authorizer \
    --api-id a1b2c3d4 \
    --authorizer-id a1b2c3
```

Output:

```
{
    "AuthorizerId": "a1b2c3",
    "AuthorizerType": "JWT",
    "IdentitySource": [
        "$request.header.Authorization"
    ],
    "JwtConfiguration": {
        "Audience": [
            "123456abc"
        ],
        "Issuer": "https://cognito-idp.us-west-2.amazonaws.com/us-west-2_abc123"
    },
    "Name": "my-jwt-authorizer"
}
```

Per ulteriori informazioni, consulta Controllare l'accesso a HTTP APIs con gli autorizzatori JWT nella Amazon API Gateway Developer Guide.

- Per i dettagli sull'API, consulta AWS CLI Command GetAuthorizerReference.

`get-authorizers`

Il seguente esempio di codice mostra come utilizzare`get-authorizers`.

AWS CLI

Come recuperare un elenco di sistemi di autorizzazione per un'API

L'esempio `get-authorizers` seguente mostra l'elenco di tutti i sistemi di autorizzazione per un'API.

```
aws apigatewayv2 get-authorizers \
```

```
    --api-id a1b2c3d4
```

Output:

```
{
    "Items": [
        {
            "AuthorizerId": "a1b2c3",
            "AuthorizerType": "JWT",
            "IdentitySource": [
                "$request.header.Authorization"
            ],
            "JwtConfiguration": {
                "Audience": [
                    "123456abc"
                ],
                "Issuer": "https://cognito-idp.us-west-2.amazonaws.com/us-
west-2_abc123"
            },
            "Name": "my-jwt-authorizer"
        },
        {
            "AuthorizerId": "a1b2c4",
            "AuthorizerType": "JWT",
            "IdentitySource": [
                "$request.header.Authorization"
            ],
            "JwtConfiguration": {
                "Audience": [
                    "6789abcde"
                ],
                "Issuer": "https://cognito-idp.us-west-2.amazonaws.com/us-
west-2_abc234"
            },
            "Name": "new-jwt-authorizer"
        }
    ]
}
```

Per ulteriori informazioni, consulta Controllare l'accesso a HTTP APIs con gli autorizzatori JWT nella Amazon API Gateway Developer Guide.

- Per i dettagli sull'API, consulta AWS CLI Command GetAuthorizersReference.

`get-deployment`

Il seguente esempio di codice mostra come utilizzare`get-deployment`.

AWS CLI

Come recuperare le informazioni relative a un'implementazione

L'esempio `get-deployment` seguente mostra le informazioni relative a un'implementazione.

```
aws apigatewayv2 get-deployment \
    --api-id a1b2c3d4 \
    --deployment-id abcdef
```

Output:

```
{
    "AutoDeployed": true,
    "CreatedDate": "2020-04-07T23:58:40Z",
    "DeploymentId": "abcdef",
    "DeploymentStatus": "DEPLOYED",
    "Description": "Automatic deployment triggered by changes to the Api
 configuration"
}
```

Per ulteriori informazioni, consulta la definizione di Distribuzione API nella Guida per gli sviluppatori di Gateway Amazon API.

- Per i dettagli sull'API, consulta GetDeployment AWS CLICommand Reference.

`get-deployments`

Il seguente esempio di codice mostra come utilizzare`get-deployments`.

AWS CLI

Come recuperare un elenco di implementazioni

L'esempio `get-deployments` seguente mostra l'elenco di tutte le implementazioni di un'API.

```
aws apigatewayv2 get-deployments \
    --api-id a1b2c3d4
```

Output:

```
{
    "Items": [
        {
            "AutoDeployed": true,
            "CreatedDate": "2020-04-07T23:58:40Z",
            "DeploymentId": "abcdef",
            "DeploymentStatus": "DEPLOYED",
            "Description": "Automatic deployment triggered by changes to the Api
 configuration"
        },
        {
            "AutoDeployed": true,
            "CreatedDate": "2020-04-06T00:33:00Z",
            "DeploymentId": "bcdefg",
            "DeploymentStatus": "DEPLOYED",
            "Description": "Automatic deployment triggered by changes to the Api
 configuration"
        }
    ]
}
```

Per ulteriori informazioni, consulta la definizione di Distribuzione API nella Guida per gli sviluppatori di Gateway Amazon API.

- Per i dettagli sull'API, consulta GetDeployments AWS CLICommand Reference.

`get-domain-name`

Il seguente esempio di codice mostra come utilizzare`get-domain-name`.

AWS CLI

Come recuperare le informazioni relative a un nome di dominio personalizzato

L'esempio `get-domain-name` seguente mostra le informazioni relative a un nome di dominio personalizzato.

```
aws apigatewayv2 get-domain-name \
    --domain-name api.example.com
```

Output:

```
{
    "ApiMappingSelectionExpression": "$request.basepath",
    "DomainName": "api.example.com",
    "DomainNameConfigurations": [
        {
            "ApiGatewayDomainName": "d-1234.execute-api.us-west-2.amazonaws.com",
            "CertificateArn": "arn:aws:acm:us-
west-2:123456789012:certificate/123456789012-1234-1234-1234-12345678",
            "EndpointType": "REGIONAL",
            "HostedZoneId": "123456789111",
            "SecurityPolicy": "TLS_1_2",
            "DomainNameStatus": "AVAILABLE"
        }
    ],
    "Tags": {}
}
```

Per ulteriori informazioni, consulta Configurazione di un nome di dominio personalizzato regionale in Gateway API nella Guida per gli sviluppatori di Gateway Amazon API.

- Per i dettagli sull'API, consulta GetDomainName AWS CLICommand Reference.

`get-domain-names`

Il seguente esempio di codice mostra come utilizzare`get-domain-names`.

AWS CLI

Come recuperare un elenco di nomi di dominio personalizzati

L'esempio `get-domain-names` seguente mostra l'elenco di tutti i nomi di dominio personalizzati per l'utente attuale.

```
aws apigatewayv2 get-domain-names
```

Output:

```
{
    "Items": [
```

```
        {
            "ApiMappingSelectionExpression": "$request.basepath",
            "DomainName": "api.example.com",
            "DomainNameConfigurations": [
                {
                    "ApiGatewayDomainName": "d-1234.execute-api.us-
west-2.amazonaws.com",
                    "CertificateArn": "arn:aws:acm:us-
west-2:123456789012:certificate/123456789012-1234-1234-1234-12345678",
                    "EndpointType": "REGIONAL",
                    "HostedZoneId": "123456789111",
                    "SecurityPolicy": "TLS_1_2",
                    "DomainNameStatus": "AVAILABLE"
                }
            ]
        },
        {
            "ApiMappingSelectionExpression": "$request.basepath",
            "DomainName": "newApi.example.com",
            "DomainNameConfigurations": [
                {
                    "ApiGatewayDomainName": "d-5678.execute-api.us-
west-2.amazonaws.com",
                    "CertificateArn": "arn:aws:acm:us-
west-2:123456789012:certificate/123456789012-1234-1234-1234-12345678",
                    "EndpointType": "REGIONAL",
                    "HostedZoneId": "123456789222",
                    "SecurityPolicy": "TLS_1_2",
                    "DomainNameStatus": "AVAILABLE"
                }
            ]
        }
    ]
}
```

Per ulteriori informazioni, consulta Configurazione di un nome di dominio personalizzato regionale in Gateway API nella Guida per gli sviluppatori di Gateway Amazon API.

- Per i dettagli sull'API, consulta GetDomainNames AWS CLICommand Reference.

`get-integration`

Il seguente esempio di codice mostra come utilizzare`get-integration`.

AWS CLI

Come recuperare le informazioni relative a un'integrazione

L'esempio `get-integration` seguente mostra le informazioni relative a un'integrazione.

```
aws apigatewayv2 get-integration \
    --api-id a1b2c3d4 \
    --integration-id a1b2c3
```

Output:

```
{
    "ApiGatewayManaged": true,
    "ConnectionType": "INTERNET",
    "IntegrationId": "a1b2c3",
    "IntegrationMethod": "POST",
    "IntegrationType": "AWS_PROXY",
    "IntegrationUri": "arn:aws:lambda:us-west-2:12356789012:function:hello12",
    "PayloadFormatVersion": "2.0",
    "TimeoutInMillis": 30000
}
```

Per ulteriori informazioni, consulta Configurazione delle integrazioni per HTTP APIs e Configurazione delle integrazioni WebSocket API nella Amazon API Gateway Developer Guide.

- Per i dettagli sull'API, consulta Command GetIntegrationReference AWS CLI .

`get-integrations`

Il seguente esempio di codice mostra come utilizzare`get-integrations`.

AWS CLI

Come recuperare un elenco di integrazioni

L'esempio `get-integrations` seguente mostra l'elenco di tutte le integrazioni di un'API.

```
aws apigatewayv2 get-integrations \
    --api-id a1b2c3d4
```

Output:

```
{
    "Items": [
        {
            "ApiGatewayManaged": true,
            "ConnectionType": "INTERNET",
            "IntegrationId": "a1b2c3",
            "IntegrationMethod": "POST",
            "IntegrationType": "AWS_PROXY",
            "IntegrationUri": "arn:aws:lambda:us-west-2:123456789012:function:my-
function",
            "PayloadFormatVersion": "2.0",
            "TimeoutInMillis": 30000
        },
        {
            "ConnectionType": "INTERNET",
            "IntegrationId": "a1b2c4",
            "IntegrationMethod": "ANY",
            "IntegrationType": "HTTP_PROXY",
            "IntegrationUri": "https://www.example.com",
            "PayloadFormatVersion": "1.0",
            "TimeoutInMillis": 30000
        }
    ]
}
```

Per ulteriori informazioni, consulta Configurazione delle integrazioni per HTTP APIs e Configurazione delle integrazioni WebSocket API nella Amazon API Gateway Developer Guide.

- Per i dettagli sull'API, consulta Command GetIntegrationsReference AWS CLI .

`get-route`

Il seguente esempio di codice mostra come utilizzare`get-route`.

AWS CLI

Come recuperare le informazioni relative a una route

L'esempio `get-route` seguente mostra le informazioni relative a una route.

```
aws apigatewayv2 get-route \
    --api-id a1b2c3d4 \
```

```
    --route-id 72jz1wk
```

Output:

```
{
    "ApiKeyRequired": false,
    "AuthorizationType": "NONE",
    "RouteId": "72jz1wk",
    "RouteKey": "ANY /pets",
    "Target": "integrations/a1b2c3"
}
```

Per ulteriori informazioni, consulta Working with routes for HTTP APIs nella Amazon API Gateway Developer Guide

- Per i dettagli sull'API, consulta GetRoute AWS CLICommand Reference.

`get-routes`

Il seguente esempio di codice mostra come utilizzare`get-routes`.

AWS CLI

Come recuperare un elenco di route

L'esempio `get-routes` seguente mostra l'elenco di tutte le route di un'API.

```
aws apigatewayv2 get-routes \
    --api-id a1b2c3d4
```

Output:

```
{
    "Items": [
        {
            "ApiKeyRequired": false,
            "AuthorizationType": "NONE",
            "RouteId": "72jz1wk",
            "RouteKey": "ANY /admin",
            "Target": "integrations/a1b2c3"
        },
        {
```

```
            "ApiGatewayManaged": true,
            "ApiKeyRequired": false,
            "AuthorizationType": "NONE",
            "RouteId": "go65gqi",
            "RouteKey": "$default",
            "Target": "integrations/a1b2c4"
        }
    ]
}
```

Per ulteriori informazioni, consulta Working with routes for HTTP APIs nella Amazon API Gateway Developer Guide

- Per i dettagli sull'API, consulta GetRoutes AWS CLICommand Reference.

`get-routing-rule`

Il seguente esempio di codice mostra come utilizzare`get-routing-rule`.

AWS CLI

Come ottenere una regola di routing

L'esempio `get-routing-rule` seguente ottiene una regola di routing per un nome di dominio.

```
aws apigatewayv2 get-routing-rule \
    --domain-name 'regional.example.com' \
    --routing-rule-id aaa111
```

Output:

```
{
    "Actions": [
        {
            "InvokeApi": {
                "ApiId": "abcd1234",
                "Stage": "prod",
                "StripBasePath": false
            }
        }
    ],
    "Conditions": [
```

```
        {
            "MatchBasePaths": {
                "AnyOf": [
                    "PetStoreShopper"
                ]
            }
        }
    ],
    "Priority": 50,
    "RoutingRuleArn": "arn:aws:apigateway:us-east-2:123456789012:/domainnames/
regional.example.com/routingrules/aaa111",
    "RoutingRuleId": "aaa111"
}
```

Per ulteriori informazioni, consulta Regole di routing per connettere le fasi dell'API a un nome di dominio personalizzato per REST APIs nella Amazon API Gateway Developer Guide.

- Per i dettagli sull'API, consulta GetRoutingRule AWS CLICommand Reference.

`get-stage`

Il seguente esempio di codice mostra come utilizzare`get-stage`.

AWS CLI

Come recuperare le informazioni relative a una fase

L'esempio `get-stage` seguente mostra le informazioni relative alla fase `prod` di un'API.

```
aws apigatewayv2 get-stage \
    --api-id a1b2c3d4 \
    --stage-name prod
```

Output:

```
{
    "CreatedDate": "2020-04-08T00:36:05Z",
    "DefaultRouteSettings": {
        "DetailedMetricsEnabled": false
    },
    "DeploymentId": "x1zwyv",
    "LastUpdatedDate": "2020-04-08T00:36:13Z",
```

```
    "RouteSettings": {},
    "StageName": "prod",
    "StageVariables": {
        "function": "my-prod-function"
    },
    "Tags": {}
}
```

Per ulteriori informazioni, consulta Working with stages for HTTP APIs nella Amazon API Gateway Developer Guide.

- Per i dettagli sull'API, consulta GetStage AWS CLICommand Reference.

`get-stages`

Il seguente esempio di codice mostra come utilizzare`get-stages`.

AWS CLI

Come recuperare un elenco di fasi

L'esempio `get-stages` seguente elenca tutte le fasi di un'API.

```
aws apigatewayv2 get-stages \
    --api-id a1b2c3d4
```

Output:

```
{
    "Items": [
        {
            "ApiGatewayManaged": true,
            "AutoDeploy": true,
            "CreatedDate": "2020-04-08T00:08:44Z",
            "DefaultRouteSettings": {
                "DetailedMetricsEnabled": false
            },
            "DeploymentId": "dty748",
            "LastDeploymentStatusMessage": "Successfully deployed stage with
 deployment ID 'dty748'",
            "LastUpdatedDate": "2020-04-08T00:09:49Z",
            "RouteSettings": {},
```

```
            "StageName": "$default",
            "StageVariables": {},
            "Tags": {}
        },
        {
            "AutoDeploy": true,
            "CreatedDate": "2020-04-08T00:35:06Z",
            "DefaultRouteSettings": {
                "DetailedMetricsEnabled": false
            },
            "LastUpdatedDate": "2020-04-08T00:35:48Z",
            "RouteSettings": {},
            "StageName": "dev",
            "StageVariables": {
                "function": "my-dev-function"
            },
            "Tags": {}
        },
        {
            "CreatedDate": "2020-04-08T00:36:05Z",
            "DefaultRouteSettings": {
                "DetailedMetricsEnabled": false
            },
            "DeploymentId": "x1zwyv",
            "LastUpdatedDate": "2020-04-08T00:36:13Z",
            "RouteSettings": {},
            "StageName": "prod",
            "StageVariables": {
                "function": "my-prod-function"
            },
            "Tags": {}
        }
    ]
}
```

Per ulteriori informazioni, consulta Working with stages for HTTP APIs nella Amazon API Gateway Developer Guide.

- Per i dettagli sull'API, consulta GetStages AWS CLICommand Reference.

`get-tags`

Il seguente esempio di codice mostra come utilizzare`get-tags`.

AWS CLI

Come recuperare un elenco di tag per una risorsa

L'esempio `get-tags` seguente elenca tutti i tag di un'API.

```
aws apigatewayv2 get-tags \
    --resource-arn arn:aws:apigateway:us-west-2::/apis/a1b2c3d4
```

Output:

```
{
    "Tags": {
        "owner": "dev-team",
        "environment": "prod"
    }
}
```

Per ulteriori informazioni, consulta Tagging delle risorse API Gateway nella Guida per gli sviluppatori di Gateway Amazon API.

- Per i dettagli sull'API, consulta GetTags AWS CLICommand Reference.

`get-vpc-link`

Il seguente esempio di codice mostra come utilizzare`get-vpc-link`.

AWS CLI

Come recuperare le informazioni relative a un link VPC

L'esempio `get-vpc-link` seguente mostra le informazioni relative a un collegamento VPC.

```
aws apigatewayv2 get-vpc-link \
    --vpc-link-id abcd123
```

Output:

```
{
    "CreatedDate": "2020-04-07T00:27:47Z",
    "Name": "MyVpcLink",
    "SecurityGroupIds": [
```

```
        "sg1234",
        "sg5678"
    ],
    "SubnetIds": [
        "subnet-aaaa",
        "subnet-bbbb"
    ],
    "Tags": {},
    "VpcLinkId": "abcd123",
    "VpcLinkStatus": "AVAILABLE",
    "VpcLinkStatusMessage": "VPC link is ready to route traffic",
    "VpcLinkVersion": "V2"
}
```

Per ulteriori informazioni, consulta Working with VPC links for HTTP APIs nella Amazon API Gateway Developer Guide.

- Per i dettagli sull'API, consulta GetVpcLink AWS CLICommand Reference.

`get-vpc-links`

Il seguente esempio di codice mostra come utilizzare`get-vpc-links`.

AWS CLI

Come recuperare un elenco di link VPC

L'esempio `get-vpc-links` seguente mostra l'elenco di tutti i collegamenti VPC per l'utente attuale.

```
aws apigatewayv2 get-vpc-links
```

Output:

```
{
    "Items": [
        {
            "CreatedDate": "2020-04-07T00:27:47Z",
            "Name": "MyVpcLink",
            "SecurityGroupIds": [
                "sg1234",
                "sg5678"
```

```
            ],
            "SubnetIds": [
                "subnet-aaaa",
                "subnet-bbbb"
            ],
            "Tags": {},
            "VpcLinkId": "abcd123",
            "VpcLinkStatus": "AVAILABLE",
            "VpcLinkStatusMessage": "VPC link is ready to route traffic",
            "VpcLinkVersion": "V2"
        }
        {
            "CreatedDate": "2020-04-07T00:27:47Z",
            "Name": "MyOtherVpcLink",
            "SecurityGroupIds": [
                "sg1234",
                "sg5678"
            ],
            "SubnetIds": [
                "subnet-aaaa",
                "subnet-bbbb"
            ],
            "Tags": {},
            "VpcLinkId": "abcd456",
            "VpcLinkStatus": "AVAILABLE",
            "VpcLinkStatusMessage": "VPC link is ready to route traffic",
            "VpcLinkVersion": "V2"
        }
    ]
}
```

Per ulteriori informazioni, consulta Working with VPC links for HTTP APIs nella Amazon API Gateway Developer Guide.

- Per i dettagli sull'API, consulta GetVpcLinks AWS CLICommand Reference.

`import-api`

Il seguente esempio di codice mostra come utilizzare`import-api`.

AWS CLI

Come importare un'API HTTP

L'esempio `import-api` seguente crea un'API HTTP da un file di definizione OpenAPI 3.0 denominato `api-definition.yaml`.

```
aws apigatewayv2 import-api \
    --body file://api-definition.yaml
```

Contenuto di `api-definition.yaml`:

```
openapi: 3.0.1
info:
    title: My Lambda API
    version: v1.0
paths:
    /hello:
        x-amazon-apigateway-any-method:
            x-amazon-apigateway-integration:
                payloadFormatVersion: 2.0
                type: aws_proxy
                httpMethod: POST
                uri: arn:aws:apigateway:us-west-2:lambda:path/2015-03-31/functions/
arn:aws:lambda:us-west-2:123456789012:function:hello/invocations
                connectionType: INTERNET
```

Output:

```
{
    "ApiEndpoint": "https://a1b2c3d4.execute-api.us-west-2.amazonaws.com",
    "ApiId": "a1b2c3d4",
    "ApiKeySelectionExpression": "$request.header.x-api-key",
    "CreatedDate": "2020-04-08T17:19:38+00:00",
    "Name": "My Lambda API",
    "ProtocolType": "HTTP",
    "RouteSelectionExpression": "$request.method $request.path",
    "Tags": {},
    "Version": "v1.0"
}
```

Per ulteriori informazioni, consulta Lavorare con le definizioni di OpenAPI per HTTP APIs nella Amazon API Gateway Developer Guide.

- Per i dettagli sull'API, consulta ImportApi AWS CLICommand Reference.

`list-routing-rules`

Il seguente esempio di codice mostra come utilizzare`list-routing-rules`.

AWS CLI

Come elencare le regole di routing

L'esempio `list-routing-rules` seguente elenca le regole di routing per un nome di dominio.

```
aws apigatewayv2 list-routing-rules \
    --domain-name 'regional.example.com'
```

Output:

```
{
    "RoutingRules": [
        {
            "Actions": [
                {
                    "InvokeApi": {
                        "ApiId": "abcd1234",
                        "Stage": "prod",
                        "StripBasePath": false
                    }
                }
            ],
            "Conditions": [
                {
                    "MatchBasePaths": {
                        "AnyOf": [
                            "PetStoreShopper"
                        ]
                    }
                }
            ],
            "Priority": 150,
            "RoutingRuleArn": "arn:aws:apigateway:us-east-1:123456789012:/
domainnames/regional.example.com/routingrules/aaa111",
            "RoutingRuleId": "aaa111"
        }
    ]
}
```

Per ulteriori informazioni, consulta Regole di routing per connettere le fasi dell'API a un nome di dominio personalizzato per REST APIs nella Amazon API Gateway Developer Guide.

- Per i dettagli sull'API, consulta ListRoutingRules AWS CLICommand Reference.

`put-routing-rule`

Il seguente esempio di codice mostra come utilizzare`put-routing-rule`.

AWS CLI

Come aggiornare una regola di routing

L'esempio `put-routing-rule` seguente aggiorna la priorità di una regola di routing.

```
aws apigatewayv2 put-routing-rule \
    --domain-name 'regional.example.com' \
    --priority 150 \
    --conditions '[ \
        { \
            "MatchBasePaths": { \
                "AnyOf": [ \
                    "PetStoreShopper" \
                ] \
            } \
        } \
    ]' \
    --actions '[ \
        { \
            "InvokeApi": { \
                "ApiId": "abcd1234", \
                "Stage": "prod" \
            } \
        } \
    ]'
```

Output:

```
{
    "Actions": [
        {
            "InvokeApi": {
                "ApiId": "abcd1234",
```

```
                "Stage": "prod",
                "StripBasePath": false
            }
        }
    ],
    "Conditions": [
        {
            "MatchBasePaths": {
                "AnyOf": [
                    "PetStoreShopper"
                ]
            }
        }
    ],
    "Priority": 150,
    "RoutingRuleArn": "arn:aws:apigateway:us-east-2:123456789012:/domainnames/
regional.example.com/routingrules/aaa111",
    "RoutingRuleId": "aaa111"
}
```

Per ulteriori informazioni, consulta Regole di routing per connettere le fasi dell'API a un nome di dominio personalizzato per REST APIs nella Amazon API Gateway Developer Guide.

- Per i dettagli sull'API, consulta PutRoutingRule AWS CLICommand Reference.

`reimport-api`

Il seguente esempio di codice mostra come utilizzare`reimport-api`.

AWS CLI

Come reimportare un'API HTTP

L'esempio `reimport-api` seguente aggiorna un'API HTTP esistente in modo che venga utilizzata la definizione OpenAPI 3.0 specificata in `api-definition.yaml`.

```
aws apigatewayv2 reimport-api \
    --body file://api-definition.yaml \
    --api-id a1b2c3d4
```

Contenuto di `api-definition.yaml`:

```
openapi: 3.0.1
```

```
info:
    title: My Lambda API
    version: v1.0
paths:
    /hello:
        x-amazon-apigateway-any-method:
            x-amazon-apigateway-integration:
                payloadFormatVersion: 2.0
                type: aws_proxy
                httpMethod: POST
                uri: arn:aws:apigateway:us-west-2:lambda:path/2015-03-31/functions/
arn:aws:lambda:us-west-2:12356789012:function:hello/invocations
                connectionType: INTERNET
```

Output:

```
{
    "ApiEndpoint": "https://a1b2c3d4.execute-api.us-west-2.amazonaws.com",
    "ApiId": "a1b2c3d4",
    "ApiKeySelectionExpression": "$request.header.x-api-key",
    "CreatedDate": "2020-04-08T17:19:38+00:00",
    "Name": "My Lambda API",
    "ProtocolType": "HTTP",
    "RouteSelectionExpression": "$request.method $request.path",
    "Tags": {},
    "Version": "v1.0"
}
```

Per ulteriori informazioni, consulta Lavorare con le definizioni di OpenAPI per HTTP APIs nella Amazon API Gateway Developer Guide.

- Per i dettagli sull'API, consulta ReimportApi AWS CLICommand Reference.

`tag-resource`

Il seguente esempio di codice mostra come utilizzare`tag-resource`.

AWS CLI

Come taggare a una risorsa

L'esempio `tag-resource` seguente aggiunge un tag con il nome di chiave `Department` e il valore `Accounting` all'API specificata.

```
aws apigatewayv2 tag-resource \
    --resource-arn arn:aws:apigateway:us-west-2::/apis/a1b2c3d4 \
    --tags Department=Accounting
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Tagging delle risorse API Gateway nella Guida per gli sviluppatori di Gateway Amazon API.

- Per i dettagli sull'API, consulta TagResource AWS CLICommand Reference.

`untag-resource`

Il seguente esempio di codice mostra come utilizzare`untag-resource`.

AWS CLI

Come rimuovere tag da una risorsa

L'esempio `untag-resource` seguente rimuove i tag con i nomi di chiave `Project` e `Owner` dall'API specificata.

```
aws apigatewayv2 untag-resource \
    --resource-arn arn:aws:apigateway:us-west-2::/apis/a1b2c3d4 \
    --tag-keys Project Owner
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Tagging delle risorse API Gateway nella Guida per gli sviluppatori di Gateway Amazon API.

- Per i dettagli sull'API, consulta UntagResource AWS CLICommand Reference.

`update-api-mapping`

Il seguente esempio di codice mostra come utilizzare`update-api-mapping`.

AWS CLI

Come aggiornare una mappatura API

L'esempio `update-api-mapping` seguente modifica una mappatura API per un nome di dominio personalizzato. Di conseguenza, l'URL di base che utilizza il nome del dominio personalizzato per l'API e la fase specificate diventa `https://api.example.com/dev`.

```
aws apigatewayv2 update-api-mapping \
    --api-id a1b2c3d4 \
    --stage dev \
    --domain-name api.example.com \
    --api-mapping-id 0qzs2sy7bh \
    --api-mapping-key dev
```

Output:

```
{
    "ApiId": "a1b2c3d4",
    "ApiMappingId": "0qzs2sy7bh",
    "ApiMappingKey": "dev"
    "Stage": "dev"
}
```

Per ulteriori informazioni, consulta Configurazione di un nome di dominio personalizzato regionale in Gateway API nella Guida per gli sviluppatori di Gateway Amazon API.

- Per i dettagli sull'API, consulta UpdateApiMapping AWS CLICommand Reference.

`update-api`

Il seguente esempio di codice mostra come utilizzare`update-api`.

AWS CLI

Come abilitare CORS per un'API HTTP

L'esempio `update-api` seguente aggiorna la configurazione CORS dell'API specificata per consentire le richieste da `https://www.example.com`.

```
aws apigatewayv2 update-api \
    --api-id a1b2c3d4 \
    --cors-configuration AllowOrigins=https://www.example.com
```

Output:

```
{
    "ApiEndpoint": "https://a1b2c3d4.execute-api.us-west-2.amazonaws.com",
    "ApiId": "a1b2c3d4",
    "ApiKeySelectionExpression": "$request.header.x-api-key",
    "CorsConfiguration": {
        "AllowCredentials": false,
        "AllowHeaders": [
            "header1",
            "header2"
        ],
        "AllowMethods": [
            "GET",
            "OPTIONS"
        ],
        "AllowOrigins": [
            "https://www.example.com"
        ]
    },
    "CreatedDate": "2020-04-08T18:39:37+00:00",
    "Name": "my-http-api",
    "ProtocolType": "HTTP",
    "RouteSelectionExpression": "$request.method $request.path",
    "Tags": {},
    "Version": "v1.0"
}
```

Per ulteriori informazioni, consulta Configurazione di CORS per le API HTTP in Gateway API nella Guida per gli sviluppatori di Gateway Amazon API.

- Per i dettagli sull'API, consulta UpdateApi AWS CLICommand Reference.

`update-authorizer`

Il seguente esempio di codice mostra come utilizzare`update-authorizer`.

AWS CLI

Come aggiornare un sistema di autorizzazione

L'esempio `update-authorizer` seguente modifica l'origine dell'identità di un sistema di autorizzazione JWT in un'intestazione denominata `Authorization`.

```
aws apigatewayv2 update-authorizer \
```

```
    --api-id a1b2c3d4 \
    --authorizer-id a1b2c3 \
    --identity-source '$request.header.Authorization'
```

Output:

```
{
    "AuthorizerId": "a1b2c3",
    "AuthorizerType": "JWT",
    "IdentitySource": [
        "$request.header.Authorization"
    ],
    "JwtConfiguration": {
        "Audience": [
            "123456abc"
        ],
        "Issuer": "https://cognito-idp.us-west-2.amazonaws.com/us-west-2_abc123"
    },
    "Name": "my-jwt-authorizer"
}
```

Per ulteriori informazioni, consulta Controllare l'accesso a HTTP APIs con gli autorizzatori JWT nella Amazon API Gateway Developer Guide.

- Per i dettagli sull'API, consulta AWS CLI Command UpdateAuthorizerReference.

`update-deployment`

Il seguente esempio di codice mostra come utilizzare`update-deployment`.

AWS CLI

Come modificare la descrizione di un'implementazione

L'esempio `update-deployment` seguente aggiorna la descrizione di un'implementazione.

```
aws apigatewayv2 update-deployment \
    --api-id a1b2c3d4 \
    --deployment-id abcdef \
    --description 'Manual deployment to fix integration test failures.'
```

Output:

```
{
    "AutoDeployed": false,
    "CreatedDate": "2020-02-05T16:21:48+00:00",
    "DeploymentId": "abcdef",
    "DeploymentStatus": "DEPLOYED",
    "Description": "Manual deployment to fix integration test failures."
}
```

Per ulteriori informazioni, consulta Sviluppo di API HTTP in Gateway API nella Guida per gli sviluppatori di Gateway Amazon API.

- Per i dettagli sull'API, consulta UpdateDeployment AWS CLICommand Reference.

`update-domain-name`

Il seguente esempio di codice mostra come utilizzare`update-domain-name`.

AWS CLI

Come aggiornare un nome di dominio personalizzato

L'esempio `update-domain-name` seguente specifica un nuovo certificato ACM per il nome di dominio personalizzato `api.example.com`.

```
aws apigatewayv2 update-domain-name \
    --domain-name api.example.com \
    --domain-name-configurations CertificateArn=arn:aws:acm:us-
west-2:123456789012:certificate/123456789012-1234-1234-1234-12345678
```

Output:

```
{
    "ApiMappingSelectionExpression": "$request.basepath",
    "DomainName": "regional.example.com",
    "DomainNameConfigurations": [
        {
            "ApiGatewayDomainName": "d-id.execute-api.us-west-2.amazonaws.com",
            "CertificateArn": "arn:aws:acm:us-
west-2:123456789012:certificate/123456789012-1234-1234-1234-12345678",
            "EndpointType": "REGIONAL",
            "HostedZoneId": "123456789111",
```

```
            "SecurityPolicy": "TLS_1_2",
            "DomainNameStatus": "AVAILABLE"
        }
    ]
}
```

Per ulteriori informazioni, consulta Configurazione di un nome di dominio personalizzato regionale in Gateway API nella Guida per gli sviluppatori di Gateway Amazon API.

- Per i dettagli sull'API, consulta UpdateDomainName AWS CLICommand Reference.

`update-integration`

Il seguente esempio di codice mostra come utilizzare`update-integration`.

AWS CLI

Come aggiornare un'integrazione Lambda

L'`update-integration`esempio seguente aggiorna un'integrazione AWS Lambda esistente per utilizzare la funzione Lambda specificata.

```
aws apigatewayv2 update-integration \
    --api-id a1b2c3d4 \
    --integration-id a1b2c3 \
    --integration-uri arn:aws:apigateway:us-west-2:lambda:path/2015-03-31/functions/
arn:aws:lambda:us-west-2:123456789012:function:my-new-function/invocations
```

Output:

```
{
    "ConnectionType": "INTERNET",
    "IntegrationId": "a1b2c3",
    "IntegrationMethod": "POST",
    "IntegrationType": "AWS_PROXY",
    "IntegrationUri": "arn:aws:apigateway:us-west-2:lambda:path/2015-03-31/
functions/arn:aws:lambda:us-west-2:123456789012:function:my-new-function/
invocations",
    "PayloadFormatVersion": "2.0",
    "TimeoutInMillis": 5000
}
```

Per ulteriori informazioni, consulta Configurazione delle integrazioni per HTTP APIs e Configurazione delle integrazioni WebSocket API nella Amazon API Gateway Developer Guide.

- Per i dettagli sull'API, consulta Command UpdateIntegrationReference AWS CLI .

`update-route`

Il seguente esempio di codice mostra come utilizzare`update-route`.

AWS CLI

Esempio 1: come aggiornare l'integrazione di una route

L'esempio `update-route` seguente aggiorna l'integrazione di una route specificata.

```
aws apigatewayv2 update-route \
    --api-id a1b2c3d4 \
    --route-id a1b2c3 \
    --target integrations/a1b2c6
```

Output:

```
{
    "ApiKeyRequired": false,
    "AuthorizationType": "NONE",
    "RouteId": "a1b2c3",
    "RouteKey": "ANY /pets",
    "Target": "integrations/a1b2c6"
}
```

Esempio 2: come aggiungere un sistema di autorizzazione a una route

L'esempio `update-route` seguente aggiorna la route specificata in modo da utilizzare un sistema di autorizzazione JWT.

```
aws apigatewayv2 update-route \
    --api-id a1b2c3d4  \
    --route-id a1b2c3  \
    --authorization-type JWT \
    --authorizer-id a1b2c5 \
    --authorization-scopes user.id user.email
```

Output:

```
{
    "ApiKeyRequired": false,
    "AuthorizationScopes": [
        "user.id",
        "user.email"
    ],
    "AuthorizationType": "JWT",
    "AuthorizerId": "a1b2c5",
    "OperationName": "GET HTTP",
    "RequestParameters": {},
    "RouteId": "a1b2c3",
    "RouteKey": "GET /pets",
    "Target": "integrations/a1b2c6"
}
```

Per ulteriori informazioni, consulta Controllare l'accesso a HTTP APIs con gli autorizzatori JWT nella Amazon API Gateway Developer Guide.

- Per i dettagli sull'API, consulta AWS CLI Command UpdateRouteReference.

`update-stage`

Il seguente esempio di codice mostra come utilizzare`update-stage`.

AWS CLI

Come configurare una limitazione (della larghezza di banda della rete) personalizzata

L'esempio `update-stage` seguente configura una limitazione (della larghezza di banda della rete) personalizzata per la fase e la route specificate di un'API.

```
aws apigatewayv2 update-stage \
    --api-id a1b2c3d4 \
    --stage-name dev \
    --route-settings '{"GET /pets":
{"ThrottlingBurstLimit":100,"ThrottlingRateLimit":2000}}'
```

Output:

```
{
```

```
    "CreatedDate": "2020-04-05T16:21:16+00:00",
    "DefaultRouteSettings": {
        "DetailedMetricsEnabled": false
    },
    "DeploymentId": "shktxb",
    "LastUpdatedDate": "2020-04-08T22:23:17+00:00",
    "RouteSettings": {
        "GET /pets": {
            "ThrottlingBurstLimit": 100,
            "ThrottlingRateLimit": 2000.0
        }
    },
    "StageName": "dev",
    "StageVariables": {},
    "Tags": {}
}
```

Per ulteriori informazioni, consulta Protezione delle API HTTP in Gateway API nella Guida per gli sviluppatori di Gateway Amazon API.

- Per i dettagli sull'API, consulta UpdateStage AWS CLICommand Reference.

`update-vpc-link`

Il seguente esempio di codice mostra come utilizzare`update-vpc-link`.

AWS CLI

Come aggiornare un link VPC

L'esempio `update-vpc-link` seguente aggiorna il nome di un collegamento VPC. Dopo aver creato un link VPC, non è possibile modificarne le sottoreti o i gruppi di sicurezza.

```
aws apigatewayv2 update-vpc-link \
    --vpc-link-id abcd123 \
    --name MyUpdatedVpcLink
```

Output:

```
{
    "CreatedDate": "2020-04-07T00:27:47Z",
    "Name": "MyUpdatedVpcLink",
```

```
    "SecurityGroupIds": [
        "sg1234",
        "sg5678"
    ],
    "SubnetIds": [
        "subnet-aaaa",
        "subnet-bbbb"
    ],
    "Tags": {},
    "VpcLinkId": "abcd123",
    "VpcLinkStatus": "AVAILABLE",
    "VpcLinkStatusMessage": "VPC link is ready to route traffic",
    "VpcLinkVersion": "V2"
}
```

Per ulteriori informazioni, consulta Working with VPC links for HTTP APIs nella Amazon API Gateway Developer Guide.

- Per i dettagli sull'API, consulta UpdateVpcLink AWS CLICommand Reference.

Esempi di API Gateway Management che utilizzano AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando l'API AWS Command Line Interface with API Gateway Management.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`delete-connection`

Il seguente esempio di codice mostra come utilizzare`delete-connection`.

AWS CLI

Per eliminare una WebSocket connessione

L'`delete-connection`esempio seguente disconnette un client dall' WebSocket API specificata.

```
aws apigatewaymanagementapi delete-connection \
    --connection-id L0SM9cOFvHcCIhw= \
    --endpoint-url https://aabbccddee.execute-api.us-west-2.amazonaws.com/prod
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Utilizzo di comandi @connections nel servizio di backend nella Guida per gli sviluppatori di Gateway Amazon API.

- Per i dettagli sull'API, vedere DeleteConnectionin AWS CLI Command Reference.

`get-connection`

Il seguente esempio di codice mostra come utilizzare`get-connection`.

AWS CLI

Per ottenere informazioni su una WebSocket connessione

L'`get-connection`esempio seguente descrive una connessione all' WebSocket API specificata.

```
aws apigatewaymanagementapi get-connection \
    --connection-id L0SM9cOFvHcCIhw= \
    --endpoint-url https://aabbccddee.execute-api.us-west-2.amazonaws.com/prod
```

Output:

```
{
    "ConnectedAt": "2020-04-30T20:10:33.236Z",
    "Identity": {
        "SourceIp": "192.0.2.1"
    },
    "LastActiveAt": "2020-04-30T20:10:42.997Z"
}
```

Per ulteriori informazioni, consulta Utilizzo di comandi @connections nel servizio di backend nella Guida per gli sviluppatori di Gateway Amazon API.

- Per i dettagli sull'API, vedere GetConnectionin AWS CLI Command Reference.

`post-to-connection`

Il seguente esempio di codice mostra come utilizzare`post-to-connection`.

AWS CLI

Per inviare dati a una WebSocket connessione

L'`post-to-connection`esempio seguente invia un messaggio a un client connesso all' WebSocket API specificata.

```
aws apigatewaymanagementapi post-to-connection \
    --connection-id L0SM9cOFvHcCIhw= \
    --data "Hello from API Gateway!" \
    --endpoint-url https://aabbccddee.execute-api.us-west-2.amazonaws.com/prod
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Utilizzo di comandi @connections nel servizio di backend nella Guida per gli sviluppatori di Gateway Amazon API.

- Per i dettagli sull'API, vedere PostToConnectionin AWS CLI Command Reference.

Esempi di App Mesh utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with App Mesh.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`create-mesh`

Il seguente esempio di codice mostra come utilizzare`create-mesh`.

AWS CLI

Esempio 1: come creare una nuova mesh di servizi

L'esempio `create-mesh` seguente crea una mesh di servizi.

```
aws appmesh create-mesh \
    --mesh-name app1
```

Output:

```
{
    "mesh":{
        "meshName":"app1",
        "metadata":{
            "arn":"arn:aws:appmesh:us-east-1:123456789012:mesh/app1",
            "createdAt":1563809909.282,
            "lastUpdatedAt":1563809909.282,
            "uid":"a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
            "version":1
        },
        "spec":{},
        "status":{
            "status":"ACTIVE"
        }
    }
}
```

Esempio 2: come creare una nuova mesh di servizi con più tag

L'esempio `create-mesh` seguente crea una mesh di servizi con più tag.

```
aws appmesh create-mesh \
    --mesh-name app2 \
    --tags key=key1,value=value1 key=key2,value=value2 key=key3,value=value3
```

Output:

```
{
    "mesh":{
        "meshName":"app2",
        "metadata":{
            "arn":"arn:aws:appmesh:us-east-1:123456789012:mesh/app2",
            "createdAt":1563822121.877,
            "lastUpdatedAt":1563822121.877,
            "uid":"a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
            "version":1
        },
        "spec":{},
        "status":{
            "status":"ACTIVE"
        }
    }
}
```

Per ulteriori informazioni, consulta Mesh dei servizi nella Guida per l'utente di AWS App Mesh.

- Per i dettagli sull'API, consulta CreateMesh AWS CLICommand Reference.

`create-route`

Il seguente esempio di codice mostra come utilizzare`create-route`.

AWS CLI

Come creare una nuova route gRPC

L'esempio `create-route` seguente utilizza un file di input JSON per creare una route gRPC. Il traffico GRPC con metadati che iniziano con 123 viene instradato a un nodo virtuale denominato serviceBgrpc. Se durante il tentativo di comunicare con la destinazione della route si verificano errori gRPC, HTTP o TCP specifici, il tentativo di instradamento viene ripetuto tre volte, con un ritardo di 15 secondi tra ogni tentativo.

```
aws appmesh create-route \
    --cli-input-json file://create-route-grpc.json
```

Contenuto di `create-route-grpc.json`:

```
{
    "meshName" : "apps",
    "routeName" : "grpcRoute",
    "spec" : {
       "grpcRoute" : {
          "action" : {
             "weightedTargets" : [
                {
                   "virtualNode" : "serviceBgrpc",
                   "weight" : 100
                }
             ]
          },
          "match" : {
             "metadata" : [
                {
                   "invert" : false,
                   "match" : {
                      "prefix" : "123"
                   },
                   "name" : "myMetadata"
                }
             ],
             "methodName" : "GetColor",
             "serviceName" : "com.amazonaws.services.ColorService"
          },
          "retryPolicy" : {
             "grpcRetryEvents" : [ "deadline-exceeded" ],
             "httpRetryEvents" : [ "server-error", "gateway-error" ],
             "maxRetries" : 3,
             "perRetryTimeout" : {
                "unit" : "s",
                "value" : 15
             },
             "tcpRetryEvents" : [ "connection-error" ]
          }
       },
       "priority" : 100
    },
    "virtualRouterName" : "serviceBgrpc"
}
```

Output:

```
{
    "route": {
        "meshName": "apps",
        "metadata": {
            "arn": "arn:aws:appmesh:us-west-2:123456789012:mesh/apps/virtualRouter/
serviceBgrpc/route/grpcRoute",
            "createdAt": 1572010806.008,
            "lastUpdatedAt": 1572010806.008,
            "uid": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
            "version": 1
        },
        "routeName": "grpcRoute",
        "spec": {
            "grpcRoute": {
                "action": {
                    "weightedTargets": [
                        {
                            "virtualNode": "serviceBgrpc",
                            "weight": 100
                        }
                    ]
                },
                "match": {
                    "metadata": [
                        {
                            "invert": false,
                            "match": {
                                "prefix": "123"
                            },
                            "name": "mymetadata"
                        }
                    ],
                    "methodName": "GetColor",
                    "serviceName": "com.amazonaws.services.ColorService"
                },
                "retryPolicy": {
                    "grpcRetryEvents": [
                        "deadline-exceeded"
                    ],
                    "httpRetryEvents": [
                        "server-error",
                        "gateway-error"
                    ],
```

```
                    "maxRetries": 3,
                    "perRetryTimeout": {
                        "unit": "s",
                        "value": 15
                    },
                    "tcpRetryEvents": [
                        "connection-error"
                    ]
                }
            },
            "priority": 100
        },
        "status": {
            "status": "ACTIVE"
        },
        "virtualRouterName": "serviceBgrpc"
    }
}
```

Come creare una nuova route HTTP o HTTP/2

L'esempio `create-route` seguente utilizza un file di input JSON per creare una route HTTP/2. Per creare una route HTTP, sostituisci http2Route con httpRoute in base alle specifiche. Tutto il traffico HTTP/2 indirizzato a qualsiasi prefisso URL con un valore di intestazione che inizia con 123 viene instradato verso un nodo virtuale denominato serviceBhttp2. Se durante il tentativo di comunicare con la destinazione della route si verificano errori HTTP o TCP specifici, il tentativo di instradamento viene ripetuto tre volte, con un ritardo di 15 secondi tra ogni tentativo.

```
aws appmesh create-route \
    --cli-input-json file://create-route-http2.json
```

Contenuto di `create-route-http2.json`:

```
{
    "meshName": "apps",
    "routeName": "http2Route",
    "spec": {
        "http2Route": {
            "action": {
                "weightedTargets": [
                    {
                        "virtualNode": "serviceBhttp2",
```

```
                        "weight": 100
                    }
                ]
            },
            "match": {
                "headers": [
                    {
                        "invert": false,
                        "match": {
                            "prefix": "123"
                        },
                        "name": "clientRequestId"
                    }
                ],
                "method": "POST",
                "prefix": "/",
                "scheme": "http"
            },
            "retryPolicy": {
                "httpRetryEvents": [
                    "server-error",
                    "gateway-error"
                ],
                "maxRetries": 3,
                "perRetryTimeout": {
                    "unit": "s",
                    "value": 15
                },
                "tcpRetryEvents": [
                    "connection-error"
                ]
            }
        },
        "priority": 200
    },
    "virtualRouterName": "serviceBhttp2"
}
```

Output:

```
{
    "route": {
        "meshName": "apps",
```

```
        "metadata": {
            "arn": "arn:aws:appmesh:us-west-2:123456789012:mesh/apps/virtualRouter/
serviceBhttp2/route/http2Route",
            "createdAt": 1572011008.352,
            "lastUpdatedAt": 1572011008.352,
            "uid": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
            "version": 1
        },
        "routeName": "http2Route",
        "spec": {
            "http2Route": {
                "action": {
                    "weightedTargets": [
                        {
                            "virtualNode": "serviceBhttp2",
                            "weight": 100
                        }
                    ]
                },
                "match": {
                    "headers": [
                        {
                            "invert": false,
                            "match": {
                                "prefix": "123"
                            },
                            "name": "clientRequestId"
                        }
                    ],
                    "method": "POST",
                    "prefix": "/",
                    "scheme": "http"
                },
                "retryPolicy": {
                    "httpRetryEvents": [
                        "server-error",
                        "gateway-error"
                    ],
                    "maxRetries": 3,
                    "perRetryTimeout": {
                        "unit": "s",
                        "value": 15
                    },
                    "tcpRetryEvents": [
```

```
                        "connection-error"
                    ]
                }
            },
            "priority": 200
        },
        "status": {
            "status": "ACTIVE"
        },
        "virtualRouterName": "serviceBhttp2"
    }
}
```

Come creare una nuova route TCP

L'esempio `create-route` seguente utilizza un file di input JSON per creare una route TCP. Il 75% del traffico viene instradato verso un nodo virtuale denominato serviceBtcp e il 25% verso un nodo virtuale denominato serviceBv2tcp. Specificare ponderazioni diverse per obiettivi differenti è un modo efficace per implementare una nuova versione di un'applicazione. È possibile modificare i pesi in modo che alla fine il 100% di tutto il traffico venga instradato verso una destinazione che dispone della nuova versione di un'applicazione.

```
aws appmesh create-route \
    --cli-input-json file://create-route-tcp.json
```

Contenuto di create-route-tcp .json:

```
{
    "meshName": "apps",
    "routeName": "tcpRoute",
    "spec": {
        "priority": 300,
        "tcpRoute": {
            "action": {
                "weightedTargets": [
                    {
                        "virtualNode": "serviceBtcp",
                        "weight": 75
                    },
                    {
                        "virtualNode": "serviceBv2tcp",
                        "weight": 25
```

```
                    }
                ]
            }
        }
    },
    "virtualRouterName": "serviceBtcp"
}
```

Output:

```
{
    "route": {
        "meshName": "apps",
        "metadata": {
            "arn": "arn:aws:appmesh:us-west-2:123456789012:mesh/apps/virtualRouter/
serviceBtcp/route/tcpRoute",
            "createdAt": 1572011436.26,
            "lastUpdatedAt": 1572011436.26,
            "uid": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
            "version": 1
        },
        "routeName": "tcpRoute",
        "spec": {
            "priority": 300,
            "tcpRoute": {
                "action": {
                    "weightedTargets": [
                        {
                            "virtualNode": "serviceBtcp",
                            "weight": 75
                        },
                        {
                            "virtualNode": "serviceBv2tcp",
                            "weight": 25
                        }
                    ]
                }
            }
        },
        "status": {
            "status": "ACTIVE"
        },
        "virtualRouterName": "serviceBtcp"
```

```
    }
}
```

Per ulteriori informazioni, consulta Route nella Guida per l'utente di AWS App Mesh.

- Per i dettagli sull'API, consulta AWS CLI Command CreateRouteReference.

`create-virtual-gateway`

Il seguente esempio di codice mostra come utilizzare`create-virtual-gateway`.

AWS CLI

Come creare un nuovo gateway virtuale

L'esempio `create-virtual-gateway` seguente utilizza un file di input JSON per creare un gateway virtuale con un listener per HTTP sulla porta 9080.

```
aws appmesh create-virtual-gateway \
    --mesh-name meshName \
    --virtual-gateway-name virtualGatewayName \
    --cli-input-json file://create-virtual-gateway.json
```

Contenuto di `create-virtual-gateway.json`:

```
{
    "spec": {
      "listeners": [
        {
          "portMapping": {
            "port": 9080,
            "protocol": "http"
          }
        }
      ]
    }
}
```

Output:

```
{
```

```
    "virtualGateway": {
        "meshName": "meshName",
        "metadata": {
            "arn": "arn:aws:appmesh:us-west-2:123456789012:mesh/meshName/
virtualGateway/virtualGatewayName",
            "createdAt": "2022-04-06T10:42:42.015000-05:00",
            "lastUpdatedAt": "2022-04-06T10:42:42.015000-05:00",
            "meshOwner": "123456789012",
            "resourceOwner": "123456789012",
            "uid": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
            "version": 1
        },
        "spec": {
            "listeners": [
                {
                    "portMapping": {
                        "port": 9080,
                        "protocol": "http"
                    }
                }
            ]
        },
        "status": {
            "status": "ACTIVE"
        },
        "virtualGatewayName": "virtualGatewayName"
    }
}
```

Per ulteriori informazioni, consulta Gateway virtuali nella Guida per l'utente di AWS App Mesh.

- Per i dettagli sull'API, consulta CreateVirtualGateway AWS CLICommand Reference.

`create-virtual-node`

Il seguente esempio di codice mostra come utilizzare`create-virtual-node`.

AWS CLI

Esempio 1: come creare un nuovo nodo virtuale che utilizza DNS per il rilevamento

L'esempio `create-virtual-node` seguente utilizza un file di input JSON per creare un nodo virtuale che utilizza DNS per il rilevamento servizi.

```
aws appmesh create-virtual-node \
    --cli-input-json file://create-virtual-node-dns.json
```

Contenuto di `create-virtual-node-dns.json`:

```
{
    "meshName": "app1",
    "spec": {
        "listeners": [
            {
                "portMapping": {
                    "port": 80,
                    "protocol": "http"
                }
            }
        ],
        "serviceDiscovery": {
            "dns": {
                "hostname": "serviceBv1.svc.cluster.local"
            }
        }
    },
    "virtualNodeName": "vnServiceBv1"
}
```

Output:

```
{
    "virtualNode": {
        "meshName": "app1",
        "metadata": {
            "arn": "arn:aws:appmesh:us-east-1:123456789012:mesh/app1/virtualNode/
vnServiceBv1",
            "createdAt": 1563810019.874,
            "lastUpdatedAt": 1563810019.874,
            "uid": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
            "version": 1
        },
        "spec": {
            "listeners": [
                {
                    "portMapping": {
```

```
                        "port": 80,
                        "protocol": "http"
                    }
                }
            ],
            "serviceDiscovery": {
                "dns": {
                    "hostname": "serviceBv1.svc.cluster.local"
                }
            }
        },
        "status": {
            "status": "ACTIVE"
        },
        "virtualNodeName": "vnServiceBv1"
    }
}
```

Esempio 2: creare un nuovo nodo virtuale che utilizza AWS Cloud Map per il rilevamento

L'`create-virtual-node`esempio seguente utilizza un file di input JSON per creare un nodo virtuale che utilizza AWS Cloud Map per il rilevamento dei servizi.

```
aws appmesh create-virtual-node \
    --cli-input-json file://create-virtual-node-cloud-map.json
```

Contenuto di `create-virtual-node-cloud-map.json`:

```
{
    "meshName": "app1",
    "spec": {
        "backends": [
            {
                "virtualService": {
                    "virtualServiceName": "serviceA.svc.cluster.local"
                }
            }
        ],
        "listeners": [
            {
                "portMapping": {
                    "port": 80,
                    "protocol": "http"
```

```
                }
            }
        ],
        "serviceDiscovery": {
            "awsCloudMap": {
                "attributes": [
                    {
                        "key": "Environment",
                        "value": "Testing"
                    }
                ],
                "namespaceName": "namespace1",
                "serviceName": "serviceA"
            }
        }
    },
    "virtualNodeName": "vnServiceA"
}
```

Output:

```
{
    "virtualNode": {
        "meshName": "app1",
        "metadata": {
            "arn": "arn:aws:appmesh:us-east-1:123456789012:mesh/app1/virtualNode/
vnServiceA",
            "createdAt": 1563810859.465,
            "lastUpdatedAt": 1563810859.465,
            "uid": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
            "version": 1
        },
        "spec": {
            "backends": [
                {
                    "virtualService": {
                        "virtualServiceName": "serviceA.svc.cluster.local"
                    }
                }
            ],
            "listeners": [
                {
                    "portMapping": {
```

```
                        "port": 80,
                        "protocol": "http"
                    }
                }
            ],
            "serviceDiscovery": {
                "awsCloudMap": {
                    "attributes": [
                        {
                            "key": "Environment",
                            "value": "Testing"
                        }
                    ],
                    "namespaceName": "namespace1",
                    "serviceName": "serviceA"
                }
            }
        },
        "status": {
            "status": "ACTIVE"
        },
        "virtualNodeName": "vnServiceA"
    }
}
```

Per ulteriori informazioni, consulta Nodi virtuali nella Guida per l'utente di AWS App Mesh.

- Per i dettagli sull'API, consulta CreateVirtualNode AWS CLICommand Reference.

`create-virtual-router`

Il seguente esempio di codice mostra come utilizzare`create-virtual-router`.

AWS CLI

Come creare un nuovo router virtuale

L'esempio `create-virtual-router` seguente utilizza un file di input JSON per creare un router virtuale con un listener per HTTP sulla porta 80.

```
aws appmesh create-virtual-router \
    --cli-input-json file://create-virtual-router.json
```

Contenuto di `create-virtual-router.json`:

```
{
    "meshName": "app1",
    "spec": {
        "listeners": [
            {
                "portMapping": {
                    "port": 80,
                    "protocol": "http"
                }
            }
        ]
    },
    "virtualRouterName": "vrServiceB"
}
```

Output:

```
{
    "virtualRouter": {
        "meshName": "app1",
        "metadata": {
            "arn": "arn:aws:appmesh:us-east-1:123456789012:mesh/app1/virtualRouter/
vrServiceB",
            "createdAt": 1563810546.59,
            "lastUpdatedAt": 1563810546.59,
            "uid": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
            "version": 1
        },
        "spec": {
            "listeners": [
                {
                    "portMapping": {
                        "port": 80,
                        "protocol": "http"
                    }
                }
            ]
        },
        "status": {
            "status": "ACTIVE"
        },
```

```
        "virtualRouterName": "vrServiceB"
    }
}
```

Per ulteriori informazioni, consulta Router virtuali nella Guida per l'utente di AWS App Mesh.

- Per i dettagli sull'API, consulta CreateVirtualRouter AWS CLICommand Reference.

`create-virtual-service`

Il seguente esempio di codice mostra come utilizzare`create-virtual-service`.

AWS CLI

Esempio 1: come creare un nuovo servizio virtuale con un provider di nodi virtuali

L'esempio `create-virtual-service` seguente utilizza un file di input JSON per creare un servizio virtuale con un provider di nodi virtuali.

```
aws appmesh create-virtual-service \
    --cli-input-json file://create-virtual-service-virtual-node.json
```

Contenuto di `create-virtual-service-virtual-node.json`:

```
{
    "meshName": "app1",
    "spec": {
        "provider": {
            "virtualNode": {
                "virtualNodeName": "vnServiceA"
            }
        }
    },
    "virtualServiceName": "serviceA.svc.cluster.local"
}
```

Output:

```
{
    "virtualService": {
        "meshName": "app1",
```

```
        "metadata": {
            "arn": "arn:aws:appmesh:us-east-1:123456789012:mesh/app1/virtualService/
serviceA.svc.cluster.local",
            "createdAt": 1563810859.474,
            "lastUpdatedAt": 1563810967.179,
            "uid": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
            "version": 2
        },
        "spec": {
            "provider": {
                "virtualNode": {
                    "virtualNodeName": "vnServiceA"
                }
            }
        },
        "status": {
            "status": "ACTIVE"
        },
        "virtualServiceName": "serviceA.svc.cluster.local"
    }
}
```

Per ulteriori informazioni, consulta Nodo virtuale nella Guida per l'utente di AWS App Mesh.

Esempio 2: come creare un nuovo servizio virtuale con un provider di router virtuali

L'esempio `create-virtual-service` seguente utilizza un file di input JSON per creare un servizio virtuale con un provider di router virtuali.

```
aws appmesh create-virtual-service \
    --cli-input-json file://create-virtual-service-virtual-router.json
```

Contenuto di `create-virtual-service-virtual-router.json`:

```
{
    "meshName": "app1",
    "spec": {
        "provider": {
            "virtualRouter": {
                "virtualRouterName": "vrServiceB"
            }
        }
    },
```

```
    "virtualServiceName": "serviceB.svc.cluster.local"
}
```

Output:

```
{
    "virtualService": {
        "meshName": "app1",
        "metadata": {
            "arn": "arn:aws:appmesh:us-east-1:123456789012:mesh/app1/virtualService/
serviceB.svc.cluster.local",
            "createdAt": 1563908363.999,
            "lastUpdatedAt": 1563908363.999,
            "uid": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
            "version": 1
        },
        "spec": {
            "provider": {
                "virtualRouter": {
                    "virtualRouterName": "vrServiceB"
                }
            }
        },
        "status": {
            "status": "ACTIVE"
        },
        "virtualServiceName": "serviceB.svc.cluster.local"
    }
}
```

Per ulteriori informazioni, consulta Virtual Services< https://docs.aws.amazon.com/app-mesh/latest/userguide/virtual _services.html> nella App Mesh User Guide AWS

- Per i dettagli sull'API, consulta Command Reference. CreateVirtualServiceAWS CLI

`delete-mesh`

Il seguente esempio di codice mostra come utilizzare`delete-mesh`.

AWS CLI

Come eliminare una mesh di servizi

L'esempio `delete-mesh` seguente elimina la mesh di servizi specificata.

```
aws appmesh delete-mesh \
    --mesh-name app1
```

Output:

```
{
    "mesh": {
        "meshName": "app1",
        "metadata": {
            "arn": "arn:aws:appmesh:us-east-1:123456789012:mesh/app1",
            "createdAt": 1563809909.282,
            "lastUpdatedAt": 1563824981.248,
            "uid": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
            "version": 2
        },
        "spec": {
            "egressFilter": {
                "type": "ALLOW_ALL"
            }
        },
        "status": {
            "status": "DELETED"
        }
    }
}
```

Per ulteriori informazioni, consulta Mesh dei servizi nella Guida per l'utente di AWS App Mesh.

- Per i dettagli sull'API, consulta DeleteMesh AWS CLICommand Reference.

`delete-route`

Il seguente esempio di codice mostra come utilizzare`delete-route`.

AWS CLI

Come eliminare una route

L'esempio `delete-route` seguente elimina la route specificata.

```
aws appmesh delete-route \
```

```
    --mesh-name app1 \
    --virtual-router-name vrServiceB \
    --route-name toVnServiceB-weighted
```

Output:

```
{
    "route": {
        "meshName": "app1",
        "metadata": {
            "arn": "arn:aws:appmesh:us-east-1:123456789012:mesh/app1/virtualRouter/
vrServiceB/route/toVnServiceB-weighted",
            "createdAt": 1563811384.015,
            "lastUpdatedAt": 1563823915.936,
            "uid": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
            "version": 3
        },
        "routeName": "toVnServiceB-weighted",
        "spec": {
            "httpRoute": {
                "action": {
                    "weightedTargets": [
                        {
                            "virtualNode": "vnServiceBv1",
                            "weight": 80
                        },
                        {
                            "virtualNode": "vnServiceBv2",
                            "weight": 20
                        }
                    ]
                },
                "match": {
                    "prefix": "/"
                }
            }
        },
        "status": {
            "status": "DELETED"
        },
        "virtualRouterName": "vrServiceB"
    }
}
```

Per ulteriori informazioni, consulta Route nella Guida per l'utente di AWS App Mesh.

- Per i dettagli sull'API, consulta DeleteRoute AWS CLICommand Reference.

`delete-virtual-node`

Il seguente esempio di codice mostra come utilizzare`delete-virtual-node`.

AWS CLI

Come eliminare un nodo virtuale

L'esempio `delete-virtual-node` seguente elimina il nodo virtuale specificato.

```
aws appmesh delete-virtual-node \
    --mesh-name app1 \
    --virtual-node-name vnServiceBv2
```

Output:

```
{
    "virtualNode": {
        "meshName": "app1",
        "metadata": {
            "arn": "arn:aws:appmesh:us-east-1:123456789012:mesh/app1/virtualNode/
vnServiceBv2",
            "createdAt": 1563810117.297,
            "lastUpdatedAt": 1563824700.678,
            "uid": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
            "version": 2
        },
        "spec": {
            "backends": [],
            "listeners": [
                {
                    "portMapping": {
                        "port": 80,
                        "protocol": "http"
                    }
                }
            ],
            "serviceDiscovery": {
```

```
                "dns": {
                    "hostname": "serviceBv2.svc.cluster.local"
                }
            }
        },
        "status": {
            "status": "DELETED"
        },
        "virtualNodeName": "vnServiceBv2"
    }
}
```

Per ulteriori informazioni, consulta Nodi virtuali nella Guida per l'utente di AWS App Mesh.

- Per i dettagli sull'API, consulta DeleteVirtualNode AWS CLICommand Reference.

`delete-virtual-router`

Il seguente esempio di codice mostra come utilizzare`delete-virtual-router`.

AWS CLI

Come eliminare un router virtuale

L'esempio `delete-virtual-router` seguente elimina il router virtuale specificato.

```
aws appmesh delete-virtual-router \
    --mesh-name app1 \
    --virtual-router-name vrServiceB
```

Output:

```
{
    "virtualRouter": {
        "meshName": "app1",
        "metadata": {
            "arn": "arn:aws:appmesh:us-east-1:123456789012:mesh/app1/virtualRouter/
vrServiceB",
            "createdAt": 1563810546.59,
            "lastUpdatedAt": 1563824253.467,
            "uid": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
            "version": 3
```

```
        },
        "spec": {
            "listeners": [
                {
                    "portMapping": {
                        "port": 80,
                        "protocol": "http"
                    }
                }
            ]
        },
        "status": {
            "status": "DELETED"
        },
        "virtualRouterName": "vrServiceB"
    }
}
```

Per ulteriori informazioni, consulta Router virtuali nella Guida per l'utente di AWS App Mesh.

- Per i dettagli sull'API, consulta DeleteVirtualRouter AWS CLICommand Reference.

`delete-virtual-service`

Il seguente esempio di codice mostra come utilizzare`delete-virtual-service`.

AWS CLI

Come eliminare un servizio virtuale

L'esempio `delete-virtual-service` seguente elimina il servizio virtuale specificato.

```
aws appmesh delete-virtual-service \
    --mesh-name app1 \
    --virtual-service-name serviceB.svc.cluster.local
```

Output:

```
{
    "virtualService": {
        "meshName": "app1",
        "metadata": {
```

```
            "arn": "arn:aws:appmesh:us-east-1:123456789012:mesh/app1/virtualService/
serviceB.svc.cluster.local",
            "createdAt": 1563908363.999,
            "lastUpdatedAt": 1563913940.866,
            "uid": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
            "version": 3
        },
        "spec": {},
        "status": {
            "status": "DELETED"
        },
        "virtualServiceName": "serviceB.svc.cluster.local"
    }
}
```

Per ulteriori informazioni, consulta Servizio virtuale nella Guida per l'utente di AWS App Mesh.

- Per i dettagli sull'API, consulta DeleteVirtualService AWS CLICommand Reference.

`describe-mesh`

Il seguente esempio di codice mostra come utilizzare`describe-mesh`.

AWS CLI

Come descrivere una mesh di servizi

L'esempio `describe-mesh` seguente restituisce i dettagli della mesh di servizi specificata.

```
aws appmesh describe-mesh \
    --mesh-name app1
```

Output:

```
{
    "mesh": {
        "meshName": "app1",
        "metadata": {
            "arn": "arn:aws:appmesh:us-east-1:123456789012:mesh/app1",
            "createdAt": 1563809909.282,
            "lastUpdatedAt": 1563809909.282,
            "uid": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
```

```
            "version": 1
        },
        "spec": {},
        "status": {
            "status": "ACTIVE"
        }
    }
}
```

Per ulteriori informazioni, consulta Mesh dei servizi nella Guida per l'utente di AWS App Mesh.

- Per i dettagli sull'API, consulta DescribeMesh AWS CLICommand Reference.

`describe-route`

Il seguente esempio di codice mostra come utilizzare`describe-route`.

AWS CLI

Come descrivere una route

L'esempio `describe-route` seguente restituisce i dettagli della route specificata.

```
aws appmesh describe-route \
    --mesh-name app1 \
    --virtual-router-name vrServiceB \
    --route-name toVnServiceB-weighted
```

Output:

```
{
    "route": {
        "meshName": "app1",
        "metadata": {
            "arn": "arn:aws:appmesh:us-east-1:123456789012:mesh/app1/virtualRouter/
vrServiceB/route/toVnServiceB-weighted",
            "createdAt": 1563811384.015,
            "lastUpdatedAt": 1563811384.015,
            "uid": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
            "version": 1
        },
        "routeName": "toVnServiceB-weighted",
```

```
        "spec": {
            "httpRoute": {
                "action": {
                    "weightedTargets": [
                        {
                            "virtualNode": "vnServiceBv1",
                            "weight": 90
                        },
                        {
                            "virtualNode": "vnServiceBv2",
                            "weight": 10
                        }
                    ]
                },
                "match": {
                    "prefix": "/"
                }
            }
        },
        "status": {
            "status": "ACTIVE"
        },
        "virtualRouterName": "vrServiceB"
    }
}
```

Per ulteriori informazioni, consulta Route nella Guida per l'utente di AWS App Mesh.

- Per i dettagli sull'API, consulta DescribeRoute AWS CLICommand Reference.

`describe-virtual-node`

Il seguente esempio di codice mostra come utilizzare`describe-virtual-node`.

AWS CLI

Come descrivere un nodo virtuale

L'esempio `describe-virtual-node` seguente restituisce i dettagli del nodo virtuale specificato.

```
aws appmesh describe-virtual-node \
    --mesh-name app1 \
    --virtual-node-name vnServiceBv1
```

Output:

```
{
    "virtualNode": {
        "meshName": "app1",
        "metadata": {
            "arn": "arn:aws:appmesh:us-east-1:123456789012:mesh/app1/virtualNode/
vnServiceBv1",
            "createdAt": 1563810019.874,
            "lastUpdatedAt": 1563810019.874,
            "uid": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
            "version": 1
        },
        "spec": {
            "backends": [],
            "listeners": [
                {
                    "portMapping": {
                        "port": 80,
                        "protocol": "http"
                    }
                }
            ],
            "serviceDiscovery": {
                "dns": {
                    "hostname": "serviceBv1.svc.cluster.local"
                }
            }
        },
        "status": {
            "status": "ACTIVE"
        },
        "virtualNodeName": "vnServiceBv1"
    }
}
```

Per ulteriori informazioni, consulta Nodi virtuali nella Guida per l'utente di AWS App Mesh.

- Per i dettagli sull'API, consulta DescribeVirtualNode AWS CLICommand Reference.

`describe-virtual-router`

Il seguente esempio di codice mostra come utilizzare`describe-virtual-router`.

AWS CLI

Come descrivere un router virtuale

L'esempio `describe-virtual-router` seguente restituisce i dettagli del router virtuale specificato.

```
aws appmesh describe-virtual-router \
    --mesh-name app1 \
    --virtual-router-name vrServiceB
```

Output:

```
{
    "virtualRouter": {
        "meshName": "app1",
        "metadata": {
            "arn": "arn:aws:appmesh:us-east-1:123456789012:mesh/app1/virtualRouter/
vrServiceB",
            "createdAt": 1563810546.59,
            "lastUpdatedAt": 1563810546.59,
            "uid": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
            "version": 1
        },
        "spec": {
            "listeners": [
                {
                    "portMapping": {
                        "port": 80,
                        "protocol": "http"
                    }
                }
            ]
        },
        "status": {
            "status": "ACTIVE"
        },
        "virtualRouterName": "vrServiceB"
    }
}
```

Per ulteriori informazioni, consulta Router virtuali nella Guida per l'utente di AWS App Mesh.

- Per i dettagli sull'API, consulta DescribeVirtualRouter AWS CLICommand Reference.

`describe-virtual-service`

Il seguente esempio di codice mostra come utilizzare`describe-virtual-service`.

AWS CLI

Come descrivere un servizio virtuale

L'esempio `describe-virtual-service` seguente restituisce i dettagli del servizio virtuale specificato.

```
aws appmesh describe-virtual-service \
    --mesh-name app1 \
    --virtual-service-name serviceB.svc.cluster.local
```

Output:

```
{
    "virtualService": {
        "meshName": "app1",
        "metadata": {
            "arn": "arn:aws:appmesh:us-east-1:123456789012:mesh/app1/virtualService/
serviceB.svc.cluster.local",
            "createdAt": 1563908363.999,
            "lastUpdatedAt": 1563908363.999,
            "uid": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
            "version": 1
        },
        "spec": {
            "provider": {
                "virtualRouter": {
                    "virtualRouterName": "vrServiceB"
                }
            }
        },
        "status": {
            "status": "ACTIVE"
        },
        "virtualServiceName": "serviceB.svc.cluster.local"
    }
```

```
}
```

Per ulteriori informazioni, consulta Servizio virtuale nella Guida per l'utente di AWS App Mesh.

- Per i dettagli sull'API, consulta DescribeVirtualService AWS CLICommand Reference.

`list-meshes`

Il seguente esempio di codice mostra come utilizzare`list-meshes`.

AWS CLI

Come elencare le mesh di servizi

L'`list-meshes`esempio seguente elenca tutte le mesh di servizio nella AWS regione corrente.

```
aws appmesh list-meshes
```

Output:

```
{
    "meshes": [
        {
            "arn": "arn:aws:appmesh:us-east-1:123456789012:mesh/app1",
            "meshName": "app1"
        }
    ]
}
```

Per ulteriori informazioni, consulta Mesh dei servizi nella Guida per l'utente di AWS App Mesh.

- Per i dettagli sull'API, vedere ListMeshesin AWS CLI Command Reference.

`list-routes`

Il seguente esempio di codice mostra come utilizzare`list-routes`.

AWS CLI

Come elencare le route

L'esempio `list-routes` seguente elenca tutte le route del router virtuale specificato.

```
aws appmesh list-routes \
    --mesh-name app1 \
    --virtual-router-name vrServiceB
```

Output:

```
{
    "routes": [
        {
            "arn": "arn:aws:appmesh:us-east-1:123456789012:mesh/app1/virtualRouter/
vrServiceB/route/toVnServiceB",
            "meshName": "app1",
            "routeName": "toVnServiceB-weighted",
            "virtualRouterName": "vrServiceB"
        }
    ]
}
```

Per ulteriori informazioni, consulta Route nella Guida per l'utente di AWS App Mesh.

- Per i dettagli sull'API, consulta ListRoutes AWS CLICommand Reference.

`list-tags-for-resource`

Il seguente esempio di codice mostra come utilizzare`list-tags-for-resource`.

AWS CLI

Come elencare i tag per una risorsa

L'esempio `list-tags-for-resource` seguente elenca tutti i tag assegnati alla risorsa specificata.

```
aws appmesh list-tags-for-resource \
    --resource-arn arn:aws:appmesh:us-east-1:123456789012:mesh/app1
```

Output:

```
{
    "tags": [
```

```
        {
            "key": "key1",
            "value": "value1"
        },
        {
            "key": "key2",
            "value": "value2"
        },
        {
            "key": "key3",
            "value": "value3"
        }
    ]
}
```

- Per i dettagli sull'API, consulta ListTagsForResource AWS CLICommand Reference.

`list-virtual-nodes`

Il seguente esempio di codice mostra come utilizzare`list-virtual-nodes`.

AWS CLI

Come elencare i nodi virtuali

L'esempio `list-virtual-nodes` seguente elenca tutti i nodi virtuali della mesh di servizi specificata.

```
aws appmesh list-virtual-nodes \
    --mesh-name app1
```

Output:

```
{
    "virtualNodes": [
        {
            "arn": "arn:aws:appmesh:us-east-1:123456789012:mesh/app1/virtualNode/
vnServiceBv1",
            "meshName": "app1",
            "virtualNodeName": "vnServiceBv1"
        },
```

```
        {
            "arn": "arn:aws:appmesh:us-east-1:123456789012:mesh/app1/virtualNode/
vnServiceBv2",
            "meshName": "app1",
            "virtualNodeName": "vnServiceBv2"
        }
    ]
}
```

Per ulteriori informazioni, consulta Nodi virtuali nella Guida per l'utente di AWS App Mesh.

- Per i dettagli sull'API, consulta ListVirtualNodes AWS CLICommand Reference.

`list-virtual-routers`

Il seguente esempio di codice mostra come utilizzare`list-virtual-routers`.

AWS CLI

Come elencare i router virtuali

L'esempio `list-virtual-routers` seguente elenca tutti i router virtuali nella mesh di servizi specificata.

```
aws appmesh list-virtual-routers \
    --mesh-name app1
```

Output:

```
{
    "virtualRouters": [
        {
            "arn": "arn:aws:appmesh:us-east-1:123456789012:mesh/app1/virtualRouter/
vrServiceB",
            "meshName": "app1",
            "virtualRouterName": "vrServiceB"
        }
    ]
}
```

Per ulteriori informazioni, consulta Router virtuali nella Guida per l'utente di AWS App Mesh.

- Per i dettagli sull'API, consulta ListVirtualRouters AWS CLICommand Reference.

`list-virtual-services`

Il seguente esempio di codice mostra come utilizzare`list-virtual-services`.

AWS CLI

Come elencare i servizi virtuali

L'esempio `list-virtual-services` seguente elenca tutti i servizi virtuali nella mesh di servizi specificata.

```
aws appmesh list-virtual-services \
    --mesh-name app1
```

Output:

```
{
    "virtualServices": [
        {
            "arn": "arn:aws:appmesh:us-east-1:123456789012:mesh/app1/virtualService/
serviceA.svc.cluster.local",
            "meshName": "app1",
            "virtualServiceName": "serviceA.svc.cluster.local"
        },
        {
            "arn": "arn:aws:appmesh:us-east-1:123456789012:mesh/app1/virtualService/
serviceB.svc.cluster.local",
            "meshName": "app1",
            "virtualServiceName": "serviceB.svc.cluster.local"
        }
    ]
}
```

Per ulteriori informazioni, consulta Servizio virtuale nella Guida per l'utente di AWS App Mesh.

- Per i dettagli sull'API, consulta ListVirtualServices AWS CLICommand Reference.

`tag-resource`

Il seguente esempio di codice mostra come utilizzare`tag-resource`.

AWS CLI

Come taggare a una risorsa

L'esempio `tag-resource` seguente aggiunge il tag `key1` con il valore `value1` alla risorsa specificata.

```
aws appmesh tag-resource \
    --resource-arn arn:aws:appmesh:us-east-1:123456789012:mesh/app1 \
    --tags key=key1,value=value1
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta TagResource AWS CLICommand Reference.

`untag-resource`

Il seguente esempio di codice mostra come utilizzare`untag-resource`.

AWS CLI

Come rimuovere un tag da una risorsa

L'esempio `untag-resource` seguente rimuove un tag con la chiave `key1` dalla risorsa specificata.

```
aws appmesh untag-resource \
    --resource-arn arn:aws:appmesh:us-east-1:123456789012:mesh/app1 \
    --tag-keys key1
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta UntagResource AWS CLICommand Reference.

`update-mesh`

Il seguente esempio di codice mostra come utilizzare`update-mesh`.

AWS CLI

Come aggiornare una mesh di servizi

L'esempio `update-mesh` seguente utilizza un file di input JSON per aggiornare una mesh di servizi per consentire l'inoltro inalterato di tutto il traffico esterno in uscita attraverso il proxy Envoy.

```
aws appmesh update-mesh \
    --cli-input-json file://update-mesh.json
```

Contenuto di `update-mesh.json`:

```
{
    "meshName": "app1",
    "spec": {
        "egressFilter": {
            "type": "ALLOW_ALL"
        }
    }
}
```

Output:

```
{
    "mesh": {
        "meshName": "app1",
        "metadata": {
            "arn": "arn:aws:appmesh:us-east-1:123456789012:mesh/app1",
            "createdAt": 1563809909.282,
            "lastUpdatedAt": 1563812829.687,
            "uid": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
            "version": 2
        },
        "spec": {
            "egressFilter": {
                "type": "ALLOW_ALL"
            }
        },
        "status": {
            "status": "ACTIVE"
        }
    }
}
```

Per ulteriori informazioni, consulta Mesh dei servizi nella Guida per l'utente di AWS App Mesh.

- Per i dettagli sull'API, consulta UpdateMesh AWS CLICommand Reference.

`update-route`

Il seguente esempio di codice mostra come utilizzare`update-route`.

AWS CLI

Come aggiornare una route

L'esempio `update-route` seguente utilizza un file di input JSON per aggiornare i pesi di una route.

```
aws appmesh update-route \
    --cli-input-json file://update-route-weighted.json
```

Contenuto di `update-route-weighted.json`:

```
{
    "meshName": "app1",
    "routeName": "toVnServiceB-weighted",
    "spec": {
        "httpRoute": {
            "action": {
                "weightedTargets": [
                    {
                        "virtualNode": "vnServiceBv1",
                        "weight": 80
                    },
                    {
                        "virtualNode": "vnServiceBv2",
                        "weight": 20
                    }
                ]
            },
            "match": {
                "prefix": "/"
            }
        }
    },
    "virtualRouterName": "vrServiceB"
}
```

Output:

```
{
    "route": {
        "meshName": "app1",
        "metadata": {
            "arn": "arn:aws:appmesh:us-east-1:123456789012:mesh/app1/virtualRouter/
vrServiceB/route/toVnServiceB-weighted",
            "createdAt": 1563811384.015,
            "lastUpdatedAt": 1563819600.022,
            "uid": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
            "version": 2
        },
        "routeName": "toVnServiceB-weighted",
        "spec": {
            "httpRoute": {
                "action": {
                    "weightedTargets": [
                        {
                            "virtualNode": "vnServiceBv1",
                            "weight": 80
                        },
                        {
                            "virtualNode": "vnServiceBv2",
                            "weight": 20
                        }
                    ]
                },
                "match": {
                    "prefix": "/"
                }
            }
        },
        "status": {
            "status": "ACTIVE"
        },
        "virtualRouterName": "vrServiceB"
    }
}
```

Per ulteriori informazioni, consulta Route nella Guida per l'utente di AWS App Mesh.

- Per i dettagli sull'API, consulta UpdateRoute AWS CLICommand Reference.

`update-virtual-node`

Il seguente esempio di codice mostra come utilizzare`update-virtual-node`.

AWS CLI

Come aggiornare un nodo virtuale

L'esempio `update-virtual-node` seguente utilizza un file di input JSON per aggiungere un controllo dell'integrità a un nodo virtuale.

```
aws appmesh update-virtual-node \
    --cli-input-json file://update-virtual-node.json
```

Contenuto di `update-virtual-node.json`:

```
{
    "clientToken": "500",
    "meshName": "app1",
    "spec": {
        "listeners": [
            {
                "healthCheck": {
                    "healthyThreshold": 5,
                    "intervalMillis": 10000,
                    "path": "/",
                    "port": 80,
                    "protocol": "http",
                    "timeoutMillis": 3000,
                    "unhealthyThreshold": 3
                },
                "portMapping": {
                    "port": 80,
                    "protocol": "http"
                }
            }
        ],
        "serviceDiscovery": {
            "dns": {
                "hostname": "serviceBv1.svc.cluster.local"
            }
        }
    },
```

```
    "virtualNodeName": "vnServiceBv1"
}
```

Output:

```
{
    "virtualNode": {
        "meshName": "app1",
        "metadata": {
            "arn": "arn:aws:appmesh:us-east-1:123456789012:mesh/app1/virtualNode/
vnServiceBv1",
            "createdAt": 1563810019.874,
            "lastUpdatedAt": 1563819234.825,
            "uid": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
            "version": 2
        },
        "spec": {
            "listeners": [
                {
                    "healthCheck": {
                        "healthyThreshold": 5,
                        "intervalMillis": 10000,
                        "path": "/",
                        "port": 80,
                        "protocol": "http",
                        "timeoutMillis": 3000,
                        "unhealthyThreshold": 3
                    },
                    "portMapping": {
                        "port": 80,
                        "protocol": "http"
                    }
                }
            ],
            "serviceDiscovery": {
                "dns": {
                    "hostname": "serviceBv1.svc.cluster.local"
                }
            }
        },
        "status": {
            "status": "ACTIVE"
        },
```

```
        "virtualNodeName": "vnServiceBv1"
    }
}
```

Per ulteriori informazioni, consulta Nodi virtuali nella Guida per l'utente di AWS App Mesh.

- Per i dettagli sull'API, consulta UpdateVirtualNode AWS CLICommand Reference.

`update-virtual-router`

Il seguente esempio di codice mostra come utilizzare`update-virtual-router`.

AWS CLI

Come aggiornare un router virtuale

L'esempio `update-virtual-router` seguente utilizza un file di input JSON per aggiornare la porta listener di un router virtuale.

```
aws appmesh update-virtual-router \
    --cli-input-json file://update-virtual-router.json
```

Contenuto di `update-virtual-router.json`:

```
{
    "meshName": "app1",
    "spec": {
        "listeners": [
            {
                "portMapping": {
                    "port": 8080,
                    "protocol": "http"
                }
            }
        ]
    },
    "virtualRouterName": "vrServiceB"
}
```

Output:

```
{
```

```
    "virtualRouter": {
        "meshName": "app1",
        "metadata": {
            "arn": "arn:aws:appmesh:us-east-1:123456789012:mesh/app1/virtualRouter/
vrServiceB",
            "createdAt": 1563810546.59,
            "lastUpdatedAt": 1563819431.352,
            "uid": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
            "version": 2
        },
        "spec": {
            "listeners": [
                {
                    "portMapping": {
                        "port": 8080,
                        "protocol": "http"
                    }
                }
            ]
        },
        "status": {
            "status": "ACTIVE"
        },
        "virtualRouterName": "vrServiceB"
    }
}
```

Per ulteriori informazioni, consulta Router virtuali nella Guida per l'utente di AWS App Mesh.

- Per i dettagli sull'API, consulta UpdateVirtualRouter AWS CLICommand Reference.

`update-virtual-service`

Il seguente esempio di codice mostra come utilizzare`update-virtual-service`.

AWS CLI

Come aggiornare un servizio virtuale

L'esempio `update-virtual-service` seguente utilizza un file di input JSON per aggiornare un servizio virtuale in modo che utilizzi un provider di router virtuali.

```
aws appmesh update-virtual-service \
```

```
    --cli-input-json file://update-virtual-service.json
```

Contenuto di `update-virtual-service.json`:

```
{
    "meshName": "app1",
    "spec": {
        "provider": {
            "virtualRouter": {
                "virtualRouterName": "vrServiceA"
            }
        }
    },
    "virtualServiceName": "serviceA.svc.cluster.local"
}
```

Output:

```
{
    "virtualService": {
        "meshName": "app1",
        "metadata": {
            "arn": "arn:aws:appmesh:us-east-1:123456789012:mesh/app1/virtualService/
serviceA.svc.cluster.local",
            "createdAt": 1563810859.474,
            "lastUpdatedAt": 1563820257.411,
            "uid": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
            "version": 3
        },
        "spec": {
            "provider": {
                "virtualRouter": {
                    "virtualRouterName": "vrServiceA"
                }
            }
        },
        "status": {
            "status": "ACTIVE"
        },
        "virtualServiceName": "serviceA.svc.cluster.local"
    }
}
```

Per ulteriori informazioni, consulta Servizio virtuale nella Guida per l'utente di AWS App Mesh.

- Per i dettagli sull'API, consulta UpdateVirtualService AWS CLICommand Reference.

Esempi di App Runner utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface con App Runner.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`associate-custom-domain`

Il seguente esempio di codice mostra come utilizzare`associate-custom-domain`.

AWS CLI

Come associare un nome di dominio e il sottodominio www a un servizio

L'esempio `associate-custom-domain` seguente associa un nome di dominio personalizzato controllato dall'utente a un servizio App Runner. Il nome di dominio è il dominio root `example.com`, incluso il sottodominio speciale `www.example.com`.

```
aws apprunner associate-custom-domain \
    --cli-input-json file://input.json
```

Contenuto di `input.json`:

```
{
    "ServiceArn": "arn:aws:apprunner:us-east-1:123456789012:service/python-
app/8fe1e10304f84fd2b0df550fe98a71fa",
```

```
    "DomainName": "example.com",
    "EnableWWWSubdomain": true
}
```

Output:

```
{
    "CustomDomain": {
        "CertificateValidationRecords": [
            {
                "Name": "_70d3f50a94f7c72dc28784cf55db2f6b.example.com",
                "Status": "PENDING_VALIDATION",
                "Type": "CNAME",
                "Value": "_1270c137383c6307b6832db02504c4b0.bsgbmzkfwj.acm-
validations.aws."
            },
            {
                "Name": "_287870d3f50a94f7c72dc4cf55db2f6b.www.example.com",
                "Status": "PENDING_VALIDATION",
                "Type": "CNAME",
                "Value": "_832db01270c137383c6307b62504c4b0.mzkbsgbfwj.acm-
validations.aws."
            }
        ],
        "DomainName": "example.com",
        "EnableWWWSubdomain": true,
        "Status": "CREATING"
    },
    "DNSTarget": "psbqam834h.us-east-1.awsapprunner.com",
    "ServiceArn": "arn:aws:apprunner:us-east-1:123456789012:service/python-
app/8fe1e10304f84fd2b0df550fe98a71fa"
}
```

- Per i dettagli sull'API, consulta AssociateCustomDomain AWS CLICommand Reference.

`create-auto-scaling-configuration`

Il seguente esempio di codice mostra come utilizzare`create-auto-scaling-configuration`.

AWS CLI

Come creare una configurazione di dimensionamento automatico ad elevata disponibilità

L'esempio `create-auto-scaling-configuration` seguente crea una configurazione del dimensionamento automatico ottimizzata per la disponibilità elevata impostando `MinSize` su 5. Con questa configurazione, App Runner tenta di distribuire le istanze del servizio nella maggior parte delle zone di disponibilità possibili, fino a cinque, a seconda della AWS regione.

La chiamata restituisce un oggetto `AutoScalingConfiguration` con le altre impostazioni configurate sui valori predefiniti. Nell'esempio, questa è la prima chiamata per creare una configurazione denominata `high-availability`. La revisione è impostata su 1 ed è l'ultima revisione.

```
aws apprunner create-auto-scaling-configuration \
    --cli-input-json file://input.json
```

Contenuto di `input.json`:

```
{
    "AutoScalingConfigurationName": "high-availability",
    "MinSize": 5
}
```

Output:

```
{
    "AutoScalingConfiguration": {
        "AutoScalingConfigurationArn": "arn:aws:apprunner:us-
east-1:123456789012:autoscalingconfiguration/high-
availability/1/2f50e7656d7819fead0f59672e68042e",
        "AutoScalingConfigurationName": "high-availability",
        "AutoScalingConfigurationRevision": 1,
        "CreatedAt": "2020-11-03T00:29:17Z",
        "Latest": true,
        "Status": "ACTIVE",
        "MaxConcurrency": 100,
        "MaxSize": 50,
        "MinSize": 5
    }
}
```

- Per i dettagli sull'API, consulta AWS CLI Command CreateAutoScalingConfigurationReference.

`create-connection`

Il seguente esempio di codice mostra come utilizzare`create-connection`.

AWS CLI

Per creare una GitHub connessione

L'`create-connection`esempio seguente crea una connessione a un archivio di GitHub codice privato. Lo stato della connessione dopo una chiamata riuscita è `PENDING_HANDSHAKE`. Questo perché l'handshake di autenticazione con il provider non è ancora avvenuto. Completa l'handshake utilizzando la console App Runner.

```
aws apprunner create-connection \
    --cli-input-json file://input.json
```

Contenuto di `input.json`:

```
{
    "ConnectionName": "my-github-connection",
    "ProviderType": "GITHUB"
}
```

Output:

```
{
    "Connection": {
        "ConnectionArn": "arn:aws:apprunner:us-east-1:123456789012:connection/my-
github-connection",
        "ConnectionName": "my-github-connection",
        "Status": "PENDING_HANDSHAKE",
        "CreatedAt": "2020-11-03T00:32:51Z",
        "ProviderType": "GITHUB"
    }
}
```

Per ulteriori informazioni, consulta Gestione delle connessioni di App Runner nella Guida per gli sviluppatori di AWS App Runner.

- Per i dettagli sull'API, vedere CreateConnectionin AWS CLI Command Reference.

`create-service`

Il seguente esempio di codice mostra come utilizzare`create-service`.

AWS CLI

Esempio 1: come creare un servizio di repository di codice sorgente

L'esempio `create-service` seguente crea un servizio App Runner basato su un repository di codice sorgente Python.

```
aws apprunner create-service \
    --cli-input-json file://input.json
```

Contenuto di `input.json`:

```
{
    "ServiceName": "python-app",
    "SourceConfiguration": {
        "AuthenticationConfiguration": {
            "ConnectionArn": "arn:aws:apprunner:us-east-1:123456789012:connection/
my-github-connection/e7656250f67242d7819feade6800f59e"
        },
        "AutoDeploymentsEnabled": true,
        "CodeRepository": {
            "RepositoryUrl": "https://github.com/my-account/python-hello",
            "SourceCodeVersion": {
                "Type": "BRANCH",
                "Value": "main"
            },
            "CodeConfiguration": {
                "ConfigurationSource": "API",
                "CodeConfigurationValues": {
                    "Runtime": "PYTHON_3",
                    "BuildCommand": "pip install -r requirements.txt",
                    "StartCommand": "python server.py",
                    "Port": "8080",
                    "RuntimeEnvironmentVariables": [
                        {
                            "NAME": "Jane"
                        }
                    ]
                }
```

```
            }
        }
    },
    "InstanceConfiguration": {
        "CPU": "1 vCPU",
        "Memory": "3 GB"
    }
}
```

Output:

```
{
    "OperationId": "17fe9f55-7e91-4097-b243-fcabbb69a4cf",
    "Service": {
        "CreatedAt": "2020-11-20T19:05:25Z",
        "UpdatedAt": "2020-11-20T19:05:25Z",
        "ServiceArn": "arn:aws:apprunner:us-east-1:123456789012:service/python-
app/8fe1e10304f84fd2b0df550fe98a71fa",
        "ServiceId": "8fe1e10304f84fd2b0df550fe98a71fa",
        "ServiceName": "python-app",
        "ServiceUrl": "psbqam834h.us-east-1.awsapprunner.com",
        "SourceConfiguration": {
            "AuthenticationConfiguration": {
                "ConnectionArn": "arn:aws:apprunner:us-
east-1:123456789012:connection/my-github-connection/
e7656250f67242d7819feade6800f59e"
            },
            "AutoDeploymentsEnabled": true,
            "CodeRepository": {
                "CodeConfiguration": {
                    "CodeConfigurationValues": {
                        "BuildCommand": "pip install -r requirements.txt",
                        "Port": "8080",
                        "Runtime": "PYTHON_3",
                        "RuntimeEnvironmentVariables": [
                            {
                                "NAME": "Jane"
                            }
                        ],
                        "StartCommand": "python server.py"
                    },
                    "ConfigurationSource": "Api"
                },
```

```
                "RepositoryUrl": "https://github.com/my-account/python-hello",
                "SourceCodeVersion": {
                    "Type": "BRANCH",
                    "Value": "main"
                }
            }
        },
        "Status": "OPERATION_IN_PROGRESS",
        "InstanceConfiguration": {
            "CPU": "1 vCPU",
            "Memory": "3 GB"
        }
    }
}
```

Esempio 2: come creare un servizio di repository di codice sorgente

L'esempio `create-service` seguente crea un servizio App Runner basato su un repository di codice sorgente Python.

```
aws apprunner create-service \
    --cli-input-json file://input.json
```

Contenuto di `input.json`:

```
{
    "ServiceName": "python-app",
    "SourceConfiguration": {
        "AuthenticationConfiguration": {
            "ConnectionArn": "arn:aws:apprunner:us-east-1:123456789012:connection/
my-github-connection/e7656250f67242d7819feade6800f59e"
        },
        "AutoDeploymentsEnabled": true,
        "CodeRepository": {
            "RepositoryUrl": "https://github.com/my-account/python-hello",
            "SourceCodeVersion": {
                "Type": "BRANCH",
                "Value": "main"
            },
            "CodeConfiguration": {
                "ConfigurationSource": "API",
                "CodeConfigurationValues": {
                    "Runtime": "PYTHON_3",
```

```
                    "BuildCommand": "pip install -r requirements.txt",
                    "StartCommand": "python server.py",
                    "Port": "8080",
                    "RuntimeEnvironmentVariables": [
                        {
                            "NAME": "Jane"
                        }
                    ]
                }
            }
        },
    },
    "InstanceConfiguration": {
        "CPU": "1 vCPU",
        "Memory": "3 GB"
    }
}
```

Output:

```
{
    "OperationId": "17fe9f55-7e91-4097-b243-fcabbb69a4cf",
    "Service": {
        "CreatedAt": "2020-11-20T19:05:25Z",
        "UpdatedAt": "2020-11-20T19:05:25Z",
        "ServiceArn": "arn:aws:apprunner:us-east-1:123456789012:service/python-
app/8fe1e10304f84fd2b0df550fe98a71fa",
        "ServiceId": "8fe1e10304f84fd2b0df550fe98a71fa",
        "ServiceName": "python-app",
        "ServiceUrl": "psbqam834h.us-east-1.awsapprunner.com",
        "SourceConfiguration": {
            "AuthenticationConfiguration": {
                "ConnectionArn": "arn:aws:apprunner:us-
east-1:123456789012:connection/my-github-connection/
e7656250f67242d7819feade6800f59e"
            },
            "AutoDeploymentsEnabled": true,
            "CodeRepository": {
                "CodeConfiguration": {
                    "CodeConfigurationValues": {
                        "BuildCommand": "pip install -r requirements.txt",
                        "Port": "8080",
                        "Runtime": "PYTHON_3",
```

```
                        "RuntimeEnvironmentVariables": [
                            {
                                "NAME": "Jane"
                            }
                        ],
                        "StartCommand": "python server.py"
                    },
                    "ConfigurationSource": "Api"
                },
                "RepositoryUrl": "https://github.com/my-account/python-hello",
                "SourceCodeVersion": {
                    "Type": "BRANCH",
                    "Value": "main"
                }
            }
        },
        "Status": "OPERATION_IN_PROGRESS",
        "InstanceConfiguration": {
            "CPU": "1 vCPU",
            "Memory": "3 GB"
        }
    }
}
```

Esempio 3: come creare un servizio di repository di immagini di origine

L'esempio `create-service` seguente crea un servizio App Runner basato su un'immagine archiviata in Elastic Container Registry (ECR).

```
aws apprunner create-service \
    --cli-input-json file://input.json
```

Contenuto di `input.json`:

```
{
    "ServiceName": "golang-container-app",
    "SourceConfiguration": {
        "AuthenticationConfiguration": {
            "AccessRoleArn": "arn:aws:iam::123456789012:role/my-ecr-role"
        },
        "AutoDeploymentsEnabled": true,
        "ImageRepository": {
```

```
            "ImageIdentifier": "123456789012.dkr.ecr.us-east-1.amazonaws.com/golang-
app:latest",
            "ImageConfiguration": {
                "Port": "8080",
                "RuntimeEnvironmentVariables": [
                    {
                        "NAME": "Jane"
                    }
                ]
            },
            "ImageRepositoryType": "ECR"
        }
    },
    "InstanceConfiguration": {
        "CPU": "1 vCPU",
        "Memory": "3 GB"
    }
}
```

Output:

```
{
    "OperationId": "17fe9f55-7e91-4097-b243-fcabbb69a4cf",
    "Service": {
        "CreatedAt": "2020-11-06T23:15:30Z",
        "UpdatedAt": "2020-11-06T23:15:30Z",
        "ServiceArn": "arn:aws:apprunner:us-east-1:123456789012:service/golang-
container-app/51728f8a20ce46d39b25398a6c8e9d1a",
        "ServiceId": "51728f8a20ce46d39b25398a6c8e9d1a",
        "ServiceName": "golang-container-app",
        "ServiceUrl": "psbqam834h.us-east-1.awsapprunner.com",
        "SourceConfiguration": {
            "AuthenticationConfiguration": {
                "AccessRoleArn": "arn:aws:iam::123456789012:role/my-ecr-role"
            },
            "AutoDeploymentsEnabled": true,
            "ImageRepository": {
                "ImageIdentifier": "123456789012.dkr.ecr.us-east-1.amazonaws.com/
golang-app:latest",
                "ImageConfiguration": {
                    "Port": "8080",
                    "RuntimeEnvironmentVariables": [
                        {
```

```
                            "NAME": "Jane"
                        }
                    ]
                },
                "ImageRepositoryType": "ECR"
            }
        },
        "Status": "OPERATION_IN_PROGRESS",
        "InstanceConfiguration": {
            "CPU": "1 vCPU",
            "Memory": "3 GB"
        }
    }
}
```

- Per i dettagli sull'API, consulta CreateService AWS CLICommand Reference.

`delete-auto-scaling-configuration`

Il seguente esempio di codice mostra come utilizzare`delete-auto-scaling-configuration`.

AWS CLI

Esempio 1: come eliminare l'ultima revisione attiva di una configurazione di dimensionamento automatico

L'esempio `delete-auto-scaling-configuration` seguente elimina l'ultima revisione attiva di una configurazione di dimensionamento automatico di App Runner. Per eliminare l'ultima revisione attiva, specifica un nome della risorsa Amazon (ARN) che termini con il nome della configurazione, senza includere il componente della revisione.

Nell'esempio, prima di questa azione esistono due revisioni. Pertanto, viene eliminata la revisione 2 (l'ultima). Tuttavia, ora viene mostra `"Latest": false`, perché, dopo essere stata eliminata, non è più l'ultima revisione attiva.

```
aws apprunner delete-auto-scaling-configuration \
    --cli-input-json file://input.json
```

Contenuto di `input.json`:

```
{
```

```
    "AutoScalingConfigurationArn": "arn:aws:apprunner:us-
east-1:123456789012:autoscalingconfiguration/high-availability"
}
```

Output:

```
{
    "AutoScalingConfiguration": {
        "AutoScalingConfigurationArn": "arn:aws:apprunner:us-
east-1:123456789012:autoscalingconfiguration/high-availability/2/
e76562f50d78042e819fead0f59672e6",
        "AutoScalingConfigurationName": "high-availability",
        "AutoScalingConfigurationRevision": 2,
        "CreatedAt": "2021-02-25T17:42:59Z",
        "DeletedAt": "2021-03-02T08:07:06Z",
        "Latest": false,
        "Status": "INACTIVE",
        "MaxConcurrency": 30,
        "MaxSize": 90,
        "MinSize": 5
    }
}
```

Esempio 2: come eliminare una revisione specifica di una configurazione di dimensionamento automatico

L'esempio `delete-auto-scaling-configuration` seguente elimina una revisione specifica di una configurazione di dimensionamento automatico di App Runner. Per eliminare una revisione specifica, specifica un ARN che includa il numero della revisione.

Nell'esempio, prima di questa azione esistono diverse revisioni. L'azione elimina la revisione 1.

```
aws apprunner delete-auto-scaling-configuration \
    --cli-input-json file://input.json
```

Contenuto di `input.json`:

```
{
    "AutoScalingConfigurationArn": "arn:aws:apprunner:us-
east-1:123456789012:autoscalingconfiguration/high-availability/1"
}
```

Output:

```
{
    "AutoScalingConfiguration": {
        "AutoScalingConfigurationArn": "arn:aws:apprunner:us-
east-1:123456789012:autoscalingconfiguration/high-
availability/1/2f50e7656d7819fead0f59672e68042e",
        "AutoScalingConfigurationName": "high-availability",
        "AutoScalingConfigurationRevision": 1,
        "CreatedAt": "2020-11-03T00:29:17Z",
        "DeletedAt": "2021-03-02T08:07:06Z",
        "Latest": false,
        "Status": "INACTIVE",
        "MaxConcurrency": 100,
        "MaxSize": 50,
        "MinSize": 5
    }
}
```

- Per i dettagli sull'API, consulta DeleteAutoScalingConfiguration AWS CLICommand Reference.

`delete-connection`

Il seguente esempio di codice mostra come utilizzare`delete-connection`.

AWS CLI

Come eliminare una connessione

L'esempio `delete-connection` seguente elimina una connessione App Runner. Lo stato della connessione dopo una chiamata riuscita è `DELETED`. Questo perché la connessione non è più disponibile.

```
aws apprunner delete-connection \
    --cli-input-json file://input.json
```

Contenuto di `input.json`:

```
{
    "ConnectionArn": "arn:aws:apprunner:us-east-1:123456789012:connection/my-github-
connection"
```

```
}
```

Output:

```
{
    "Connection": {
        "ConnectionArn": "arn:aws:apprunner:us-east-1:123456789012:connection/my-
github-connection",
        "ConnectionName": "my-github-connection",
        "Status": "DELETED",
        "CreatedAt": "2020-11-03T00:32:51Z",
        "ProviderType": "GITHUB"
    }
}
```

- Per i dettagli sull'API, consulta DeleteConnection AWS CLICommand Reference.

`delete-service`

Il seguente esempio di codice mostra come utilizzare`delete-service`.

AWS CLI

Come eliminare un servizio

L'esempio `delete-service` seguente elimina un servizio App Runner.

```
aws apprunner delete-service \
    --cli-input-json file://input.json
```

Contenuto di `input.json`:

```
{
    "ServiceArn": "arn:aws:apprunner:us-east-1:123456789012:service/python-
app/8fe1e10304f84fd2b0df550fe98a71fa"
}
```

Output:

```
{
    "OperationId": "17fe9f55-7e91-4097-b243-fcabbb69a4cf",
    "Service": {
```

```
        "CreatedAt": "2020-11-20T19:05:25Z",
        "UpdatedAt": "2020-11-20T19:05:25Z",
        "ServiceArn": "arn:aws:apprunner:us-east-1:123456789012:service/python-
app/8fe1e10304f84fd2b0df550fe98a71fa",
        "ServiceId": "8fe1e10304f84fd2b0df550fe98a71fa",
        "ServiceName": "python-app",
        "ServiceUrl": "psbqam834h.us-east-1.awsapprunner.com",
        "SourceConfiguration": {
            "AuthenticationConfiguration": {
                "ConnectionArn": "arn:aws:apprunner:us-
east-1:123456789012:connection/my-github-connection/
e7656250f67242d7819feade6800f59e"
            },
            "AutoDeploymentsEnabled": true,
            "CodeRepository": {
                "CodeConfiguration": {
                    "CodeConfigurationValues": {
                        "BuildCommand": "pip install -r requirements.txt",
                        "Port": "8080",
                        "Runtime": "PYTHON_3",
                        "RuntimeEnvironmentVariables": [
                            {
                                "NAME": "Jane"
                            }
                        ],
                        "StartCommand": "python server.py"
                    },
                    "ConfigurationSource": "Api"
                },
                "RepositoryUrl": "https://github.com/my-account/python-hello",
                "SourceCodeVersion": {
                    "Type": "BRANCH",
                    "Value": "main"
                }
            }
        },
        "Status": "OPERATION_IN_PROGRESS",
        "InstanceConfiguration": {
            "CPU": "1 vCPU",
            "Memory": "3 GB"
        }
    }
}
```

- Per i dettagli sull'API, consulta DeleteService AWS CLICommand Reference.

`describe-auto-scaling-configuration`

Il seguente esempio di codice mostra come utilizzare`describe-auto-scaling-configuration`.

AWS CLI

Esempio 1: come descrivere l'ultima revisione attiva di una configurazione di dimensionamento automatico

L'esempio `describe-auto-scaling-configuration` seguente ottiene una descrizione dell'ultima revisione attiva di una configurazione di dimensionamento automatico di App Runner. Per descrivere l'ultima revisione attiva, specifica un ARN che termini con il nome della configurazione, senza includere il componente della revisione.

Nell'esempio, esistono due revisioni. Pertanto, viene descritta la revisione 2 (l'ultima). L'oggetto risultante mostra `"Latest": true`.

```
aws apprunner describe-auto-scaling-configuration \
    --cli-input-json file://input.json
```

Contenuto di `input.json`:

```
{
    "AutoScalingConfigurationArn": "arn:aws:apprunner:us-
east-1:123456789012:autoscalingconfiguration/high-availability"
}
```

Output:

```
{
    "AutoScalingConfiguration": {
        "AutoScalingConfigurationArn": "arn:aws:apprunner:us-
east-1:123456789012:autoscalingconfiguration/high-availability/2/
e76562f50d78042e819fead0f59672e6",
        "AutoScalingConfigurationName": "high-availability",
        "AutoScalingConfigurationRevision": 2,
        "CreatedAt": "2021-02-25T17:42:59Z",
        "Latest": true,
        "Status": "ACTIVE",
```

```
        "MaxConcurrency": 30,
        "MaxSize": 90,
        "MinSize": 5
    }
}
```

Esempio 2: come descrivere una revisione specifica di una configurazione di dimensionamento automatico

L'esempio `describe-auto-scaling-configuration` seguente ottiene una descrizione di una revisione specifica di una configurazione di dimensionamento automatico di App Runner. Per descrivere una revisione specifica, specifica un ARN che includa il numero della revisione.

Nell'esempio, esistono diverse revisioni e viene richiesta la revisione 1. L'oggetto risultante mostra `"Latest": false`.

```
aws apprunner describe-auto-scaling-configuration \
    --cli-input-json file://input.json
```

Contenuto di `input.json`:

```
{
    "AutoScalingConfigurationArn": "arn:aws:apprunner:us-
east-1:123456789012:autoscalingconfiguration/high-availability/1"
}
```

Output:

```
{
    "AutoScalingConfiguration": {
        "AutoScalingConfigurationArn": "arn:aws:apprunner:us-
east-1:123456789012:autoscalingconfiguration/high-
availability/1/2f50e7656d7819fead0f59672e68042e",
        "AutoScalingConfigurationName": "high-availability",
        "AutoScalingConfigurationRevision": 1,
        "CreatedAt": "2020-11-03T00:29:17Z",
        "Latest": false,
        "Status": "ACTIVE",
        "MaxConcurrency": 100,
        "MaxSize": 50,
        "MinSize": 5
    }
```

```
}
```

- Per i dettagli sull'API, consulta DescribeAutoScalingConfiguration AWS CLICommand Reference.

`describe-custom-domains`

Il seguente esempio di codice mostra come utilizzare`describe-custom-domains`.

AWS CLI

Come ottenere le descrizioni dei nomi di dominio personalizzati associati a un servizio

L'esempio `describe-custom-domains` seguente ottiene le descrizioni e lo stato dei nomi di dominio personalizzati associati a un servizio App Runner.

```
aws apprunner describe-custom-domains \
    --cli-input-json file://input.json
```

Contenuto di `input.json`:

```
{
    "ServiceArn": "arn:aws:apprunner:us-east-1:123456789012:service/python-
app/8fe1e10304f84fd2b0df550fe98a71fa",
    "DomainName": "example.com",
    "EnableWWWSubdomain": true
}
```

Output:

```
{
    "CustomDomains": [
        {
            "CertificateValidationRecords": [
                {
                    "Name": "_70d3f50a94f7c72dc28784cf55db2f6b.example.com",
                    "Status": "PENDING_VALIDATION",
                    "Type": "CNAME",
                    "Value": "_1270c137383c6307b6832db02504c4b0.bsgbmzkfwj.acm-
validations.aws."
                },
```

```
                {
                    "Name": "_287870d3f50a94f7c72dc4cf55db2f6b.www.example.com",
                    "Status": "PENDING_VALIDATION",
                    "Type": "CNAME",
                    "Value": "_832db01270c137383c6307b62504c4b0.mzkbsgbfwj.acm-
validations.aws."
                }
            ],
            "DomainName": "example.com",
            "EnableWWWSubdomain": true,
            "Status": "PENDING_CERTIFICATE_DNS_VALIDATION"
        },
        {
            "CertificateValidationRecords": [
                {
                    "Name": "_a94f784c70d3f507c72dc28f55db2f6b.deals.example.com",
                    "Status": "SUCCESS",
                    "Type": "CNAME",
                    "Value": "_2db02504c1270c137383c6307b6834b0.bsgbmzkfwj.acm-
validations.aws."
                }
            ],
            "DomainName": "deals.example.com",
            "EnableWWWSubdomain": false,
            "Status": "ACTIVE"
        }
    ],
    "DNSTarget": "psbqam834h.us-east-1.awsapprunner.com",
    "ServiceArn": "arn:aws:apprunner:us-east-1:123456789012:service/python-
app/8fe1e10304f84fd2b0df550fe98a71fa"
}
```

- Per i dettagli sull'API, consulta DescribeCustomDomains AWS CLICommand Reference.

`describe-service`

Il seguente esempio di codice mostra come utilizzare`describe-service`.

AWS CLI

Come descrivere un servizio

L'esempio `describe-service` seguente ottiene una descrizione di un servizio App Runner.

```
aws apprunner describe-service \
    --cli-input-json file://input.json
```

Contenuto di `input.json`:

```
{
    "ServiceArn": "arn:aws:apprunner:us-east-1:123456789012:service/python-
app/8fe1e10304f84fd2b0df550fe98a71fa"
}
```

Output:

```
{
    "Service": {
        "CreatedAt": "2020-11-20T19:05:25Z",
        "UpdatedAt": "2020-11-20T19:05:25Z",
        "ServiceArn": "arn:aws:apprunner:us-east-1:123456789012:service/python-
app/8fe1e10304f84fd2b0df550fe98a71fa",
        "ServiceId": "8fe1e10304f84fd2b0df550fe98a71fa",
        "ServiceName": "python-app",
        "ServiceUrl": "psbqam834h.us-east-1.awsapprunner.com",
        "SourceConfiguration": {
            "AuthenticationConfiguration": {
                "ConnectionArn": "arn:aws:apprunner:us-
east-1:123456789012:connection/my-github-connection/
e7656250f67242d7819feade6800f59e"
            },
            "AutoDeploymentsEnabled": true,
            "CodeRepository": {
                "CodeConfiguration": {
                    "CodeConfigurationValues": {
                        "BuildCommand": "pip install -r requirements.txt",
                        "Port": "8080",
                        "Runtime": "PYTHON_3",
                        "RuntimeEnvironmentVariables": [
                            {
                                "NAME": "Jane"
                            }
                        ],
                        "StartCommand": "python server.py"
                    },
                    "ConfigurationSource": "Api"
```

```
                },
                "RepositoryUrl": "https://github.com/my-account/python-hello",
                "SourceCodeVersion": {
                    "Type": "BRANCH",
                    "Value": "main"
                }
            }
        },
        "Status": "RUNNING",
        "InstanceConfiguration": {
            "CPU": "1 vCPU",
            "Memory": "3 GB"
        }
    }
}
```

- Per i dettagli sull'API, consulta DescribeService AWS CLICommand Reference.

`disassociate-custom-domain`

Il seguente esempio di codice mostra come utilizzare`disassociate-custom-domain`.

AWS CLI

Come annullare l'associazione di un nome di dominio da un servizio

L'esempio `disassociate-custom-domain` seguente annulla l'associazione tra il dominio `example.com` e un servizio App Runner. La chiamata annulla anche l'associazione del sottodominio `www.example.com` che era associato al dominio root.

```
aws apprunner disassociate-custom-domain \
    --cli-input-json file://input.json
```

Contenuto di `input.json`:

```
{
    "ServiceArn": "arn:aws:apprunner:us-east-1:123456789012:service/python-
app/8fe1e10304f84fd2b0df550fe98a71fa",
    "DomainName": "example.com"
}
```

Output:

```
{
    "CustomDomain": {
        "CertificateValidationRecords": [
            {
                "Name": "_70d3f50a94f7c72dc28784cf55db2f6b.example.com",
                "Status": "PENDING_VALIDATION",
                "Type": "CNAME",
                "Value": "_1270c137383c6307b6832db02504c4b0.bsgbmzkfwj.acm-
validations.aws."
            },
            {
                "Name": "_287870d3f50a94f7c72dc4cf55db2f6b.www.example.com",
                "Status": "PENDING_VALIDATION",
                "Type": "CNAME",
                "Value": "_832db01270c137383c6307b62504c4b0.mzkbsgbfwj.acm-
validations.aws."
            }
        ],
        "DomainName": "example.com",
        "EnableWWWSubdomain": true,
        "Status": "DELETING"
    },
    "DNSTarget": "psbqam834h.us-east-1.awsapprunner.com",
    "ServiceArn": "arn:aws:apprunner:us-east-1:123456789012:service/python-
app/8fe1e10304f84fd2b0df550fe98a71fa"
}
```

- Per i dettagli sull'API, consulta DisassociateCustomDomain AWS CLICommand Reference.

`list-auto-scaling-configurations`

Il seguente esempio di codice mostra come utilizzare`list-auto-scaling-configurations`.

AWS CLI

Come ottenere un elenco impaginato delle configurazioni di dimensionamento automatico di App Runner

L'`list-auto-scaling-configurations`esempio seguente elenca tutte le configurazioni di scalabilità automatica di App Runner presenti nell'account. AWS In ogni risposta sono elencate fino a cinque configurazioni di dimensionamento automatico.

`AutoScalingConfigurationName` e `LatestOnly` non sono specificati. I loro valori predefiniti fanno sì che venga elencata l'ultima revisione di tutte le configurazioni attive.

In questo esempio, la risposta include due risultati e, non essendocene altri, non viene restituito alcun `NextToken`.

```
aws apprunner list-auto-scaling-configurations \
    --cli-input-json file://input.json
```

Contenuto di `input.json`:

```
{
    "MaxResults": 5
}
```

Output:

```
{
    "AutoScalingConfigurationSummaryList": [
        {
            "AutoScalingConfigurationArn": "arn:aws:apprunner:us-
east-1:123456789012:autoscalingconfiguration/high-availability/2/
e76562f50d78042e819fead0f59672e6",
            "AutoScalingConfigurationName": "high-availability",
            "AutoScalingConfigurationRevision": 2
        },
        {
            "AutoScalingConfigurationArn": "arn:aws:apprunner:us-
east-1:123456789012:autoscalingconfiguration/low-
cost/1/50d7804e7656fead0f59672e62f2e819",
            "AutoScalingConfigurationName": "low-cost",
            "AutoScalingConfigurationRevision": 1
        }
    ]
}
```

- Per i dettagli sull'API, consulta Command ListAutoScalingConfigurationsReference AWS CLI .

`list-connections`

Il seguente esempio di codice mostra come utilizzare`list-connections`.

AWS CLI

Esempio 1: come elencare tutte le connessioni

L'`list-connections`esempio seguente elenca tutte le connessioni App Runner nell' AWS account.

```
aws apprunner list-connections
```

Output:

```
{
    "ConnectionSummaryList": [
        {
            "ConnectionArn": "arn:aws:apprunner:us-east-1:123456789012:connection/
my-github-connection",
            "ConnectionName": "my-github-connection",
            "Status": "AVAILABLE",
            "CreatedAt": "2020-11-03T00:32:51Z",
            "ProviderType": "GITHUB"
        },
        {
            "ConnectionArn": "arn:aws:apprunner:us-east-1:123456789012:connection/
my-github-org-connection",
            "ConnectionName": "my-github-org-connection",
            "Status": "AVAILABLE",
            "CreatedAt": "2020-11-03T02:54:17Z",
            "ProviderType": "GITHUB"
        }
    ]
}
```

Esempio 2: come elencare una connessione in base al nome

L'esempio `list-connections` seguente elenca una connessione in base al relativo nome.

```
aws apprunner list-connections \
    --cli-input-json file://input.json
```

Contenuto di `input.json`:

```
{
```

```
    "ConnectionName": "my-github-org-connection"
}
```

Output:

```
{
    "ConnectionSummaryList": [
        {
            "ConnectionArn": "arn:aws:apprunner:us-east-1:123456789012:connection/
my-github-org-connection",
            "ConnectionName": "my-github-org-connection",
            "Status": "AVAILABLE",
            "CreatedAt": "2020-11-03T02:54:17Z",
            "ProviderType": "GITHUB"
        }
    ]
}
```

- Per i dettagli sull'API, consulta ListConnections AWS CLICommand Reference.

`list-operations`

Il seguente esempio di codice mostra come utilizzare`list-operations`.

AWS CLI

Come elencare le operazioni eseguite su un servizio

L'esempio `list-operations` seguente elenca tutte le operazioni eseguite finora su un servizio App Runner. In questo esempio, il servizio è nuovo ed è stata eseguita solo una singola operazione di tipo `CREATE_SERVICE`.

```
aws apprunner list-operations \
    --cli-input-json file://input.json
```

Contenuto di `input.json`:

```
{
    "ServiceArn": "arn:aws:apprunner:us-east-1:123456789012:service/python-
app/8fe1e10304f84fd2b0df550fe98a71fa"
}
```

Output:

```
{
    "OperationSummaryList": [
        {
            "EndedAt": 1606156217,
            "Id": "17fe9f55-7e91-4097-b243-fcabbb69a4cf",
            "StartedAt": 1606156014,
            "Status": "SUCCEEDED",
            "TargetArn": "arn:aws:apprunner:us-east-1:123456789012:service/python-
app/8fe1e10304f84fd2b0df550fe98a71fa",
            "Type": "CREATE_SERVICE",
            "UpdatedAt": 1606156217
        }
    ]
}
```

- Per i dettagli sull'API, consulta ListOperations AWS CLICommand Reference.

`list-services`

Il seguente esempio di codice mostra come utilizzare`list-services`.

AWS CLI

Come ottenere un elenco impaginato dei servizi App Runner

L'`list-services`esempio seguente elenca tutti i servizi App Runner dell' AWS account. In ogni risposta sono elencati fino a due servizi. In questo esempio viene mostrata la prima richiesta. La risposta include due risultati e un token che può essere utilizzato nella richiesta successiva. Quando una risposta successiva non include un token, significa che tutti i servizi sono stati elencati.

```
aws apprunner list-services \
    --cli-input-json file://input.json
```

Contenuto di `input.json`:

```
{
    "MaxResults": 2
```

```
}
```

Output:

```
{
    "NextToken":
 "eyJDdXN0b21lckFjY291bnRJZCI6IjI3MDIwNTQwMjg0NSIsIlNlcnZpY2VTdGF0dXNDb2RlIjoiUFJPVklTU9OU
    "ServiceSummaryList": [
        {
            "CreatedAt": "2020-11-20T19:05:25Z",
            "UpdatedAt": "2020-11-23T12:41:37Z",
            "ServiceArn": "arn:aws:apprunner:us-east-1:123456789012:service/python-
app/8fe1e10304f84fd2b0df550fe98a71fa",
            "ServiceId": "8fe1e10304f84fd2b0df550fe98a71fa",
            "ServiceName": "python-app",
            "ServiceUrl": "psbqam834h.us-east-1.awsapprunner.com",
            "Status": "RUNNING"
        },
        {
            "CreatedAt": "2020-11-06T23:15:30Z",
            "UpdatedAt": "2020-11-23T13:21:22Z",
            "ServiceArn": "arn:aws:apprunner:us-east-1:123456789012:service/golang-
container-app/ab8f94cfe29a460fb8760afd2ee87555",
            "ServiceId": "ab8f94cfe29a460fb8760afd2ee87555",
            "ServiceName": "golang-container-app",
            "ServiceUrl": "e2m8rrrx33.us-east-1.awsapprunner.com",
            "Status": "RUNNING"
        }
    ]
}
```

- Per i dettagli sull'API, consulta ListServices AWS CLICommand Reference.

`list-tags-for-resource`

Il seguente esempio di codice mostra come utilizzare`list-tags-for-resource`.

AWS CLI

Come elencare tag associati a un servizio App Runner

L'esempio `list-tags-for-resource` seguente elenca tutti i tag associati a un servizio.

```
aws apprunner list-tags-for-resource \
    --cli-input-json file://input.json
```

Contenuto di `input.json`:

```
{
    "ResourceArn": "arn:aws:apprunner:us-east-1:123456789012:service/python-
app/8fe1e10304f84fd2b0df550fe98a71fa"
}
```

Output:

```
{
    "Tags": [
        {
            "Key": "Department",
            "Value": "Retail"
        },
        {
            "Key": "CustomerId",
            "Value": "56439872357912"
        }
    ]
}
```

- Per i dettagli sull'API, consulta ListTagsForResource AWS CLICommand Reference.

`pause-service`

Il seguente esempio di codice mostra come utilizzare`pause-service`.

AWS CLI

Come sospendere un servizio

L'esempio `pause-service` seguente mette in pausa un servizio App Runner.

```
aws apprunner pause-service \
    --cli-input-json file://input.json
```

Contenuto di `input.json`:

```
{
    "ServiceArn": "arn:aws:apprunner:us-east-1:123456789012:service/python-
app/8fe1e10304f84fd2b0df550fe98a71fa"
}
```

Output:

```
{
    "OperationId": "17fe9f55-7e91-4097-b243-fcabbb69a4cf",
    "Service": {
        "CreatedAt": "2020-11-20T19:05:25Z",
        "UpdatedAt": "2020-11-23T12:41:37Z",
        "ServiceArn": "arn:aws:apprunner:us-east-1:123456789012:service/python-
app/8fe1e10304f84fd2b0df550fe98a71fa",
        "ServiceId": "8fe1e10304f84fd2b0df550fe98a71fa",
        "ServiceName": "python-app",
        "ServiceUrl": "psbqam834h.us-east-1.awsapprunner.com",
        "SourceConfiguration": {
            "AuthenticationConfiguration": {
                "ConnectionArn": "arn:aws:apprunner:us-
east-1:123456789012:connection/my-github-connection/
e7656250f67242d7819feade6800f59e"
            },
            "AutoDeploymentsEnabled": true,
            "CodeRepository": {
                "CodeConfiguration": {
                    "CodeConfigurationValues": {
                        "BuildCommand": "pip install -r requirements.txt",
                        "Port": "8080",
                        "Runtime": "PYTHON_3",
                        "RuntimeEnvironmentVariables": [
                            {
                                "NAME": "Jane"
                            }
                        ],
                        "StartCommand": "python server.py"
                    },
                    "ConfigurationSource": "Api"
                },
                "RepositoryUrl": "https://github.com/my-account/python-hello",
                "SourceCodeVersion": {
```

```
                    "Type": "BRANCH",
                    "Value": "main"
                }
            }
        },
        "Status": "OPERATION_IN_PROGRESS",
        "InstanceConfiguration": {
            "CPU": "1 vCPU",
            "Memory": "3 GB"
        }
    }
}
```

- Per i dettagli sull'API, consulta PauseService AWS CLICommand Reference.

`resume-service`

Il seguente esempio di codice mostra come utilizzare`resume-service`.

AWS CLI

Come riprendere un servizio

L'esempio `resume-service` seguente riprende l'esecuzione di un servizio App Runner.

```
aws apprunner resume-service \
    --cli-input-json file://input.json
```

Contenuto di `input.json`:

```
{
    "ServiceArn": "arn:aws:apprunner:us-east-1:123456789012:service/python-
app/8fe1e10304f84fd2b0df550fe98a71fa"
}
```

Output:

```
{
    "OperationId": "17fe9f55-7e91-4097-b243-fcabbb69a4cf",
    "Service": {
        "CreatedAt": "2020-11-20T19:05:25Z",
```

```
        "UpdatedAt": "2020-11-23T12:41:37Z",
        "ServiceArn": "arn:aws:apprunner:us-east-1:123456789012:service/python-
app/8fe1e10304f84fd2b0df550fe98a71fa",
        "ServiceId": "8fe1e10304f84fd2b0df550fe98a71fa",
        "ServiceName": "python-app",
        "ServiceUrl": "psbqam834h.us-east-1.awsapprunner.com",
        "SourceConfiguration": {
            "AuthenticationConfiguration": {
                "ConnectionArn": "arn:aws:apprunner:us-
east-1:123456789012:connection/my-github-connection/
e7656250f67242d7819feade6800f59e"
            },
            "AutoDeploymentsEnabled": true,
            "CodeRepository": {
                "CodeConfiguration": {
                    "CodeConfigurationValues": {
                        "BuildCommand": "pip install -r requirements.txt",
                        "Port": "8080",
                        "Runtime": "PYTHON_3",
                        "RuntimeEnvironmentVariables": [
                            {
                                "NAME": "Jane"
                            }
                        ],
                        "StartCommand": "python server.py"
                    },
                    "ConfigurationSource": "Api"
                },
                "RepositoryUrl": "https://github.com/my-account/python-hello",
                "SourceCodeVersion": {
                    "Type": "BRANCH",
                    "Value": "main"
                }
            }
        },
        "Status": "OPERATION_IN_PROGRESS",
        "InstanceConfiguration": {
            "CPU": "1 vCPU",
            "Memory": "3 GB"
        }
    }
}
```

- Per i dettagli sull'API, consulta ResumeService AWS CLICommand Reference.

`start-deployment`

Il seguente esempio di codice mostra come utilizzare`start-deployment`.

AWS CLI

Come avviare un'implementazione manuale

L'esempio `start-deployment` seguente esegue un'implementazione manuale in un servizio App Runner.

```
aws apprunner start-deployment \
    --cli-input-json file://input.json
```

Contenuto di `input.json`:

```
{
    "ServiceArn": "arn:aws:apprunner:us-east-1:123456789012:service/python-
app/8fe1e10304f84fd2b0df550fe98a71fa"
}
```

Output:

```
{
    "OperationId": "853a7d5b-fc9f-4730-831b-fd8037ab832a"
}
```

- Per i dettagli sull'API, consulta StartDeployment AWS CLICommand Reference.

`tag-resource`

Il seguente esempio di codice mostra come utilizzare`tag-resource`.

AWS CLI

Come aggiungere tag a un servizio App Runner

L'esempio `tag-resource` seguente aggiunge due tag a un servizio App Runner.

```
aws apprunner tag-resource \
    --cli-input-json file://input.json
```

Contenuto di `input.json`:

```
{
    "ResourceArn": "arn:aws:apprunner:us-east-1:123456789012:service/python-
app/8fe1e10304f84fd2b0df550fe98a71fa",
    "Tags": [
        {
            "Key": "Department",
            "Value": "Retail"
        },
        {
            "Key": "CustomerId",
            "Value": "56439872357912"
        }
    ]
}
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta TagResource AWS CLICommand Reference.

`untag-resource`

Il seguente esempio di codice mostra come utilizzare`untag-resource`.

AWS CLI

Come rimuovere tag da un servizio App Runner

L'esempio `untag-resource` seguente rimuove due tag da un servizio App Runner.

```
aws apprunner untag-resource \
    --cli-input-json file://input.json
```

Contenuto di `input.json`:

```
{
    "ResourceArn": "arn:aws:apprunner:us-east-1:123456789012:service/python-
app/8fe1e10304f84fd2b0df550fe98a71fa",
    "TagKeys": [
        "Department",
        "CustomerId"
```

```
    ]
}
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta UntagResource AWS CLICommand Reference.

`update-service`

Il seguente esempio di codice mostra come utilizzare`update-service`.

AWS CLI

Come aggiornare la dimensione della memoria

L'esempio `update-service` seguente aggiorna la dimensione della memoria delle istanze (unità di dimensionamento) di un servizio App Runner a 2048 MiB.

Quando la chiamata ha esito positivo, App Runner avvia un processo di aggiornamento asincrono. La struttura `Service` restituita dalla chiamata riflette il nuovo valore di memoria applicato da questa chiamata.

```
aws apprunner update-service \
    --cli-input-json file://input.json
```

Contenuto di `input.json`:

```
{
    "ServiceArn": "arn:aws:apprunner:us-east-1:123456789012:service/python-
app/8fe1e10304f84fd2b0df550fe98a71fa",
    "InstanceConfiguration": {
        "Memory": "4 GB"
    }
}
```

Output:

```
{
    "OperationId": "17fe9f55-7e91-4097-b243-fcabbb69a4cf",
    "Service": {
        "CreatedAt": "2020-11-20T19:05:25Z",
```

```
        "UpdatedAt": "2020-11-23T12:41:37Z",
        "ServiceArn": "arn:aws:apprunner:us-east-1:123456789012:service/python-
app/8fe1e10304f84fd2b0df550fe98a71fa",
        "ServiceId": "8fe1e10304f84fd2b0df550fe98a71fa",
        "ServiceName": "python-app",
        "ServiceUrl": "psbqam834h.us-east-1.awsapprunner.com",
        "SourceConfiguration": {
            "AuthenticationConfiguration": {
                "ConnectionArn": "arn:aws:apprunner:us-
east-1:123456789012:connection/my-github-connection/
e7656250f67242d7819feade6800f59e"
            },
            "AutoDeploymentsEnabled": true,
            "CodeRepository": {
                "CodeConfiguration": {
                    "CodeConfigurationValues": {
                        "BuildCommand": "pip install -r requirements.txt",
                        "Port": "8080",
                        "Runtime": "PYTHON_3",
                        "RuntimeEnvironmentVariables": [
                            {
                                "NAME": "Jane"
                            }
                        ],
                        "StartCommand": "python server.py"
                    },
                    "ConfigurationSource": "Api"
                },
                "RepositoryUrl": "https://github.com/my-account/python-hello",
                "SourceCodeVersion": {
                    "Type": "BRANCH",
                    "Value": "main"
                }
            }
        },
        "Status": "OPERATION_IN_PROGRESS",
        "InstanceConfiguration": {
            "CPU": "1 vCPU",
            "Memory": "4 GB"
        }
    }
}
```

- Per i dettagli sull'API, consulta UpdateService AWS CLICommand Reference.

AWS AppConfig esempi utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with AWS AppConfig.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`create-application`

Il seguente esempio di codice mostra come utilizzare`create-application`.

AWS CLI

Come creare un'applicazione

L'`create-application`esempio seguente crea un'applicazione in AWS AppConfig.

```
aws appconfig create-application \
    --name "example-application" \
    --description "An application used for creating an example."
```

Output:

```
{
    "Description": "An application used for creating an example.",
    "Id": "339ohji",
    "Name": "example-application"
}
```

Per ulteriori informazioni, vedere Fase 1: Creazione di un' AWS AppConfig applicazione nella Guida AWS AppConfig per l'utente.

- Per i dettagli sull'API, consulta CreateApplication AWS CLICommand Reference.

`create-configuration-profile`

Il seguente esempio di codice mostra come utilizzare`create-configuration-profile`.

AWS CLI

Come creare un profilo di configurazione

L'esempio `create-configuration-profile` seguente crea un profilo di configurazione utilizzando una configurazione archiviata in Parameter Store, una funzionalità di Systems Manager.

```
aws appconfig create-configuration-profile \
    --application-id "339ohji" \
    --name "Example-Configuration-Profile" \
    --location-uri "ssm-parameter://Example-Parameter" \
    --retrieval-role-arn "arn:aws:iam::111122223333:role/Example-App-Config-Role"
```

Output:

```
{
    "ApplicationId": "339ohji",
    "Description": null,
    "Id": "ur8hx2f",
    "LocationUri": "ssm-parameter://Example-Parameter",
    "Name": "Example-Configuration-Profile",
    "RetrievalRoleArn": "arn:aws:iam::111122223333:role/Example-App-Config-Role",
    "Type": null,
    "Validators": null
}
```

Per ulteriori informazioni, vedere Fase 3: Creazione di una configurazione e di un profilo di configurazione nella Guida per l'AWS AppConfig utente.

- Per i dettagli sull'API, consulta CreateConfigurationProfile AWS CLICommand Reference.

`create-environment`

Il seguente esempio di codice mostra come utilizzare`create-environment`.

AWS CLI

Come creare un ambiente

L'`create-environment`esempio seguente crea un AWS AppConfig ambiente denominato Example-Environment utilizzando l'applicazione creata utilizzando create-application.

```
aws appconfig create-environment \
    --application-id "339ohji" \
    --name "Example-Environment"
```

Output:

```
{
    "ApplicationId": "339ohji",
    "Description": null,
    "Id": "54j1r29",
    "Monitors": null,
    "Name": "Example-Environment",
    "State": "ReadyForDeployment"
}
```

Per ulteriori informazioni, consulta Fase 2: Creazione di un ambiente nella Guida per l'utente.AWS AppConfig

- Per i dettagli sull'API, consulta CreateEnvironment AWS CLICommand Reference.

`create-extension-association`

Il seguente esempio di codice mostra come utilizzare`create-extension-association`.

AWS CLI

Come creare un'associazione di estensioni

L'`create-extension-association`esempio seguente crea una nuova associazione di estensioni in AWS AppConfig.

```
aws appconfig create-extension-association \
    --region us-west-2 \
    --extension-identifier S3-backup-extension \
```

```
    --resource-identifier "arn:aws:appconfig:us-west-2:123456789012:application/
Finance" \
    --parameters S3bucket=FinanceConfigurationBackup
```

Output:

```
{
  "Id": "a1b2c3d4",
  "ExtensionArn": "arn:aws:appconfig:us-west-2:123456789012:extension/S3-backup-
extension/1",
  "ResourceArn": "arn:aws:appconfig:us-west-2:123456789012:application/Finance",
  "Parameters": {
    "S3bucket": "FinanceConfigurationBackup"
  },
  "ExtensionVersionNumber": 1
}
```

Per ulteriori informazioni, consultate Lavorare con AWS AppConfig le estensioni nella Guida AWS AppConfig per l'utente.

- Per i dettagli sull'API, consulta CreateExtensionAssociation AWS CLICommand Reference.

`create-extension`

Il seguente esempio di codice mostra come utilizzare`create-extension`.

AWS CLI

Come creare un'estensione

L'`create-extension`esempio seguente crea una nuova estensione in AWS AppConfig.

```
aws appconfig create-extension \
    --region us-west-2 \
    --name S3-backup-extension \
    --
actions PRE_CREATE_HOSTED_CONFIGURATION_VERSION=[{Name=S3backup,Uri=arn:aws:lambda:us-
west-2:123456789012:function:s3backupfunction,RoleArn=arn:aws:iam::123456789012:role/
appconfigextensionrole}] \
    --parameters S3bucket={Required=true}
```

Output:

```
{
    "Id": "1A2B3C4D",
    "Name": "S3-backup-extension",
    "VersionNumber": 1,
    "Arn": "arn:aws:appconfig:us-west-2:123456789012:extension/1A2B3C4D/1",
    "Actions": {
        "PRE_CREATE_HOSTED_CONFIGURATION_VERSION": [
            {
                "Name": "S3backup",
                "Uri": "arn:aws:lambda:us-
west-2:123456789012:function:s3backupfunction",
                "RoleArn": "arn:aws:iam::123456789012:role/appconfigextensionrole"
            }
        ]
    },
    "Parameters": {
        "S3bucket": {
            "Required": true
        }
    }
}
```

Per ulteriori informazioni, vedete Lavorare con AWS AppConfig le estensioni nella Guida AWS AppConfig per l'utente.

- Per i dettagli sull'API, consulta CreateExtension AWS CLICommand Reference.

`create-hosted-configuration-version`

Il seguente esempio di codice mostra come utilizzare`create-hosted-configuration-version`.

AWS CLI

Come creare una versione della configurazione ospitata

L'`create-hosted-configuration-version`esempio seguente crea una nuova configurazione nell'archivio di configurazione AWS AppConfig ospitato. Il contenuto della configurazione deve essere prima convertito in Base64.

```
aws appconfig create-hosted-configuration-version \
    --application-id "339ohji" \
    --configuration-profile-id "ur8hx2f" \
```

```
    --
content eyAiTmFtZSI6ICJFeGFtcGxlQXBwbGljYXRpb24iLCAiSWQiOiBFeGFtcGxlSUQsICJSYW5rIjogNyB9
 \
    --content-type "application/json" \
    configuration_version_output_file
```

Contenuto di `configuration_version_output_file`:

```
{ "Name": "ExampleApplication", "Id": ExampleID, "Rank": 7 }
```

Output:

```
{
    "ApplicationId": "339ohji",
    "ConfigurationProfileId": "ur8hx2f",
    "VersionNumber": "1",
    "ContentType": "application/json"
}
```

Per ulteriori informazioni, consulta Informazioni sull'archivio di configurazione AWS AppConfig ospitato nella Guida per l'utente di AWS Appconfig.

- Per i dettagli sull'API, consulta AWS CLI Command CreateHostedConfigurationVersionReference.

`delete-application`

Il seguente esempio di codice mostra come utilizzare`delete-application`.

AWS CLI

Come eliminare un'applicazione

L'esempio `delete-application` seguente elimina l'applicazione specificata.

```
aws appconfig delete-application \
--application-id 339ohji
```

Questo comando non produce alcun output.

Per ulteriori informazioni, vedere Fase 1: Creazione di un' AWS AppConfig applicazione nella Guida AWS AppConfig per l'utente.

- Per i dettagli sull'API, consulta DeleteApplication AWS CLICommand Reference.

`delete-configuration-profile`

Il seguente esempio di codice mostra come utilizzare`delete-configuration-profile`.

AWS CLI

Come eliminare un profilo di configurazione

L'esempio `delete-configuration-profile` seguente elimina il profilo di configurazione specificato.

```
aws appconfig delete-configuration-profile \
    --application-id 339ohji \
    --configuration-profile-id ur8hx2f
```

Questo comando non produce alcun output.

Per ulteriori informazioni, vedere Fase 3: Creazione di una configurazione e di un profilo di configurazione nella Guida per l'AWS AppConfig utente.

- Per i dettagli sull'API, consulta DeleteConfigurationProfile AWS CLICommand Reference.

`delete-deployment-strategy`

Il seguente esempio di codice mostra come utilizzare`delete-deployment-strategy`.

AWS CLI

Come eliminare una strategia di implementazione

L'esempio `delete-deployment-strategy` seguente elimina la strategia di implementazione specificata.

```
aws appconfig delete-deployment-strategy \
    --deployment-strategy-id 1225qzk
```

Questo comando non produce alcun output.

Per ulteriori informazioni, vedere Fase 4: Creazione di una strategia di distribuzione nella Guida AWS AppConfig per l'utente.

- Per i dettagli sull'API, consulta DeleteDeploymentStrategy AWS CLICommand Reference.

`delete-environment`

Il seguente esempio di codice mostra come utilizzare`delete-environment`.

AWS CLI

Come eliminare un ambiente

L'esempio `delete-environment` seguente elimina l'ambiente dell'applicazione specificato.

```
aws appconfig delete-environment \
    --application-id 339ohji \
    --environment-id 54j1r29
```

Questo comando non produce alcun output.

Per ulteriori informazioni, vedere Fase 2: Creazione di un ambiente nella Guida AWS AppConfig per l'utente.

- Per i dettagli sull'API, consulta DeleteEnvironment AWS CLICommand Reference.

`delete-extension-association`

Il seguente esempio di codice mostra come utilizzare`delete-extension-association`.

AWS CLI

Come eliminare un'associazione di estensioni

L'`delete-extension-association`esempio seguente elimina un'associazione di estensioni da AWS AppConfig.

```
aws appconfig delete-extension-association \
    --region us-west-2 \
    --extension-association-id a1b2c3d4
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consultate Lavorare con AWS AppConfig le estensioni nella Guida per l'AWS AppConfig utente.

- Per i dettagli sull'API, consulta DeleteExtensionAssociation AWS CLICommand Reference.

`delete-extension`

Il seguente esempio di codice mostra come utilizzare`delete-extension`.

AWS CLI

Come eliminare un'estensione

L'`delete-extension`esempio seguente elimina un'estensione da AWS AppConfig.

```
aws appconfig delete-extension \
    --region us-west-2 \
    --extension-identifier S3-backup-extension
```

Questo comando non produce alcun output.

Per ulteriori informazioni, vedete Lavorare con AWS AppConfig le estensioni nella Guida per l'AWS AppConfig utente.

- Per i dettagli sull'API, consulta DeleteExtension AWS CLICommand Reference.

`delete-hosted-configuration-version`

Il seguente esempio di codice mostra come utilizzare`delete-hosted-configuration-version`.

AWS CLI

Come eliminare una versione della configurazione ospitata

L'`delete-hosted-configuration-version`esempio seguente elimina una versione di configurazione ospitata nell'archivio di configurazione AWS AppConfig ospitato.

```
aws appconfig delete-hosted-configuration-version \
    --application-id 339ohji \
    --configuration-profile-id ur8hx2f \
    --version-number 1
```

Output: questo comando non produce alcun output.

Per ulteriori informazioni, consulta Fase 3: Creazione di una configurazione e di un profilo di configurazione nella Guida per l'AWS AppConfig utente.

- Per i dettagli sull'API, consulta DeleteHostedConfigurationVersion AWS CLICommand Reference.

`get-application`

Il seguente esempio di codice mostra come utilizzare`get-application`.

AWS CLI

Come elencare i dettagli di un'applicazione

L'esempio `get-application` seguente elenca i dettagli dell'applicazione specificata.

```
aws appconfig get-application \
    --application-id 339ohji
```

Output:

```
{
    "Description": "An application used for creating an example.",
    "Id": "339ohji",
    "Name": "example-application"
}
```

Per ulteriori informazioni, consulta How AWS AppConfig works in the AWS AppConfig User Guide.

- Per i dettagli sull'API, consulta GetApplication AWS CLICommand Reference.

`get-configuration-profile`

Il seguente esempio di codice mostra come utilizzare`get-configuration-profile`.

AWS CLI

Come recuperare i dettagli del profilo di configurazione

L'esempio `get-configuration-profile` seguente restituisce i dettagli del profilo di configurazione specificato.

```
aws appconfig get-configuration-profile \
    --application-id 339ohji \
    --configuration-profile-id ur8hx2f
```

Output:

```
{
    "ApplicationId": "339ohji",
    "Id": "ur8hx2f",
    "Name": "Example-Configuration-Profile",
    "LocationUri": "ssm-parameter://Example-Parameter",
    "RetrievalRoleArn": "arn:aws:iam::111122223333:role/Example-App-Config-Role"
}
```

Per ulteriori informazioni, vedere Fase 3: Creazione di una configurazione e di un profilo di configurazione nella Guida per l'AWS AppConfig utente.

- Per i dettagli sull'API, consulta GetConfigurationProfile AWS CLICommand Reference.

`get-configuration`

Il seguente esempio di codice mostra come utilizzare`get-configuration`.

AWS CLI

Come recuperare i dettagli della configurazione

L'esempio `get-configuration` seguente restituisce i dettagli della configurazione dell'applicazione di esempio. Nelle chiamate successive a get-configuration, utilizza il parametro `client-configuration-version` per aggiornare la configurazione dell'applicazione solo se è cambiata la versione. Se la configurazione viene aggiornata solo quando è cambiata la versione, si evitano costi in eccesso derivanti dalle chiamate a get-configuration.

```
aws appconfig get-configuration \
    --application "example-application" \
    --environment "Example-Environment" \
    --configuration "Example-Configuration-Profile" \
    --client-id "test-id" \
    configuration-output-file
```

Contenuto di `configuration-output-file`:

```
{ "Name": "ExampleApplication", "Id": ExampleID, "Rank": 7 }
```

Output:

```
{
    "ConfigurationVersion": "1",
    "ContentType": "application/json"
}
```

Per ulteriori informazioni, vedere Fase 6: Ricezione della configurazione nella Guida per l'AWS AppConfig utente.

- Per i dettagli sull'API, consulta GetConfiguration AWS CLICommand Reference.

`get-deployment-strategy`

Il seguente esempio di codice mostra come utilizzare`get-deployment-strategy`.

AWS CLI

Come recuperare i dettagli di una strategia di implementazione

L'esempio `get-deployment-strategy` seguente elenca i dettagli della strategia di implementazione specificata.

```
aws appconfig get-deployment-strategy \
    --deployment-strategy-id 1225qzk
```

Output:

```
{
    "Id": "1225qzk",
    "Name": "Example-Deployment",
    "DeploymentDurationInMinutes": 15,
    "GrowthType": "LINEAR",
    "GrowthFactor": 25.0,
    "FinalBakeTimeInMinutes": 0,
    "ReplicateTo": "SSM_DOCUMENT"
}
```

Per ulteriori informazioni, vedere Fase 4: Creazione di una strategia di distribuzione nella Guida AWS AppConfig per l'utente.

- Per i dettagli sull'API, consulta GetDeploymentStrategy AWS CLICommand Reference.

`get-deployment`

Il seguente esempio di codice mostra come utilizzare`get-deployment`.

AWS CLI

Come recuperare i dettagli dell'implementazione

L'esempio `get-deployment` seguente elenca i dettagli dell'implementazione nell'applicazione nell'ambiente e nell'implementazione specificati.

```
aws appconfig get-deployment \
    --application-id 339ohji \
    --environment-id 54j1r29 \
    --deployment-number 1
```

Output:

```
{
    "ApplicationId": "339ohji",
    "EnvironmentId": "54j1r29",
    "DeploymentStrategyId": "1225qzk",
    "ConfigurationProfileId": "ur8hx2f",
    "DeploymentNumber": 1,
    "ConfigurationName": "Example-Configuration-Profile",
    "ConfigurationLocationUri": "ssm-parameter://Example-Parameter",
    "ConfigurationVersion": "1",
    "DeploymentDurationInMinutes": 15,
    "GrowthType": "LINEAR",
    "GrowthFactor": 25.0,
    "FinalBakeTimeInMinutes": 0,
    "State": "COMPLETE",
    "EventLog": [
        {
            "EventType": "DEPLOYMENT_COMPLETED",
            "TriggeredBy": "APPCONFIG",
            "Description": "Deployment completed",
            "OccurredAt": "2021-09-17T21:59:03.888000+00:00"
```

```
        },
        {
            "EventType": "BAKE_TIME_STARTED",
            "TriggeredBy": "APPCONFIG",
            "Description": "Deployment bake time started",
            "OccurredAt": "2021-09-17T21:58:57.722000+00:00"
        },
        {
            "EventType": "PERCENTAGE_UPDATED",
            "TriggeredBy": "APPCONFIG",
            "Description": "Configuration available to 100.00% of clients",
            "OccurredAt": "2021-09-17T21:55:56.816000+00:00"
        },
        {
            "EventType": "PERCENTAGE_UPDATED",
            "TriggeredBy": "APPCONFIG",
            "Description": "Configuration available to 75.00% of clients",
            "OccurredAt": "2021-09-17T21:52:56.567000+00:00"
        },
        {
            "EventType": "PERCENTAGE_UPDATED",
            "TriggeredBy": "APPCONFIG",
            "Description": "Configuration available to 50.00% of clients",
            "OccurredAt": "2021-09-17T21:49:55.737000+00:00"
        },
        {
            "EventType": "PERCENTAGE_UPDATED",
            "TriggeredBy": "APPCONFIG",
            "Description": "Configuration available to 25.00% of clients",
            "OccurredAt": "2021-09-17T21:46:55.187000+00:00"
        },
        {
            "EventType": "DEPLOYMENT_STARTED",
            "TriggeredBy": "USER",
            "Description": "Deployment started",
            "OccurredAt": "2021-09-17T21:43:54.205000+00:00"
        }
    ],
    "PercentageComplete": 100.0,
    "StartedAt": "2021-09-17T21:43:54.205000+00:00",
    "CompletedAt": "2021-09-17T21:59:03.888000+00:00"
}
```

Per ulteriori informazioni, vedere Fase 5: Distribuzione di una configurazione nella Guida per l'AWS AppConfig utente.

- Per i dettagli sull'API, consulta GetDeployment AWS CLICommand Reference.

`get-environment`

Il seguente esempio di codice mostra come utilizzare`get-environment`.

AWS CLI

Come recuperare i dettagli dell'ambiente

L'esempio `get-environment` seguente restituisce i dettagli e lo stato dell'ambiente specificato.

```
aws appconfig get-environment \
    --application-id 339ohji \
    --environment-id 54j1r29
```

Output:

```
{
    "ApplicationId": "339ohji",
    "Id": "54j1r29",
    "Name": "Example-Environment",
    "State": "ReadyForDeployment"
}
```

Per ulteriori informazioni, vedere Fase 2: Creazione di un ambiente nella Guida AWS AppConfig per l'utente.

- Per i dettagli sull'API, consulta GetEnvironment AWS CLICommand Reference.

`get-extension-association`

Il seguente esempio di codice mostra come utilizzare`get-extension-association`.

AWS CLI

Come ottenere informazioni dettagliate su un'associazione di estensioni

L'esempio `get-extension-association` seguente visualizza informazioni su un'associazione di estensioni.

```
aws appconfig get-extension-association \
    --region us-west-2 \
    --extension-association-id a1b2c3d4
```

Output:

```
{
    "Id": "a1b2c3d4",
    "ExtensionArn": "arn:aws:appconfig:us-west-2:123456789012:extension/S3-backup-
extension/1",
    "ResourceArn": "arn:aws:appconfig:us-west-2:123456789012:application/Finance",
    "Parameters": {
        "S3bucket": "FinanceConfigurationBackup"
    },
    "ExtensionVersionNumber": 1
}
```

Per ulteriori informazioni, consulta Lavorare con AWS AppConfig le estensioni nella Guida AWS AppConfig per l'utente.

- Per i dettagli sull'API, consulta GetExtensionAssociation AWS CLICommand Reference.

`get-extension`

Il seguente esempio di codice mostra come utilizzare`get-extension`.

AWS CLI

Come ottenere informazioni dettagliate su un'estensione

L'esempio `get-extension` seguente visualizza informazioni su un'estensione.

```
aws appconfig get-extension \
    --region us-west-2 \
    --extension-identifier S3-backup-extension
```

Output:

```
{
    "Id": "1A2B3C4D",
```

```
    "Name": "S3-backup-extension",
    "VersionNumber": 1,
    "Arn": "arn:aws:appconfig:us-west-2:123456789012:extension/S3-backup-
extension/1",
    "Actions": {
        "PRE_CREATE_HOSTED_CONFIGURATION_VERSION": [
            {
                "Name": "S3backup",
                "Uri": "arn:aws:lambda:us-
west-2:123456789012:function:S3backupfunction",
                "RoleArn": "arn:aws:iam::123456789012:role/appconfigextensionrole"
            }
        ]
    },
    "Parameters": {
        "S3bucket": {
            "Required": true
        }
    }
}
```

Per ulteriori informazioni, consulta Lavorare con AWS AppConfig le estensioni nella Guida AWS AppConfig per l'utente.

- Per i dettagli sull'API, consulta GetExtension AWS CLICommand Reference.

`get-hosted-configuration-version`

Il seguente esempio di codice mostra come utilizzare`get-hosted-configuration-version`.

AWS CLI

Come recuperare informazioni dettagliate su una configurazione ospitata

L'`get-hosted-configuration-version`esempio seguente recupera i dettagli di configurazione della configurazione AWS AppConfig ospitata.

```
aws appconfig get-hosted-configuration-version \
    --application-id 339ohji \
    --configuration-profile-id ur8hx2f \
    --version-number 1 \
    hosted-configuration-version-output
```

Contenuto di `hosted-configuration-version-output`:

```
{ "Name": "ExampleApplication", "Id": ExampleID, "Rank": 7 }
```

Output:

```
{
    "ApplicationId": "339ohji",
    "ConfigurationProfileId": "ur8hx2f",
    "VersionNumber": "1",
    "ContentType": "application/json"
}
```

Per ulteriori informazioni, vedere Informazioni sull'archivio di configurazione AWS AppConfig ospitato nella Guida per l'AWS AppConfig utente.

- Per i dettagli sull'API, consulta GetHostedConfigurationVersion AWS CLICommand Reference.

`list-applications`

Il seguente esempio di codice mostra come utilizzare`list-applications`.

AWS CLI

Come elencare le applicazioni disponibili

L'`list-applications`esempio seguente elenca le applicazioni disponibili nell' AWS account.

```
aws appconfig list-applications
```

Output:

```
{
    "Items": [
        {
            "Id": "339ohji",
            "Name": "test-application",
            "Description": "An application used for creating an example."
        },
        {
            "Id": "rwalwu7",
```

```
            "Name": "Test-Application"
        }
    ]
}
```

Per ulteriori informazioni, vedere Fase 1: Creazione di un' AWS AppConfig applicazione nella Guida AWS AppConfig per l'utente.

- Per i dettagli sull'API, consulta ListApplications AWS CLICommand Reference.

`list-configuration-profiles`

Il seguente esempio di codice mostra come utilizzare`list-configuration-profiles`.

AWS CLI

Come elencare i profili di configurazione disponibili

L'esempio `list-configuration-profiles` seguente elenca i profili di configurazione disponibili per l'applicazione specificata.

```
aws appconfig list-configuration-profiles \
    --application-id 339ohji
```

Output:

```
{
    "Items": [
        {
            "ApplicationId": "339ohji",
            "Id": "ur8hx2f",
            "Name": "Example-Configuration-Profile",
            "LocationUri": "ssm-parameter://Example-Parameter"
        }
    ]
}
```

Per ulteriori informazioni, vedere Fase 3: Creazione di una configurazione e di un profilo di configurazione nella Guida per l'AWS AppConfig utente.

- Per i dettagli sull'API, consulta ListConfigurationProfiles AWS CLICommand Reference.

`list-deployment-strategies`

Il seguente esempio di codice mostra come utilizzare`list-deployment-strategies`.

AWS CLI

Come elencare le strategie di implementazione disponibili

L'`list-deployment-strategies`esempio seguente elenca le strategie di distribuzione disponibili nel tuo AWS account.

```
aws appconfig list-deployment-strategies
```

Output:

```
{
    "Items": [
        {
            "Id": "1225qzk",
            "Name": "Example-Deployment",
            "DeploymentDurationInMinutes": 15,
            "GrowthType": "LINEAR",
            "GrowthFactor": 25.0,
            "FinalBakeTimeInMinutes": 0,
            "ReplicateTo": "SSM_DOCUMENT"
        },
        {
            "Id": "AppConfig.AllAtOnce",
            "Name": "AppConfig.AllAtOnce",
            "Description": "Quick",
            "DeploymentDurationInMinutes": 0,
            "GrowthType": "LINEAR",
            "GrowthFactor": 100.0,
            "FinalBakeTimeInMinutes": 10,
            "ReplicateTo": "NONE"
        },
        {
            "Id": "AppConfig.Linear50PercentEvery30Seconds",
            "Name": "AppConfig.Linear50PercentEvery30Seconds",
            "Description": "Test/Demo",
            "DeploymentDurationInMinutes": 1,
            "GrowthType": "LINEAR",
            "GrowthFactor": 50.0,
```

```
            "FinalBakeTimeInMinutes": 1,
            "ReplicateTo": "NONE"
        },
        {
            "Id": "AppConfig.Canary10Percent20Minutes",
            "Name": "AppConfig.Canary10Percent20Minutes",
            "Description": "AWS Recommended",
            "DeploymentDurationInMinutes": 20,
            "GrowthType": "EXPONENTIAL",
            "GrowthFactor": 10.0,
            "FinalBakeTimeInMinutes": 10,
            "ReplicateTo": "NONE"
        }
    ]
}
```

Per ulteriori informazioni, consulta Fase 4: Creazione di una strategia di distribuzione nella Guida AWS AppConfig per l'utente.

- Per i dettagli sull'API, consulta ListDeploymentStrategies AWS CLICommand Reference.

`list-deployments`

Il seguente esempio di codice mostra come utilizzare`list-deployments`.

AWS CLI

Come elencare le implementazioni disponibili

L'`list-deployments`esempio seguente elenca le distribuzioni disponibili nell' AWS account per l'applicazione e l'ambiente specificati.

```
aws appconfig list-deployments \
    --application-id 339ohji \
    --environment-id 54j1r29
```

Output:

```
{
    "Items": [
        {
            "DeploymentNumber": 1,
```

```
            "ConfigurationName": "Example-Configuration-Profile",
            "ConfigurationVersion": "1",
            "DeploymentDurationInMinutes": 15,
            "GrowthType": "LINEAR",
            "GrowthFactor": 25.0,
            "FinalBakeTimeInMinutes": 0,
            "State": "COMPLETE",
            "PercentageComplete": 100.0,
            "StartedAt": "2021-09-17T21:43:54.205000+00:00",
            "CompletedAt": "2021-09-17T21:59:03.888000+00:00"
        }
    ]
}
```

Per ulteriori informazioni, vedere Fase 5: Distribuzione di una configurazione nella Guida per l'AWS AppConfig utente.

- Per i dettagli sull'API, consulta ListDeployments AWS CLICommand Reference.

`list-environments`

Il seguente esempio di codice mostra come utilizzare`list-environments`.

AWS CLI

Come elencare gli ambienti disponibili

L'`list-environments`esempio seguente elenca gli ambienti disponibili nell' AWS account per l'applicazione specificata.

```
aws appconfig list-environments \
    --application-id 339ohji
```

Output:

```
{
    "Items": [
        {
            "ApplicationId": "339ohji",
            "Id": "54j1r29",
            "Name": "Example-Environment",
            "State": "ReadyForDeployment"
```

```
        }
    ]
}
```

Per ulteriori informazioni, vedere Fase 2: Creazione di un ambiente nella Guida AWS AppConfig per l'utente.

- Per i dettagli sull'API, consulta ListEnvironments AWS CLICommand Reference.

`list-extension-associations`

Il seguente esempio di codice mostra come utilizzare`list-extension-associations`.

AWS CLI

Per elencare tutte le associazioni di AWS AppConfig estensioni presenti nel tuo AWS account per una AWS regione

L'`list-extension-associations`esempio seguente elenca tutte le associazioni di AWS AppConfig estensione per l' AWS account corrente in una AWS regione specifica.

```
aws appconfig list-extension-associations \
    --region us-west-2
```

Output:

```
{
    "Items": [
        {
            "Id": "a1b2c3d4",
            "ExtensionArn": "arn:aws:appconfig:us-west-2:123456789012:extension/S3-
backup-extension/1",
            "ResourceArn": "arn:aws:appconfig:us-west-2:123456789012:application/
Finance"
        }
    ]
}
```

Per ulteriori informazioni, consulta Lavorare con AWS AppConfig le estensioni nella Guida AWS AppConfig per l'utente.

- Per i dettagli sull'API, consulta ListExtensionAssociations AWS CLICommand Reference.

`list-extensions`

Il seguente esempio di codice mostra come utilizzare`list-extensions`.

AWS CLI

Per elencare tutte AWS AppConfig le estensioni del tuo AWS account per una AWS regione

L'`list-extensions`esempio seguente elenca tutte AWS AppConfig le estensioni per l' AWS account corrente in una AWS regione specifica. Il comando restituisce estensioni personalizzate e AWS create.

```
aws appconfig list-extensions \
    --region us-west-2
```

Output:

```
{
    "Items": [
        {
            "Id": "1A2B3C4D",
            "Name": "S3-backup-extension",
            "VersionNumber": 1,
            "Arn": "arn:aws:appconfig:us-west-2:123456789012:extension/1A2B3C4D/1"
        },
        {
            "Id": "AWS.AppConfig.FeatureFlags",
            "Name": "AppConfig Feature Flags Helper",
            "VersionNumber": 1,
            "Arn": "arn:aws:appconfig:us-west-2::extension/
AWS.AppConfig.FeatureFlags/1",
            "Description": "Validates AppConfig feature flag data automatically
 against a JSON schema that includes structure and constraints. Also transforms
 feature flag data prior to sending to the client. This extension is automatically
 associated to configuration profiles with type \"AWS.AppConfig.FeatureFlags\"."
        },
        {
            "Id": "AWS.AppConfig.JiraIntegration",
            "Name": "AppConfig integration with Atlassian Jira",
            "VersionNumber": 1,
            "Arn": "arn:aws:appconfig:us-west-2::extension/
AWS.AppConfig.JiraIntegration/1",
```

```
            "Description": "Exports feature flag data from AWS AppConfig into
 Jira. The lifecycle of each feature flag in AppConfig is tracked in Jira as an
 individual issue. Customers can see in Jira when flags are updated, turned on or
 off. Works in conjunction with the AppConfig app in the Atlassian Marketplace and
 is automatically associated to configuration profiles configured within that app."
        },
        {
            "Id": "AWS.AppConfig.DeploymentNotificationsToEventBridge",
            "Name": "AppConfig deployment events to Amazon EventBridge",
            "VersionNumber": 1,
            "Arn": "arn:aws:appconfig:us-west-2::extension/
AWS.AppConfig.DeploymentNotificationsToEventBridge/1",
            "Description": "Sends events to Amazon EventBridge when a deployment
 of configuration data in AppConfig is started, completed, or rolled back. Can
 be associated to the following resources in AppConfig: Application, Environment,
 Configuration Profile."
        },
        {
            "Id": "AWS.AppConfig.DeploymentNotificationsToSqs",
            "Name": "AppConfig deployment events to Amazon SQS",
            "VersionNumber": 1,
            "Arn": "arn:aws:appconfig:us-west-2::extension/
AWS.AppConfig.DeploymentNotificationsToSqs/1",
            "Description": "Sends messages to the configured Amazon SQS queue when
 a deployment of configuration data in AppConfig is started, completed, or rolled
 back. Can be associated to the following resources in AppConfig: Application,
 Environment, Configuration Profile."
        },
        {
            "Id": "AWS.AppConfig.DeploymentNotificationsToSns",
            "Name": "AppConfig deployment events to Amazon SNS",
            "VersionNumber": 1,
            "Description": "Sends events to the configured Amazon SNS topic when
 a deployment of configuration data in AppConfig is started, completed, or rolled
 back. Can be associated to the following resources in AppConfig: Application,
 Environment, Configuration Profile."
        }
    ]
}
```

Per ulteriori informazioni, consulta Lavorare con AWS AppConfig le estensioni nella Guida per l'AWS AppConfig utente.

- Per i dettagli sull'API, consulta ListExtensions AWS CLICommand Reference.

`list-hosted-configuration-versions`

Il seguente esempio di codice mostra come utilizzare`list-hosted-configuration-versions`.

AWS CLI

Come elencare le versioni delle configurazioni ospitate disponibili

L'`list-hosted-configuration-versions`esempio seguente elenca le versioni delle configurazioni ospitate nell'archivio di configurazione AWS AppConfig ospitato per l'applicazione e il profilo di configurazione specificati.

```
aws appconfig list-hosted-configuration-versions \
    --application-id 339ohji \
    --configuration-profile-id ur8hx2f
```

Output:

```
{
    "Items": [
        {
            "ApplicationId": "339ohji",
            "ConfigurationProfileId": "ur8hx2f",
            "VersionNumber": 1,
            "ContentType": "application/json"
        }
    ]
}
```

Per ulteriori informazioni, vedere Informazioni sull'archivio di configurazione AWS AppConfig ospitato nella Guida per l'AWS AppConfig utente.

- Per i dettagli sull'API, consulta ListHostedConfigurationVersions AWS CLICommand Reference.

`list-tags-for-resource`

Il seguente esempio di codice mostra come utilizzare`list-tags-for-resource`.

AWS CLI

Come elencare i tag di un'applicazione

L'esempio `list-tags-for-resource` seguente elenca i tag dell'applicazione specificata.

```
aws appconfig list-tags-for-resource \
    --resource-arn arn:aws:appconfig:us-east-1:682428703967:application/339ohji
```

Output:

```
{
    "Tags": {
        "group1": "1"
    }
}
```

Per ulteriori informazioni, vedere Fase 1: Creazione di un' AWS AppConfig applicazione nella Guida AWS AppConfig per l'utente.

- Per i dettagli sull'API, consulta ListTagsForResource AWS CLICommand Reference.

`start-deployment`

Il seguente esempio di codice mostra come utilizzare`start-deployment`.

AWS CLI

Come avviare l'implementazione di una configurazione

L'esempio `start-deployment` seguente avvia un'implementazione nell'applicazione utilizzando l'ambiente, la strategia di implementazione e il profilo di configurazione specificati.

```
aws appconfig start-deployment \
    --application-id 339ohji \
    --environment-id 54j1r29 \
    --deployment-strategy-id 1225qzk \
    --configuration-profile-id ur8hx2f \
    --configuration-version 1
```

Output:

```
{
    "ApplicationId": "339ohji",
```

```
    "EnvironmentId": "54j1r29",
    "DeploymentStrategyId": "1225qzk",
    "ConfigurationProfileId": "ur8hx2f",
    "DeploymentNumber": 1,
    "ConfigurationName": "Example-Configuration-Profile",
    "ConfigurationLocationUri": "ssm-parameter://Example-Parameter",
    "ConfigurationVersion": "1",
    "DeploymentDurationInMinutes": 15,
    "GrowthType": "LINEAR",
    "GrowthFactor": 25.0,
    "FinalBakeTimeInMinutes": 0,
    "State": "DEPLOYING",
    "EventLog": [
        {
            "EventType": "DEPLOYMENT_STARTED",
            "TriggeredBy": "USER",
            "Description": "Deployment started",
            "OccurredAt": "2021-09-17T21:43:54.205000+00:00"
        }
    ],
    "PercentageComplete": 0.0,
    "StartedAt": "2021-09-17T21:43:54.205000+00:00"
}
```

Per ulteriori informazioni, vedere Fase 5: Distribuzione di una configurazione nella Guida per l'AWS AppConfig utente.

- Per i dettagli sull'API, consulta StartDeployment AWS CLICommand Reference.

`stop-deployment`

Il seguente esempio di codice mostra come utilizzare`stop-deployment`.

AWS CLI

Come arrestare l'implementazione di una configurazione

L'esempio `stop-deployment` seguente arresta l'implementazione della configurazione di un'applicazione nell'ambiente specificato.

```
aws appconfig stop-deployment \
    --application-id 339ohji \
    --environment-id 54j1r29 \
```

```
    --deployment-number 2
```

Output:

```
{
    "DeploymentNumber": 0,
    "DeploymentDurationInMinutes": 0,
    "GrowthFactor": 0.0,
    "FinalBakeTimeInMinutes": 0,
    "PercentageComplete": 0.0
}
```

Per ulteriori informazioni, vedere Fase 5: Distribuzione di una configurazione nella Guida per l'AWS AppConfig utente.

- Per i dettagli sull'API, consulta StopDeployment AWS CLICommand Reference.

`tag-resource`

Il seguente esempio di codice mostra come utilizzare`tag-resource`.

AWS CLI

Come taggare un'applicazione

L'esempio `tag-resource` seguente tagga una risorsa dell'applicazione.

```
aws appconfig tag-resource \
    --resource-arn arn:aws:appconfig:us-east-1:682428703967:application/339ohji \
    --tags '{"group1" : "1"}'
```

Questo comando non produce alcun output.

Per ulteriori informazioni, vedere Fase 1: Creazione di un' AWS AppConfig applicazione nella Guida AWS AppConfig per l'utente.

- Per i dettagli sull'API, consulta TagResource AWS CLICommand Reference.

`untag-resource`

Il seguente esempio di codice mostra come utilizzare`untag-resource`.

AWS CLI

Come rimuovere un tag da un'applicazione

L'esempio `untag-resource` seguente rimuove il tag group1 dall'applicazione specificata.

```
aws appconfig untag-resource \
    --resource-arn arn:aws:appconfig:us-east-1:111122223333:application/339ohji \
    --tag-keys '["group1"]'
```

Questo comando non produce alcun output.

Per ulteriori informazioni, vedere Fase 1: Creazione di un' AWS AppConfig applicazione nella Guida AWS AppConfig per l'utente.

- Per i dettagli sull'API, consulta UntagResource AWS CLICommand Reference.

`update-application`

Il seguente esempio di codice mostra come utilizzare`update-application`.

AWS CLI

Come aggiornare un'applicazione

L'esempio `update-application` seguente aggiorna il nome dell'applicazione specificata.

```
aws appconfig update-application \
    --application-id 339ohji \
    --name "Example-Application"
```

Output:

```
{
    "Id": "339ohji",
    "Name": "Example-Application",
    "Description": "An application used for creating an example."
}
```

Per ulteriori informazioni, vedere Fase 1: Creazione di un' AWS AppConfig applicazione nella Guida AWS AppConfig per l'utente.

- Per i dettagli sull'API, consulta UpdateApplication AWS CLICommand Reference.

`update-configuration-profile`

Il seguente esempio di codice mostra come utilizzare`update-configuration-profile`.

AWS CLI

Come aggiornare un profilo di configurazione

L'esempio `update-configuration-profile` seguente aggiorna la descrizione del profilo di configurazione specificato.

```
aws appconfig update-configuration-profile \
    --application-id 339ohji \
    --configuration-profile-id ur8hx2f \
    --description "Configuration profile used for examples."
```

Output:

```
{
    "ApplicationId": "339ohji",
    "Id": "ur8hx2f",
    "Name": "Example-Configuration-Profile",
    "Description": "Configuration profile used for examples.",
    "LocationUri": "ssm-parameter://Example-Parameter",
    "RetrievalRoleArn": "arn:aws:iam::111122223333:role/Example-App-Config-Role"
}
```

Per ulteriori informazioni, vedere Fase 3: Creazione di una configurazione e di un profilo di configurazione nella Guida per l'AWS AppConfig utente.

- Per i dettagli sull'API, consulta UpdateConfigurationProfile AWS CLICommand Reference.

`update-deployment-strategy`

Il seguente esempio di codice mostra come utilizzare`update-deployment-strategy`.

AWS CLI

Come aggiornare una strategia di implementazione

L'esempio `update-deployment-strategy` seguente aggiorna il tempo di incorporamento finale a 20 minuti nella strategia di implementazione specificata.

```
aws appconfig update-deployment-strategy \
    --deployment-strategy-id 1225qzk \
    --final-bake-time-in-minutes 20
```

Output:

```
{
    "Id": "1225qzk",
    "Name": "Example-Deployment",
    "DeploymentDurationInMinutes": 15,
    "GrowthType": "LINEAR",
    "GrowthFactor": 25.0,
    "FinalBakeTimeInMinutes": 20,
    "ReplicateTo": "SSM_DOCUMENT"
}
```

Per ulteriori informazioni, vedere Fase 4: Creazione di una strategia di distribuzione nella Guida AWS AppConfig per l'utente.

- Per i dettagli sull'API, consulta UpdateDeploymentStrategy AWS CLICommand Reference.

`update-environment`

Il seguente esempio di codice mostra come utilizzare`update-environment`.

AWS CLI

Come aggiornare un ambiente

L'esempio `update-environment` seguente aggiorna la descrizione di un ambiente.

```
aws appconfig update-environment \
    --application-id 339ohji \
    --environment-id 54j1r29 \
    --description "An environment for examples."
```

Output:

```
{
    "ApplicationId": "339ohji",
    "Id": "54j1r29",
```

```
    "Name": "Example-Environment",
    "Description": "An environment for examples.",
    "State": "RolledBack"
}
```

Per ulteriori informazioni, vedere Fase 2: Creazione di un ambiente nella Guida AWS AppConfig per l'utente.

- Per i dettagli sull'API, consulta UpdateEnvironment AWS CLICommand Reference.

`update-extension-association`

Il seguente esempio di codice mostra come utilizzare`update-extension-association`.

AWS CLI

Per aggiornare un'associazione di AWS AppConfig estensioni

L'`update-extension-association`esempio seguente aggiunge un nuovo valore di parametro a un'associazione di estensione in AWS AppConfig.

```
aws appconfig update-extension-association \
    --region us-west-2 \
    --extension-association-id a1b2c3d4 \
    --parameters S3bucket=FinanceMobileApp
```

Output:

```
{
    "Id": "a1b2c3d4",
    "ExtensionArn": "arn:aws:appconfig:us-west-2:123456789012:extension/S3-backup-
extension/1",
    "ResourceArn": "arn:aws:appconfig:us-west-2:123456789012:application/Finance",
    "Parameters": {
        "S3bucket": "FinanceMobileApp"
    },
    "ExtensionVersionNumber": 1
}
```

Per ulteriori informazioni, vedete Lavorare con AWS AppConfig le estensioni nella Guida AWS AppConfig per l'utente.

- Per i dettagli sull'API, consulta UpdateExtensionAssociation AWS CLICommand Reference.

`update-extension`

Il seguente esempio di codice mostra come utilizzareu`pdate-extension`.

AWS CLI

Per aggiornare un' AWS AppConfig estensione

L'`update-extension`esempio seguente aggiunge un parametro aggiuntivo Key a un'estensione in AWS AppConfig.

```
aws appconfig update-extension \
    --region us-west-2 \
    --extension-identifier S3-backup-extension \
    --parameters S3bucket={Required=true},CampaignID={Required=false}
```

Output:

```
{
    "Id": "1A2B3C4D",
    "Name": "S3-backup-extension",
    "VersionNumber": 1,
    "Arn": "arn:aws:appconfig:us-west-2:123456789012:extension/1A2B3C4D/1",
    "Actions": {
        "PRE_CREATE_HOSTED_CONFIGURATION_VERSION": [
            {
                "Name": "S3backup",
                "Uri": "arn:aws:lambda:us-
west-2:123456789012:function:S3backupfunction",
                "RoleArn": "arn:aws:iam::123456789012:role/appconfigextensionrole"
            }
        ]
    },
    "Parameters": {
        "CampaignID": {
            "Required": false
        },
        "S3bucket": {
            "Required": true
        }
```

```
    }
}
```

Per ulteriori informazioni, consultate Lavorare con AWS AppConfig le estensioni nella Guida AWS AppConfig per l'utente.

- Per i dettagli sull'API, consulta UpdateExtension AWS CLICommand Reference.

`validate-configuration`

Il seguente esempio di codice mostra come utilizzare`validate-configuration`.

AWS CLI

Come convalidare una configurazione

L'esempio `validate-configuration` seguente utilizza i validatori in un profilo di configurazione per convalidare una configurazione.

```
aws appconfig validate-configuration \
    --application-id abc1234 \
    --configuration-profile-id ur8hx2f \
    --configuration-version 1
```

Il comando non produce alcun output.

Per ulteriori informazioni, vedere Fase 3: Creazione di una configurazione e di un profilo di configurazione nella Guida per l'AWS AppConfig utente.

- Per i dettagli sull'API, consulta ValidateConfiguration AWS CLICommand Reference.

Esempi di Application Auto Scaling utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with Application Auto Scaling.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`delete-scaling-policy`

Il seguente esempio di codice mostra come utilizzare`delete-scaling-policy`.

AWS CLI

Come eliminare una policy di dimensionamento

Questo esempio elimina una policy di dimensionamento per l'app web del servizio Amazon ECS in esecuzione nel cluster predefinito.

Comando:

```
aws application-autoscaling delete-scaling-policy --policy-name web-app-cpu-lt-25 --
scalable-dimension ecs:service:DesiredCount --resource-id service/default/web-app --
service-namespace ecs
```

- Per i dettagli sull'API, consulta DeleteScalingPolicy AWS CLICommand Reference.

`delete-scheduled-action`

Il seguente esempio di codice mostra come utilizzare`delete-scheduled-action`.

AWS CLI

Come eliminare un'operazione pianificata

L'`delete-scheduled-action`esempio seguente elimina l'azione pianificata specificata dalla flotta Amazon AppStream 2.0 specificata:

```
aws application-autoscaling delete-scheduled-action \
    --service-namespace appstream \
    --scalable-dimension appstream:fleet:DesiredCapacity \
    --resource-id fleet/sample-fleet \
    --scheduled-action-name my-recurring-action
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Dimensionamento pianificato nella Guida per l'utente di Application Auto Scaling.

- Per i dettagli sull'API, consulta AWS CLI Command DeleteScheduledActionReference.

`deregister-scalable-target`

Il seguente esempio di codice mostra come utilizzare`deregister-scalable-target`.

AWS CLI

Come annullare la registrazione di una destinazione dimensionabile

Questo esempio annulla la registrazione di una destinazione dimensionabile di un servizio Amazon ECS denominato web-app in esecuzione nel cluster predefinito.

Comando:

```
aws application-autoscaling deregister-scalable-target --service-namespace ecs --
scalable-dimension ecs:service:DesiredCount --resource-id service/default/web-app
```

Questo esempio annulla la registrazione di una destinazione dimensionabile di una risorsa personalizzata. Il custom-resource-id file.txt contiene una stringa che identifica l'ID risorsa, che, per una risorsa personalizzata, è il percorso della risorsa personalizzata attraverso l'endpoint Amazon API Gateway.

Comando:

```
aws application-autoscaling deregister-scalable-target --service-namespace custom-
resource --scalable-dimension custom-resource:ResourceType:Property --resource-
id file://~/custom-resource-id.txt
```

Contenuto del file.txt: custom-resource-id

```
https://example.execute-api.us-west-2.amazonaws.com/prod/
scalableTargetDimensions/1-23456789
```

- Per i dettagli sull'API, consulta AWS CLI Command DeregisterScalableTargetReference.

`describe-scalable-targets`

Il seguente esempio di codice mostra come utilizzare`describe-scalable-targets`.

AWS CLI

Come descrivere le destinazioni dimensionabili

L'esempio `describe-scalable-targets` seguente descrive le destinazioni dimensionabili del namespace del servizio `ecs`.

```
aws application-autoscaling describe-scalable-targets \
    --service-namespace ecs
```

Output:

```
{
    "ScalableTargets": [
        {
            "ServiceNamespace": "ecs",
            "ScalableDimension": "ecs:service:DesiredCount",
            "ResourceId": "service/default/web-app",
            "MinCapacity": 1,
            "MaxCapacity": 10,
            "RoleARN": "arn:aws:iam::123456789012:role/
aws-service-role/ecs.application-autoscaling.amazonaws.com/
AWSServiceRoleForApplicationAutoScaling_ECSService",
            "CreationTime": 1462558906.199,
            "SuspendedState": {
                "DynamicScalingOutSuspended": false,
                "ScheduledScalingSuspended": false,
                "DynamicScalingInSuspended": false
            },
            "ScalableTargetARN": "arn:aws:application-autoscaling:us-
west-2:123456789012:scalable-target/1234abcd56ab78cd901ef1234567890ab123"
        }
    ]
}
```

Per ulteriori informazioni, consulta Servizi AWS che è possibile utilizzare con Application Auto Scaling nella Guida per l'utente di Application Auto Scaling.

- Per i dettagli sull'API, consulta DescribeScalableTargets AWS CLICommand Reference.

`describe-scaling-activities`

Il seguente esempio di codice mostra come utilizzare`describe-scaling-activities`.

AWS CLI

Esempio 1: come descrivere le attività di dimensionamento per il servizio Amazon ECS specificato

L'esempio `describe-scaling-activities` seguente descrive le attività di dimensionamento per un servizio Amazon ECS denominato `web-app` in esecuzione nel cluster `default`. L'output mostra un'attività di dimensionamento avviata da una policy di dimensionamento.

```
aws application-autoscaling describe-scaling-activities \
    --service-namespace ecs \
    --resource-id service/default/web-app
```

Output:

```
{
    "ScalingActivities": [
        {
            "ScalableDimension": "ecs:service:DesiredCount",
            "Description": "Setting desired count to 1.",
            "ResourceId": "service/default/web-app",
            "ActivityId": "e6c5f7d1-dbbb-4a3f-89b2-51f33e766399",
            "StartTime": 1462575838.171,
            "ServiceNamespace": "ecs",
            "EndTime": 1462575872.111,
            "Cause": "monitor alarm web-app-cpu-lt-25 in state ALARM triggered
 policy web-app-cpu-lt-25",
            "StatusMessage": "Successfully set desired count to 1. Change
 successfully fulfilled by ecs.",
            "StatusCode": "Successful"
        }
    ]
}
```

Per ulteriori informazioni, consulta Attività di dimensionamento per Application Auto Scaling nella Guida per l'utente di Application Auto Scaling.

Esempio 2: come descrivere le attività di dimensionamento per la tabella DynamoDB specificata

L'esempio `describe-scaling-activities` seguente descrive le attività di dimensionamento per una tabella DynamoDB denominata `TestTable`. L'output mostra le attività di dimensionamento avviate da due diverse azioni pianificate.

```
aws application-autoscaling describe-scaling-activities \
    --service-namespace dynamodb \
    --resource-id table/TestTable
```

Output:

```
{
    "ScalingActivities": [
        {
            "ScalableDimension": "dynamodb:table:WriteCapacityUnits",
            "Description": "Setting write capacity units to 10.",
            "ResourceId": "table/my-table",
            "ActivityId": "4d1308c0-bbcf-4514-a673-b0220ae38547",
            "StartTime": 1561574415.086,
            "ServiceNamespace": "dynamodb",
            "EndTime": 1561574449.51,
            "Cause": "maximum capacity was set to 10",
            "StatusMessage": "Successfully set write capacity units to 10. Change
 successfully fulfilled by dynamodb.",
            "StatusCode": "Successful"
        },
        {
            "ScalableDimension": "dynamodb:table:WriteCapacityUnits",
            "Description": "Setting min capacity to 5 and max capacity to 10",
            "ResourceId": "table/my-table",
            "ActivityId": "f2b7847b-721d-4e01-8ef0-0c8d3bacc1c7",
            "StartTime": 1561574414.644,
            "ServiceNamespace": "dynamodb",
            "Cause": "scheduled action name my-second-scheduled-action was
 triggered",
            "StatusMessage": "Successfully set min capacity to 5 and max capacity to
 10",
            "StatusCode": "Successful"
        },
        {
            "ScalableDimension": "dynamodb:table:WriteCapacityUnits",
            "Description": "Setting write capacity units to 15.",
            "ResourceId": "table/my-table",
            "ActivityId": "d8ea4de6-9eaa-499f-b466-2cc5e681ba8b",
```

```
            "StartTime": 1561574108.904,
            "ServiceNamespace": "dynamodb",
            "EndTime": 1561574140.255,
            "Cause": "minimum capacity was set to 15",
            "StatusMessage": "Successfully set write capacity units to 15. Change
 successfully fulfilled by dynamodb.",
            "StatusCode": "Successful"
        },
        {
            "ScalableDimension": "dynamodb:table:WriteCapacityUnits",
            "Description": "Setting min capacity to 15 and max capacity to 20",
            "ResourceId": "table/my-table",
            "ActivityId": "3250fd06-6940-4e8e-bb1f-d494db7554d2",
            "StartTime": 1561574108.512,
            "ServiceNamespace": "dynamodb",
            "Cause": "scheduled action name my-first-scheduled-action was
 triggered",
            "StatusMessage": "Successfully set min capacity to 15 and max capacity
 to 20",
            "StatusCode": "Successful"
        }
    ]
}
```

Per ulteriori informazioni, consulta Attività di dimensionamento per Application Auto Scaling nella Guida per l'utente di Application Auto Scaling.

- Per i dettagli sull'API, consulta DescribeScalingActivities AWS CLICommand Reference.

`describe-scaling-policies`

Il seguente esempio di codice mostra come utilizzare`describe-scaling-policies`.

AWS CLI

Come descrivere le policy di dimensionamento

Questo comando di esempio descrive le policy di dimensionamento per il namespace del servizio ecs.

Comando:

```
aws application-autoscaling describe-scaling-policies --service-namespace ecs
```

Output:


```
{
    "ScalingPolicies": [
        {
            "PolicyName": "web-app-cpu-gt-75",
            "ScalableDimension": "ecs:service:DesiredCount",
            "ResourceId": "service/default/web-app",
            "CreationTime": 1462561899.23,
            "StepScalingPolicyConfiguration": {
                "Cooldown": 60,
                "StepAdjustments": [
                    {
                        "ScalingAdjustment": 200,
                        "MetricIntervalLowerBound": 0.0
                    }
                ],
                "AdjustmentType": "PercentChangeInCapacity"
            },
            "PolicyARN": "arn:aws:autoscaling:us-
west-2:012345678910:scalingPolicy:6d8972f3-efc8-437c-92d1-6270f29a66e7:resource/ecs/
service/default/web-app:policyName/web-app-cpu-gt-75",
            "PolicyType": "StepScaling",
            "Alarms": [
                {
                    "AlarmName": "web-app-cpu-gt-75",
                    "AlarmARN": "arn:aws:cloudwatch:us-
west-2:012345678910:alarm:web-app-cpu-gt-75"
                }
            ],
            "ServiceNamespace": "ecs"
        },
        {
            "PolicyName": "web-app-cpu-lt-25",
            "ScalableDimension": "ecs:service:DesiredCount",
            "ResourceId": "service/default/web-app",
            "CreationTime": 1462562575.099,
            "StepScalingPolicyConfiguration": {
                "Cooldown": 1,
                "StepAdjustments": [
                    {
                        "ScalingAdjustment": -50,
                        "MetricIntervalUpperBound": 0.0
                    }
```

```
                ],
                "AdjustmentType": "PercentChangeInCapacity"
            },
            "PolicyARN": "arn:aws:autoscaling:us-
west-2:012345678910:scalingPolicy:6d8972f3-efc8-437c-92d1-6270f29a66e7:resource/ecs/
service/default/web-app:policyName/web-app-cpu-lt-25",
            "PolicyType": "StepScaling",
            "Alarms": [
                {
                    "AlarmName": "web-app-cpu-lt-25",
                    "AlarmARN": "arn:aws:cloudwatch:us-
west-2:012345678910:alarm:web-app-cpu-lt-25"
                }
            ],
            "ServiceNamespace": "ecs"
        }
    ]
}
```

- Per i dettagli sull'API, consulta DescribeScalingPolicies AWS CLICommand Reference.

`describe-scheduled-actions`

Il seguente esempio di codice mostra come utilizzare`describe-scheduled-actions`.

AWS CLI

Come descrivere le azioni pianificate

L'esempio `describe-scheduled-actions` seguente visualizza i dettagli delle azioni pianificate per il namespace del servizio specificato:

```
aws application-autoscaling describe-scheduled-actions \
    --service-namespace dynamodb
```

Output:

```
{
    "ScheduledActions": [
        {
            "ScalableDimension": "dynamodb:table:WriteCapacityUnits",
```

```
            "Schedule": "at(2019-05-20T18:35:00)",
            "ResourceId": "table/my-table",
            "CreationTime": 1561571888.361,
            "ScheduledActionARN": "arn:aws:autoscaling:us-
west-2:123456789012:scheduledAction:2d36aa3b-cdf9-4565-b290-81db519b227d:resource/
dynamodb/table/my-table:scheduledActionName/my-first-scheduled-action",
            "ScalableTargetAction": {
                "MinCapacity": 15,
                "MaxCapacity": 20
            },
            "ScheduledActionName": "my-first-scheduled-action",
            "ServiceNamespace": "dynamodb"
        },
        {
            "ScalableDimension": "dynamodb:table:WriteCapacityUnits",
            "Schedule": "at(2019-05-20T18:40:00)",
            "ResourceId": "table/my-table",
            "CreationTime": 1561571946.021,
            "ScheduledActionARN": "arn:aws:autoscaling:us-
west-2:123456789012:scheduledAction:2d36aa3b-cdf9-4565-b290-81db519b227d:resource/
dynamodb/table/my-table:scheduledActionName/my-second-scheduled-action",
            "ScalableTargetAction": {
                "MinCapacity": 5,
                "MaxCapacity": 10
            },
            "ScheduledActionName": "my-second-scheduled-action",
            "ServiceNamespace": "dynamodb"
        }
    ]
}
```

Per ulteriori informazioni, consulta Dimensionamento pianificato nella Guida per l'utente di Application Auto Scaling.

- Per i dettagli sull'API, consulta DescribeScheduledActions AWS CLICommand Reference.

`list-tags-for-resource`

Il seguente esempio di codice mostra come utilizzare`list-tags-for-resource`.

AWS CLI

Come elencare i tag per una destinazione dimensionabile

L'esempio `list-tags-for-resource` seguente elenca i nomi e i valori delle chiavi dei tag associati alla destinazione dimensionabile specificata dal relativo ARN.

```
aws application-autoscaling list-tags-for-resource \
    --resource-arn arn:aws:application-autoscaling:us-west-2:123456789012:scalable-target/1234abcd56ab78cd901ef1234567890ab123
```

Output:

```
{
    "Tags": {
        "environment": "production"
    }
}
```

Per ulteriori informazioni, consulta Service Quotas di Application Auto Scaling nella Guida per l'utente di Application Auto Scaling.

- Per i dettagli sull'API, consulta ListTagsForResource AWS CLICommand Reference.

`put-scaling-policy`

Il seguente esempio di codice mostra come utilizzare`put-scaling-policy`.

AWS CLI

Esempio 1: come applicare una policy di dimensionamento di monitoraggio obiettivi con uno specifico parametro predefinito

L'esempio `put-scaling-policy` seguente applica una policy di dimensionamento con monitoraggio delle destinazioni con una specifica di metrica predefinita a un servizio Amazon ECS denominato web-app nel cluster predefinito. La policy mantiene l'utilizzo medio della CPU del servizio al 75%, con tempi di raffreddamento delle operazioni di dimensionamento orizzontale di 60 secondi. L'output contiene i nomi ARNs e i nomi dei due CloudWatch allarmi creati per tuo conto.

```
aws application-autoscaling put-scaling-policy --service-namespace ecs \
--scalable-dimension ecs:service:DesiredCount \
--resource-id service/default/web-app \
--policy-name cpu75-target-tracking-scaling-policy --policy-
type TargetTrackingScaling \
```

```
--target-tracking-scaling-policy-configuration file://config.json
```

Questo esempio presuppone che nella directory corrente sia presente un file config.json con il seguente contenuto:

```
{
     "TargetValue": 75.0,
     "PredefinedMetricSpecification": {
         "PredefinedMetricType": "ECSServiceAverageCPUUtilization"
     },
     "ScaleOutCooldown": 60,
    "ScaleInCooldown": 60
}
```

Output:

```
{
    "PolicyARN": "arn:aws:autoscaling:us-west-2:012345678910:scalingPolicy:6d8972f3-
efc8-437c-92d1-6270f29a66e7:resource/ecs/service/default/web-app:policyName/cpu75-
target-tracking-scaling-policy",
    "Alarms": [
        {
            "AlarmARN": "arn:aws:cloudwatch:us-
west-2:012345678910:alarm:TargetTracking-service/default/web-app-AlarmHigh-d4f0770c-
b46e-434a-a60f-3b36d653feca",
            "AlarmName": "TargetTracking-service/default/web-app-AlarmHigh-d4f0770c-
b46e-434a-a60f-3b36d653feca"
        },
        {
            "AlarmARN": "arn:aws:cloudwatch:us-
west-2:012345678910:alarm:TargetTracking-service/default/web-app-AlarmLow-1b437334-
d19b-4a63-a812-6c67aaf2910d",
            "AlarmName": "TargetTracking-service/default/web-app-AlarmLow-1b437334-
d19b-4a63-a812-6c67aaf2910d"
        }
    ]
}
```

Esempio 2: come applicare una policy di dimensionamento di monitoraggio obiettivi con uno specifico parametro personalizzato

L'esempio `put-scaling-policy` seguente applica una policy di dimensionamento con monitoraggio delle destinazioni con una specifica di metrica personalizzata a un servizio Amazon ECS denominato web-app nel cluster predefinito. La policy mantiene l'utilizzo medio del servizio al 75%, con tempi di raffreddamento delle operazioni di dimensionamento orizzontale di 60 secondi. L'output contiene i nomi ARNs e i nomi dei due CloudWatch allarmi creati per tuo conto.

```
aws application-autoscaling put-scaling-policy --service-namespace ecs \
--scalable-dimension ecs:service:DesiredCount \
--resource-id service/default/web-app \
--policy-name cms75-target-tracking-scaling-policy
--policy-type TargetTrackingScaling \
--target-tracking-scaling-policy-configuration file://config.json
```

Questo esempio presuppone che nella directory corrente sia presente un file config.json con il seguente contenuto:

```
{
    "TargetValue":75.0,
    "CustomizedMetricSpecification":{
        "MetricName":"MyUtilizationMetric",
        "Namespace":"MyNamespace",
        "Dimensions": [
            {
                "Name":"MyOptionalMetricDimensionName",
                "Value":"MyOptionalMetricDimensionValue"
            }
        ],
        "Statistic":"Average",
        "Unit":"Percent"
    },
    "ScaleOutCooldown": 60,
    "ScaleInCooldown": 60
}
```

Output:

```
{
    "PolicyARN": "arn:aws:autoscaling:us-west-2:012345678910:scalingPolicy:
 8784a896-b2ba-47a1-b08c-27301cc499a1:resource/ecs/service/default/web-
app:policyName/cms75-target-tracking-scaling-policy",
    "Alarms": [
```

```
        {
            "AlarmARN": "arn:aws:cloudwatch:us-
west-2:012345678910:alarm:TargetTracking-service/default/web-app-
AlarmHigh-9bc77b56-0571-4276-ba0f-d4178882e0a0",
            "AlarmName": "TargetTracking-service/default/web-app-
AlarmHigh-9bc77b56-0571-4276-ba0f-d4178882e0a0"
        },
        {
            "AlarmARN": "arn:aws:cloudwatch:us-
west-2:012345678910:alarm:TargetTracking-service/default/web-app-
AlarmLow-9b6ad934-6d37-438e-9e05-02836ddcbdc4",
            "AlarmName": "TargetTracking-service/default/web-app-
AlarmLow-9b6ad934-6d37-438e-9e05-02836ddcbdc4"
        }
    ]
}
```

Esempio 3: come applicare una policy di dimensionamento di monitoraggio obiettivi per il solo incremento

L'esempio `put-scaling-policy` seguente applica una policy di dimensionamento con monitoraggio delle destinazioni a un servizio denominato `web-app` nel cluster predefinito. La policy viene utilizzata per aumentare orizzontalmente il servizio ECS quando la metrica `RequestCountPerTarget` dell'Application Load Balancer supera la soglia. L'output contiene l'ARN e il nome dell' CloudWatch allarme creato per tuo conto.

```
aws application-autoscaling put-scaling-policy \
    --service-namespace ecs \
    --scalable-dimension ecs:service:DesiredCount \
    --resource-id service/default/web-app \
    --policy-name alb-scale-out-target-tracking-scaling-policy \
    --policy-type TargetTrackingScaling \
    --target-tracking-scaling-policy-configuration file://config.json
```

Contenuto di `config.json`:

```
{
     "TargetValue": 1000.0,
     "PredefinedMetricSpecification": {
         "PredefinedMetricType": "ALBRequestCountPerTarget",
         "ResourceLabel": "app/EC2Co-EcsEl-1TKLTMITMM0EO/f37c06a68c1748aa/
targetgroup/EC2Co-Defau-LDNM7Q3ZH1ZN/6d4ea56ca2d6a18d"
```

```
     },
     "ScaleOutCooldown": 60,
    "ScaleInCooldown": 60,
    "DisableScaleIn": true
}
```

Output:

```
{
    "PolicyARN": "arn:aws:autoscaling:us-west-2:123456789012:scalingPolicy:6d8972f3-
efc8-437c-92d1-6270f29a66e7:resource/ecs/service/default/web-app:policyName/alb-
scale-out-target-tracking-scaling-policy",
    "Alarms": [
        {
            "AlarmName": "TargetTracking-service/default/web-app-AlarmHigh-d4f0770c-
b46e-434a-a60f-3b36d653feca",
            "AlarmARN": "arn:aws:cloudwatch:us-
west-2:123456789012:alarm:TargetTracking-service/default/web-app-AlarmHigh-d4f0770c-
b46e-434a-a60f-3b36d653feca"
        }
    ]
}
```

Per ulteriori informazioni, consulta Policy di dimensionamento del monitoraggio delle destinazioni per Application Auto Scaling nella Guida per l'utente di AWS Application Auto Scaling.

- Per i dettagli sull'API, consulta PutScalingPolicy AWS CLICommand Reference.

`put-scheduled-action`

Il seguente esempio di codice mostra come utilizzare`put-scheduled-action`.

AWS CLI

Come aggiungere un'azione pianificata a una tabella DynamoDB

Questo esempio aggiunge un'azione pianificata a una tabella DynamoDB TestTable chiamata per scalare out in base a una pianificazione ricorrente. Nella pianificazione specificata (ogni giorno alle 12:15 UTC), se la capacità corrente è inferiore al valore specificato per MinCapacity, Application Auto Scaling viene ridimensionato fino al valore specificato da. MinCapacity

Comando:

```
aws application-autoscaling put-scheduled-action --service-namespace dynamodb
 --scheduled-action-name my-recurring-action --schedule "cron(15 12 * * ? *)" --
resource-id table/TestTable --scalable-dimension dynamodb:table:WriteCapacityUnits
 --scalable-target-action MinCapacity=6
```

Per ulteriori informazioni, consulta Dimensionamento pianificato nella Guida per l'utente di Application Auto Scaling.

- Per i dettagli sull'API, consulta PutScheduledActionCommand Reference.AWS CLI

`register-scalable-target`

Il seguente esempio di codice mostra come utilizzare`register-scalable-target`.

AWS CLI

Esempio 1: come registrare un servizio ECS come destinazione dimensionabile

L'esempio `register-scalable-target` seguente registra un servizio Amazon ECS con Application Auto Scaling. Alla destinazione dimensionabile aggiunge anche un tag con il nome della chiave `environment` e il valore `production`.

```
aws application-autoscaling register-scalable-target \
    --service-namespace ecs \
    --scalable-dimension ecs:service:DesiredCount \
    --resource-id service/default/web-app \
    --min-capacity 1 --max-capacity 10 \
    --tags environment=production
```

Output:

```
{
    "ScalableTargetARN": "arn:aws:application-autoscaling:us-
west-2:123456789012:scalable-target/1234abcd56ab78cd901ef1234567890ab123"
}
```

Per esempi di altri AWS servizi e risorse personalizzate, consultate gli argomenti relativi AWS ai servizi che è possibile utilizzare con Application Auto Scaling nella Application Auto Scaling User Guide.

Esempio 2: come sospendere le attività di dimensionamento per una destinazione dimensionabile

L'esempio `register-scalable-target` seguente sospende le attività di dimensionamento per una destinazione dimensionabile esistente.

```
aws application-autoscaling register-scalable-target \
    --service-namespace dynamodb \
    --scalable-dimension dynamodb:table:ReadCapacityUnits \
    --resource-id table/my-table \
    --suspended-
state DynamicScalingInSuspended=true,DynamicScalingOutSuspended=true,ScheduledScalingSuspend
```

Output:

```
{
    "ScalableTargetARN": "arn:aws:application-autoscaling:us-
west-2:123456789012:scalable-target/1234abcd56ab78cd901ef1234567890ab123"
}
```

Per ulteriori informazioni, consulta Sospensione e ripresa del dimensionamento per Application Auto Scaling nella Guida per l'utente di Application Auto Scaling.

Esempio 3: come sospendere le attività di dimensionamento per una destinazione dimensionabile

L'esempio `register-scalable-target` seguente sospende le attività di dimensionamento per una destinazione dimensionabile esistente.

```
aws application-autoscaling register-scalable-target \
    --service-namespace dynamodb \
    --scalable-dimension dynamodb:table:ReadCapacityUnits \
    --resource-id table/my-table \
    --suspended-
state DynamicScalingInSuspended=false,DynamicScalingOutSuspended=false,ScheduledScalingSuspe
```

Output:

```
{
    "ScalableTargetARN": "arn:aws:application-autoscaling:us-
west-2:123456789012:scalable-target/1234abcd56ab78cd901ef1234567890ab123"
}
```

Per ulteriori informazioni, consulta Sospensione e ripresa del dimensionamento per Application Auto Scaling nella Guida per l'utente di Application Auto Scaling.

- Per i dettagli sull'API, consulta AWS CLI Command RegisterScalableTargetReference.

`tag-resource`

Il seguente esempio di codice mostra come utilizzare`tag-resource`.

AWS CLI

Come aggiungere un tag a una destinazione dimensionabile

L'esempio `tag-resource` seguente aggiunge un tag con il nome della chiave `environment` e il valore `production` alla destinazione dimensionabile specificata dal relativo ARN.

```
aws application-autoscaling tag-resource \
    --resource-arn arn:aws:application-autoscaling:us-west-2:123456789012:scalable-target/1234abcd56ab78cd901ef1234567890ab123 \
    --tags environment=production
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Service Quotas di Application Auto Scaling nella Guida per l'utente di Application Auto Scaling.

- Per i dettagli sull'API, consulta TagResource AWS CLICommand Reference.

`untag-resource`

Il seguente esempio di codice mostra come utilizzare`untag-resource`.

AWS CLI

Come rimuovere un tag da una destinazione dimensionabile

L'esempio `untag-resource` seguente rimuove la coppia di tag con il nome di chiave `environment` dalla destinazione dimensionabile specificata dal relativo ARN.

```
aws application-autoscaling untag-resource \
    --resource-arn arn:aws:application-autoscaling:us-west-2:123456789012:scalable-target/1234abcd56ab78cd901ef1234567890ab123 \
    --tag-keys "environment"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Service Quotas di Application Auto Scaling nella Guida per l'utente di Application Auto Scaling.

- Per i dettagli sull'API, consulta UntagResource AWS CLICommand Reference.

Esempi di Application Discovery Service utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando il servizio AWS Command Line Interface with Application Discovery Service.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`describe-agents`

Il seguente esempio di codice mostra come utilizzare`describe-agents`.

AWS CLI

Descrivi gli agenti con stati collectionStatus specificati

Questo comando di esempio descrive gli agenti di raccolta con lo stato di raccolta "STARTED" o "STOPPED".

Comando:

```
aws discovery describe-agents --filters
 name="collectionStatus",values="STARTED","STOPPED",condition="EQUALS" --max-
results 3
```

Output:

```
{
       "Snapshots": [
      {
            "version": "1.0.40.0",
            "agentType": "EC2",
            "hostName": "ip-172-31-40-234",
            "collectionStatus": "STOPPED",
            "agentNetworkInfoList": [
                {
                    "macAddress": "06:b5:97:14:fc:0d",
                    "ipAddress": "172.31.40.234"
                }
            ],
            "health": "UNKNOWN",
            "agentId": "i-003305c02a776e883",
            "registeredTime": "2016-12-09T19:05:06Z",
            "lastHealthPingTime": "2016-12-09T19:05:10Z"
        },
        {
            "version": "1.0.40.0",
            "agentType": "EC2",
            "hostName": "ip-172-31-39-64",
            "collectionStatus": "STARTED",
            "agentNetworkInfoList": [
                {
                    "macAddress": "06:a1:0e:c7:b2:73",
                    "ipAddress": "172.31.39.64"
                }
            ],
            "health": "SHUTDOWN",
            "agentId": "i-003a5e5e2b36cf8bd",
            "registeredTime": "2016-11-16T16:36:25Z",
            "lastHealthPingTime": "2016-11-16T16:47:37Z"
        }
    ]
}
```

- Per i dettagli sull'API, consulta DescribeAgents AWS CLICommand Reference.

`describe-configurations`

Il seguente esempio di codice mostra come utilizzare`describe-configurations`.

AWS CLI

Descrivi le configurazioni degli asset selezionati

Questo comando di esempio descrive le configurazioni di due server specificati. L'azione rileva il tipo di asset dall'ID di configurazione. È consentito un solo tipo di asset per comando.

Comando:

```
aws discovery describe-configurations --configuration-ids "d-
server-099385097ef9fbcfb" "d-server-0c4f2dd1fee22c6c1"
```

Output:

```
{
    "configurations": [
        {
                "server.performance.maxCpuUsagePct": "0.0",
                "server.performance.maxDiskReadIOPS": "0.0",
                "server.performance.avgCpuUsagePct": "0.0",
                "server.type": "EC2",
                "server.performance.maxNetworkReadsPerSecondInKB": "0.19140625",
                "server.hostName": "ip-172-31-35-152",
                "server.configurationId": "d-server-0c4f2dd1fee22c6c1",
                "server.tags.hasMoreValues": "false",
                "server.performance.minFreeRAMInKB": "1543496.0",
                "server.osVersion": "3.14.48-33.39.amzn1.x86_64",
                "server.performance.maxDiskReadsPerSecondInKB": "0.0",
                "server.applications": "[]",
                "server.performance.numDisks": "1",
                "server.performance.numCpus": "1",
                "server.performance.numCores": "1",
                "server.performance.maxDiskWriteIOPS": "0.0",
                "server.performance.maxNetworkWritesPerSecondInKB": "0.82421875",
                "server.performance.avgDiskWritesPerSecondInKB": "0.0",
                "server.networkInterfaceInfo": "[{\"name\":\"eth0\",
\"macAddress\":\"06:A7:7D:3F:54:57\",\"ipAddress\":\"172.31.35.152\",\"netMask\":
\"255.255.240.0\"},{\"name\":\"lo\",\"macAddress\":\"00:00:00:00:00:00\",\"ipAddress
\":\"127.0.0.1\",\"netMask\":\"255.0.0.0\"},{\"name\":\"eth0\",\"macAddress\":
\"06:A7:7D:3F:54:57\",\"ipAddress\":\"fe80::4a7:7dff:fe3f:5457\"},{\"name\":\"lo\",
\"macAddress\":\"00:00:00:00:00:00\",\"ipAddress\":\"::1\"}]",
                "server.performance.avgNetworkReadsPerSecondInKB":
 "0.04915364583333333",
```

```
                "server.tags": "[]",
                "server.applications.hasMoreValues": "false",
                "server.timeOfCreation": "2016-10-28 23:44:00.0",
                "server.agentId": "i-4447bc1b",
                "server.performance.maxDiskWritesPerSecondInKB": "0.0",
                "server.performance.avgDiskReadIOPS": "0.0",
                "server.performance.avgFreeRAMInKB": "1547210.1333333333",
                "server.performance.avgDiskReadsPerSecondInKB": "0.0",
                "server.performance.avgDiskWriteIOPS": "0.0",
                "server.performance.numNetworkCards": "2",
                "server.hypervisor": "xen",
                "server.networkInterfaceInfo.hasMoreValues": "false",
                "server.performance.avgNetworkWritesPerSecondInKB": "0.1380859375",
                "server.osName": "Linux - Amazon Linux AMI release 2015.03",
                "server.performance.totalRAMInKB": "1694732.0",
                "server.cpuType": "x64"
        },
        {
                "server.performance.maxCpuUsagePct": "100.0",
                "server.performance.maxDiskReadIOPS": "0.0",
                "server.performance.avgCpuUsagePct": "14.733333333333338",
                "server.type": "EC2",
                "server.performance.maxNetworkReadsPerSecondInKB": "13.400390625",
                "server.hostName": "ip-172-31-42-208",
                "server.configurationId": "d-server-099385097ef9fbcfb",
                "server.tags.hasMoreValues": "false",
                "server.performance.minFreeRAMInKB": "1531104.0",
                "server.osVersion": "3.14.48-33.39.amzn1.x86_64",
                "server.performance.maxDiskReadsPerSecondInKB": "0.0",
                "server.applications": "[]",
                "server.performance.numDisks": "1",
                "server.performance.numCpus": "1",
                "server.performance.numCores": "1",
                "server.performance.maxDiskWriteIOPS": "1.0",
                "server.performance.maxNetworkWritesPerSecondInKB": "12.271484375",
                "server.performance.avgDiskWritesPerSecondInKB":
 "0.5333333333333334",
                "server.networkInterfaceInfo": "[{\"name\":\"eth0\",
\"macAddress\":\"06:4A:79:60:75:61\",\"ipAddress\":\"172.31.42.208\",\"netMask
\":\"255.255.240.0\"},{\"name\":\"eth0\",\"macAddress\":\"06:4A:79:60:75:61\",
\"ipAddress\":\"fe80::44a:79ff:fe60:7561\"},{\"name\":\"lo\",\"macAddress\":
\"00:00:00:00:00:00\",\"ipAddress\":\"::1\"},{\"name\":\"lo\",\"macAddress\":
\"00:00:00:00:00:00\",\"ipAddress\":\"127.0.0.1\",\"netMask\":\"255.0.0.0\"}]",
```

```
                "server.performance.avgNetworkReadsPerSecondInKB":
 "2.8720052083333334",
                "server.tags": "[]",
                "server.applications.hasMoreValues": "false",
                "server.timeOfCreation": "2016-10-28 23:44:30.0",
                "server.agentId": "i-c142b99e",
                "server.performance.maxDiskWritesPerSecondInKB": "4.0",
                "server.performance.avgDiskReadIOPS": "0.0",
                "server.performance.avgFreeRAMInKB": "1534946.4",
                "server.performance.avgDiskReadsPerSecondInKB": "0.0",
                "server.performance.avgDiskWriteIOPS": "0.13333333333333336",
                "server.performance.numNetworkCards": "2",
                "server.hypervisor": "xen",
                "server.networkInterfaceInfo.hasMoreValues": "false",
                "server.performance.avgNetworkWritesPerSecondInKB":
 "1.7977864583333332",
                "server.osName": "Linux - Amazon Linux AMI release 2015.03",
                "server.performance.totalRAMInKB": "1694732.0",
                "server.cpuType": "x64"
        }
    ]
}
```

Descrivi le configurazioni degli asset selezionati

Questo comando di esempio descrive le configurazioni di due applicazioni specificate. L'azione rileva il tipo di asset dall'ID di configurazione. È consentito un solo tipo di asset per comando.

Comando:

```
aws discovery describe-configurations --configuration-ids "d-
application-0ac39bc0e4fad0e42" "d-application-02444a45288013764q"
```

Output:

```
{
    "configurations": [
        {
                "application.serverCount": "0",
                "application.name": "Application-12345",
                "application.lastModifiedTime": "2016-12-13 23:53:27.0",
                "application.description": "",
                "application.timeOfCreation": "2016-12-13 23:53:27.0",
```

```
                "application.configurationId": "d-application-0ac39bc0e4fad0e42"
        },
        {
                "application.serverCount": "0",
                "application.name": "Application-67890",
                "application.lastModifiedTime": "2016-12-13 23:53:33.0",
                "application.description": "",
                "application.timeOfCreation": "2016-12-13 23:53:33.0",
                "application.configurationId": "d-application-02444a45288013764"
         }
    ]
}
```

- Per i dettagli sull'API, consulta DescribeConfigurations AWS CLICommand Reference.

`list-configurations`

Il seguente esempio di codice mostra come utilizzare`list-configurations`.

AWS CLI

Come elencare tutti i server rilevati che soddisfano una serie di condizioni di filtro

Questo comando di esempio elenca i server rilevati che corrispondono a uno dei due modelli di hostname e che non eseguono Ubuntu.

Comando:

```
aws discovery list-configurations --configuration-type SERVER --filters
 name="server.hostName",values="172-31-35","172-31-42",condition="CONTAINS"
 name="server.osName",values="Ubuntu",condition="NOT_CONTAINS"
```

Output:

```
{
    "configurations": [
      {
            "server.osVersion": "3.14.48-33.39.amzn1.x86_64",
            "server.type": "EC2",
            "server.hostName": "ip-172-31-42-208",
            "server.timeOfCreation": "2016-10-28 23:44:30.0",
            "server.configurationId": "d-server-099385097ef9fbcfb",
            "server.osName": "Linux - Amazon Linux AMI release 2015.03",
```

```
            "server.agentId": "i-c142b99e"
        },
        {
            "server.osVersion": "3.14.48-33.39.amzn1.x86_64",
            "server.type": "EC2",
            "server.hostName": "ip-172-31-35-152",
            "server.timeOfCreation": "2016-10-28 23:44:00.0",
            "server.configurationId": "d-server-0c4f2dd1fee22c6c1",
            "server.osName": "Linux - Amazon Linux AMI release 2015.03",
            "server.agentId": "i-4447bc1b"
        }
    ]
}
```

- Per i dettagli sull'API, consulta ListConfigurations AWS CLICommand Reference.

Esempi di segnali applicativi utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando Application Signals. AWS Command Line Interface

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`batch-get-service-level-objective-budget-report`

Il seguente esempio di codice mostra come utilizzare`batch-get-service-level-objective-budget-report`.

AWS CLI

Come recuperare uno o più report sul budget dell'obiettivo del livello di servizio (SLO).

L'esempio `batch-get-service-level-objective-budget-report` seguente recupera uno o più report sul budget dell'obiettivo del livello di servizio (SLO).

```
aws application-signals batch-get-service-level-objective-budget-report \
--timestamp 1735059869 \
--slo-ids "arn:aws:application-signals:us-east-1:123456789101:slo/
SLOName1" "arn:aws:application-signals:us-east-1:123456789101:slo/SLOName2"
```

Output:

```
{
    "Timestamp": "2024-12-24T22:34:29+05:30",
    "Reports": [{
            "Arn": "arn:aws:application-signals:us-east-1:123456789101:slo/
SLOName1",
            "Name": "Your SLO Name",
            "EvaluationType": "PeriodBased",
            "BudgetStatus": "OK",
            "Attainment": 100.0,
            "TotalBudgetSeconds": 6048,
            "BudgetSecondsRemaining": 6048,
            "Sli": {
                "SliMetric": {
                    "MetricDataQueries": [{
                        "Id": "m1",
                        "MetricStat": {
                            "Metric": {
                                "Namespace": "AWS/EC2",
                                "MetricName": "CPUUtilization",
                                "Dimensions": [{
                                    "Name": "InstanceId",
                                    "Value": "i-0e098765432522"
                                }]
                            },
                            "Period": 60,
                            "Stat": "Average"
                        },
                        "ReturnData": true
                    }]
                },
                "MetricThreshold": 200.0,
                "ComparisonOperator": "LessThanOrEqualTo"
            },
```

```
            "Goal": {
                "Interval": {
                    "RollingInterval": {
                        "DurationUnit": "DAY",
                        "Duration": 7
                    }
                },
                "AttainmentGoal": 99.0,
                "WarningThreshold": 50.0
            }
        },
        {
            "Arn": "arn:aws:application-signals:us-east-1:123456789101:slo/
SLOName2",
            "Name": "test",
            "EvaluationType": "PeriodBased",
            "BudgetStatus": "BREACHED",
            "Attainment": 97.39583275,
            "TotalBudgetSeconds": 86,
            "BudgetSecondsRemaining": -2154,
            "Sli": {
                "SliMetric": {
                    "MetricDataQueries": [{
                        "Id": "cwMetric",
                        "MetricStat": {
                            "Metric": {
                                "Namespace": "AWS/EC2",
                                "MetricName": "CPUUtilization",
                                "Dimensions": [{
                                    "Name": "InstanceId",
                                    "Value": "i-0e12345678922"
                                }]
                            },
                            "Period": 300,
                            "Stat": "Average"
                        },
                        "ReturnData": true
                    }]
                },
                "MetricThreshold": 5.0,
                "ComparisonOperator": "GreaterThan"
            },
            "Goal": {
                "Interval": {
```

```
                    "RollingInterval": {
                        "DurationUnit": "DAY",
                        "Duration": 1
                    }
                },
                "AttainmentGoal": 99.9,
                "WarningThreshold": 30.0
            }
        }
    ],
    "Errors": []
}
```

Per ulteriori informazioni, consulta Application Signals nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta BatchGetServiceLevelObjectiveBudgetReport AWS CLICommand Reference.

`create-service-level-objective`

Il seguente esempio di codice mostra come utilizzare`create-service-level-objective`.

AWS CLI

Come creare un obiettivo del livello di servizio (SLO)

L'esempio `create-service-level-objective` seguente crea un obiettivo del livello di servizio (SLO), che può aiutarti a garantire che le operazioni aziendali critiche soddisfino le aspettative dei clienti.

```
aws application-signals create-service-level-objective \
    --name "SLOName" \
    --description "Description of your SLO" \
    --sli-config file://sli-config.json
```

Contenuto di `sli-config.json`:

```
{
    "SliMetricConfig": {
        "MetricDataQueries": [
            {
                "Id": "m1",
```

```
                "MetricStat": {
                    "Metric": {
                        "Namespace": "AWS/EC2",
                        "MetricName": "CPUUtilization",
                        "Dimensions": [
                            {
                                "Name": "InstanceId",
                                "Value": "i-0e5a1234561522"
                            }
                        ]
                    },
                    "Period": 60,
                    "Stat": "Average"
                },
                "ReturnData": true
            }
        ]
    },
    "MetricThreshold": 200,
    "ComparisonOperator": "LessThanOrEqualTo"
}
```

Output:

```
{
    "Slo": {
    "Arn": "arn:aws:application-signals:us-east-1:123456789101:slo/SLOName",
    "Name": "SLOName",
    "Description": "Description of your SLO",
    "CreatedTime": "2024-12-27T08:16:09.032000+05:30",
    "LastUpdatedTime": "2024-12-27T08:16:09.032000+05:30",
    "Sli": {
        "SliMetric": {
            "MetricDataQueries": [
                {
                    "Id": "m1",
                    "MetricStat": {
                        "Metric": {
                            "Namespace": "AWS/EC2",
                            "MetricName": "CPUUtilization",
                            "Dimensions": [
                                {
                                    "Name": "InstanceId",
```

```
                                    "Value": "i-0e59876543234522"
                                }
                            ]
                        },
                        "Period": 60,
                        "Stat": "Average"
                    },
                    "ReturnData": true
                }
            ]
        },
        "MetricThreshold": 200.0,
        "ComparisonOperator": "LessThanOrEqualTo"
    },
    "EvaluationType": "PeriodBased",
    "Goal": {
        "Interval": {
            "RollingInterval": {
                "DurationUnit": "DAY",
                "Duration": 7
            }
        },
        "AttainmentGoal": 99.0,
        "WarningThreshold": 50.0
      }
    }
}
```

Per ulteriori informazioni, consulta Application Signals nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta CreateServiceLevelObjective AWS CLICommand Reference.

`delete-service-level-objective`

Il seguente esempio di codice mostra come utilizzare`delete-service-level-objective`.

AWS CLI

Come eliminare l'obiettivo del livello di servizio specificato.

L'esempio `delete-service-level-objective` seguente elimina l'obiettivo del livello di servizio specificato.

```
aws application-signals delete-service-level-objective \
    --id "arn:aws:application-signals:us-east-1:123456789101:slo/SLOName"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Application Signals nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta DeleteServiceLevelObjective AWS CLICommand Reference.

`get-service-level-objective`

Il seguente esempio di codice mostra come utilizzare`get-service-level-objective`.

AWS CLI

Come restituire informazioni su uno SLO creato nell'account

L'esempio `get-service-level-objective` seguente restituisce informazioni su un obiettivo del livello di servizio (SLO) creato nell'account.

```
aws application-signals get-service-level-objective \
    --id "arn:aws:application-signals:us-east-1:123456789101:slo/SLOName"
```

Output:

```
{
    "Slo": {
        "Arn": "arn:aws:application-signals:us-east-1:123456789101:slo/SLOName",
        "Name": "SLOName",
        "Description": "Description of your SLO",
        "CreatedTime": "2024-12-24T22:19:18.624000+05:30",
        "LastUpdatedTime": "2024-12-24T22:19:55.280000+05:30",
        "Sli": {
            "SliMetric": {
                "MetricDataQueries": [{
                    "Id": "m1",
                    "MetricStat": {
                        "Metric": {
                            "Namespace": "AWS/EC2",
                            "MetricName": "CPUUtilization",
                            "Dimensions": [{
                                "Name": "InstanceId",
                                "Value": "i-0e0987654321522"
```

```
                            }]
                        },
                        "Period": 60,
                        "Stat": "Average"
                    },
                    "ReturnData": true
                }]
            },
            "MetricThreshold": 200.0,
            "ComparisonOperator": "LessThanOrEqualTo"
        },
        "EvaluationType": "PeriodBased",
        "Goal": {
            "Interval": {
                "RollingInterval": {
                    "DurationUnit": "DAY",
                    "Duration": 7
                }
            },
            "AttainmentGoal": 99.0,
            "WarningThreshold": 50.0
        }
    }
}
```

Per ulteriori informazioni, consulta Application Signals nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta GetServiceLevelObjective AWS CLICommand Reference.

`get-service`

Il seguente esempio di codice mostra come utilizzare`get-service`.

AWS CLI

Come restituire informazioni su un servizio rilevato da Application Signals

L'esempio `get-service` seguente restituisce informazioni su un servizio rilevato da Application Signals.

```
aws application-signals get-service \
    --start-time 1732704000 \
    --end-time 1732714500 \
```

```
    --key-attributes Environment=lambda:default,Name=hello-world-python,Type=Service
```

Output:

```
{
    "Service": {
        "KeyAttributes": {
            "Environment": "lambda:default",
            "Name": "hello-world-python",
            "Type": "Service"
        },
        "AttributeMaps": [{
            "Lambda.Function.Name": "hello-world-python",
            "PlatformType": "AWS::Lambda"
        }],
        "MetricReferences": [{
            "Namespace": "ApplicationSignals",
            "MetricType": "LATENCY",
            "Dimensions": [{
                "Name": "Environment",
                "Value": "lambda:default"
            }, {
                "Name": "Service",
                "Value": "hello-world-python"
            }],
            "MetricName": "Latency"
        }, {
            "Namespace": "ApplicationSignals",
            "MetricType": "FAULT",
            "Dimensions": [{
                "Name": "Environment",
                "Value": "lambda:default"
            }, {
                "Name": "Service",
                "Value": "hello-world-python"
            }],
            "MetricName": "Fault"
        }, {
            "Namespace": "ApplicationSignals",
            "MetricType": "ERROR",
            "Dimensions": [{
                "Name": "Environment",
                "Value": "lambda:default"
```

```
            }, {
                "Name": "Service",
                "Value": "hello-world-python"
            }],
            "MetricName": "Error"
        }],
        "LogGroupReferences": [{
            "Identifier": "/aws/lambda/hello-world-python",
            "ResourceType": "AWS::Logs::LogGroup",
            "Type": "AWS::Resource"
        }]
    },
    "StartTime": "2024-11-27T10:00:00+00:00",
    "EndTime": "2024-11-27T14:00:01+00:00",
    "LogGroupReferences": [{
        "Identifier": "/aws/lambda/hello-world-python",
        "ResourceType": "AWS::Logs::LogGroup",
        "Type": "AWS::Resource"
    }]
}
```

Per ulteriori informazioni, consulta Application Signals nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta GetService AWS CLICommand Reference.

`list-service-dependencies`

Il seguente esempio di codice mostra come utilizzare`list-service-dependencies`.

AWS CLI

Come restituire un elenco delle dipendenze del servizio specificato

L'esempio `list-service-dependencies` seguente restituisce un elenco di dipendenze del servizio specificato.

```
aws application-signals list-service-dependencies \
    --start-time 1732021200 \
    --end-time 1732107600 \
    --key-attributes Environment=api-
gateway:prod, Name=PetAdoptionStatusUpdater,Type=Service
```

Output:

```
{
    "ServiceDependencies": [{
        "OperationName": "PUT /prod",
        "DependencyKeyAttributes": {
            "Environment": "lambda:default",
            "Name": "Services-name",
            "Type": "Service"
        },
        "DependencyOperationName": "Invoke",
        "MetricReferences": [{
            "Namespace": "ApplicationSignals",
            "MetricType": "LATENCY",
            "Dimensions": [{
                "Name": "Environment",
                "Value": "api-gateway:prod"
            }, {
                "Name": "Operation",
                "Value": "PUT /prod"
            }, {
                "Name": "RemoteEnvironment",
                "Value": "lambda:default"
            }, {
                "Name": "RemoteOperation",
                "Value": "Invoke"
            }, {
                "Name": "RemoteService",
                "Value": "Services-name"
            }, {
                "Name": "Service",
                "Value": "PetAdoptionStatusUpdater"
            }],
            "MetricName": "Latency"
        }, {
            "Namespace": "ApplicationSignals",
            "MetricType": "FAULT",
            "Dimensions": [{
                "Name": "Environment",
                "Value": "api-gateway:prod"
            }, {
                "Name": "Operation",
                "Value": "PUT /prod"
            }, {
                "Name": "RemoteEnvironment",
```

```
                "Value": "lambda:default"
            }, {
                "Name": "RemoteOperation",
                "Value": "Invoke"
            }, {
                "Name": "RemoteService",
                "Value": "Services-name"
            }, {
                "Name": "Service",
                "Value": "PetAdoptionStatusUpdater"
            }],
            "MetricName": "Fault"
        }, {
            "Namespace": "ApplicationSignals",
            "MetricType": "ERROR",
            "Dimensions": [{
                "Name": "Environment",
                "Value": "api-gateway:prod"
            }, {
                "Name": "Operation",
                "Value": "PUT /prod"
            }, {
                "Name": "RemoteEnvironment",
                "Value": "lambda:default"
            }, {
                "Name": "RemoteOperation",
                "Value": "Invoke"
            }, {
                "Name": "RemoteService",
                "Value": "Services-name"
            }, {
                "Name": "Service",
                "Value": "PetAdoptionStatusUpdater"
            }],
            "MetricName": "Error"
        }]
    }],
    "StartTime": "2024-11-19T13:00:00+00:00",
    "EndTime": "2024-11-20T13:00:01+00:00"
}
```

Per ulteriori informazioni, consulta Application Signals nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta ListServiceDependencies AWS CLICommand Reference.

`list-service-dependents`

Il seguente esempio di codice mostra come utilizzare`list-service-dependents`.

AWS CLI

Come restituire l'elenco dei dipendenti che hanno invocato il servizio specificato nell'intervallo di tempo fornito

L'esempio `list-service-dependents` seguente restituisce l'elenco dei servizi dipendenti che hanno invocato il servizio specificato durante l'intervallo di tempo fornito.

```
aws application-signals list-service-dependents \
    --start-time 1732021200 \
    --end-time 1732107600 \
    --key-attributes Environment=generic:default,Name=PetSite,Type=Service
```

Output:

```
{
    "ServiceDependents": [{
        "OperationName": "",
        "DependentKeyAttributes": {
            "Identifier": "pet-api-canary-hao",
            "ResourceType": "AWS::Synthetics::Canary",
            "Type": "AWS::Resource"
        },
        "DependentOperationName": "",
        "MetricReferences": []
    }, {
        "OperationName": "",
        "DependentKeyAttributes": {
            "Identifier": "PetSite",
            "ResourceType": "AWS::Synthetics::Canary",
            "Type": "AWS::Resource"
        },
        "DependentOperationName": "",
        "MetricReferences": []
    }],
    "StartTime": "2024-12-24T05:00:00+00:00",
    "EndTime": "2024-12-25T06:00:01+00:00"
}
```

Per ulteriori informazioni, consulta Application Signals nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta ListServiceDependents AWS CLICommand Reference.

`list-service-level-objectives`

Il seguente esempio di codice mostra come utilizzare`list-service-level-objectives`.

AWS CLI

Per restituire un elenco di quelli SLOs creati in questo account.

L'`list-service-level-objectives`esempio seguente restituisce un elenco di SLOs quelli creati in questo account.

```
aws application-signals list-service-level-objectives
```

Output:

```
{
    "SloSummaries": [{
        "Arn": "arn:aws:application-signals:us-east-1:123456789101:slo/test",
        "Name": "test",
        "CreatedTime": "2024-12-24T22:01:21.116000+05:30"
    }]
}
```

Per ulteriori informazioni, consulta Application Signals nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta ListServiceLevelObjectives AWS CLICommand Reference.

`list-service-operations`

Il seguente esempio di codice mostra come utilizzare`list-service-operations`.

AWS CLI

Come restituire un elenco delle operazioni di questo servizio che sono state rilevate da Application Signals

L'esempio `list-service-operations` seguente restituisce un elenco delle operazioni di questo servizio che sono state rilevate da Application Signals.

```
aws application-signals list-service-operations \
    --start-time 1735017423 \
    --end-time 1735103823 \
    --key-attributes Environment=generic:default,Name=payforadoption,Type=Service
```

Output:

```
{
    "ServiceOperations": [{
        "Name": "POST /api",
        "MetricReferences": [{
            "Namespace": "ApplicationSignals",
            "MetricType": "LATENCY",
            "Dimensions": [{
                "Name": "Environment",
                "Value": "generic:default"
            }, {
                "Name": "Operation",
                "Value": "POST /api"
            }, {
                "Name": "Service",
                "Value": "payforadoption"
            }],
            "MetricName": "Latency"
        }, {
            "Namespace": "ApplicationSignals",
            "MetricType": "FAULT",
            "Dimensions": [{
                "Name": "Environment",
                "Value": "generic:default"
            }, {
                "Name": "Operation",
                "Value": "POST /api"
            }, {
                "Name": "Service",
                "Value": "payforadoption"
            }],
            "MetricName": "Fault"
        }, {
            "Namespace": "ApplicationSignals",
            "MetricType": "ERROR",
            "Dimensions": [{
                "Name": "Environment",
```

```
                "Value": "generic:default"
            }, {
                "Name": "Operation",
                "Value": "POST /api"
            }, {
                "Name": "Service",
                "Value": "payforadoption"
            }],
            "MetricName": "Error"
        }]
    }],
    "StartTime": "2024-12-24T05:00:00+00:00",
    "EndTime": "2024-12-25T06:00:01+00:00"
}
```

Per ulteriori informazioni, consulta Application Signals nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta ListServiceOperations AWS CLICommand Reference.

`list-services`

Il seguente esempio di codice mostra come utilizzare`list-services`.

AWS CLI

Come restituire un elenco di servizi rilevati da Application Signals

L'esempio `list-services` seguente restituisce un elenco di servizi rilevati da Application Signals.

```
aws application-signals list-services \
    --start-time 1734918791 \
    --end-time 1734965591
```

Output:

```
{
    "ServiceSummaries": [{
        "KeyAttributes": {
            "Environment": "lambda:default",
            "Name": "hello-world-python",
            "Type": "Service"
        },
```

```
        "AttributeMaps": [{
            "Lambda.Function.Name": "hello-world-python",
            "PlatformType": "AWS::Lambda"
        }],
        "MetricReferences": [{
            "Namespace": "ApplicationSignals",
            "MetricType": "LATENCY",
            "Dimensions": [{
                "Name": "Environment",
                "Value": "lambda:default"
            }, {
                "Name": "Service",
                "Value": "hello-world-python"
            }],
            "MetricName": "Latency"
        }, {
            "Namespace": "ApplicationSignals",
            "MetricType": "FAULT",
            "Dimensions": [{
                "Name": "Environment",
                "Value": "lambda:default"
            }, {
                "Name": "Service",
                "Value": "hello-world-python"
            }],
            "MetricName": "Fault"
        }, {
            "Namespace": "ApplicationSignals",
            "MetricType": "ERROR",
            "Dimensions": [{
                "Name": "Environment",
                "Value": "lambda:default"
            }, {
                "Name": "Service",
                "Value": "hello-world-python"
            }],
            "MetricName": "Error"
        }]
    }],
    "StartTime": "2024-11-27T10:00:00+00:00",
    "EndTime": "2024-11-27T14:00:01+00:00"
}
```

Per ulteriori informazioni, consulta Application Signals nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta ListServices AWS CLICommand Reference.

`list-tags-for-resource`

Il seguente esempio di codice mostra come utilizzare`list-tags-for-resource`.

AWS CLI

Per visualizzare i tag associati a una CloudWatch risorsa

L'`list-tags-for-resource`esempio seguente visualizza i tag associati a una CloudWatch risorsa.

```
aws application-signals list-tags-for-resource \
    --resource-arn "arn:aws:application-signals:us-east-1:123456789101:slo/SLOName"
```

Output:

```
{
    "Tags": [{
        "Key": "test",
        "Value": "value"
    }]
}
```

Per ulteriori informazioni, consulta Application Signals nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta ListTagsForResource AWS CLICommand Reference.

`start-discovery`

Il seguente esempio di codice mostra come utilizzare`start-discovery`.

AWS CLI

Per abilitare questo account Amazon Web Services per poter utilizzare CloudWatch Application Signals

L'`start-discovery`esempio seguente consente a questo account Amazon Web Services di utilizzare CloudWatch Application Signals creando il ruolo AWS ServiceRoleForCloudWatchApplicationSignalscollegato al servizio.

```
aws application-signals start-discovery
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Application Signals nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta StartDiscovery AWS CLICommand Reference.

`tag-resource`

Il seguente esempio di codice mostra come utilizzare`tag-resource`.

AWS CLI

Assegna uno o più tag (coppie chiave-valore) alla CloudWatch risorsa specificata, ad esempio un obiettivo del livello di servizio

L'`tag-resource`esempio seguente assegna uno o più tag (coppie chiave-valore) alla CloudWatch risorsa specificata, ad esempio un obiettivo del livello di servizio.

```
aws application-signals tag-resource \
    --resource-arn "arn:aws:application-signals:us-east-1:123456789101:slo/SLOName"
 \
    --tags '{"Key":"test","Value":"value"}'
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Application Signals nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta TagResource AWS CLICommand Reference.

`untag-resource`

Il seguente esempio di codice mostra come utilizzare`untag-resource`.

AWS CLI

Come rimuovere uno o più tag dalla risorsa specificata

L'esempio `untag-resource` seguente rimuove uno o più tag dalla risorsa specificata.

```
aws application-signals untag-resource \
    --resource-arn "arn:aws:application-signals:us-east-1:123456789101:slo/SLOName"
 \
    --tag-keys "test"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Application Signals nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta UntagResource AWS CLICommand Reference.

`update-service-level-objective`

Il seguente esempio di codice mostra come utilizzare`update-service-level-objective`.

AWS CLI

Come aggiornare un obiettivo del livello di servizio (SLO) esistente

L'esempio `update-service-level-objective` seguente aggiorna un obiettivo del livello di servizio (SLO) esistente.

```
aws application-signals update-service-level-objective \
--cli-input-json file://update-slo.json
```

Contenuto di `update-slo.json`:

```
{
    "id": "arn:aws:application-signals:us-east-1:123456789101:slo/SLOName",
    "goal": {
        "Interval": {
            "RollingInterval": {
                "DurationUnit": "DAY",
                "Duration": 7
            }
        },
        "AttainmentGoal": 90.0,
        "WarningThreshold": 50.0
    }
```

```
}
```

Output:

```
{
    "Slo": {
        "Arn": "arn:aws:application-signals:us-east-1:123456789101:slo/SLOName",
        "Name": "SLOName",
        "Description": "Description of your SLO",
        "CreatedTime": "2024-12-24T22:19:18.624000+05:30",
        "LastUpdatedTime": "2024-12-27T08:51:38.278000+05:30",
        "Sli": {
            "SliMetric": {
                "MetricDataQueries": [{
                    "Id": "m1",
                    "MetricStat": {
                        "Metric": {
                            "Namespace": "AWS/EC2",
                            "MetricName": "CPUUtilization",
                            "Dimensions": [{
                                "Name": "InstanceId",
                                "Value": "i-00987654345222"
                            }]
                        },
                        "Period": 60,
                        "Stat": "Average"
                    },
                    "ReturnData": true
                }]
            },
            "MetricThreshold": 200.0,
            "ComparisonOperator": "LessThanOrEqualTo"
        },
        "EvaluationType": "PeriodBased",
        "Goal": {
            "Interval": {
                "RollingInterval": {
                    "DurationUnit": "DAY",
                    "Duration": 7
                }
            },
            "AttainmentGoal": 90.0,
            "WarningThreshold": 50.0
```

```
        }
    }
}
```

Per ulteriori informazioni, consulta Application Signals nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta UpdateServiceLevelObjective AWS CLICommand Reference.

AppRegistry esempi utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with AppRegistry.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`associate-attribute-group`

Il seguente esempio di codice mostra come utilizzare`associate-attribute-group`.

AWS CLI

Come associare un gruppo di attributi

L'`associate-attribute-group`esempio seguente associa un gruppo di attributi specifico AWS dell'account a un'applicazione specifica dell' AWS account.

```
aws servicecatalog-appregistry associate-attribute-group \
    --application "ExampleApplication" \
    --attribute-group "ExampleAttributeGroup"
```

Output:

```
{
    "applicationArn": "arn:aws:servicecatalog:us-west-2:813737243517:/
applications/0ars38r6btoohvpvd9gqrptt9l",
    "attributeGroupArn": "arn:aws:servicecatalog:us-west-2:813737243517:/attribute-
groups/01sj5xdwhbw54kejwnt09fnpcl"
}
```

Per ulteriori informazioni, vedere Associating and Dissociating Attributing Groups nella AWS Service Catalog Administrator Guide. AppRegistry

- Per i dettagli sull'API, vedere AssociateAttributeGroupin AWS CLI Command Reference.

`create-application`

Il seguente esempio di codice mostra come utilizzare`create-application`.

AWS CLI

Come creare un'applicazione

L'`create-application`esempio seguente crea una nuova applicazione nel tuo AWS account.

```
aws servicecatalog-appregistry create-application \
    --name "ExampleApplication"
```

Output:

```
{
    "application": {
        "id": "0ars38r6btoohvpvd9gqrptt9l",
        "arn": "arn:aws:servicecatalog:us-west-2:813737243517:/
applications/0ars38r6btoohvpvd9gqrptt9l",
        "name": "ExampleApplication",
        "creationTime": "2023-02-28T21:10:10.820000+00:00",
        "lastUpdateTime": "2023-02-28T21:10:10.820000+00:00",
        "tags": {}
    }
}
```

Per ulteriori informazioni, vedere Creazione di applicazioni nella AWS Service Catalog AppRegistry Administrator Guide.

- Per i dettagli sull'API, vedere CreateApplicationin AWS CLI Command Reference.

`create-attribute-group`

Il seguente esempio di codice mostra come utilizzare`create-attribute-group`.

AWS CLI

Come creare un gruppo di attributi

L'`create-attribute-group`esempio seguente crea un nuovo gruppo di attributi nel tuo AWS account.

```
aws servicecatalog-appregistry create-attribute-group \
    --name "ExampleAttributeGroup" \
    --attributes '{"SomeKey1":"SomeValue1","SomeKey2":"SomeValue2"}'
```

Output:

```
{
    "attributeGroup": {
        "id": "01sj5xdwhbw54kejwnt09fnpcl",
        "arn": "arn:aws:servicecatalog:us-west-2:813737243517:/attribute-
groups/01sj5xdwhbw54kejwnt09fnpcl",
        "name": "ExampleAttributeGroup",
        "creationTime": "2023-02-28T20:38:01.389000+00:00",
        "lastUpdateTime": "2023-02-28T20:38:01.389000+00:00",
        "tags": {}
    }
}
```

Per ulteriori informazioni, vedere Creazione di gruppi di attributi nella AWS Service Catalog AppRegistry Administrator Guide.

- Per i dettagli sull'API, vedere CreateAttributeGroupin AWS CLI Command Reference.

`delete-application`

Il seguente esempio di codice mostra come utilizzare`delete-application`.

AWS CLI

Come eliminare un'applicazione

L'`delete-application`esempio seguente elimina un'applicazione specifica dall' AWS account.

```
aws servicecatalog-appregistry delete-application \
    --application "ExampleApplication3"
```

Output:

```
{
    "application": {
        "id": "055gw7aynr1i5mbv7kjwzx5945",
        "arn": "arn:aws:servicecatalog:us-west-2:813737243517:/
applications/055gw7aynr1i5mbv7kjwzx5945",
        "name": "ExampleApplication3",
        "creationTime": "2023-02-28T22:06:28.228000+00:00",
        "lastUpdateTime": "2023-02-28T22:06:28.228000+00:00"
    }
}
```

Per ulteriori informazioni, vedere Eliminazione di applicazioni nella AWS Service Catalog AppRegistry Administrator Guide.

- Per i dettagli sull'API, vedere DeleteApplicationin AWS CLI Command Reference.

`delete-attribute-group`

Il seguente esempio di codice mostra come utilizzare`delete-attribute-group`.

AWS CLI

Esempio 8: come eliminare un gruppo di attributi

L'`delete-attribute-group`esempio seguente elimina un gruppo di attributi specifico nel tuo AWS account.

```
aws servicecatalog-appregistry delete-attribute-group \
    --attribute-group "ExampleAttributeGroup3"
```

Output:

```
{
    "attributeGroup": {
        "id": "011ge6y3emyjijt8dw8jn6r0hv",
        "arn": "arn:aws:servicecatalog:us-west-2:813737243517:/attribute-
groups/011ge6y3emyjijt8dw8jn6r0hv",
        "name": "ExampleAttributeGroup3",
        "creationTime": "2023-02-28T22:05:35.224000+00:00",
        "lastUpdateTime": "2023-02-28T22:05:35.224000+00:00"
    }
}
```

Per ulteriori informazioni, vedere Eliminazione dei gruppi di attributi nella AWS Service Catalog AppRegistry Administrator Guide.

- Per i dettagli sull'API, vedere DeleteAttributeGroupin AWS CLI Command Reference.

`get-application`

Il seguente esempio di codice mostra come utilizzare`get-application`.

AWS CLI

Come ottenere un'applicazione

L'`get-application`esempio seguente recupera le informazioni sui metadati relative a un'applicazione specifica dell'account AWS .

```
aws servicecatalog-appregistry get-application \
    --application "ExampleApplication"
```

Output:

```
{
    "id": "0ars38r6btoohvpvd9gqrptt9l",
    "arn": "arn:aws:servicecatalog:us-west-2:813737243517:/
applications/0ars38r6btoohvpvd9gqrptt9l",
    "name": "ExampleApplication",
    "creationTime": "2023-02-28T21:10:10.820000+00:00",
    "lastUpdateTime": "2023-02-28T21:10:10.820000+00:00",
```

```
    "associatedResourceCount": 0,
    "tags": {
        "aws:servicecatalog:applicationName": "ExampleApplication"
    },
    "integrations": {
        "resourceGroup": {
            "state": "CREATE_COMPLETE",
            "arn": "arn:aws:resource-groups:us-west-2:813737243517:group/
AWS_AppRegistry_Application-ExampleApplication"
        }
    }
}
```

Per ulteriori informazioni, vedere Using Application details nella AWS Service Catalog AppRegistry Administrator Guide.

- Per i dettagli sull'API, vedere GetApplicationin AWS CLI Command Reference.

`get-attribute-group`

Il seguente esempio di codice mostra come utilizzare`get-attribute-group`.

AWS CLI

Come ottenere un gruppo di attributi

L'`get-attribute-group`esempio seguente recupera un gruppo di attributi specifico nel tuo AWS account.

```
aws servicecatalog-appregistry get-attribute-group \
    --attribute-group "ExampleAttributeGroup"
```

Output:

```
{
    "id": "01sj5xdwhbw54kejwnt09fnpcl",
    "arn": "arn:aws:servicecatalog:us-west-2:813737243517:/attribute-
groups/01sj5xdwhbw54kejwnt09fnpcl",
    "name": "ExampleAttributeGroup",
    "attributes": "{\"SomeKey1\":\"SomeValue1\",\"SomeKey2\":\"SomeValue2\"}",
    "creationTime": "2023-02-28T20:38:01.389000+00:00",
```

```
    "lastUpdateTime": "2023-02-28T20:38:01.389000+00:00",
    "tags": {
        "aws:servicecatalog:attributeGroupName": "ExampleAttributeGroup"
    }
}
```

Per ulteriori informazioni, vedere Gestione dei metadati per i gruppi di attributi nella AWS Service Catalog AppRegistry Administrator Guide.

- Per i dettagli sull'API, vedere GetAttributeGroupin AWS CLI Command Reference.

`list-applications`

Il seguente esempio di codice mostra come utilizzare`list-applications`.

AWS CLI

Come elencare le applicazioni

L'`list-applications`esempio seguente recupera un elenco di tutte le applicazioni del tuo AWS account.

```
aws servicecatalog-appregistry list-applications
```

Output:

```
{
    "applications": [
        {
            "id": "03axw94pjfj3uan00tcgbrxnkw",
            "arn": "arn:aws:servicecatalog:us-west-2:813737243517:/
applications/03axw94pjfj3uan00tcgbrxnkw",
            "name": "ExampleApplication2",
            "creationTime": "2023-02-28T21:59:34.094000+00:00",
            "lastUpdateTime": "2023-02-28T21:59:34.094000+00:00"
        },
        {
            "id": "055gw7aynr1i5mbv7kjwzx5945",
            "arn": "arn:aws:servicecatalog:us-west-2:813737243517:/
applications/055gw7aynr1i5mbv7kjwzx5945",
            "name": "ExampleApplication3",
```

```
            "creationTime": "2023-02-28T22:06:28.228000+00:00",
            "lastUpdateTime": "2023-02-28T22:06:28.228000+00:00"
        },
        {
            "id": "0ars38r6btoohvpvd9gqrptt9l",
            "arn": "arn:aws:servicecatalog:us-west-2:813737243517:/
applications/0ars38r6btoohvpvd9gqrptt9l",
            "name": "ExampleApplication",
            "description": "This is an example application",
            "creationTime": "2023-02-28T21:10:10.820000+00:00",
            "lastUpdateTime": "2023-02-28T21:24:19.729000+00:00"
        }
    ]
}
```

Per ulteriori informazioni, vedere Visualizzazione dei dettagli dell'applicazione nella AWS Service Catalog AppRegistry Administrator Guide.

- Per i dettagli sull'API, vedere ListApplicationsin AWS CLI Command Reference.

`list-associated-attribute-groups`

Il seguente esempio di codice mostra come utilizzare`list-associated-attribute-groups`.

AWS CLI

Come elencare i gruppi di attributi associati

L'`list-associated-attribute-groups`esempio seguente recupera un elenco di tutti i gruppi di attributi AWS dell'account associati a un'applicazione specifica dell' AWS account.

```
aws servicecatalog-appregistry list-associated-attribute-groups \
    --application "ExampleApplication"
```

Output:

```
{
    "attributeGroups": [
        "01sj5xdwhbw54kejwnt09fnpcl"
    ]
}
```

Per ulteriori informazioni, vedere Associating and Dissociating Attribating Groups nella AWS Service Catalog Administrator Guide. AppRegistry

- Per i dettagli sull'API, vedere ListAssociatedAttributeGroupsin AWS CLI Command Reference.

`list-attribute-groups-for-application`

Il seguente esempio di codice mostra come utilizzare`list-attribute-groups-for-application`.

AWS CLI

Come elencare i gruppi di attributi per un'applicazione

L'`list-attribute-groups-for-application`esempio seguente elenca i dettagli di tutti i gruppi di attributi AWS dell'account associati a un'applicazione specifica dell' AWS account.

```
aws servicecatalog-appregistry list-attribute-groups-for-application \
    --application "ExampleApplication"
```

Output:

```
{
    "attributeGroupsDetails": [
        {
            "id": "01sj5xdwhbw54kejwnt09fnpcl",
            "arn": "arn:aws:servicecatalog:us-west-2:813737243517:/attribute-
groups/01sj5xdwhbw54kejwnt09fnpcl",
            "name": "ExampleAttributeGroup"
        }
    ]
}
```

Per ulteriori informazioni, vedere Visualizzazione dei dettagli del gruppo di attributi nella AWS Service Catalog AppRegistry Administrator Guide.

- Per i dettagli sull'API, vedere ListAttributeGroupsForApplicationin AWS CLI Command Reference.

`list-attribute-groups`

Il seguente esempio di codice mostra come utilizzare`list-attribute-groups`.

AWS CLI

Come elencare i gruppi di attributi

L'`list-attribute-groups`esempio seguente recupera un elenco di tutti i gruppi di attributi presenti nell' AWS account.

```
aws servicecatalog-appregistry list-attribute-groups
```

Output:

```
{
    "attributeGroups": [
        {
            "id": "011ge6y3emyjijt8dw8jn6r0hv",
            "arn": "arn:aws:servicecatalog:us-west-2:813737243517:/attribute-
groups/011ge6y3emyjijt8dw8jn6r0hv",
            "name": "ExampleAttributeGroup3",
            "creationTime": "2023-02-28T22:05:35.224000+00:00",
            "lastUpdateTime": "2023-02-28T22:05:35.224000+00:00"
        },
        {
            "id": "01sj5xdwhbw54kejwnt09fnpcl",
            "arn": "arn:aws:servicecatalog:us-west-2:813737243517:/attribute-
groups/01sj5xdwhbw54kejwnt09fnpcl",
            "name": "ExampleAttributeGroup",
            "description": "This is an example attribute group",
            "creationTime": "2023-02-28T20:38:01.389000+00:00",
            "lastUpdateTime": "2023-02-28T21:02:04.559000+00:00"
        },
        {
            "id": "03n1yffgq6d18vwrzxf0c70nm3",
            "arn": "arn:aws:servicecatalog:us-west-2:813737243517:/attribute-
groups/03n1yffgq6d18vwrzxf0c70nm3",
            "name": "ExampleAttributeGroup2",
            "creationTime": "2023-02-28T21:57:30.687000+00:00",
            "lastUpdateTime": "2023-02-28T21:57:30.687000+00:00"
        }
    ]
}
```

Per ulteriori informazioni, vedere Visualizzazione dei dettagli del gruppo di attributi nella AWS Service Catalog AppRegistry Administrator Guide.

- Per i dettagli sull'API, vedere ListAttributeGroupsin AWS CLI Command Reference.

`update-application`

Il seguente esempio di codice mostra come utilizzareupdate-application.

AWS CLI

Come aggiornare un'applicazione

L'update-applicationesempio seguente aggiorna un'applicazione specifica nell' AWS account per includere una descrizione.

```
aws servicecatalog-appregistry update-application \
    --application "ExampleApplication" \
    --description "This is an example application"
```

Output:

```
{
    "application": {
        "id": "0ars38r6btoohvpvd9gqrptt9l",
        "arn": "arn:aws:servicecatalog:us-west-2:813737243517:/
applications/0ars38r6btoohvpvd9gqrptt9l",
        "name": "ExampleApplication",
        "description": "This is an example application",
        "creationTime": "2023-02-28T21:10:10.820000+00:00",
        "lastUpdateTime": "2023-02-28T21:24:19.729000+00:00",
        "tags": {
            "aws:servicecatalog:applicationName": "ExampleApplication"
        }
    }
}
```

Per ulteriori informazioni, vedere Modifica delle applicazioni nella AWS Service Catalog AppRegistry Administrator Guide.

- Per i dettagli sull'API, vedere UpdateApplicationin AWS CLI Command Reference.

`update-attribute-group`

Il seguente esempio di codice mostra come utilizzareupdate-attribute-group.

AWS CLI

Come aggiornare un gruppo di attributi

L'`update-attribute-group`esempio seguente aggiorna un gruppo di attributi specifico nell' AWS account per includere una descrizione.

```
aws servicecatalog-appregistry update-attribute-group \
    --attribute-group "ExampleAttributeGroup" \
    --description "This is an example attribute group"
```

Output:

```
{
    "attributeGroup": {
        "id": "01sj5xdwhbw54kejwnt09fnpcl",
        "arn": "arn:aws:servicecatalog:us-west-2:813737243517:/attribute-
groups/01sj5xdwhbw54kejwnt09fnpcl",
        "name": "ExampleAttributeGroup",
        "description": "This is an example attribute group",
        "creationTime": "2023-02-28T20:38:01.389000+00:00",
        "lastUpdateTime": "2023-02-28T21:02:04.559000+00:00",
        "tags": {
            "aws:servicecatalog:attributeGroupName": "ExampleAttributeGroup"
        }
    }
}
```

Per ulteriori informazioni, vedere Modifica dei gruppi di attributi nella AWS Service Catalog AppRegistry Administrator Guide.

- Per i dettagli sull'API, vedere UpdateAttributeGroupin AWS CLI Command Reference.

Esempi di Athena che utilizzano AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with Athena.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`batch-get-named-query`

Il seguente esempio di codice mostra come utilizzare`batch-get-named-query`.

AWS CLI

Come restituire informazioni su più di una query

L'`batch-get-named-query`esempio seguente restituisce informazioni sulle interrogazioni denominate che hanno il valore specificato IDs.

```
aws athena batch-get-named-query \
    --named-query-ids a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 a1b2c3d4-5678-90ab-cdef-EXAMPLE22222 a1b2c3d4-5678-90ab-cdef-EXAMPLE33333
```

Output:

```
{
    "NamedQueries": [
        {
            "Name": "Flights Select Query",
            "Description": "Sample query to get the top 10 airports with the most number of departures since 2000",
            "Database": "sampledb",
            "QueryString": "SELECT origin, count(*) AS total_departures\nFROM \nflights_parquet\nWHERE year >= '2000'\nGROUP BY origin\nORDER BY total_departures DESC\nLIMIT 10;",
            "NamedQueryId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
            "WorkGroup": "primary"
        },
        {
            "Name": "Load flights table partitions",
            "Description": "Sample query to load flights table partitions using MSCK REPAIR TABLE statement",
```

```
            "Database": "sampledb",
            "QueryString": "MSCK REPAIR TABLE flights_parquet;",
            "NamedQueryId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
            "WorkGroup": "primary"
        },
        {
            "Name": "CloudFront Select Query",
            "Description": "Sample query to view requests per operating system
 during a particular time frame",
            "Database": "sampledb",
            "QueryString": "SELECT os, COUNT(*) count FROM cloudfront_logs WHERE
 date BETWEEN date '2014-07-05' AND date '2014-08-05' GROUP BY os;",
            "NamedQueryId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE33333",
            "WorkGroup": "primary"
        }
    ],
    "UnprocessedNamedQueryIds": []
}
```

Per ulteriori informazioni, consulta Esecuzione di query SQL con Amazon Athena nella Guida per l'utente di Amazon Athena.

- Per i dettagli sull'API, vedere BatchGetNamedQueryin AWS CLI Command Reference.

`batch-get-query-execution`

Il seguente esempio di codice mostra come utilizzare`batch-get-query-execution`.

AWS CLI

Come restituire informazioni su una o più esecuzioni di query

L'`batch-get-query-execution`esempio seguente restituisce le informazioni sull'esecuzione delle query per le quali è stata specificata la query IDs.

```
aws athena batch-get-query-execution \
    --query-execution-ids a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 a1b2c3d4-5678-90ab-
cdef-EXAMPLE22222
```

Output:

```
{
    "QueryExecutions": [
```

```
        {
            "QueryExecutionId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
            "Query": "create database if not exists webdata",
            "StatementType": "DDL",
            "ResultConfiguration": {
                "OutputLocation": "s3://amzn-s3-demo-bucket/a1b2c3d4-5678-90ab-cdef-
EXAMPLE11111.txt"
            },
            "QueryExecutionContext": {},
            "Status": {
                "State": "SUCCEEDED",
                "SubmissionDateTime": 1593470720.592,
                "CompletionDateTime": 1593470720.902
            },
            "Statistics": {
                "EngineExecutionTimeInMillis": 232,
                "DataScannedInBytes": 0,
                "TotalExecutionTimeInMillis": 310,
            "ResultConfiguration": {

                "QueryQueueTimeInMillis": 50,
                "ServiceProcessingTimeInMillis": 28
            },
            "WorkGroup": "AthenaAdmin"
        },
        {
            "QueryExecutionId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
            "Query": "select date, location, browser, uri, status from
 cloudfront_logs where method = 'GET' and status = 200 and location like 'SFO%'
 limit 10",
            "StatementType": "DML",
            "ResultConfiguration": {
                "OutputLocation": "s3://amzn-s3-demo-bucket/a1b2c3d4-5678-90ab-cdef-
EXAMPLE22222.csv"
            },
            "QueryExecutionContext": {
                "Database": "mydatabase",
                "Catalog": "awsdatacatalog"
            },
            "Status": {
                "State": "SUCCEEDED",
                "SubmissionDateTime": 1593469842.665,
                "CompletionDateTime": 1593469846.486
            },
```

```
            "Statistics": {
                "EngineExecutionTimeInMillis": 3600,
                "DataScannedInBytes": 203089,
                "TotalExecutionTimeInMillis": 3821,
                "QueryQueueTimeInMillis": 267,
                "QueryPlanningTimeInMillis": 1175
            },
            "WorkGroup": "AthenaAdmin"
        }
    ],
    "UnprocessedQueryExecutionIds": []
}
```

Per ulteriori informazioni, consulta Esecuzione di query SQL con Amazon Athena nella Guida per l'utente di Amazon Athena.

- Per i dettagli sull'API, vedere BatchGetQueryExecutionin AWS CLI Command Reference.

`create-data-catalog`

Il seguente esempio di codice mostra come utilizzare`create-data-catalog`.

AWS CLI

Come creare un catalogo di dati

L'esempio `create-data-catalog` seguente crea il catalogo dati `dynamo_db_catalog`.

```
aws athena create-data-catalog \
    --name dynamo_db_catalog \
    --type LAMBDA \
    --description "DynamoDB Catalog" \
    --parameters function=arn:aws:lambda:us-
west-2:111122223333:function:dynamo_db_lambda
```

Questo comando non produce alcun output. Per visualizzare il risultato, utilizza `aws athena get-data-catalog --name dynamo_db_catalog`.

Per ulteriori informazioni, consulta Registrazione di un catalogo: create-data-catalog nella Guida per l'utente di Amazon Athena.

- Per i dettagli sull'API, consulta AWS CLI Command CreateDataCatalogReference.

`create-named-query`

Il seguente esempio di codice mostra come utilizzare`create-named-query`.

AWS CLI

Come creare una query denominata

L'esempio `create-named-query` seguente crea una query salvata nel gruppo di lavoro `AthenaAdmin` che esegue query sulla tabella `flights_parquet` relativa ai voli Seattle-New York nel gennaio 2016 caratterizzati da ritardi alla partenza e all'arrivo di oltre dieci minuti. Poiché i valori dei codici aeroportuali nella tabella sono stringhe che includono virgolette doppie (ad esempio, "SEA"), vengono preceduti da barre rovesciate e racchiusi tra virgolette singole.

```
aws athena create-named-query \
    --name "SEA to JFK delayed flights Jan 2016" \
    --description "Both arrival and departure delayed more than 10 minutes." \
    --database sampledb \
    --query-string "SELECT flightdate, carrier, flightnum, origin, dest,
 depdelayminutes, arrdelayminutes FROM sampledb.flights_parquet WHERE yr = 2016 AND
 month = 1 AND origin = '\"SEA\"' AND dest = '\"JFK\"' AND depdelayminutes > 10 AND
 arrdelayminutes > 10" \
    --work-group AthenaAdmin
```

Output:

```
{
    "NamedQueryId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
}
```

Per ulteriori informazioni, consulta Esecuzione di query SQL con Amazon Athena nella Guida per l'utente di Amazon Athena.

- Per i dettagli sull'API, consulta CreateNamedQuery AWS CLICommand Reference.

`create-work-group`

Il seguente esempio di codice mostra come utilizzare`create-work-group`.

AWS CLI

Come creare un gruppo di lavoro

L'esempio `create-work-group` seguente crea un gruppo di lavoro denominato `Data_Analyst_Group`, il cui output dei risultati delle query si trova in `s3://amzn-s3-demo-bucket`. Il comando crea un gruppo di lavoro che sostituisce le impostazioni di configurazione del client, il che include la posizione di output dei risultati della query. Il comando abilita inoltre le CloudWatch metriche e aggiunge tre coppie di tag chiave-valore al gruppo di lavoro per distinguerlo dagli altri gruppi di lavoro. Nota che l'argomento `--configuration` non ha spazi prima delle virgole che ne separano le opzioni.

```
aws athena create-work-group \
    --name Data_Analyst_Group \
    --configuration ResultConfiguration={OutputLocation="s3://amzn-s3-demo-
bucket"},EnforceWorkGroupConfiguration="true",PublishCloudWatchMetricsEnabled="true"
 \
    --description "Workgroup for data analysts" \
    --tags Key=Division,Value=West Key=Location,Value=Seattle Key=Team,Value="Big
 Data"
```

Questo comando non produce alcun output. Per visualizzare i risultati, utilizza `aws athena get-work-group --work-group Data_Analyst_Group`.

Per ulteriori informazioni, consulta Gestione dei gruppi di lavoro nella Guida per l'utente di Amazon Athena.

- Per i dettagli sull'API, consulta Command Reference. CreateWorkGroupAWS CLI

`delete-data-catalog`

Il seguente esempio di codice mostra come utilizzare`delete-data-catalog`.

AWS CLI

Come eliminare un catalogo di dati

L'esempio `delete-data-catalog` seguente elimina il catalogo dati `UnusedDataCatalog`.

```
aws athena delete-data-catalog \
    --name UnusedDataCatalog
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Eliminazione di un catalogo: delete-data-catalog nella Guida per l'utente di Amazon Athena.

- Per i dettagli sull'API, consulta AWS CLI Command DeleteDataCatalogReference.

`delete-named-query`

Il seguente esempio di codice mostra come utilizzare`delete-named-query`.

AWS CLI

Come eliminare una query denominata

L'esempio `delete-named-query` seguente elimina la query denominata con l'ID specificato.

```
aws athena delete-named-query \
    --named-query-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Esecuzione di query SQL con Amazon Athena nella Guida per l'utente di Amazon Athena.

- Per i dettagli sull'API, consulta DeleteNamedQuery AWS CLICommand Reference.

`delete-work-group`

Il seguente esempio di codice mostra come utilizzare`delete-work-group`.

AWS CLI

Come eliminare un gruppo di lavoro

L'esempio `delete-work-group` seguente elimina il gruppo di lavoro `TeamB`.

```
aws athena delete-work-group \
    --work-group TeamB
```

Questo comando non produce alcun output. Per confermare l'eliminazione, utilizza `aws athena list-work-groups`.

Per ulteriori informazioni, consulta Gestione dei gruppi di lavoro nella Guida per l'utente di Amazon Athena.

- Per i dettagli sull'API, consulta DeleteWorkGroup AWS CLICommand Reference.

`get-data-catalog`

Il seguente esempio di codice mostra come utilizzare`get-data-catalog`.

AWS CLI

Come restituire le informazioni su un catalogo dati

L'esempio `get-data-catalog` seguente restituisce informazioni sul catalogo dati `dynamo_db_catalog`.

```
aws athena get-data-catalog \
    --name dynamo_db_catalog
```

Output:

```
{
    "DataCatalog": {
        "Name": "dynamo_db_catalog",
        "Description": "DynamoDB Catalog",
        "Type": "LAMBDA",
        "Parameters": {
            "catalog": "dynamo_db_catalog",
            "metadata-function": "arn:aws:lambda:us-
west-2:111122223333:function:dynamo_db_lambda",
            "record-function": "arn:aws:lambda:us-
west-2:111122223333:function:dynamo_db_lambda"
        }
    }
}
```

Per ulteriori informazioni, consulta Mostrare i dettagli del catalogo: get-data-catalog nella Guida per l'utente di Amazon Athena.

- Per i dettagli sull'API, consulta GetDataCatalog AWS CLICommand Reference.

`get-database`

Il seguente esempio di codice mostra come utilizzare`get-database`.

AWS CLI

Come restituire le informazioni su un database in un catalogo dati

L'esempio `get-database` seguente restituisce informazioni sul database `sampledb` nel catalogo dati `AwsDataCatalog`.

```
aws athena get-database \
    --catalog-name AwsDataCatalog \
    --database-name sampledb
```

Output:

```
{
    "Database": {
        "Name": "sampledb",
        "Description": "Sample database",
        "Parameters": {
            "CreatedBy": "Athena",
            "EXTERNAL": "TRUE"
        }
    }
}
```

Per ulteriori informazioni, consulta Visualizzazione dei dettagli del database: get-database nella Guida per l'utente di Amazon Athena.

- Per i dettagli sull'API, consulta GetDatabase AWS CLICommand Reference.

`get-named-query`

Il seguente esempio di codice mostra come utilizzare`get-named-query`.

AWS CLI

Come restituire una query denominata

L'esempio `get-named-query` seguente restituisce informazioni sulle query con l'ID specificato.

```
aws athena get-named-query \
    --named-query-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Output:

```
{
    "NamedQuery": {
        "Name": "CloudFront Logs - SFO",
        "Description": "Shows successful GET request data for SFO",
        "Database": "default",
        "QueryString": "select date, location, browser, uri, status from
 cloudfront_logs where method = 'GET' and status = 200 and location like 'SFO%'
 limit 10",
        "NamedQueryId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
        "WorkGroup": "AthenaAdmin"
    }
}
```

Per ulteriori informazioni, consulta Esecuzione di query SQL con Amazon Athena nella Guida per l'utente di Amazon Athena.

- Per i dettagli sull'API, consulta GetNamedQuery AWS CLICommand Reference.

`get-query-execution`

Il seguente esempio di codice mostra come utilizzare`get-query-execution`.

AWS CLI

Come restituire informazioni su un'esecuzione di query

L'esempio `get-query-execution` seguente restituisce informazioni sulle query con l'ID query specificato.

```
aws athena get-query-execution \
    --query-execution-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Output:

```
{
    "QueryExecution": {
        "QueryExecutionId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
        "Query": "select date, location, browser, uri, status from cloudfront_logs
 where method = 'GET
' and status = 200 and location like 'SFO%' limit 10",
```

```
        "StatementType": "DML",
        "ResultConfiguration": {
            "OutputLocation": "s3://amzn-s3-demo-bucket/a1b2c3d4-5678-90ab-cdef-
EXAMPLE11111.csv"
        },
        "QueryExecutionContext": {
            "Database": "mydatabase",
            "Catalog": "awsdatacatalog"
        },
        "Status": {
            "State": "SUCCEEDED",
            "SubmissionDateTime": 1593469842.665,
            "CompletionDateTime": 1593469846.486
        },
        "Statistics": {
            "EngineExecutionTimeInMillis": 3600,
            "DataScannedInBytes": 203089,
            "TotalExecutionTimeInMillis": 3821,
            "QueryQueueTimeInMillis": 267,
            "QueryPlanningTimeInMillis": 1175
        },
        "WorkGroup": "AthenaAdmin"
    }
}
```

Per ulteriori informazioni, consulta Esecuzione di query SQL con Amazon Athena nella Guida per l'utente di Amazon Athena.

- Per i dettagli sull'API, consulta GetQueryExecution AWS CLICommand Reference.

`get-query-results`

Il seguente esempio di codice mostra come utilizzare`get-query-results`.

AWS CLI

Come restituire i risultati di una query

L'esempio `get-query-results` seguente restituisce i risultati della query con l'ID query specificato.

```
aws athena get-query-results \
    --query-execution-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Output:


```
{
    "ResultSet": {
        "Rows": [
            {
                "Data": [
                    {
                        "VarCharValue": "date"
                    },
                    {
                        "VarCharValue": "location"
                    },
                    {
                        "VarCharValue": "browser"
                    },
                    {
                        "VarCharValue": "uri"
                    },
                    {
                        "VarCharValue": "status"
                    }
                ]
            },
            {
                "Data": [
                    {
                        "VarCharValue": "2014-07-05"
                    },
                    {
                        "VarCharValue": "SFO4"
                    },
                    {
                        "VarCharValue": "Safari"
                    },
                    {
                        "VarCharValue": "/test-image-2.jpeg"
                    },
                    {
                        "VarCharValue": "200"
                    }
                ]
            },
            {
```

```
                "Data": [
                    {
                        "VarCharValue": "2014-07-05"
                    },
                    {
                        "VarCharValue": "SFO4"
                    },
                    {
                        "VarCharValue": "Opera"
                    },
                    {
                        "VarCharValue": "/test-image-2.jpeg"
                    },
                    {
                        "VarCharValue": "200"
                    }
                ]
            },
            {
                "Data": [
                    {
                        "VarCharValue": "2014-07-05"
                    },
                    {
                        "VarCharValue": "SFO4"
                    },
                    {
                        "VarCharValue": "Firefox"
                    },
                    {
                        "VarCharValue": "/test-image-3.jpeg"
                    },
                    {
                        "VarCharValue": "200"
                    }
                ]
            },
            {
                "Data": [
                    {
                        "VarCharValue": "2014-07-05"
                    },
                    {
                        "VarCharValue": "SFO4"
```

```
                    },
                    {
                        "VarCharValue": "Lynx"
                    },
                    {
                        "VarCharValue": "/test-image-3.jpeg"
                    },
                    {
                        "VarCharValue": "200"
                    }
                ]
            },
            {
                "Data": [
                    {
                        "VarCharValue": "2014-07-05"
                    },
                    {
                        "VarCharValue": "SFO4"
                    },
                    {
                        "VarCharValue": "IE"
                    },
                    {
                        "VarCharValue": "/test-image-2.jpeg"
                    },
                    {
                        "VarCharValue": "200"
                    }
                ]
            },
            {
                "Data": [
                    {
                        "VarCharValue": "2014-07-05"
                    },
                    {
                        "VarCharValue": "SFO4"
                    },
                    {
                        "VarCharValue": "Opera"
                    },
                    {
                        "VarCharValue": "/test-image-1.jpeg"
```

```
                    },
                    {
                        "VarCharValue": "200"
                    }
                ]
            },
            {
                "Data": [
                    {
                        "VarCharValue": "2014-07-05"
                    },
                    {
                        "VarCharValue": "SFO4"
                    },
                    {
                        "VarCharValue": "Chrome"
                    },
                    {
                        "VarCharValue": "/test-image-3.jpeg"
                    },
                    {
                        "VarCharValue": "200"
                    }
                ]
            },
            {
                "Data": [
                    {
                        "VarCharValue": "2014-07-05"
                    },
                    {
                        "VarCharValue": "SFO4"
                    },
                    {
                        "VarCharValue": "Firefox"
                    },
                    {
                        "VarCharValue": "/test-image-2.jpeg"
                    },
                    {
                        "VarCharValue": "200"
                    }
                ]
            },
```

```
            {
                "Data": [
                    {
                        "VarCharValue": "2014-07-05"
                    },
                    {
                        "VarCharValue": "SFO4"
                    },
                    {
                        "VarCharValue": "Chrome"
                    },
                    {
                        "VarCharValue": "/test-image-3.jpeg"
                    },
                    {
                        "VarCharValue": "200"
                    }
                ]
            },
            {
                "Data": [
                    {
                        "VarCharValue": "2014-07-05"
                    },
                    {
                        "VarCharValue": "SFO4"
                    },
                    {
                        "VarCharValue": "IE"
                    },
                    {
                        "VarCharValue": "/test-image-2.jpeg"
                    },
                    {
                        "VarCharValue": "200"
                    }
                ]
            }
        ],
        "ResultSetMetadata": {
            "ColumnInfo": [
                {
                    "CatalogName": "hive",
                    "SchemaName": "",
```

```
                    "TableName": "",
                    "Name": "date",
                    "Label": "date",
                    "Type": "date",
                    "Precision": 0,
                    "Scale": 0,
                    "Nullable": "UNKNOWN",
                    "CaseSensitive": false
                },
                {
                    "CatalogName": "hive",
                    "SchemaName": "",
                    "TableName": "",
                    "Name": "location",
                    "Label": "location",
                    "Type": "varchar",
                    "Precision": 2147483647,
                "Data": [

                    "Scale": 0,
                    "Nullable": "UNKNOWN",
                    "CaseSensitive": true
                },
                {
                    "CatalogName": "hive",
                    "SchemaName": "",
                    "TableName": "",
                    "Name": "browser",
                    "Label": "browser",
                    "Type": "varchar",
                    "Precision": 2147483647,
                    "Scale": 0,
                    "Nullable": "UNKNOWN",
                    "CaseSensitive": true
                },
                {
                    "CatalogName": "hive",
                    "SchemaName": "",
                    "TableName": "",
                    "Name": "uri",
                    "Label": "uri",
                    "Type": "varchar",
                    "Precision": 2147483647,
                    "Scale": 0,
```

```
                    "Nullable": "UNKNOWN",
                    "CaseSensitive": true
                },
                {
                    "CatalogName": "hive",
                    "SchemaName": "",
                    "TableName": "",
                    "Name": "status",
                    "Label": "status",
                    "Type": "integer",
                    "Precision": 10,
                    "Scale": 0,
                    "Nullable": "UNKNOWN",
                    "CaseSensitive": false
                }
            ]
        }
    },
    "UpdateCount": 0
}
```

Per ulteriori informazioni, consulta Utilizzo di risultati delle query, file di output e cronologia delle query nella Guida per l'utente di Amazon Athena.

- Per i dettagli sull'API, consulta GetQueryResults AWS CLICommand Reference.

`get-table-metadata`

Il seguente esempio di codice mostra come utilizzare`get-table-metadata`.

AWS CLI

Come restituire le informazioni sui metadati relativi a una tabella

L'esempio `get-table-metadata` seguente restituisce informazioni sui metadati della tabella `counties`, inclusi i nomi di colonna e i relativi tipi di dati, dal database `sampledb` del catalogo dati `AwsDataCatalog`.

```
aws athena get-table-metadata \
    --catalog-name AwsDataCatalog \
    --database-name sampledb \
    --table-name counties
```

Output:

```
{
    "TableMetadata": {
        "Name": "counties",
        "CreateTime": 1593559968.0,
        "LastAccessTime": 0.0,
        "TableType": "EXTERNAL_TABLE",
        "Columns": [
            {
                "Name": "name",
                "Type": "string",
                "Comment": "from deserializer"
            },
            {
                "Name": "boundaryshape",
                "Type": "binary",
                "Comment": "from deserializer"
            },
            {
                "Name": "motto",
                "Type": "string",
                "Comment": "from deserializer"
            },
            {
                "Name": "population",
                "Type": "int",
                "Comment": "from deserializer"
            }
        ],
        "PartitionKeys": [],
        "Parameters": {
            "EXTERNAL": "TRUE",
            "inputformat": "com.esri.json.hadoop.EnclosedJsonInputFormat",
            "location": "s3://amzn-s3-demo-bucket/json",
            "outputformat":
 "org.apache.hadoop.hive.ql.io.HiveIgnoreKeyTextOutputFormat",
            "serde.param.serialization.format": "1",
            "serde.serialization.lib": "com.esri.hadoop.hive.serde.JsonSerde",
            "transient_lastDdlTime": "1593559968"
        }
    }
}
```

Per ulteriori informazioni, consulta Mostrare i dettagli della tabella: get-table-metadata nella Guida per l'utente di Amazon Athena.

- Per i dettagli sull'API, consulta GetTableMetadata AWS CLICommand Reference.

`get-work-group`

Il seguente esempio di codice mostra come utilizzare`get-work-group`.

AWS CLI

Come restituire le informazioni su un gruppo di lavoro

L'esempio `get-work-group` seguente restituisce informazioni sul gruppo di lavoro `AthenaAdmin`.

```
aws athena get-work-group \
    --work-group AthenaAdmin
```

Output:

```
{
    "WorkGroup": {
        "Name": "AthenaAdmin",
        "State": "ENABLED",
        "Configuration": {
            "ResultConfiguration": {
                "OutputLocation": "s3://amzn-s3-demo-bucket/"
            },
            "EnforceWorkGroupConfiguration": false,
            "PublishCloudWatchMetricsEnabled": true,
            "RequesterPaysEnabled": false
        },
        "Description": "Workgroup for Athena administrators",
        "CreationTime": 1573677174.105
    }
}
```

Per ulteriori informazioni, consulta Gestione dei gruppi di lavoro nella Guida per l'utente di Amazon Athena.

- Per i dettagli sull'API, consulta GetWorkGroup AWS CLICommand Reference.

`list-data-catalogs`

Il seguente esempio di codice mostra come utilizzare`list-data-catalogs`.

AWS CLI

Come elencare i cataloghi dati registrati con Athena

L'esempio `list-data-catalogs` seguente elenca i cataloghi dati registrati con Athena.

```
aws athena list-data-catalogs
```

Output:

```
{
    "DataCatalogsSummary": [
        {
            "CatalogName": "AwsDataCatalog",
            "Type": "GLUE"
        },
        {
            "CatalogName": "cw_logs_catalog",
            "Type": "LAMBDA"
        },
        {
            "CatalogName": "cw_metrics_catalog",
            "Type": "LAMBDA"
        }
    ]
}
```

Per ulteriori informazioni, consulta Pubblicare cataloghi registrati: list-data-catalogs nella Guida per l'utente di Amazon Athena.

- Per i dettagli sull'API, consulta AWS CLI Command ListDataCatalogsReference.

`list-databases`

Il seguente esempio di codice mostra come utilizzare`list-databases`.

AWS CLI

Come elencare i database in un catalogo dati

L'esempio `list-databases` seguente elenca i database nel catalogo dati `AwsDataCatalog`.

```
aws athena list-databases \
    --catalog-name AwsDataCatalog
```

Output:

```
{
    "DatabaseList": [
        {
            "Name": "default"
        },
        {
            "Name": "mydatabase"
        },
        {
            "Name": "newdb"
        },
        {
            "Name": "sampledb",
            "Description": "Sample database",
            "Parameters": {
                "CreatedBy": "Athena",
                "EXTERNAL": "TRUE"
            }
        },
        {
            "Name": "webdata"
        }
    ]
}
```

Per ulteriori informazioni, consulta Elenco dei database in un catalogo: list-databases nella Guida per l'utente di Amazon Athena.

- Per i dettagli sull'API, consulta ListDatabases AWS CLICommand Reference.

`list-named-queries`

Il seguente esempio di codice mostra come utilizzare`list-named-queries`.

AWS CLI

Come elencare le query denominate per un gruppo di lavoro

L'esempio `list-named-queries` seguente elenca le query denominate per il gruppo di lavoro `AthenaAdmin`.

```
aws athena list-named-queries \
    --work-group AthenaAdmin
```

Output:

```
{
    "NamedQueryIds": [
        "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
        "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
        "a1b2c3d4-5678-90ab-cdef-EXAMPLE33333"
    ]
}
```

Per ulteriori informazioni, consulta Esecuzione di query SQL con Amazon Athena nella Guida per l'utente di Amazon Athena.

- Per i dettagli sull'API, consulta ListNamedQueries AWS CLICommand Reference.

`list-query-executions`

Il seguente esempio di codice mostra come utilizzare`list-query-executions`.

AWS CLI

Per elencare l'interrogazione IDs delle interrogazioni in un gruppo di lavoro specificato

L'`list-query-executions`esempio seguente elenca un massimo di dieci interrogazioni IDs nel gruppo di lavoro. `AthenaAdmin`

```
aws athena list-query-executions \
    --work-group AthenaAdmin \
    --max-items 10
```

Output:

```
{
    "QueryExecutionIds": [
        "a1b2c3d4-5678-90ab-cdef-EXAMPLE11110",
        "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
        "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
        "a1b2c3d4-5678-90ab-cdef-EXAMPLE33333",
        "a1b2c3d4-5678-90ab-cdef-EXAMPLE11114",
        "a1b2c3d4-5678-90ab-cdef-EXAMPLE11115",
        "a1b2c3d4-5678-90ab-cdef-EXAMPLE11116",
        "a1b2c3d4-5678-90ab-cdef-EXAMPLE11117",
        "a1b2c3d4-5678-90ab-cdef-EXAMPLE11118",
        "a1b2c3d4-5678-90ab-cdef-EXAMPLE11119"
    ],
    "NextToken": "eyJOZXh0VG9rZW4iOiBudWxsLCAiYm90b190cnVuY2F0ZV9hbW91bnQiOiAxMH0="
}
```

Per ulteriori informazioni, consulta Utilizzo di risultati delle query, file di output e cronologia delle query nella Guida per l'utente di Amazon Athena.

- Per i dettagli sull'API, vedere ListQueryExecutionsin AWS CLI Command Reference.

`list-table-metadata`

Il seguente esempio di codice mostra come utilizzare`list-table-metadata`.

AWS CLI

Come elencare i metadati per le tabelle nel database specificato di un catalogo dati

L'esempio `list-table-metadata` seguente restituisce informazioni sui metadati per un massimo di due tabelle nel database `geography` del catalogo dati `AwsDataCatalog`.

```
aws athena list-table-metadata \
    --catalog-name AwsDataCatalog \
    --database-name geography \
    --max-items 2
```

Output:

```
{
    "TableMetadataList": [
        {
```

```
            "Name": "country_codes",
            "CreateTime": 1586553454.0,
            "TableType": "EXTERNAL_TABLE",
            "Columns": [
                {
                    "Name": "country",
                    "Type": "string",
                    "Comment": "geo id"
                },
                {
                    "Name": "alpha-2 code",
                    "Type": "string",
                    "Comment": "geo id2"
                },
                {
                    "Name": "alpha-3 code",
                    "Type": "string",
                    "Comment": "state name"
                },
                {
                    "Name": "numeric code",
                    "Type": "bigint",
                    "Comment": ""
                },
                {
                    "Name": "latitude",
                    "Type": "bigint",
                    "Comment": "location (latitude)"
                },
                {
                    "Name": "longitude",
                    "Type": "bigint",
                    "Comment": "location (longitude)"
                }
            ],
            "Parameters": {
                "areColumnsQuoted": "false",
                "classification": "csv",
                "columnsOrdered": "true",
                "delimiter": ",",
                "has_encrypted_data": "false",
                "inputformat": "org.apache.hadoop.mapred.TextInputFormat",
                "location": "s3://amzn-s3-demo-bucket/csv/countrycode",
```

```
                "outputformat":
"org.apache.hadoop.hive.ql.io.HiveIgnoreKeyTextOutputFormat",
                "serde.param.field.delim": ",",
                "serde.serialization.lib":
"org.apache.hadoop.hive.serde2.lazy.LazySimpleSerDe",
                "skip.header.line.count": "1",
                "typeOfData": "file"
            }
        },
        {
            "Name": "county_populations",
            "CreateTime": 1586553446.0,
            "TableType": "EXTERNAL_TABLE",
            "Columns": [
                {
                    "Name": "id",
                    "Type": "string",
                    "Comment": "geo id"
                },
                {
                    "Name": "country",

                    "Name": "id2",
                    "Type": "string",
                    "Comment": "geo id2"
                },
                {
                    "Name": "county",
                    "Type": "string",
                    "Comment": "county name"
                },
                {
                    "Name": "state",
                    "Type": "string",
                    "Comment": "state name"
                },
                {
                    "Name": "population estimate 2018",
                    "Type": "string",
                    "Comment": ""
                }
            ],
            "Parameters": {
                "areColumnsQuoted": "false",
```

```
                "classification": "csv",
                "columnsOrdered": "true",
                "delimiter": ",",
                "has_encrypted_data": "false",
                "inputformat": "org.apache.hadoop.mapred.TextInputFormat",
                "location": "s3://amzn-s3-demo-bucket/csv/CountyPopulation",
                "outputformat":
 "org.apache.hadoop.hive.ql.io.HiveIgnoreKeyTextOutputFormat",
                "serde.param.field.delim": ",",
                "serde.serialization.lib":
 "org.apache.hadoop.hive.serde2.lazy.LazySimpleSerDe",
                "skip.header.line.count": "1",
                "typeOfData": "file"
            }
        }
    ],
    "NextToken": "eyJOZXh0VG9rZW4iOiBudWxsLCAiYm90b190cnVuY2F0ZV9hbW91bnQiOiAyfQ=="
}
```

Per ulteriori informazioni, consulta Mostrare i metadati per tutte le tabelle in un database: list-table-metadata nella Amazon Athena User Guide.

- Per i dettagli sull'API, consulta AWS CLI Command ListTableMetadataReference.

`list-tags-for-resource`

Il seguente esempio di codice mostra come utilizzare`list-tags-for-resource`.

AWS CLI

Esempio 1: come elencare i tag per un gruppo di lavoro

L'esempio `list-tags-for-resource` seguente elenca i tag per il gruppo di lavoro `Data_Analyst_Group`.

```
aws athena list-tags-for-resource \
    --resource-arn arn:aws:athena:us-west-2:111122223333:workgroup/
Data_Analyst_Group
```

Output:

```
{
```

```
    "Tags": [
        {
            "Key": "Division",
            "Value": "West"
        },
        {
            "Key": "Team",
            "Value": "Big Data"
        },
        {
            "Key": "Location",
            "Value": "Seattle"
        }
    ]
}
```

Esempio 2: come elencare i tag per un catalogo dati

L'esempio `list-tags-for-resource` seguente elenca i tag per il catalogo dati `dynamo_db_catalog`.

```
aws athena list-tags-for-resource \
    --resource-arn arn:aws:athena:us-west-2:111122223333:datacatalog/
dynamo_db_catalog
```

Output:

```
{
    "Tags": [
        {
            "Key": "Division",
            "Value": "Mountain"
        },
        {
            "Key": "Organization",
            "Value": "Retail"
        },
        {
            "Key": "Product_Line",
            "Value": "Shoes"
        },
        {
            "Key": "Location",
```

```
            "Value": "Denver"
        }
    ]
}
```

Per ulteriori informazioni, consulta Elencare i tag per una risorsa: list-tags-for-resource nella Amazon Athena User Guide.

- Per i dettagli sull'API, consulta ListTagsForResource AWS CLICommand Reference.

`list-work-groups`

Il seguente esempio di codice mostra come utilizzare`list-work-groups`.

AWS CLI

Come elencare i gruppi di lavoro

L'esempio `list-work-groups` seguente elenca i gruppi di lavoro nell'account corrente.

```
aws athena list-work-groups
```

Output:

```
{
    "WorkGroups": [
        {
            "Name": "Data_Analyst_Group",
            "State": "ENABLED",
            "Description": "",
            "CreationTime": 1578006683.016
        },
        {
            "Name": "AthenaAdmin",
            "State": "ENABLED",
            "Description": "",
            "CreationTime": 1573677174.105
        },
        {
            "Name": "primary",
            "State": "ENABLED",
            "Description": "",
```

```
            "CreationTime": 1567465222.723
        }
    ]
}
```

Per ulteriori informazioni, consulta Gestione dei gruppi di lavoro nella Guida per l'utente di Amazon Athena.

- Per i dettagli sull'API, consulta ListWorkGroups AWS CLICommand Reference.

`start-query-execution`

Il seguente esempio di codice mostra come utilizzare`start-query-execution`.

AWS CLI

Esempio 1: come eseguire una query in un gruppo di lavoro sulla tabella specificata nel database e nel catalogo dati specificati

L'esempio `start-query-execution` seguente utilizza il gruppo di lavoro `AthenaAdmin` per eseguire una query sulla tabella `cloudfront_logs` in `cflogsdatabase` nel catalogo dati `AwsDataCatalog`.

```
aws athena start-query-execution \
    --query-string "select date, location, browser, uri, status from cloudfront_logs
 where method = 'GET' and status = 200 and location like 'SFO%' limit 10" \
    --work-group "AthenaAdmin" \
    --query-execution-context Database=cflogsdatabase,Catalog=AwsDataCatalog
```

Output:

```
{
"QueryExecutionId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
}
```

Per ulteriori informazioni, consulta Esecuzione di query SQL con Amazon Athena nella Guida per l'utente di Amazon Athena.

Esempio 2: come eseguire una query che utilizza un gruppo di lavoro specificato per creare un database nel catalogo dati specificato

L'esempio `start-query-execution` seguente utilizza il gruppo di lavoro `AthenaAdmin` per creare il database `newdb` nel catalogo dati `AwsDataCatalog` predefinito.

```
aws athena start-query-execution \
    --query-string "create database if not exists newdb" \
    --work-group "AthenaAdmin"
```

Output:

```
{
"QueryExecutionId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11112"
}
```

Per ulteriori informazioni, consulta Esecuzione di query SQL con Amazon Athena nella Guida per l'utente di Amazon Athena.

Esempio 3: come eseguire una query che crea una vista su una tabella nel database e nel catalogo dati specificati

L'esempio `start-query-execution` seguente utilizza un'istruzione `SELECT` nella tabella `cloudfront_logs` in `cflogsdatabase` per creare la vista `cf10`.

```
aws athena start-query-execution \
    --query-string  "CREATE OR REPLACE VIEW cf10 AS SELECT * FROM cloudfront_logs
 limit 10" \
    --query-execution-context Database=cflogsdatabase
```

Output:

```
{
"QueryExecutionId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11113"
}
```

Per ulteriori informazioni, consulta Esecuzione di query SQL con Amazon Athena nella Guida per l'utente di Amazon Athena.

- Per i dettagli sull'API, consulta StartQueryExecution AWS CLICommand Reference.

`stop-query-execution`

Il seguente esempio di codice mostra come utilizzare`stop-query-execution`.

AWS CLI

Come arrestare una query in esecuzione

L'esempio `stop-query-execution` seguente arresta la query con l'ID query specificato.

```
aws athena stop-query-execution \
    --query-execution-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Esecuzione di query SQL con Amazon Athena nella Guida per l'utente di Amazon Athena.

- Per i dettagli sull'API, consulta StopQueryExecution AWS CLICommand Reference.

`tag-resource`

Il seguente esempio di codice mostra come utilizzare`tag-resource`.

AWS CLI

Come aggiungere un tag a una risorsa

L'esempio `tag-resource` seguente aggiunge tre tag al catalogo dati `dynamo_db_catalog`.

```
aws athena tag-resource \
    --resource-arn arn:aws:athena:us-west-2:111122223333:datacatalog/
dynamo_db_catalog \
    --
tags Key=Organization,Value=Retail Key=Division,Value=Mountain Key=Product_Line,Value=Shoes
```

Questo comando non produce alcun output. Per visualizzare il risultato, utilizza `aws athena list-tags-for-resource --resource-arn arn:aws:athena:us-west-2:111122223333:datacatalog/dynamo_db_catalog`.

Per ulteriori informazioni, consulta Aggiunta di tag associati a una risorsa: tag-resource nella Guida per l'utente di Amazon Athena.

- Per i dettagli sull'API, consulta TagResource AWS CLICommand Reference.

`untag-resource`

Il seguente esempio di codice mostra come utilizzare`untag-resource`.

AWS CLI

Come rimuovere un tag da una risorsa

L'esempio `untag-resource` seguente rimuove le chiavi `Focus` e `Specialization` e i relativi valori dalla risorsa del catalogo dati `dynamo_db_catalog`.

```
aws athena untag-resource \
    --resource-arn arn:aws:athena:us-west-2:111122223333:datacatalog/
dynamo_db_catalog \
    --tag-keys Specialization Focus
```

Questo comando non produce alcun output. Per visualizzare i risultati, utilizza il comando `list-tags-for-resource`.

Per ulteriori informazioni, consulta Rimozione di tag associati a una risorsa: untag-resource nella Guida per l'utente di Amazon Athena.

- Per i dettagli sull'API, consulta UntagResource AWS CLICommand Reference.

`update-data-catalog`

Il seguente esempio di codice mostra come utilizzare`update-data-catalog`.

AWS CLI

Come aggiornare un catalogo di dati

L'esempio `update-data-catalog` seguente aggiorna la funzione Lambda e la descrizione del catalogo dati `cw_logs_catalog`.

```
aws athena update-data-catalog \
    --name cw_logs_catalog \
    --type LAMBDA \
    --description "New CloudWatch Logs Catalog" \
    --function=arn:aws:lambda:us-west-2:111122223333:function:new_cw_logs_lambda
```

Questo comando non produce alcun output. Per visualizzare il risultato, utilizza `aws athena get-data-catalog --name cw_logs_catalog`.

Per ulteriori informazioni, consulta Aggiornamento di un catalogo: update-data-catalog nella Guida per l'utente di Amazon Athena.

- Per i dettagli sull'API, consulta UpdateDataCatalog AWS CLICommand Reference.

`update-work-group`

Il seguente esempio di codice mostra come utilizzare`update-work-group`.

AWS CLI

Come aggiornare un gruppo di lavoro

L'esempio `update-work-group` seguente disabilita il gruppo di lavoro `Data_Analyst_Group`. Gli utenti non possono eseguire o creare query nel gruppo di lavoro disattivato, ma possono comunque visualizzare le metriche, i controlli dei limiti di utilizzo dei dati, le impostazioni del gruppo di lavoro, la cronologia delle query e le query salvate.

```
aws athena update-work-group \
    --work-group Data_Analyst_Group \
    --state DISABLED
```

Questo comando non produce alcun output. Per verificare la variazione di stato, utilizza `aws athena get-work-group --work-group Data_Analyst_Group` e controlla la proprietà `State` nell'output.

Per ulteriori informazioni, consulta Gestione dei gruppi di lavoro nella Guida per l'utente di Amazon Athena.

- Per i dettagli sull'API, consulta UpdateWorkGroup AWS CLICommand Reference.

Esempi di Auto Scaling utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando l' AWS Command Line Interface Auto Scaling.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`attach-instances`

Il seguente esempio di codice mostra come utilizzare`attach-instances`.

AWS CLI

Come collegare un'istanza a un gruppo Auto Scaling

Questo esempio collega l'istanza specificata al gruppo Auto Scaling specificato.

```
aws autoscaling attach-instances \
    --instance-ids i-061c63c5eb45f0416 \
    --auto-scaling-group-name my-asg
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta AttachInstances AWS CLICommand Reference.

`attach-load-balancer-target-groups`

Il seguente esempio di codice mostra come utilizzare`attach-load-balancer-target-groups`.

AWS CLI

Come collegare un gruppo di destinazione a un gruppo Auto Scaling

Questo esempio collega il gruppo di destinazione specificato al gruppo Auto Scaling specificato.

```
aws autoscaling attach-load-balancer-target-groups \
    --auto-scaling-group-name my-asg \
```

```
    --target-group-arns arn:aws:elasticloadbalancing:us-
west-2:123456789012:targetgroup/my-targets/73e2d6bc24d8a067
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Elastic Load Balancing e Amazon Auto EC2 Scaling nella Amazon Auto EC2 Scaling User Guide.

- Per i dettagli sull'API, consulta Command AttachLoadBalancerTargetGroupsReference AWS CLI .

`attach-load-balancers`

Il seguente esempio di codice mostra come utilizzare`attach-load-balancers`.

AWS CLI

Come collegare un Classic Load Balancer al gruppo Auto Scaling

Questo esempio collega il Classic Load Balancer specificato al gruppo Auto Scaling specificato.

```
aws autoscaling attach-load-balancers \
    --load-balancer-names my-load-balancer \
    --auto-scaling-group-name my-asg
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Elastic Load Balancing e Amazon Auto EC2 Scaling nella Amazon Auto EC2 Scaling User Guide.

- Per i dettagli sull'API, consulta Command AttachLoadBalancersReference AWS CLI .

`cancel-instance-refresh`

Il seguente esempio di codice mostra come utilizzare`cancel-instance-refresh`.

AWS CLI

Come annullare l'aggiornamento di un'istanza

L'esempio `cancel-instance-refresh` seguente annulla un aggiornamento dell'istanza in corso per il gruppo di dimensionamento automatico specificato.

```
aws autoscaling cancel-instance-refresh \
    --auto-scaling-group-name my-asg
```

Output:

```
{
    "InstanceRefreshId": "08b91cf7-8fa6-48af-b6a6-d227f40f1b9b"
}
```

Per ulteriori informazioni, consulta Annullare l'aggiornamento di un'istanza nella Amazon EC2 Auto Scaling User Guide.

- Per i dettagli sull'API, consulta AWS CLI Command CancelInstanceRefreshReference.

`complete-lifecycle-action`

Il seguente esempio di codice mostra come utilizzare`complete-lifecycle-action`.

AWS CLI

Come completare un'azione del ciclo di vita

Questo esempio notifica ad Amazon EC2 Auto Scaling che l'azione del ciclo di vita specificata è completa in modo che possa completare l'avvio o la chiusura dell'istanza.

```
aws autoscaling complete-lifecycle-action \
    --lifecycle-hook-name my-launch-hook \
    --auto-scaling-group-name my-asg \
    --lifecycle-action-result CONTINUE \
    --lifecycle-action-token bcd2f1b8-9a78-44d3-8a7a-4dd07d7cf635
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta i riferimenti al ciclo di vita di Amazon EC2 Auto Scaling nella Amazon Auto EC2 Scaling User Guide.

- Per i dettagli sull'API, consulta Command Reference. CompleteLifecycleActionAWS CLI

`create-auto-scaling-group`

Il seguente esempio di codice mostra come utilizzare`create-auto-scaling-group`.

AWS CLI

Esempio 1: come creare un gruppo Auto Scaling

L'esempio `create-auto-scaling-group` seguente crea un gruppo Auto Scaling in sottoreti in più zone di disponibilità all'interno di una Regione. Le istanze vengono avviate con la versione predefinita del modello di avvio specificato. Tieni presente che le impostazioni predefinite vengono utilizzate per la maggior parte delle altre impostazioni, come le policy di terminazione e la configurazione dei controlli dell'integrità.

```
aws autoscaling create-auto-scaling-group \
    --auto-scaling-group-name my-asg \
    --launch-template LaunchTemplateId=lt-1234567890abcde12 \
    --min-size 1 \
    --max-size 5 \
    --vpc-zone-identifier "subnet-5ea0c127,subnet-6194ea3b,subnet-c934b782"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta i gruppi Auto Scaling nella Amazon Auto EC2 Scaling User Guide.

Esempio 2: come collegare un'istanza di Application Load Balancer, Network Load Balancer o Gateway Load Balancer

Questo esempio specifica l'ARN di un gruppo di destinazione per un bilanciatore del carico che supporta il traffico previsto. Il tipo di controllo dell'integrità è impostato su `ELB`, ovvero quando Elastic Load Balancing segnala un'istanza non integra, il gruppo Auto Scaling la sostituisce. Il comando definisce anche un periodo di tolleranza di `600` secondi per il controllo dell'integrità. Il periodo di tolleranza consente di evitare la terminazione anticipata delle istanze appena avviate.

```
aws autoscaling create-auto-scaling-group \
    --auto-scaling-group-name my-asg \
    --launch-template LaunchTemplateId=lt-1234567890abcde12 \
    --target-group-arns arn:aws:elasticloadbalancing:us-
west-2:123456789012:targetgroup/my-targets/943f017f100becff \
    --health-check-type ELB \
    --health-check-grace-period 600 \
    --min-size 1 \
    --max-size 5 \
    --vpc-zone-identifier "subnet-5ea0c127,subnet-6194ea3b,subnet-c934b782"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Elastic Load Balancing e Amazon Auto EC2 Scaling nella Amazon Auto EC2 Scaling User Guide.

Esempio 3: come specificare un gruppo di posizionamento e utilizzare la versione più recente del modello di avvio

Questo esempio avvia le istanze in un gruppo di posizionamento all'interno di un'unica zona di disponibilità. Ciò può essere utile per gruppi a bassa latenza con carichi di lavoro basati sul calcolo ad alte prestazioni (HPC). Questo esempio specifica anche la dimensione minima, la dimensione massima e la capacità desiderata del gruppo.

```
aws autoscaling create-auto-scaling-group \
    --auto-scaling-group-name my-asg \
    --launch-template LaunchTemplateId=lt-1234567890abcde12,Version='$Latest' \
    --min-size 1 \
    --max-size 5 \
    --desired-capacity 3 \
    --placement-group my-placement-group \
    --vpc-zone-identifier "subnet-6194ea3b"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Placement groups nella Amazon EC2 User Guide for Linux Instances.

Esempio 4: come specificare un gruppo Auto Scaling a istanza singola e utilizzare una versione specifica del modello di avvio

Questo esempio crea un gruppo Auto Scaling con capacità minima e massima impostate su 1 per forzare l'esecuzione di un'istanza. Il comando specifica anche v1 come versione di un modello di avvio in cui è specificato l'ID di un'interfaccia di rete elastica (ENI) esistente. Quando si utilizza un modello di avvio che specifica una interfaccia di rete elastica (ENI) esistente per eth0, è necessario specificare una zona di disponibilità per il gruppo Auto Scaling corrispondente all'interfaccia di rete, senza specificare anche un ID di sottorete nella richiesta.

```
aws autoscaling create-auto-scaling-group \
    --auto-scaling-group-name my-asg-single-instance \
    --launch-template LaunchTemplateName=my-template-for-auto-scaling,Version='1' \
```

```
    --min-size 1 \
    --max-size 1 \
    --availability-zones us-west-2a
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta i gruppi Auto Scaling nella Amazon Auto EC2 Scaling User Guide.

Esempio 5: come specificare una policy di terminazione diversa

Questo esempio crea un gruppo Auto Scaling utilizzando una configurazione di avvio e imposta la policy di terminazione per terminare prima le istanze più vecchie. Il comando applica inoltre un tag al gruppo e alle relative istanze, con una chiave `Role` e un valore `WebServer`.

```
aws autoscaling create-auto-scaling-group \
    --auto-scaling-group-name my-asg \
    --launch-configuration-name my-lc \
    --min-size 1 \
    --max-size 5 \
    --termination-policies "OldestInstance" \
    --tags "ResourceId=my-asg,ResourceType=auto-scaling-
group,Key=Role,Value=WebServer,PropagateAtLaunch=true" \
    --vpc-zone-identifier "subnet-5ea0c127,subnet-6194ea3b,subnet-c934b782"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Working with Amazon EC2 Auto Scaling policy di terminazione nella Amazon EC2 Auto Scaling User Guide.

Esempio 6: come specificare un hook del ciclo di vita all'avvio

L'esempio seguente crea un gruppo Auto Scaling con un hook del ciclo di vita che supporta un'azione personalizzata all'avvio delle istanze.

```
aws autoscaling create-auto-scaling-group \
    --cli-input-json file://~/config.json
```

Contenuto del file `config.json`:

```
{
```

```
    "AutoScalingGroupName": "my-asg",
    "LaunchTemplate": {
        "LaunchTemplateId": "lt-1234567890abcde12"
    },
    "LifecycleHookSpecificationList": [{
        "LifecycleHookName": "my-launch-hook",
        "LifecycleTransition": "autoscaling:EC2_INSTANCE_LAUNCHING",
        "NotificationTargetARN": "arn:aws:sqs:us-west-2:123456789012:my-sqs-queue",
        "RoleARN": "arn:aws:iam::123456789012:role/my-notification-role",
        "NotificationMetadata": "SQS message metadata",
        "HeartbeatTimeout": 4800,
        "DefaultResult": "ABANDON"
    }],
    "MinSize": 1,
    "MaxSize": 5,
    "VPCZoneIdentifier": "subnet-5ea0c127,subnet-6194ea3b,subnet-c934b782",
    "Tags": [{
        "ResourceType": "auto-scaling-group",
        "ResourceId": "my-asg",
        "PropagateAtLaunch": true,
        "Value": "test",
        "Key": "environment"
    }]
}
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta i riferimenti al ciclo di vita di Amazon EC2 Auto Scaling nella Amazon Auto EC2 Scaling User Guide.

Esempio 7: come specificare un hook del ciclo di vita alla terminazione

L'esempio seguente crea un gruppo Auto Scaling con un hook del ciclo di vita che supporta un'azione personalizzata alla terminazione delle istanze.

```
aws autoscaling create-auto-scaling-group \
    --cli-input-json file://~/config.json
```

Contenuto di `config.json`:

```
{
    "AutoScalingGroupName": "my-asg",
```

```
    "LaunchTemplate": {
        "LaunchTemplateId": "lt-1234567890abcde12"
    },
    "LifecycleHookSpecificationList": [{
        "LifecycleHookName": "my-termination-hook",
        "LifecycleTransition": "autoscaling:EC2_INSTANCE_TERMINATING",
        "HeartbeatTimeout": 120,
        "DefaultResult": "CONTINUE"
    }],
    "MinSize": 1,
    "MaxSize": 5,
    "TargetGroupARNs": [
        "arn:aws:elasticloadbalancing:us-west-2:123456789012:targetgroup/my-
targets/73e2d6bc24d8a067"
    ],
    "VPCZoneIdentifier": "subnet-5ea0c127,subnet-6194ea3b,subnet-c934b782"
}
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta i riferimenti al ciclo di vita di Amazon EC2 Auto Scaling nella Amazon Auto EC2 Scaling User Guide.

Esempio 8: come specificare una policy di terminazione personalizzata

Questo esempio crea un gruppo Auto Scaling che specifica una policy di terminazione della funzione Lambda personalizzata che indica ad Amazon Auto EC2 Scaling su quali istanze è sicuro terminare su larga scala.

```
aws autoscaling create-auto-scaling-group \
    --auto-scaling-group-name my-asg-single-instance \
    --launch-template LaunchTemplateName=my-template-for-auto-scaling \
    --min-size 1 \
    --max-size 5 \
    --termination-policies "arn:aws:lambda:us-
west-2:123456789012:function:HelloFunction:prod" \
    --vpc-zone-identifier "subnet-5ea0c127,subnet-6194ea3b,subnet-c934b782"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Creazione di una politica di terminazione personalizzata con Lambda nella Amazon Auto EC2 Scaling User Guide.

- Per i dettagli sull'API, consulta Command CreateAutoScalingGroupReference AWS CLI .

`create-launch-configuration`

Il seguente esempio di codice mostra come utilizzare`create-launch-configuration`.

AWS CLI

Esempio 1: come creare una configurazione di avvio

Questo esempio crea una configurazione di avvio semplice.

```
aws autoscaling create-launch-configuration \
    --launch-configuration-name my-lc \
    --image-id ami-04d5cc9b88example \
    --instance-type m5.large
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Creazione di una configurazione di avvio nella Amazon EC2 Auto Scaling User Guide.

Esempio 2: come creare una configurazione di avvio con un gruppo di sicurezza, una coppia di chiavi e uno script di avvio

Questo esempio crea una configurazione di avvio con un gruppo di sicurezza, una coppia di chiavi e uno script di avvio contenuti nei dati utente.

```
aws autoscaling create-launch-configuration \
    --launch-configuration-name my-lc \
    --image-id ami-04d5cc9b88example \
    --instance-type m5.large \
    --security-groups sg-eb2af88example \
    --key-name my-key-pair \
    --user-data file://myuserdata.txt
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Creazione di una configurazione di avvio nella Amazon EC2 Auto Scaling User Guide.

Esempio 3: come creare una configurazione di avvio con un ruolo IAM

Questo esempio crea una configurazione di avvio con il nome del profilo dell'istanza di un ruolo IAM.

```
aws autoscaling create-launch-configuration \
    --launch-configuration-name my-lc \
    --image-id ami-04d5cc9b88example \
    --instance-type m5.large \
    --iam-instance-profile my-autoscaling-role
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta il ruolo IAM per le applicazioni eseguite su EC2 istanze Amazon nella Amazon EC2 Auto Scaling User Guide.

Esempio 4: come creare una configurazione di avvio con il monitoraggio dettagliato abilitato

Questo esempio crea una configurazione di avvio con il monitoraggio EC2 dettagliato abilitato, che invia i EC2 parametri CloudWatch in periodi di 1 minuto.

```
aws autoscaling create-launch-configuration \
    --launch-configuration-name my-lc \
    --image-id ami-04d5cc9b88example \
    --instance-type m5.large \
    --instance-monitoring Enabled=true
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Configurazione del monitoraggio per le istanze Auto Scaling nella Amazon Auto EC2 Scaling User Guide.

Esempio 5: come creare una configurazione di avvio che avvia le istanze spot

Questo esempio crea una configurazione di avvio che utilizza le istanze spot come unica opzione di acquisto.

```
aws autoscaling create-launch-configuration \
    --launch-configuration-name my-lc \
    --image-id ami-04d5cc9b88example \
    --instance-type m5.large \
    --spot-price "0.50"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta la sezione Richiesta di istanze Spot nella Amazon Auto EC2 Scaling User Guide.

Esempio 6: creare una configurazione di avvio utilizzando un'istanza EC2

Questo esempio crea una configurazione di avvio basata sugli attributi di un'istanza esistente. Sostituisce la tenancy del posizionamento e l'eventuale impostazione di un indirizzo IP pubblico includendo le opzioni `--placement-tenancy` e `--no-associate-public-ip-address`.

```
aws autoscaling create-launch-configuration \
    --launch-configuration-name my-lc-from-instance \
    --instance-id i-0123a456700123456 \
    --instance-type m5.large \
    --no-associate-public-ip-address \
    --placement-tenancy dedicated
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Creazione di una configurazione di avvio utilizzando un' EC2 istanza nella Amazon EC2 Auto Scaling User Guide.

Esempio 7: come creare una configurazione di avvio con una mappatura dei dispositivi a blocchi per un volume Amazon EBS

Questo esempio crea una configurazione di avvio con una mappatura dei dispositivi a blocchi per un volume gp3 Amazon EBS con il nome di dispositivo `/dev/sdh` e una dimensione del volume pari a 20.

```
aws autoscaling create-launch-configuration \
    --launch-configuration-name my-lc \
    --image-id ami-04d5cc9b88example \
    --instance-type m5.large \
    --block-device-mappings '[{"DeviceName":"/dev/sdh","Ebs":
{"VolumeSize":20,"VolumeType":"gp3"}}]'
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta EBS nell'Amazon EC2 Auto Scaling API Reference.

Per informazioni sulla sintassi per citare i valori dei parametri in formato JSON, vedete Uso delle virgolette con le stringhe nella CLI AWS nella Guida per l'utente dell'interfaccia a riga di comando.AWS

Esempio 8: come creare una configurazione di avvio con una mappatura dei dispositivi a blocchi per un volume dell'archivio dell'istanza

Questo esempio crea una configurazione di avvio con `ephemeral1` come volume di archivio dell'istanza con il nome di dispositivo `/dev/sdc`.

```
aws autoscaling create-launch-configuration \
    --launch-configuration-name my-lc \
    --image-id ami-04d5cc9b88example \
    --instance-type m5.large \
    --block-device-mappings '[{"DeviceName":"/dev/sdc","VirtualName":"ephemeral1"}]'
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta la BlockDeviceMappingpagina di riferimento dell'API Amazon EC2 Auto Scaling.

Per informazioni sulla sintassi per citare i valori dei parametri in formato JSON, vedete Uso delle virgolette con le stringhe nella CLI AWS nella Guida per l'utente dell'interfaccia a riga di comando.AWS

Esempio 9: come creare una configurazione di avvio e impedire a un dispositivo a blocchi di collegarsi al momento dell'avvio

Questo esempio crea una configurazione di avvio che elimina un dispositivo a blocchi specificato dalla mappatura dei dispositivi a blocchi dell'AMI (ad esempio, `/dev/sdf`).

```
aws autoscaling create-launch-configuration \
    --launch-configuration-name my-lc \
    --image-id ami-04d5cc9b88example \
    --instance-type m5.large \
    --block-device-mappings '[{"DeviceName":"/dev/sdf","NoDevice":""}]'
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta la BlockDeviceMappingpagina di riferimento dell'API Amazon EC2 Auto Scaling.

Per informazioni sulla sintassi per citare i valori dei parametri in formato JSON, vedete Uso delle virgolette con le stringhe nella CLI AWS nella Guida per l'utente dell'interfaccia a riga di comando.AWS

- Per i dettagli sull'API, consulta Command Reference. CreateLaunchConfigurationAWS CLI

`create-or-update-tags`

Il seguente esempio di codice mostra come utilizzare`create-or-update-tags`.

AWS CLI

Come creare o modificare i tag per un gruppo Auto Scaling

Questo esempio aggiunge due tag al gruppo Auto Scaling specificato.

```
aws autoscaling create-or-update-tags \
    --tags ResourceId=my-asg,ResourceType=auto-scaling-
group,Key=Role,Value=WebServer,PropagateAtLaunch=true ResourceId=my-
asg,ResourceType=auto-scaling-group,Key=Dept,Value=Research,PropagateAtLaunch=true
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta la sezione Tagging di gruppi e istanze di Auto Scaling nella Amazon Auto EC2 Scaling User Guide.

- Per i dettagli sull'API, consulta CreateOrUpdateTagsCommand Reference.AWS CLI

`delete-auto-scaling-group`

Il seguente esempio di codice mostra come utilizzare`delete-auto-scaling-group`.

AWS CLI

Esempio 1: come eliminare il gruppo Auto Scaling specificato

Questo esempio elimina il gruppo Auto Scaling specificato.

```
aws autoscaling delete-auto-scaling-group \
    --auto-scaling-group-name my-asg
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Eliminazione dell'infrastruttura Auto Scaling nella Amazon Auto EC2 Scaling User Guide.

Esempio 2: come forzare l'eliminazione del gruppo Auto Scaling specificato

Per eliminare il gruppo Auto Scaling senza attendere il termine delle istanze del gruppo, utilizza l'opzione `--force-delete`.

```
aws autoscaling delete-auto-scaling-group \
    --auto-scaling-group-name my-asg \
    --force-delete
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Eliminazione dell'infrastruttura Auto Scaling nella Amazon Auto EC2 Scaling User Guide.

- Per i dettagli sull'API, consulta Command DeleteAutoScalingGroupReference AWS CLI .

`delete-launch-configuration`

Il seguente esempio di codice mostra come utilizzare`delete-launch-configuration`.

AWS CLI

Come eliminare la configurazione di avvio

Questo esempio elimina la configurazione di avvio specificata.

```
aws autoscaling delete-launch-configuration \
    --launch-configuration-name my-launch-config
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Eliminazione dell'infrastruttura Auto Scaling nella Amazon Auto EC2 Scaling User Guide.

- Per i dettagli sull'API, consulta Command DeleteLaunchConfigurationReference AWS CLI .

`delete-lifecycle-hook`

Il seguente esempio di codice mostra come utilizzare`delete-lifecycle-hook`.

AWS CLI

Come eliminare un hook del ciclo di vita

Questo esempio elimina l'hook del ciclo di vita specificato.

```
aws autoscaling delete-lifecycle-hook \
    --lifecycle-hook-name my-lifecycle-hook \
    --auto-scaling-group-name my-asg
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta DeleteLifecycleHook AWS CLICommand Reference.

`delete-notification-configuration`

Il seguente esempio di codice mostra come utilizzare`delete-notification-configuration`.

AWS CLI

Come eliminare una notifica di Auto Scaling

Questo esempio elimina la notifica specificata dal gruppo Auto Scaling specificato.

```
aws autoscaling delete-notification-configuration \
    --auto-scaling-group-name my-asg \
    --topic-arn arn:aws:sns:us-west-2:123456789012:my-sns-topic
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Eliminare la configurazione delle notifiche nella Amazon EC2 Auto Scaling User Guide.

- Per i dettagli sull'API, consulta DeleteNotificationConfiguration AWS CLICommand Reference.

`delete-policy`

Il seguente esempio di codice mostra come utilizzare`delete-policy`.

AWS CLI

Come eliminare una policy di dimensionamento

Questo esempio elimina la policy di dimensionamento specificata.

```
aws autoscaling delete-policy \
    --auto-scaling-group-name my-asg \
```

```
    --policy-name alb1000-target-tracking-scaling-policy
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta DeletePolicy AWS CLICommand Reference.

`delete-scheduled-action`

Il seguente esempio di codice mostra come utilizzare`delete-scheduled-action`.

AWS CLI

Come eliminare un'azione pianificata da un gruppo Auto Scaling

Questo esempio elimina l'azione pianificata specificata dal gruppo Auto Scaling specificato.

```
aws autoscaling delete-scheduled-action \
    --auto-scaling-group-name my-asg \
    --scheduled-action-name my-scheduled-action
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta DeleteScheduledAction AWS CLICommand Reference.

`delete-tags`

Il seguente esempio di codice mostra come utilizzare`delete-tags`.

AWS CLI

Come eliminare un tag da un gruppo Auto Scaling

Questo esempio elimina il tag specificato dal gruppo Auto Scaling specificato.

```
aws autoscaling delete-tags \
    --tags ResourceId=my-asg,ResourceType=auto-scaling-group,Key=Dept,Value=Research
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta la sezione Tagging di gruppi e istanze di Auto Scaling nella Amazon Auto EC2 Scaling User Guide.

- Per i dettagli sull'API, consulta DeleteTagsCommand Reference.AWS CLI

`delete-warm-pool`

Il seguente esempio di codice mostra come utilizzare`delete-warm-pool`.

AWS CLI

Esempio 1: come eliminare un warm pool

L'esempio seguente elimina un warm pool per il gruppo Auto Scaling specificato.

```
aws autoscaling delete-warm-pool \
    --auto-scaling-group-name my-asg
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Warm pools for Amazon EC2 Auto Scaling nella Amazon Auto EC2 Scaling User Guide.

Esempio 2: come forzare l'eliminazione di un warm pool

Per eliminare il warm pool senza attendere la terminazione delle relative istanze, utilizza l'opzione `--force-delete`.

```
aws autoscaling delete-warm-pool \
    --auto-scaling-group-name my-asg \
    --force-delete
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Warm pools for Amazon EC2 Auto Scaling nella Amazon Auto EC2 Scaling User Guide.

- Per i dettagli sull'API, consulta AWS CLI Command DeleteWarmPoolReference.

`describe-account-limits`

Il seguente esempio di codice mostra come utilizzare`describe-account-limits`.

AWS CLI

Per descrivere i limiti del tuo EC2 account Amazon Auto Scaling

Questo esempio descrive i limiti di Amazon EC2 Auto Scaling per il tuo AWS account.

```
aws autoscaling describe-account-limits
```

Output:

```
{
    "NumberOfLaunchConfigurations": 5,
    "MaxNumberOfLaunchConfigurations": 100,
    "NumberOfAutoScalingGroups": 3,
    "MaxNumberOfAutoScalingGroups": 20
}
```

Per ulteriori informazioni, consulta le quote del servizio Amazon EC2 Auto Scaling nella Amazon EC2 Auto Scaling User Guide.

- Per i dettagli sull'API, consulta Command DescribeAccountLimitsReference AWS CLI .

`describe-adjustment-types`

Il seguente esempio di codice mostra come utilizzare`describe-adjustment-types`.

AWS CLI

Come descrivere i tipi di regolazione del dimensionamento

Questo esempio descrive i tipi di regolazione disponibili.

```
aws autoscaling describe-adjustment-types
```

Output:

```
{
    "AdjustmentTypes": [
        {
            "AdjustmentType": "ChangeInCapacity"
        },
        {
            "AdjustmentType": "ExactCapacity"
        },
        {
            "AdjustmentType": "PercentChangeInCapacity"
```

```
        }
    ]
}
```

Per ulteriori informazioni, consulta i tipi di regolazione della scalabilità nella Guida per l'utente di Amazon EC2 Auto Scaling.

- Per i dettagli sull'API, consulta AWS CLI Command DescribeAdjustmentTypesReference.

`describe-auto-scaling-groups`

Il seguente esempio di codice mostra come utilizzare`describe-auto-scaling-groups`.

AWS CLI

Esempio 1: come descrivere il gruppo Auto Scaling specificato

Questo esempio descrive il gruppo Auto Scaling specificato.

```
aws autoscaling describe-auto-scaling-groups \
    --auto-scaling-group-names my-asg
```

Output:

```
{
    "AutoScalingGroups": [
        {
            "AutoScalingGroupName": "my-asg",
            "AutoScalingGroupARN": "arn:aws:autoscaling:us-
west-2:123456789012:autoScalingGroup:930d940e-891e-4781-
a11a-7b0acd480f03:autoScalingGroupName/my-asg",
            "LaunchTemplate": {
                "LaunchTemplateName": "my-launch-template",
                "Version": "1",
                "LaunchTemplateId": "lt-1234567890abcde12"
            },
            "MinSize": 0,
            "MaxSize": 1,
            "DesiredCapacity": 1,
            "DefaultCooldown": 300,
            "AvailabilityZones": [
                "us-west-2a",
```

```
                "us-west-2b",
                "us-west-2c"
            ],
            "LoadBalancerNames": [],
            "TargetGroupARNs": [],
            "HealthCheckType": "EC2",
            "HealthCheckGracePeriod": 0,
            "Instances": [
                {
                    "InstanceId": "i-06905f55584de02da",
                    "InstanceType": "t2.micro",
                    "AvailabilityZone": "us-west-2a",
                    "HealthStatus": "Healthy",
                    "LifecycleState": "InService",
                    "ProtectedFromScaleIn": false,
                    "LaunchTemplate": {
                        "LaunchTemplateName": "my-launch-template",
                        "Version": "1",
                        "LaunchTemplateId": "lt-1234567890abcde12"
                    }
                }
            ],
            "CreatedTime": "2023-10-28T02:39:22.152Z",
            "SuspendedProcesses": [],
            "VPCZoneIdentifier": "subnet-5ea0c127,subnet-6194ea3b,subnet-c934b782",
            "EnabledMetrics": [],
            "Tags": [],
            "TerminationPolicies": [
                "Default"
            ],
            "NewInstancesProtectedFromScaleIn": false,
            "ServiceLinkedRoleARN":"arn",
            "TrafficSources": []
        }
    ]
}
```

Esempio 2: come descrivere i primi 100 gruppi Auto Scaling specificati

Questo esempio descrive i gruppi Auto Scaling specificati. Consente di specificare fino a 100 nomi di gruppo.

```
aws autoscaling describe-auto-scaling-groups \
```

```
    --max-items 100 \
    --auto-scaling-group-names "group1" "group2" "group3" "group4"
```

Vedi l'esempio 1 per un output di esempio.

Esempio 3: come descrivere un gruppo Auto Scaling nella Regione specificata

Questo esempio descrive fino a un massimo di 75 gruppi Auto Scaling nella Regione specificata.

```
aws autoscaling describe-auto-scaling-groups \
    --max-items 75 \
    --region us-east-1
```

Vedi l'esempio 1 per un output di esempio.

Esempio 4: come descrivere il numero specificato di gruppi Auto Scaling

Per restituire un numero specifico di gruppi Auto Scaling, utilizza l'opzione `--max-items`.

```
aws autoscaling describe-auto-scaling-groups \
    --max-items 1
```

Vedi l'esempio 1 per un output di esempio.

Se l'output include un campo `NextToken`, significa che sono presenti più gruppi. Per ottenere i gruppi aggiuntivi, utilizza il valore di questo campo con l'opzione `--starting-token` in una chiamata successiva come descritto di seguito.

```
aws autoscaling describe-auto-scaling-groups \
    --starting-token Z3M3LMPEXAMPLE
```

Vedi l'esempio 1 per un output di esempio.

Esempio 5: come descrivere i gruppi Auto Scaling che utilizzano configurazioni di avvio

Questo esempio utilizza l'opzione `--query` per descrivere i gruppi Auto Scaling che utilizzano configurazioni di avvio.

```
aws autoscaling describe-auto-scaling-groups \
    --query 'AutoScalingGroups[?LaunchConfigurationName!=`null`]'
```

Output:


```
[
    {
        "AutoScalingGroupName": "my-asg",
        "AutoScalingGroupARN": "arn:aws:autoscaling:us-
west-2:123456789012:autoScalingGroup:930d940e-891e-4781-
a11a-7b0acd480f03:autoScalingGroupName/my-asg",
        "LaunchConfigurationName": "my-lc",
        "MinSize": 0,
        "MaxSize": 1,
        "DesiredCapacity": 1,
        "DefaultCooldown": 300,
        "AvailabilityZones": [
            "us-west-2a",
            "us-west-2b",
            "us-west-2c"
        ],
        "LoadBalancerNames": [],
        "TargetGroupARNs": [],
        "HealthCheckType": "EC2",
        "HealthCheckGracePeriod": 0,
        "Instances": [
            {
                "InstanceId": "i-088c57934a6449037",
                "InstanceType": "t2.micro",
                "AvailabilityZone": "us-west-2c",
                "HealthStatus": "Healthy",
                "LifecycleState": "InService",
                "LaunchConfigurationName": "my-lc",
                "ProtectedFromScaleIn": false
            }
        ],
        "CreatedTime": "2023-10-28T02:39:22.152Z",
        "SuspendedProcesses": [],
        "VPCZoneIdentifier": "subnet-5ea0c127,subnet-6194ea3b,subnet-c934b782",
        "EnabledMetrics": [],
        "Tags": [],
        "TerminationPolicies": [
            "Default"
        ],
        "NewInstancesProtectedFromScaleIn": false,
        "ServiceLinkedRoleARN":"arn",
        "TrafficSources": []
```

```
    }
]
```

Per ulteriori informazioni, consulta Filter AWS CLI output nella AWS Command Line Interface User Guide.

- Per i dettagli sull'API, consulta DescribeAutoScalingGroups AWS CLICommand Reference.

`describe-auto-scaling-instances`

Il seguente esempio di codice mostra come utilizzare`describe-auto-scaling-instances`.

AWS CLI

Esempio 1: come descrivere una o più istanze

Questo esempio descrive l'istanza specificata.

```
aws autoscaling describe-auto-scaling-instances \
    --instance-ids i-06905f55584de02da
```

Output:

```
{
    "AutoScalingInstances": [
        {
            "InstanceId": "i-06905f55584de02da",
            "InstanceType": "t2.micro",
            "AutoScalingGroupName": "my-asg",
            "AvailabilityZone": "us-west-2b",
            "LifecycleState": "InService",
            "HealthStatus": "HEALTHY",
            "ProtectedFromScaleIn": false,
            "LaunchTemplate": {
                "LaunchTemplateId": "lt-1234567890abcde12",
                "LaunchTemplateName": "my-launch-template",
                "Version": "1"
            }
        }
    ]
}
```

Esempio 2: come descrivere una o più istanze

Questo esempio utilizza l'opzione `--max-items` per specificare il numero di istanze da restituire con questa chiamata.

```
aws autoscaling describe-auto-scaling-instances \
    --max-items 1
```

Se l'output include un campo `NextToken`, significa che sono presenti più istanze. Per ottenere le istanze aggiuntive, utilizza il valore di questo campo con l'opzione `--starting-token` in una chiamata successiva come descritto di seguito.

```
aws autoscaling describe-auto-scaling-instances \
    --starting-token Z3M3LMPEXAMPLE
```

Vedi l'esempio 1 per un output di esempio.

Esempio 3: come descrivere le istanze che utilizzano le configurazioni di avvio

Questo esempio utilizza l'opzione `--query` per descrivere le istanze che utilizzano configurazioni di avvio.

```
aws autoscaling describe-auto-scaling-instances \
    --query 'AutoScalingInstances[?LaunchConfigurationName!=`null`]'
```

Output:

```
[
    {
        "InstanceId": "i-088c57934a6449037",
        "InstanceType": "t2.micro",
        "AutoScalingGroupName": "my-asg",
        "AvailabilityZone": "us-west-2c",
        "LifecycleState": "InService",
        "HealthStatus": "HEALTHY",
        "LaunchConfigurationName": "my-lc",
        "ProtectedFromScaleIn": false
    }
]
```

Per ulteriori informazioni, consulta Filter AWS CLI output nella AWS Command Line Interface User Guide.

- Per i dettagli sull'API, consulta DescribeAutoScalingInstances AWS CLICommand Reference.

`describe-auto-scaling-notification-types`

Il seguente esempio di codice mostra come utilizzare`describe-auto-scaling-notification-types`.

AWS CLI

Come descrivere i tipi di notifica disponibili

Questo esempio descrive i tipi di notifica disponibili.

```
aws autoscaling describe-auto-scaling-notification-types
```

Output:

```
{
    "AutoScalingNotificationTypes": [
        "autoscaling:EC2_INSTANCE_LAUNCH",
        "autoscaling:EC2_INSTANCE_LAUNCH_ERROR",
        "autoscaling:EC2_INSTANCE_TERMINATE",
        "autoscaling:EC2_INSTANCE_TERMINATE_ERROR",
        "autoscaling:TEST_NOTIFICATION"
    ]
}
```

Per ulteriori informazioni, consulta Ricevere notifiche Amazon SNS quando il tuo gruppo Auto Scaling aumenta nella Amazon EC2 Auto Scaling User Guide.

- Per i dettagli sull'API, consulta DescribeAutoScalingNotificationTypesCommand Reference.AWS CLI

`describe-instance-refreshes`

Il seguente esempio di codice mostra come utilizzare`describe-instance-refreshes`.

AWS CLI

Come descrivere gli aggiornamenti delle istanze

L'esempio `describe-instance-refreshes` seguente restituisce una descrizione di tutte le richieste di aggiornamento delle istanze per il gruppo Auto Scaling specificato, incluso il messaggio di stato e (se disponibile) il motivo dello stato.

```
aws autoscaling describe-instance-refreshes \
    --auto-scaling-group-name my-asg
```

Output:

```
{
    "InstanceRefreshes": [
        {
            "InstanceRefreshId": "08b91cf7-8fa6-48af-b6a6-d227f40f1b9b",
            "AutoScalingGroupName": "my-asg",
            "Status": "InProgress",
            "StatusReason": "Waiting for instances to warm up before continuing. For
 example: 0e69cc3f05f825f4f is warming up.",
            "EndTime": "2023-03-23T16:42:55Z",
            "PercentageComplete": 0,
            "InstancesToUpdate": 0,
    "Preferences": {
                "MinHealthyPercentage": 100,
                "InstanceWarmup": 300,
                "CheckpointPercentages": [
                    50
                ],
                "CheckpointDelay": 3600,
                "SkipMatching": false,
                "AutoRollback": true,
                "ScaleInProtectedInstances": "Ignore",
                "StandbyInstances": "Ignore"
            }
        },
        {
            "InstanceRefreshId": "dd7728d0-5bc4-4575-96a3-1b2c52bf8bb1",
            "AutoScalingGroupName": "my-asg",
            "Status": "Successful",
            "EndTime": "2022-06-02T16:53:37Z",
            "PercentageComplete": 100,
            "InstancesToUpdate": 0,
    "Preferences": {
                "MinHealthyPercentage": 90,
                "InstanceWarmup": 300,
                "SkipMatching": true,
                "AutoRollback": true,
                "ScaleInProtectedInstances": "Ignore",
                "StandbyInstances": "Ignore"
```

```
            }
        }
    ]
}
```

Per ulteriori informazioni, consulta Verifica dello stato di un aggiornamento dell'istanza nella Amazon EC2 Auto Scaling User Guide.

- Per i dettagli sull'API, consulta AWS CLI Command DescribeInstanceRefreshesReference.

`describe-launch-configurations`

Il seguente esempio di codice mostra come utilizzare`describe-launch-configurations`.

AWS CLI

Esempio 1: come descrivere la configurazione di avvio specificata

Questo esempio descrive la configurazione di avvio specificata.

```
aws autoscaling describe-launch-configurations \
    --launch-configuration-names my-launch-config
```

Output:

```
{
    "LaunchConfigurations": [
        {
            "LaunchConfigurationName": "my-launch-config",
            "LaunchConfigurationARN": "arn:aws:autoscaling:us-
west-2:123456789012:launchConfiguration:98d3b196-4cf9-4e88-8ca1-8547c24ced8b:launchConfigura
my-launch-config",
            "ImageId": "ami-0528a5175983e7f28",
            "KeyName": "my-key-pair-uswest2",
            "SecurityGroups": [
                "sg-05eaec502fcdadc2e"
            ],
            "ClassicLinkVPCSecurityGroups": [],
            "UserData": "",
            "InstanceType": "t2.micro",
            "KernelId": "",
            "RamdiskId": "",
```

```
            "BlockDeviceMappings": [
                {
                    "DeviceName": "/dev/xvda",
                    "Ebs": {
                        "SnapshotId": "snap-06c1606ba5ca274b1",
                        "VolumeSize": 8,
                        "VolumeType": "gp2",
                        "DeleteOnTermination": true,
                        "Encrypted": false
                    }
                }
            ],
            "InstanceMonitoring": {
                "Enabled": true
            },
            "CreatedTime": "2020-10-28T02:39:22.321Z",
            "EbsOptimized": false,
            "AssociatePublicIpAddress": true,
            "MetadataOptions": {
                "HttpTokens": "required",
                "HttpPutResponseHopLimit": 1,
                "HttpEndpoint": "disabled"
            }
        }
    ]
}
```

Esempio 2: come descrivere un numero specificato di configurazioni di avvio

Per restituire un numero specifico di configurazioni di avvio, utilizza l'opzione `--max-items`.

```
aws autoscaling describe-launch-configurations \
    --max-items 1
```

Se l'output include un campo `NextToken`, significa che sono presenti più configurazioni di avvio. Per ottenere le configurazioni di avvio aggiuntive, utilizza il valore di questo campo con l'opzione `--starting-token` in una chiamata successiva come descritto di seguito.

```
aws autoscaling describe-launch-configurations \
    --starting-token Z3M3LMPEXAMPLE
```

- Per i dettagli sull'API, consulta DescribeLaunchConfigurations AWS CLICommand Reference.

`describe-lifecycle-hook-types`

Il seguente esempio di codice mostra come utilizzare`describe-lifecycle-hook-types`.

AWS CLI

Come descrivere i tipi di hook del ciclo di vita disponibili

Questo esempio descrive i tipi di hook del ciclo di vita disponibili.

```
aws autoscaling describe-lifecycle-hook-types
```

Output:

```
{
    "LifecycleHookTypes": [
        "autoscaling:EC2_INSTANCE_LAUNCHING",
        "autoscaling:EC2_INSTANCE_TERMINATING"
    ]
}
```

- Per i dettagli sull'API, consulta DescribeLifecycleHookTypes AWS CLICommand Reference.

`describe-lifecycle-hooks`

Il seguente esempio di codice mostra come utilizzare`describe-lifecycle-hooks`.

AWS CLI

Come descrivere gli hook del ciclo di vita

Questo esempio descrive gli hook del ciclo di vita del gruppo Auto Scaling specificato.

```
aws autoscaling describe-lifecycle-hooks \
    --auto-scaling-group-name my-asg
```

Output:

```
{
    "LifecycleHooks": [
        {
            "GlobalTimeout": 3000,
```

```
            "HeartbeatTimeout": 30,
            "AutoScalingGroupName": "my-asg",
            "LifecycleHookName": "my-launch-hook",
            "DefaultResult": "ABANDON",
            "LifecycleTransition": "autoscaling:EC2_INSTANCE_LAUNCHING"
        },
        {
            "GlobalTimeout": 6000,
            "HeartbeatTimeout": 60,
            "AutoScalingGroupName": "my-asg",
            "LifecycleHookName": "my-termination-hook",
            "DefaultResult": "CONTINUE",
            "LifecycleTransition": "autoscaling:EC2_INSTANCE_TERMINATING"
        }
    ]
}
```

- Per i dettagli sull'API, consulta DescribeLifecycleHooks AWS CLICommand Reference.

`describe-load-balancer-target-groups`

Il seguente esempio di codice mostra come utilizzare`describe-load-balancer-target-groups`.

AWS CLI

Come descrivere i gruppi di destinazione del bilanciatore del carico per un gruppo Auto Scaling

In questo esempio, i gruppi di destinazione del bilanciatore del carico vengono collegati al gruppo Auto Scaling specificato.

```
aws autoscaling describe-load-balancer-target-groups \
    --auto-scaling-group-name my-asg
```

Output:

```
{
    "LoadBalancerTargetGroups": [
        {
            "LoadBalancerTargetGroupARN": "arn:aws:elasticloadbalancing:us-
west-2:123456789012:targetgroup/my-targets/73e2d6bc24d8a067",
            "State": "Added"
```

```
        }
    ]
}
```

- Per i dettagli sull'API, consulta DescribeLoadBalancerTargetGroups AWS CLICommand Reference.

`describe-load-balancers`

Il seguente esempio di codice mostra come utilizzare`describe-load-balancers`.

AWS CLI

Come descrivere i Classic Load Balancer per un gruppo Auto Scaling

Questo esempio descrive i Classic Load Balancer per il gruppo Auto Scaling specificato.

```
aws autoscaling describe-load-balancers \
    --auto-scaling-group-name my-asg
```

Output:

```
{
    "LoadBalancers": [
        {
            "State": "Added",
            "LoadBalancerName": "my-load-balancer"
        }
    ]
}
```

- Per i dettagli sull'API, consulta DescribeLoadBalancers AWS CLICommand Reference.

`describe-metric-collection-types`

Il seguente esempio di codice mostra come utilizzare`describe-metric-collection-types`.

AWS CLI

Come descrivere i tipi di raccolta delle metriche disponibili

Questo esempio descrive i tipi di raccolta delle metriche disponibili.

```
aws autoscaling describe-metric-collection-types
```

Output:

```
{
    "Metrics": [
        {
            "Metric": "GroupMinSize"
        },
        {
            "Metric": "GroupMaxSize"
        },
        {
            "Metric": "GroupDesiredCapacity"
        },
        {
            "Metric": "GroupInServiceInstances"
        },
        {
            "Metric": "GroupInServiceCapacity"
        },
        {
            "Metric": "GroupPendingInstances"
        },
        {
            "Metric": "GroupPendingCapacity"
        },
        {
            "Metric": "GroupTerminatingInstances"
        },
        {
            "Metric": "GroupTerminatingCapacity"
        },
        {
            "Metric": "GroupStandbyInstances"
        },
        {
            "Metric": "GroupStandbyCapacity"
        },
        {
            "Metric": "GroupTotalInstances"
        },
        {
```

```
            "Metric": "GroupTotalCapacity"
        }
    ],
    "Granularities": [
        {
            "Granularity": "1Minute"
        }
    ]
}
```

Per ulteriori informazioni, consulta le metriche del gruppo Auto Scaling nella Amazon EC2 Auto Scaling User Guide.

- Per i dettagli sull'API, consulta Command DescribeMetricCollectionTypesReference AWS CLI .

`describe-notification-configurations`

Il seguente esempio di codice mostra come utilizzare`describe-notification-configurations`.

AWS CLI

Esempio 1: come descrivere le configurazioni di notifica di un gruppo specificato

Questo esempio descrive le configurazioni di notifica del gruppo Auto Scaling specificato.

```
aws autoscaling describe-notification-configurations \
    --auto-scaling-group-name my-asg
```

Output:

```
{
    "NotificationConfigurations": [
        {
            "AutoScalingGroupName": "my-asg",
            "NotificationType": "autoscaling:TEST_NOTIFICATION",
            "TopicARN": "arn:aws:sns:us-west-2:123456789012:my-sns-topic-2"
        },
        {
            "AutoScalingGroupName": "my-asg",
            "NotificationType": "autoscaling:TEST_NOTIFICATION",
            "TopicARN": "arn:aws:sns:us-west-2:123456789012:my-sns-topic"
        }
```

```
    ]
}
```

Per ulteriori informazioni, consulta Ricevere notifiche Amazon SNS quando il tuo gruppo Auto Scaling aumenta nella Amazon EC2 Auto Scaling User Guide.

Esempio 1: come descrivere il numero specificato di configurazioni di notifica

Per restituire un numero specifico di configurazioni di notifica, utilizza il parametro `max-items`.

```
aws autoscaling describe-notification-configurations \
    --auto-scaling-group-name my-auto-scaling-group \
    --max-items 1
```

Output:

```
{
    "NotificationConfigurations": [
        {
            "AutoScalingGroupName": "my-asg",
            "NotificationType": "autoscaling:TEST_NOTIFICATION",
            "TopicARN": "arn:aws:sns:us-west-2:123456789012:my-sns-topic-2"
        },
        {
            "AutoScalingGroupName": "my-asg",
            "NotificationType": "autoscaling:TEST_NOTIFICATION",
            "TopicARN": "arn:aws:sns:us-west-2:123456789012:my-sns-topic"
        }
    ]
}
```

Se l'output include un campo `NextToken`, significa che sono presenti più configurazioni di notifica. Per ottenere le configurazioni di notifica aggiuntive, utilizza il valore di questo campo con il parametro `starting-token` in una chiamata successiva come descritto di seguito.

```
aws autoscaling describe-notification-configurations \
    --auto-scaling-group-name my-asg \
    --starting-token Z3M3LMPEXAMPLE
```

Per ulteriori informazioni, consulta Ricevere notifiche Amazon SNS quando il tuo gruppo Auto Scaling aumenta nella Amazon EC2 Auto Scaling User Guide.

- Per i dettagli sull'API, consulta DescribeNotificationConfigurationsCommand Reference.AWS CLI

`describe-policies`

Il seguente esempio di codice mostra come utilizzare`describe-policies`.

AWS CLI

Esempio 1: come descrivere le policy di dimensionamento di un gruppo specificato

Questo esempio descrive le policy di dimensionamento del gruppo Auto Scaling specificato.

```
aws autoscaling describe-policies \
    --auto-scaling-group-name my-asg
```

Output:

```
{
    "ScalingPolicies": [
        {
            "AutoScalingGroupName": "my-asg",
            "PolicyName": "alb1000-target-tracking-scaling-policy",
            "PolicyARN": "arn:aws:autoscaling:us-
west-2:123456789012:scalingPolicy:3065d9c8-9969-4bec-
bb6a-3fbe5550fde6:autoScalingGroupName/my-asg:policyName/alb1000-target-tracking-
scaling-policy",
            "PolicyType": "TargetTrackingScaling",
            "StepAdjustments": [],
            "Alarms": [
                {
                    "AlarmName": "TargetTracking-my-asg-
AlarmHigh-924887a9-12d7-4e01-8686-6f844d13a196",
                    "AlarmARN": "arn:aws:cloudwatch:us-
west-2:123456789012:alarm:TargetTracking-my-asg-
AlarmHigh-924887a9-12d7-4e01-8686-6f844d13a196"
                },
                {
                    "AlarmName": "TargetTracking-my-asg-AlarmLow-f96f899d-b8e7-4d09-
a010-c1aaa35da296",
                    "AlarmARN": "arn:aws:cloudwatch:us-
west-2:123456789012:alarm:TargetTracking-my-asg-AlarmLow-f96f899d-b8e7-4d09-a010-
c1aaa35da296"
```

```
                }
            ],
            "TargetTrackingConfiguration": {
                "PredefinedMetricSpecification": {
                    "PredefinedMetricType": "ALBRequestCountPerTarget",
                    "ResourceLabel": "app/my-alb/778d41231b141a0f/targetgroup/my-
alb-target-group/943f017f100becff"
                },
                "TargetValue": 1000.0,
                "DisableScaleIn": false
            },
            "Enabled": true
        },
        {
            "AutoScalingGroupName": "my-asg",
            "PolicyName": "cpu40-target-tracking-scaling-policy",
            "PolicyARN": "arn:aws:autoscaling:us-
west-2:123456789012:scalingPolicy:5fd26f71-39d4-4690-82a9-
b8515c45cdde:autoScalingGroupName/my-asg:policyName/cpu40-target-tracking-scaling-
policy",
            "PolicyType": "TargetTrackingScaling",
            "StepAdjustments": [],
            "Alarms": [
                {
                    "AlarmName": "TargetTracking-my-asg-
AlarmHigh-139f9789-37b9-42ad-bea5-b5b147d7f473",
                    "AlarmARN": "arn:aws:cloudwatch:us-
west-2:123456789012:alarm:TargetTracking-my-asg-AlarmHigh-139f9789-37b9-42ad-bea5-
b5b147d7f473"
                },
                {
                    "AlarmName": "TargetTracking-my-asg-AlarmLow-bd681c67-
fc18-4c56-8468-fb8e413009c9",
                    "AlarmARN": "arn:aws:cloudwatch:us-
west-2:123456789012:alarm:TargetTracking-my-asg-AlarmLow-bd681c67-fc18-4c56-8468-
fb8e413009c9"
                }
            ],
            "TargetTrackingConfiguration": {
                "PredefinedMetricSpecification": {
                    "PredefinedMetricType": "ASGAverageCPUUtilization"
                },
                "TargetValue": 40.0,
                "DisableScaleIn": false
```

```
            },
            "Enabled": true
        }
    ]
}
```

Per ulteriori informazioni, consulta Dynamic Scaling nella Amazon EC2 Auto Scaling User Guide.

Esempio 2: come descrivere le policy di dimensionamento di un nome specificato

Per restituire policy di dimensionamento specifiche, utilizza l'opzione `--policy-names`.

```
aws autoscaling describe-policies \
    --auto-scaling-group-name my-asg \
    --policy-names cpu40-target-tracking-scaling-policy
```

Vedi l'esempio 1 per un output di esempio.

Per ulteriori informazioni, consulta Dynamic Scaling nella Amazon EC2 Auto Scaling User Guide.

Esempio 3: come descrivere una serie di policy di dimensionamento

Per restituire un numero specifico di policy, utilizza l'opzione `--max-items`.

```
aws autoscaling describe-policies \
    --auto-scaling-group-name my-asg \
    --max-items 1
```

Vedi l'esempio 1 per un output di esempio.

Se l'output include un campo `NextToken`, utilizza il valore di questo campo con l'opzione `--starting-token` in una chiamata successiva per ottenere le policy aggiuntive.

```
aws autoscaling describe-policies --auto-scaling-group-name my-asg --starting-
token Z3M3LMPEXAMPLE
```

Per ulteriori informazioni, consulta Dynamic Scaling nella Amazon EC2 Auto Scaling User Guide.

- Per i dettagli sull'API, consulta AWS CLI Command DescribePoliciesReference.

`describe-scaling-activities`

Il seguente esempio di codice mostra come utilizzare`describe-scaling-activities`.

AWS CLI

Esempio 1: come descrivere le attività di dimensionamento per il gruppo specificato

Questo esempio descrive le attività di dimensionamento del gruppo Auto Scaling specificato.

```
aws autoscaling describe-scaling-activities \
    --auto-scaling-group-name my-asg
```

Output:

```
{
    "Activities": [
        {
            "ActivityId": "f9f2d65b-f1f2-43e7-b46d-d86756459699",
            "Description": "Launching a new EC2 instance: i-0d44425630326060f",
            "AutoScalingGroupName": "my-asg",
            "Cause": "At 2020-10-30T19:35:51Z a user request update of
 AutoScalingGroup constraints to min: 0, max: 16, desired: 16 changing the desired
 capacity from 0 to 16.  At 2020-10-30T19:36:07Z an instance was started in response
 to a difference between desired and actual capacity, increasing the capacity from 0
 to 16.",
            "StartTime": "2020-10-30T19:36:09.766Z",
            "EndTime": "2020-10-30T19:36:41Z",
            "StatusCode": "Successful",
            "Progress": 100,
            "Details": "{\"Subnet ID\":\"subnet-5ea0c127\",\"Availability Zone\":
\"us-west-2b\"}"
        }
    ]
}
```

Per ulteriori informazioni, consulta Verificare un'attività di scaling per un gruppo di Auto Scaling nella Amazon Auto EC2 Scaling User Guide.

Esempio 2: come descrivere le attività di dimensionamento per il gruppo eliminato

Per visualizzare le attività di dimensionamento dopo l'eliminazione di un gruppo Auto Scaling, aggiungi l'opzione `--include-deleted-groups`.

```
aws autoscaling describe-scaling-activities \
    --auto-scaling-group-name my-asg \
```

```
    --include-deleted-groups
```

Output:

```
{
    "Activities": [
        {
            "ActivityId": "e1f5de0e-f93e-1417-34ac-092a76fba220",
            "Description": "Launching a new EC2 instance.  Status Reason: Your Spot
 request price of 0.001 is lower than the minimum required Spot request fulfillment
 price of 0.0031. Launching EC2 instance failed.",
            "AutoScalingGroupName": "my-asg",
            "Cause": "At 2021-01-13T20:47:24Z a user request update of
 AutoScalingGroup constraints to min: 1, max: 5, desired: 3 changing the desired
 capacity from 0 to 3.  At 2021-01-13T20:47:27Z an instance was started in response
 to a difference between desired and actual capacity, increasing the capacity from 0
 to 3.",
            "StartTime": "2021-01-13T20:47:30.094Z",
            "EndTime": "2021-01-13T20:47:30Z",
            "StatusCode": "Failed",
            "StatusMessage": "Your Spot request price of 0.001 is lower than the
 minimum required Spot request fulfillment price of 0.0031. Launching EC2 instance
 failed.",
            "Progress": 100,
            "Details": "{\"Subnet ID\":\"subnet-5ea0c127\",\"Availability Zone\":
\"us-west-2b\"}",
            "AutoScalingGroupState": "Deleted",
            "AutoScalingGroupARN": "arn:aws:autoscaling:us-
west-2:123456789012:autoScalingGroup:283179a2-
f3ce-423d-93f6-66bb518232f7:autoScalingGroupName/my-asg"
        }
    ]
}
```

Per ulteriori informazioni, consulta la sezione Risoluzione dei problemi di Amazon Auto EC2 Scaling nella Amazon Auto EC2 Scaling User Guide.

Esempio 3: come descrivere il numero specificato di attività di dimensionamento

Per restituire un numero specifico di attività, utilizza l'opzione `--max-items`.

```
aws autoscaling describe-scaling-activities \
```

```
    --max-items 1
```

Output:

```
{
    "Activities": [
        {
            "ActivityId": "f9f2d65b-f1f2-43e7-b46d-d86756459699",
            "Description": "Launching a new EC2 instance: i-0d44425630326060f",
            "AutoScalingGroupName": "my-asg",
            "Cause": "At 2020-10-30T19:35:51Z a user request update of
 AutoScalingGroup constraints to min: 0, max: 16, desired: 16 changing the desired
 capacity from 0 to 16.  At 2020-10-30T19:36:07Z an instance was started in response
 to a difference between desired and actual capacity, increasing the capacity from 0
 to 16.",
            "StartTime": "2020-10-30T19:36:09.766Z",
            "EndTime": "2020-10-30T19:36:41Z",
            "StatusCode": "Successful",
            "Progress": 100,
            "Details": "{\"Subnet ID\":\"subnet-5ea0c127\",\"Availability Zone\":
\"us-west-2b\"}"
        }
    ]
}
```

Se l'output include un campo `NextToken`, significa che sono presenti più attività. Per ottenere le attività aggiuntive, utilizza il valore di questo campo con l'opzione `--starting-token` in una chiamata successiva come descritto di seguito.

```
aws autoscaling describe-scaling-activities \
    --starting-token Z3M3LMPEXAMPLE
```

Per ulteriori informazioni, consulta Verificare un'attività di scaling per un gruppo di Auto Scaling nella Amazon Auto EC2 Scaling User Guide.

- Per i dettagli sull'API, consulta Command DescribeScalingActivitiesReference AWS CLI .

`describe-scaling-process-types`

Il seguente esempio di codice mostra come utilizzare`describe-scaling-process-types`.

AWS CLI

Come descrivere i tipi di processo disponibili

Questo esempio descrive i tipi di processo disponibili.

```
aws autoscaling describe-scaling-process-types
```

Output:

```
{
    "Processes": [
        {
            "ProcessName": "AZRebalance"
        },
        {
            "ProcessName": "AddToLoadBalancer"
        },
        {
            "ProcessName": "AlarmNotification"
        },
        {
            "ProcessName": "HealthCheck"
        },
        {
            "ProcessName": "InstanceRefresh"
        },
        {
            "ProcessName": "Launch"
        },
        {
            "ProcessName": "ReplaceUnhealthy"
        },
        {
            "ProcessName": "ScheduledActions"
        },
        {
            "ProcessName": "Terminate"
        }
    ]
}
```

Per ulteriori informazioni, consulta Sospensione e ripresa dei processi di scalabilità nella Amazon Auto Scaling User Guide. EC2

- Per i dettagli sull'API, consulta Command Reference. DescribeScalingProcessTypesAWS CLI

`describe-scheduled-actions`

Il seguente esempio di codice mostra come utilizzare`describe-scheduled-actions`.

AWS CLI

Esempio 1: come descrivere tutte le azioni pianificate

Questo esempio descrive tutte le azioni pianificate.

```
aws autoscaling describe-scheduled-actions
```

Output:

```
{
    "ScheduledUpdateGroupActions": [
        {
            "AutoScalingGroupName": "my-asg",
            "ScheduledActionName": "my-recurring-action",
            "Recurrence": "30 0 1 1,6,12 *",
            "ScheduledActionARN": "arn:aws:autoscaling:us-
west-2:123456789012:scheduledUpdateGroupAction:8e86b655-b2e6-4410-8f29-
b4f094d6871c:autoScalingGroupName/my-asg:scheduledActionName/my-recurring-action",
            "StartTime": "2023-12-01T04:00:00Z",
            "Time": "2023-12-01T04:00:00Z",
            "MinSize": 1,
            "MaxSize": 6,
            "DesiredCapacity": 4,
            "TimeZone": "America/New_York"
        }
    ]
}
```

Per ulteriori informazioni, consulta Scheduled Scaling nella Amazon EC2 Auto Scaling User Guide.

Esempio 2: come descrivere le azioni pianificate per il gruppo specificato

Per descrivere le azioni pianificate per un gruppo Auto Scaling specifico, utilizza l'opzione `--auto-scaling-group-name`.

```
aws autoscaling describe-scheduled-actions \
    --auto-scaling-group-name my-asg
```

Output:

```
{
    "ScheduledUpdateGroupActions": [
        {
            "AutoScalingGroupName": "my-asg",
            "ScheduledActionName": "my-recurring-action",
            "Recurrence": "30 0 1 1,6,12 *",
            "ScheduledActionARN": "arn:aws:autoscaling:us-
west-2:123456789012:scheduledUpdateGroupAction:8e86b655-b2e6-4410-8f29-
b4f094d6871c:autoScalingGroupName/my-asg:scheduledActionName/my-recurring-action",
            "StartTime": "2023-12-01T04:00:00Z",
            "Time": "2023-12-01T04:00:00Z",
            "MinSize": 1,
            "MaxSize": 6,
            "DesiredCapacity": 4,
            "TimeZone": "America/New_York"
        }
    ]
}
```

Per ulteriori informazioni, consulta Scheduled Scaling nella Amazon EC2 Auto Scaling User Guide.

Esempio 3: come descrivere l'azione pianificata specificata

Per descrivere un'azione pianificata specifica, utilizza l'opzione `--scheduled-action-names`.

```
aws autoscaling describe-scheduled-actions \
    --scheduled-action-names my-recurring-action
```

Output:

```
{
    "ScheduledUpdateGroupActions": [
```

```
        {
            "AutoScalingGroupName": "my-asg",
            "ScheduledActionName": "my-recurring-action",
            "Recurrence": "30 0 1 1,6,12 *",
            "ScheduledActionARN": "arn:aws:autoscaling:us-
west-2:123456789012:scheduledUpdateGroupAction:8e86b655-b2e6-4410-8f29-
b4f094d6871c:autoScalingGroupName/my-asg:scheduledActionName/my-recurring-action",
            "StartTime": "2023-12-01T04:00:00Z",
            "Time": "2023-12-01T04:00:00Z",
            "MinSize": 1,
            "MaxSize": 6,
            "DesiredCapacity": 4,
            "TimeZone": "America/New_York"
        }
    ]
}
```

Per ulteriori informazioni, consulta Scheduled Scaling nella Amazon EC2 Auto Scaling User Guide.

Esempio 4: come descrivere le azioni pianificate con un'ora di inizio specificata

Per descrivere le azioni pianificate che iniziano a un'ora specifica, utilizza l'opzione `--start-time`.

```
aws autoscaling describe-scheduled-actions \
    --start-time "2023-12-01T04:00:00Z"
```

Output:

```
{
    "ScheduledUpdateGroupActions": [
        {
            "AutoScalingGroupName": "my-asg",
            "ScheduledActionName": "my-recurring-action",
            "Recurrence": "30 0 1 1,6,12 *",
            "ScheduledActionARN": "arn:aws:autoscaling:us-
west-2:123456789012:scheduledUpdateGroupAction:8e86b655-b2e6-4410-8f29-
b4f094d6871c:autoScalingGroupName/my-asg:scheduledActionName/my-recurring-action",
            "StartTime": "2023-12-01T04:00:00Z",
            "Time": "2023-12-01T04:00:00Z",
            "MinSize": 1,
```

```
            "MaxSize": 6,
            "DesiredCapacity": 4,
            "TimeZone": "America/New_York"
        }
    ]
}
```

Per ulteriori informazioni, consulta Scheduled Scaling nella Amazon EC2 Auto Scaling User Guide.

Esempio 5: come descrivere le azioni pianificate che terminano a un'ora specificata

Per descrivere le azioni pianificate che terminano a un'ora specifica, utilizza l'opzione `--end-time`.

```
aws autoscaling describe-scheduled-actions \
    --end-time "2023-12-01T04:00:00Z"
```

Output:

```
{
    "ScheduledUpdateGroupActions": [
        {
            "AutoScalingGroupName": "my-asg",
            "ScheduledActionName": "my-recurring-action",
            "Recurrence": "30 0 1 1,6,12 *",
            "ScheduledActionARN": "arn:aws:autoscaling:us-
west-2:123456789012:scheduledUpdateGroupAction:8e86b655-b2e6-4410-8f29-
b4f094d6871c:autoScalingGroupName/my-asg:scheduledActionName/my-recurring-action",
            "StartTime": "2023-12-01T04:00:00Z",
            "Time": "2023-12-01T04:00:00Z",
            "MinSize": 1,
            "MaxSize": 6,
            "DesiredCapacity": 4,
            "TimeZone": "America/New_York"
        }
    ]
}
```

Per ulteriori informazioni, consulta Scheduled Scaling nella Amazon EC2 Auto Scaling User Guide.

Esempio 6: come descrivere il numero specificato di azioni pianificate

Per restituire un numero specifico di azioni pianificate, utilizza l'opzione `--max-items`.

```
aws autoscaling describe-scheduled-actions \
    --auto-scaling-group-name my-asg \
    --max-items 1
```

Output:

```
{
    "ScheduledUpdateGroupActions": [
        {
            "AutoScalingGroupName": "my-asg",
            "ScheduledActionName": "my-recurring-action",
            "Recurrence": "30 0 1 1,6,12 *",
            "ScheduledActionARN": "arn:aws:autoscaling:us-
west-2:123456789012:scheduledUpdateGroupAction:8e86b655-b2e6-4410-8f29-
b4f094d6871c:autoScalingGroupName/my-asg:scheduledActionName/my-recurring-action",
            "StartTime": "2023-12-01T04:00:00Z",
            "Time": "2023-12-01T04:00:00Z",
            "MinSize": 1,
            "MaxSize": 6,
            "DesiredCapacity": 4,
            "TimeZone": "America/New_York"
        }
    ]
}
```

Se l'output include un campo `NextToken`, significa che sono presenti più azioni pianificate. Per ottenere le azioni pianificate aggiuntive, utilizza il valore di questo campo con l'opzione `--starting-token` in una chiamata successiva come descritto di seguito.

```
aws autoscaling describe-scheduled-actions \
    --auto-scaling-group-name my-asg \
    --starting-token Z3M3LMPEXAMPLE
```

Per ulteriori informazioni, consulta Scheduled Scaling nella Amazon EC2 Auto Scaling User Guide.

- Per i dettagli sull'API, consulta AWS CLI Command DescribeScheduledActionsReference.

`describe-tags`

Il seguente esempio di codice mostra come utilizzare`describe-tags`.

AWS CLI

Come descrivere tutti i tag

Questo esempio descrive tutti i tag.

```
aws autoscaling describe-tags
```

Output:

```
{
    "Tags": [
        {
            "ResourceType": "auto-scaling-group",
            "ResourceId": "my-asg",
            "PropagateAtLaunch": true,
            "Value": "Research",
            "Key": "Dept"
        },
        {
            "ResourceType": "auto-scaling-group",
            "ResourceId": "my-asg",
            "PropagateAtLaunch": true,
            "Value": "WebServer",
            "Key": "Role"
        }
    ]
}
```

Per ulteriori informazioni, consulta la sezione Tagging di gruppi e istanze di Auto Scaling nella Amazon Auto EC2 Scaling User Guide.

Esempio 2: come descrivere i tag per un gruppo specificato

Per descrivere i tag di un gruppo Auto Scaling specifico, utilizza l'opzione `--filters`.

```
aws autoscaling describe-tags --filters Name=auto-scaling-group,Values=my-asg
```

Per ulteriori informazioni, consulta la sezione Tagging di gruppi e istanze di Auto Scaling nella Amazon Auto EC2 Scaling User Guide.

Esempio 3: come descrivere il numero specificato di tag

Per restituire un numero specifico di tag, utilizza l'opzione `--max-items`.

```
aws autoscaling describe-tags \
    --max-items 1
```

Se l'output include un campo `NextToken`, significa che sono presenti più tag. Per ottenere i tag aggiuntivi, utilizza il valore di questo campo con l'opzione `--starting-token` in una chiamata successiva come descritto di seguito.

```
aws autoscaling describe-tags \
    --filters Name=auto-scaling-group,Values=my-asg \
    --starting-token Z3M3LMPEXAMPLE
```

Per ulteriori informazioni, consulta la sezione Tagging di gruppi e istanze di Auto Scaling nella Amazon Auto EC2 Scaling User Guide.

- Per i dettagli sull'API, consulta DescribeTagsCommand Reference.AWS CLI

`describe-termination-policy-types`

Il seguente esempio di codice mostra come utilizzare`describe-termination-policy-types`.

AWS CLI

Come descrivere i tipi di policy di terminazione disponibili

Questo esempio descrive i tipi di policy di terminazione disponibili.

```
aws autoscaling describe-termination-policy-types
```

Output:

```
{
    "TerminationPolicyTypes": [
        "AllocationStrategy",
        "ClosestToNextInstanceHour",
        "Default",
```

```
        "NewestInstance",
        "OldestInstance",
        "OldestLaunchConfiguration",
        "OldestLaunchTemplate"
    ]
}
```

Per ulteriori informazioni, consulta la sezione Controllo delle istanze di Auto Scaling che terminano durante la scalabilità nella Amazon Auto Scaling EC2 User Guide.

- Per i dettagli sull'API, consulta DescribeTerminationPolicyTypesCommand Reference.AWS CLI

`describe-warm-pool`

Il seguente esempio di codice mostra come utilizzare`describe-warm-pool`.

AWS CLI

Come descrivere un warm pool

In questo esempio viene descritto il warm pool del gruppo Auto Scaling specificato.

```
aws autoscaling describe-warm-pool \
    --auto-scaling-group-name my-asg
```

Output:

```
{
    "WarmPoolConfiguration": {
        "MinSize": 2,
        "PoolState": "Stopped"
    },
    "Instances": [
        {
            "InstanceId": "i-070a5bbc7e7f40dc5",
            "InstanceType": "t2.micro",
            "AvailabilityZone": "us-west-2c",
            "LifecycleState": "Warmed:Pending",
            "HealthStatus": "Healthy",
            "LaunchTemplate": {
                "LaunchTemplateId": "lt-00a731f6e9fa48610",
                "LaunchTemplateName": "my-template-for-auto-scaling",
                "Version": "6"
```

```
            }
        },
        {
            "InstanceId": "i-0b52f061814d3bd2d",
            "InstanceType": "t2.micro",
            "AvailabilityZone": "us-west-2b",
            "LifecycleState": "Warmed:Pending",
            "HealthStatus": "Healthy",
            "LaunchTemplate": {
                "LaunchTemplateId": "lt-00a731f6e9fa48610",
                "LaunchTemplateName": "my-template-for-auto-scaling",
                "Version": "6"
            }
        }
    ]
}
```

Per ulteriori informazioni, consulta Warm pools for Amazon EC2 Auto Scaling nella Amazon Auto EC2 Scaling User Guide.

- Per i dettagli sull'API, consulta AWS CLI Command DescribeWarmPoolReference.

`detach-instances`

Il seguente esempio di codice mostra come utilizzare`detach-instances`.

AWS CLI

Come scollegare un'istanza da un gruppo Auto Scaling

Questo esempio scollega l'istanza specificata dal gruppo Auto Scaling specificato.

```
aws autoscaling detach-instances \
    --instance-ids i-030017cfa84b20135 \
    --auto-scaling-group-name my-asg \
    --should-decrement-desired-capacity
```

Output:

```
{
    "Activities": [
        {
            "ActivityId": "5091cb52-547a-47ce-a236-c9ccbc2cb2c9",
```

```
            "AutoScalingGroupName": "my-asg",
            "Description": "Detaching EC2 instance: i-030017cfa84b20135",
            "Cause": "At 2020-10-31T17:35:04Z instance i-030017cfa84b20135 was
 detached in response to a user request, shrinking the capacity from 2 to 1.",
            "StartTime": "2020-04-12T15:02:16.179Z",
            "StatusCode": "InProgress",
            "Progress": 50,
            "Details": "{\"Subnet ID\":\"subnet-6194ea3b\",\"Availability Zone\":
\"us-west-2c\"}"
        }
    ]
}
```

- Per i dettagli sull'API, consulta DetachInstances AWS CLICommand Reference.

`detach-load-balancer-target-groups`

Il seguente esempio di codice mostra come utilizzare`detach-load-balancer-target-groups`.

AWS CLI

Come scollegare un gruppo di destinazione del bilanciatore del carico da un gruppo Auto Scaling

In questo esempio, il gruppo di destinazione del bilanciatore del carico specificato viene scollegato dal gruppo Auto Scaling specificato.

```
aws autoscaling detach-load-balancer-target-groups \
    --auto-scaling-group-name my-asg \
    --target-group-arns arn:aws:elasticloadbalancing:us-
west-2:123456789012:targetgroup/my-targets/73e2d6bc24d8a067
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Collegare un sistema di bilanciamento del carico al gruppo Auto Scaling nella Amazon Auto Scaling User EC2 Guide.

- Per i dettagli sull'API, consulta DetachLoadBalancerTargetGroupsCommand Reference.AWS CLI

`detach-load-balancers`

Il seguente esempio di codice mostra come utilizzare`detach-load-balancers`.

AWS CLI

Come scollegare un Classic Load Balancer da un gruppo Auto Scaling

Questo esempio scollega il Classic Load Balancer specificato dal gruppo Auto Scaling specificato.

```
aws autoscaling detach-load-balancers \
    --load-balancer-names my-load-balancer \
    --auto-scaling-group-name my-asg
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Collegare un sistema di bilanciamento del carico al gruppo Auto Scaling nella Amazon Auto Scaling User EC2 Guide.

- Per i dettagli sull'API, consulta DetachLoadBalancersCommand Reference.AWS CLI

`disable-metrics-collection`

Il seguente esempio di codice mostra come utilizzare`disable-metrics-collection`.

AWS CLI

Come disabilitare la raccolta delle metriche per un gruppo Auto Scaling

In questo esempio viene disabilitata la raccolta della metrica `GroupDesiredCapacity` per il gruppo Auto Scaling specificato.

```
aws autoscaling disable-metrics-collection \
    --auto-scaling-group-name my-asg \
    --metrics GroupDesiredCapacity
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta i CloudWatch parametri di monitoraggio per i gruppi e le istanze di Auto Scaling nella Amazon Auto EC2 Scaling User Guide.

- Per i dettagli sull'API, consulta DisableMetricsCollectionCommand Reference.AWS CLI

`enable-metrics-collection`

Il seguente esempio di codice mostra come utilizzare`enable-metrics-collection`.

AWS CLI

Esempio 1: come abilitare la raccolta delle metriche di un gruppo Auto Scaling

Questo esempio abilita la raccolta dei dati per il gruppo Auto Scaling specificato.

```
aws autoscaling enable-metrics-collection \
    --auto-scaling-group-name my-asg \
    --granularity "1Minute"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta i CloudWatch parametri di monitoraggio per i gruppi e le istanze di Auto Scaling nella Amazon Auto EC2 Scaling User Guide.

Esempio 2: come raccogliere i dati della metrica specificata per un gruppo Auto Scaling

Per raccogliere i dati per una metrica specifica, utilizza l'opzione `--metrics`.

```
aws autoscaling enable-metrics-collection \
    --auto-scaling-group-name my-asg \
    --metrics GroupDesiredCapacity --granularity "1Minute"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta i CloudWatch parametri di monitoraggio per i gruppi e le istanze di Auto Scaling nella Amazon Auto EC2 Scaling User Guide.

- Per i dettagli sull'API, consulta EnableMetricsCollectionCommand Reference.AWS CLI

`enter-standby`

Il seguente esempio di codice mostra come utilizzare`enter-standby`.

AWS CLI

Come spostare le istanze in modalità standby

Questo esempio mette l'istanza specificata in modalità standby. Ciò è utile per aggiornare o risolvere i problemi di un'istanza attualmente in servizio.

```
aws autoscaling enter-standby \
    --instance-ids i-061c63c5eb45f0416 \
```

```
    --auto-scaling-group-name my-asg \
    --should-decrement-desired-capacity
```

Output:

```
{
    "Activities": [
        {
            "ActivityId": "ffa056b4-6ed3-41ba-ae7c-249dfae6eba1",
            "AutoScalingGroupName": "my-asg",
            "Description": "Moving EC2 instance to Standby: i-061c63c5eb45f0416",
            "Cause": "At 2020-10-31T20:31:00Z instance i-061c63c5eb45f0416 was moved
 to standby in response to a user request, shrinking the capacity from 1 to 0.",
            "StartTime": "2020-10-31T20:31:00.949Z",
            "StatusCode": "InProgress",
            "Progress": 50,
            "Details": "{\"Subnet ID\":\"subnet-6194ea3b\",\"Availability Zone\":
\"us-west-2c\"}"
        }
    ]
}
```

Per ulteriori informazioni, consulta il ciclo di vita delle istanze di Amazon EC2 Auto Scaling nella Amazon Auto Scaling EC2 User Guide.

- Per i dettagli sull'API, consulta EnterStandbyCommand Reference.AWS CLI

`execute-policy`

Il seguente esempio di codice mostra come utilizzare`execute-policy`.

AWS CLI

Come eseguire una policy di dimensionamento

In questo esempio viene descritta la policy di dimensionamento denominata `my-step-scale-out-policy` per il gruppo Auto Scaling specificato.

```
aws autoscaling execute-policy \
    --auto-scaling-group-name my-asg \
    --policy-name  my-step-scale-out-policy \
    --metric-value 95 \
```

```
    --breach-threshold 80
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Step and simple scaling policies nella Amazon EC2 Auto Scaling User Guide.

- Per i dettagli sull'API, consulta AWS CLI Command ExecutePolicyReference.

`exit-standby`

Il seguente esempio di codice mostra come utilizzare`exit-standby`.

AWS CLI

Come impostare l'uscita delle istanze dalla modalità standby

Questo esempio imposta l'uscita dell'istanza specificata dalla modalità standby.

```
aws autoscaling exit-standby \
    --instance-ids i-061c63c5eb45f0416 \
    --auto-scaling-group-name my-asg
```

Output:

```
{
    "Activities": [
        {
            "ActivityId": "142928e1-a2dc-453a-9b24-b85ad6735928",
            "AutoScalingGroupName": "my-asg",
            "Description": "Moving EC2 instance out of Standby:
 i-061c63c5eb45f0416",
            "Cause": "At 2020-10-31T20:32:50Z instance i-061c63c5eb45f0416 was moved
 out of standby in response to a user request, increasing the capacity from 0 to
 1.",
            "StartTime": "2020-10-31T20:32:50.222Z",
            "StatusCode": "PreInService",
            "Progress": 30,
            "Details": "{\"Subnet ID\":\"subnet-6194ea3b\",\"Availability Zone\":
\"us-west-2c\"}"
        }
    ]
}
```

Per ulteriori informazioni, consulta Rimuovere temporaneamente le istanze dal gruppo Auto Scaling nella Amazon Auto Scaling EC2 User Guide.

- Per i dettagli sull'API, consulta Command ExitStandbyReference AWS CLI .

`put-lifecycle-hook`

Il seguente esempio di codice mostra come utilizzare`put-lifecycle-hook`.

AWS CLI

Esempio 1: come creare un hook del ciclo di vita

Questo esempio crea un hook del ciclo di vita che verrà invocato su tutte le nuove istanze avviate, con un timeout di 4800 secondi. Ciò è utile per mantenere le istanze in uno stato di attesa fino al termine degli script dei dati utente o per richiamare una funzione Lambda utilizzando AWS . EventBridge

```
aws autoscaling put-lifecycle-hook \
    --auto-scaling-group-name my-asg \
    --lifecycle-hook-name my-launch-hook \
    --lifecycle-transition autoscaling:EC2_INSTANCE_LAUNCHING \
    --heartbeat-timeout 4800
```

Questo comando non produce alcun output. Se esiste già un hook del ciclo di vita con lo stesso nome, verrà sovrascritto dal nuovo hook del ciclo di vita.

Per ulteriori informazioni, consulta i riferimenti al ciclo di vita di Amazon EC2 Auto Scaling nella Amazon Auto EC2 Scaling User Guide.

Esempio 2: come inviare un messaggio e-mail tramite Amazon SNS per comunicare le transizioni di stato dell'istanza

Questo esempio crea un hook del ciclo di vita con l'argomento Amazon SNS e il ruolo IAM da utilizzare per ricevere una notifica all'avvio dell'istanza.

```
aws autoscaling put-lifecycle-hook \
    --auto-scaling-group-name my-asg \
    --lifecycle-hook-name my-launch-hook \
    --lifecycle-transition autoscaling:EC2_INSTANCE_LAUNCHING \
    --notification-target-arn arn:aws:sns:us-west-2:123456789012:my-sns-topic \
```

```
    --role-arn arn:aws:iam::123456789012:role/my-auto-scaling-role
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta i riferimenti al ciclo di vita di Amazon EC2 Auto Scaling nella Amazon Auto EC2 Scaling User Guide.

Esempio 3: come pubblicare un messaggio in una coda Amazon SQS

Questo esempio crea un hook del ciclo di vita che pubblica un messaggio con metadati nella coda Amazon SQS specificata.

```
aws autoscaling put-lifecycle-hook \
    --auto-scaling-group-name my-asg \
    --lifecycle-hook-name my-launch-hook \
    --lifecycle-transition autoscaling:EC2_INSTANCE_LAUNCHING \
    --notification-target-arn arn:aws:sqs:us-west-2:123456789012:my-sqs-queue \
    --role-arn arn:aws:iam::123456789012:role/my-notification-role \
    --notification-metadata "SQS message metadata"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta i riferimenti al ciclo di vita di Amazon EC2 Auto Scaling nella Amazon Auto EC2 Scaling User Guide.

- Per i dettagli sull'API, consulta Command Reference. PutLifecycleHookAWS CLI

`put-notification-configuration`

Il seguente esempio di codice mostra come utilizzare`put-notification-configuration`.

AWS CLI

Come aggiungere una notifica

Questo esempio aggiunge la notifica specificata al gruppo Auto Scaling specificato.

```
aws autoscaling put-notification-configuration \
    --auto-scaling-group-name my-asg \
    --topic-arn arn:aws:sns:us-west-2:123456789012:my-sns-topic \
    --notification-type autoscaling:TEST_NOTIFICATION
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Ricevere notifiche Amazon SNS quando il tuo gruppo Auto Scaling aumenta nella Amazon EC2 Auto Scaling User Guide.

- Per i dettagli sull'API, consulta PutNotificationConfigurationCommand Reference.AWS CLI

`put-scaling-policy`

Il seguente esempio di codice mostra come utilizzare`put-scaling-policy`.

AWS CLI

Come aggiungere una policy di dimensionamento con monitoraggio delle destinazioni a un gruppo Auto Scaling

L'esempio `put-scaling-policy` seguente applica una policy di dimensionamento con monitoraggio delle destinazioni al gruppo Auto Scaling specificato. L'output contiene i nomi ARNs e i nomi dei due CloudWatch allarmi creati per tuo conto. Se esiste già una policy di dimensionamento con lo stesso nome, tale policy verrà sovrascritta dalla nuova policy.

```
aws autoscaling put-scaling-policy --auto-scaling-group-name my-asg \
  --policy-name alb1000-target-tracking-scaling-policy \
  --policy-type TargetTrackingScaling \
  --target-tracking-configuration file://config.json
```

Contenuto di `config.json`:

```
{
     "TargetValue": 1000.0,
     "PredefinedMetricSpecification": {
          "PredefinedMetricType": "ALBRequestCountPerTarget",
          "ResourceLabel": "app/my-alb/778d41231b141a0f/targetgroup/my-alb-target-
group/943f017f100becff"
     }
}
```

Output:

```
{
     "PolicyARN": "arn:aws:autoscaling:region:account-id:scalingPolicy:228f02c2-
c665-4bfd-aaac-8b04080bea3c:autoScalingGroupName/my-asg:policyName/alb1000-target-
tracking-scaling-policy",
```

```
    "Alarms": [
        {
            "AlarmARN": "arn:aws:cloudwatch:region:account-id:alarm:TargetTracking-
my-asg-AlarmHigh-fc0e4183-23ac-497e-9992-691c9980c38e",
            "AlarmName": "TargetTracking-my-asg-AlarmHigh-
fc0e4183-23ac-497e-9992-691c9980c38e"
        },
        {
            "AlarmARN": "arn:aws:cloudwatch:region:account-id:alarm:TargetTracking-
my-asg-AlarmLow-61a39305-ed0c-47af-bd9e-471a352ee1a2",
            "AlarmName": "TargetTracking-my-asg-AlarmLow-61a39305-ed0c-47af-
bd9e-471a352ee1a2"
        }
    ]
}
```

Per altri esempi, consulta Example scaling policies for the AWS Command Line Interface (AWS CLI) nella Amazon Auto EC2 Scaling User Guide.

- Per i dettagli sull'API, consulta Command PutScalingPolicyReference AWS CLI .

`put-scheduled-update-group-action`

Il seguente esempio di codice mostra come utilizzare`put-scheduled-update-group-action`.

AWS CLI

Esempio 1: come aggiungere un'azione pianificata a un gruppo Auto Scaling

Questo esempio aggiunge l'azione pianificata specificata al gruppo Auto Scaling specificato.

```
aws autoscaling put-scheduled-update-group-action \
    --auto-scaling-group-name my-asg \
    --scheduled-action-name my-scheduled-action \
    --start-time "2023-05-12T08:00:00Z" \
    --min-size 2 \
    --max-size 6 \
    --desired-capacity 4
```

Questo comando non produce alcun output. Se esiste già un'azione pianificata con lo stesso nome, tale azione verrà sovrascritta dalla nuova azione pianificata.

Per altri esempi, consulta Scheduled Scaling nella Amazon EC2 Auto Scaling User Guide.

Esempio 2: come specificare una pianificazione ricorrente

Questo esempio crea un'azione di dimensionamento pianificata in base a una pianificazione ricorrente la cui esecuzione è programmata alle 00:30 del primo gennaio, giugno e dicembre di ogni anno.

```
aws autoscaling put-scheduled-update-group-action \
    --auto-scaling-group-name my-asg \
    --scheduled-action-name my-recurring-action \
    --recurrence "30 0 1 1,6,12 *" \
    --min-size 2 \
    --max-size 6 \
    --desired-capacity 4
```

Questo comando non produce alcun output. Se esiste già un'azione pianificata con lo stesso nome, tale azione verrà sovrascritta dalla nuova azione pianificata.

Per altri esempi, consulta Scheduled Scaling nella Amazon EC2 Auto Scaling User Guide.

- Per i dettagli sull'API, consulta AWS CLI Command PutScheduledUpdateGroupActionReference.

`put-warm-pool`

Il seguente esempio di codice mostra come utilizzare`put-warm-pool`.

AWS CLI

Come creare un warm pool

L'esempio seguente crea un warm pool per il gruppo Auto Scaling specificato.

```
aws autoscaling put-warm-pool \
    --auto-scaling-group-name my-asg \
    --min-size 2
```

Questo comando non produce alcun output. Se esiste già un warm pool, verrà aggiornato.

Per ulteriori informazioni, consulta Warm pools for Amazon EC2 Auto Scaling nella Amazon Auto EC2 Scaling User Guide.

- Per i dettagli sull'API, consulta AWS CLI Command PutWarmPoolReference.

`record-lifecycle-action-heartbeat`

Il seguente esempio di codice mostra come utilizzare`record-lifecycle-action-heartbeat`.

AWS CLI

Come registrare l'heartbeat di un'azione del ciclo di vita

Questo esempio registra l'heartbeat di un'azione del ciclo di vita per mantenere l'istanza in sospeso.

```
aws autoscaling record-lifecycle-action-heartbeat \
    --lifecycle-hook-name my-launch-hook \
    --auto-scaling-group-name my-asg \
    --lifecycle-action-token bcd2f1b8-9a78-44d3-8a7a-4dd07d7cf635
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta i riferimenti al ciclo di vita di Amazon EC2 Auto Scaling nella Amazon Auto EC2 Scaling User Guide.

- Per i dettagli sull'API, consulta Command Reference. RecordLifecycleActionHeartbeatAWS CLI

`resume-processes`

Il seguente esempio di codice mostra come utilizzare`resume-processes`.

AWS CLI

Come riprendere i processi in sospeso

Questo esempio riprende il processo di dimensionamento automatico specificato per il gruppo Auto Scaling specificato.

```
aws autoscaling resume-processes \
    --auto-scaling-group-name my-asg \
    --scaling-processes AlarmNotification
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Sospensione e ripresa dei processi di scalabilità nella Amazon Auto Scaling User Guide. EC2

- Per i dettagli sull'API, consulta Command Reference. ResumeProcessesAWS CLI

`rollback-instance-refresh`

Il seguente esempio di codice mostra come utilizzare`rollback-instance-refresh`.

AWS CLI

Come eseguire il rollback dell'aggiornamento di un'istanza

L'esempio `rollback-instance-refresh` seguente esegue il rollback un aggiornamento dell'istanza in corso per il gruppo Auto Scaling specificato.

```
aws autoscaling rollback-instance-refresh \
    --auto-scaling-group-name my-asg
```

Output:

```
{
    "InstanceRefreshId": "08b91cf7-8fa6-48af-b6a6-d227f40f1b9b"
}
```

Per ulteriori informazioni, consulta Annulla le modifiche con un rollback nella Amazon Auto EC2 Scaling User Guide.

- Per i dettagli sull'API, consulta Command RollbackInstanceRefreshReference AWS CLI .

`set-desired-capacity`

Il seguente esempio di codice mostra come utilizzare`set-desired-capacity`.

AWS CLI

Come impostare la capacità desiderata per un gruppo Auto Scaling

Questo esempio imposta la capacità desiderata per il gruppo Auto Scaling specificato.

```
aws autoscaling set-desired-capacity \
    --auto-scaling-group-name my-asg \
    --desired-capacity 2 \
    --honor-cooldown
```

In caso di esito positivo, il comando torna al prompt.

- Per i dettagli sull'API, consulta SetDesiredCapacity AWS CLICommand Reference.

`set-instance-health`

Il seguente esempio di codice mostra come utilizzare`set-instance-health`.

AWS CLI

Come impostare lo stato dell'integrità di un'istanza

In questo esempio, lo stato di integrità dell'istanza specificata viene impostato su `Unhealthy`.

```
aws autoscaling set-instance-health \
    --instance-id i-061c63c5eb45f0416 \
    --health-status Unhealthy
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta SetInstanceHealth AWS CLICommand Reference.

`set-instance-protection`

Il seguente esempio di codice mostra come utilizzare`set-instance-protection`.

AWS CLI

Esempio 1: come abilitare l'impostazione di protezione delle istanze per un'istanza

Questo esempio abilita la protezione dell'istanza specificata.

```
aws autoscaling set-instance-protection \
    --instance-ids i-061c63c5eb45f0416 \
    --auto-scaling-group-name my-asg --protected-from-scale-in
```

Questo comando non produce alcun output.

Esempio 2: come disabilitare l'impostazione di protezione delle istanze per un'istanza

Questo esempio disabilita la protezione dell'istanza specificata.

```
aws autoscaling set-instance-protection \
```

```
    --instance-ids i-061c63c5eb45f0416 \
    --auto-scaling-group-name my-asg \
    --no-protected-from-scale-in
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta SetInstanceProtection AWS CLICommand Reference.

`start-instance-refresh`

Il seguente esempio di codice mostra come utilizzare`start-instance-refresh`.

AWS CLI

Esempio 1: come avviare l'aggiornamento di un'istanza utilizzando i parametri della riga di comando

L'esempio `start-instance-refresh` seguente avvia l'aggiornamento di un'istanza utilizzando argomenti della riga di comando. Il parametro opzionale `preferences` specifica una `InstanceWarmup` di `60` secondi e una `MinHealthyPercentage` pari al `50`%.

```
aws autoscaling start-instance-refresh \
    --auto-scaling-group-name my-asg \
    --preferences '{"InstanceWarmup": 60, "MinHealthyPercentage": 50}'
```

Output:

```
{
    "InstanceRefreshId": "08b91cf7-8fa6-48af-b6a6-d227f40f1b9b"
}
```

Per ulteriori informazioni, consulta Start an istance refresh nella Amazon EC2 Auto Scaling User Guide.

Esempio 2: come avviare l'aggiornamento di un'istanza utilizzando un file JSON

L'esempio `start-instance-refresh` seguente avvia l'aggiornamento di un'istanza utilizzando un file JSON. È possibile specificare il gruppo Auto Scaling e definire la configurazione e le preferenze desiderate in un file JSON, come illustrato nell'esempio seguente.

```
aws autoscaling start-instance-refresh \
```

```
    --cli-input-json file://config.json
```

Contenuto di `config.json`:

```
{
    "AutoScalingGroupName": "my-asg",
    "DesiredConfiguration": {
        "LaunchTemplate": {
            "LaunchTemplateId": "lt-068f72b729example",
            "Version": "$Default"
        }
    },
    "Preferences": {
        "InstanceWarmup": 60,
        "MinHealthyPercentage": 50,
        "AutoRollback": true,
        "ScaleInProtectedInstances": Ignore,
        "StandbyInstances": Terminate
    }
}
```

Output:

```
{
    "InstanceRefreshId": "08b91cf7-8fa6-48af-b6a6-d227f40f1b9b"
}
```

Per ulteriori informazioni, consulta Start an istance refresh nella Amazon EC2 Auto Scaling User Guide.

- Per i dettagli sull'API, consulta AWS CLI Command StartInstanceRefreshReference.

`suspend-processes`

Il seguente esempio di codice mostra come utilizzare`suspend-processes`.

AWS CLI

Come sospendere i processi di dimensionamento automatico

Questo esempio sospende il processo di dimensionamento specificato per il gruppo Auto Scaling specificato.

```
aws autoscaling suspend-processes \
    --auto-scaling-group-name my-asg \
    --scaling-processes AlarmNotification
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Sospensione e ripresa dei processi di scalabilità nella Amazon Auto Scaling User Guide. EC2

- Per i dettagli sull'API, consulta Command Reference. SuspendProcessesAWS CLI

`terminate-instance-in-auto-scaling-group`

Il seguente esempio di codice mostra come utilizzare`terminate-instance-in-auto-scaling-group`.

AWS CLI

Come terminare un'istanza in un gruppo Auto Scaling

Questo esempio termina l'istanza specificata nel gruppo Auto Scaling specificato senza aggiornare la dimensione del gruppo. Amazon EC2 Auto Scaling avvia un'istanza sostitutiva al termine dell'istanza specificata.

```
aws autoscaling terminate-instance-in-auto-scaling-group \
    --instance-id i-061c63c5eb45f0416 \
    --no-should-decrement-desired-capacity
```

Output:

```
{
    "Activities": [
        {
            "ActivityId": "8c35d601-793c-400c-fcd0-f64a27530df7",
            "AutoScalingGroupName": "my-asg",
            "Description": "Terminating EC2 instance: i-061c63c5eb45f0416",
            "Cause": "",
            "StartTime": "2020-10-31T20:34:25.680Z",
            "StatusCode": "InProgress",
            "Progress": 0,
            "Details": "{\"Subnet ID\":\"subnet-6194ea3b\",\"Availability Zone\":
\"us-west-2c\"}"
```

```
        }
    ]
}
```

- Per i dettagli sull'API, consulta Command TerminateInstanceInAutoScalingGroupReference AWS CLI .

`update-auto-scaling-group`

Il seguente esempio di codice mostra come utilizzare`update-auto-scaling-group`.

AWS CLI

Esempio 1: come aggiornare i limiti di dimensione di un gruppo Auto Scaling

Questo esempio aggiorna il gruppo Auto Scaling specificato in base a una dimensione minima di 2 e una dimensione massima di 10.

```
aws autoscaling update-auto-scaling-group \
    --auto-scaling-group-name my-asg \
    --min-size 2 \
    --max-size 10
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Impostazione dei limiti di capacità per il tuo gruppo Auto Scaling nella Amazon Auto EC2 Scaling User Guide.

Esempio 2: come aggiungere i controlli dell'integrità di Elastic Load Balancing e specificare le zone di disponibilità e le sottoreti da utilizzare

Questo esempio aggiorna il gruppo Auto Scaling specificato per aggiungere i controlli dell'integrità di Elastic Load Balancing. Questo comando aggiorna anche il valore di `--vpc-zone-identifier` con un elenco di sottoreti IDs in più zone di disponibilità.

```
aws autoscaling update-auto-scaling-group \
    --auto-scaling-group-name my-asg \
    --health-check-type ELB \
    --health-check-grace-period 600 \
    --vpc-zone-identifier "subnet-5ea0c127,subnet-6194ea3b,subnet-c934b782"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Elastic Load Balancing e Amazon Auto EC2 Scaling nella Amazon Auto EC2 Scaling User Guide.

Esempio 3: come aggiornare il gruppo di posizionamento e la policy di terminazione

Questo esempio aggiorna il gruppo di posizionamento e la policy di terminazione da utilizzare.

```
aws autoscaling update-auto-scaling-group \
    --auto-scaling-group-name my-asg \
    --placement-group my-placement-group \
    --termination-policies "OldestInstance"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta i gruppi Auto Scaling nella Amazon Auto EC2 Scaling User Guide.

Esempio 4: come utilizzare la versione più recente del modello di avvio

Questo esempio aggiorna il gruppo Auto Scaling specificato per utilizzare la versione più recente del modello di avvio specificato.

```
aws autoscaling update-auto-scaling-group \
    --auto-scaling-group-name my-asg \
    --launch-template LaunchTemplateId=lt-1234567890abcde12,Version='$Latest'
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Launch templates nella Amazon EC2 Auto Scaling User Guide.

Esempio 5: come utilizzare una versione specifica del modello di avvio

Questo esempio aggiorna il gruppo Auto Scaling specificato per utilizzare una versione specifica di un modello di avvio anziché la versione più recente o predefinita.

```
aws autoscaling update-auto-scaling-group \
    --auto-scaling-group-name my-asg \
    --launch-template LaunchTemplateName=my-template-for-auto-scaling,Version='2'
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Launch templates nella Amazon EC2 Auto Scaling User Guide.

Esempio 6: come definire una policy di istanze miste e abilitare il ribilanciamento della capacità

Questo esempio aggiorna il gruppo Auto Scaling specificato per utilizzare una policy di istanze miste e consentire il ribilanciamento della capacità. Questa struttura consente di specificare gruppi con capacità spot e on demand e utilizzare modelli di avvio diversi per architetture diverse.

```
aws autoscaling update-auto-scaling-group \
    --cli-input-json file://~/config.json
```

Contenuto di `config.json`:

```
{
    "AutoScalingGroupName": "my-asg",
    "CapacityRebalance": true,
    "MixedInstancesPolicy": {
        "LaunchTemplate": {
            "LaunchTemplateSpecification": {
                "LaunchTemplateName": "my-launch-template-for-x86",
                "Version": "$Latest"
            },
            "Overrides": [
                {
                    "InstanceType": "c6g.large",
                    "LaunchTemplateSpecification": {
                        "LaunchTemplateName": "my-launch-template-for-arm",
                        "Version": "$Latest"
                    }
                },
                {
                    "InstanceType": "c5.large"
                },
                {
                    "InstanceType": "c5a.large"
                }
            ]
        },
        "InstancesDistribution": {
            "OnDemandPercentageAboveBaseCapacity": 50,
            "SpotAllocationStrategy": "capacity-optimized"
        }
    }
}
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta i gruppi di Auto Scaling con più tipi di istanze e opzioni di acquisto nella Amazon Auto EC2 Scaling User Guide.

- Per i dettagli sull'API, consulta AWS CLI Command UpdateAutoScalingGroupReference.

Esempi di Piani di Auto Scaling che utilizzano AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando i piani AWS Command Line Interface con Auto Scaling.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`create-scaling-plan`

Il seguente esempio di codice mostra come utilizzare`create-scaling-plan`.

AWS CLI

Come creare un piano di dimensionamento

L'esempio `create-scaling-plan` seguente crea un piano di dimensionamento denominato `my-scaling-plan` utilizzando un file JSON già creato (denominato config.json). La struttura del piano di dimensionamento include un'istruzione di dimensionamento per un gruppo Auto Scaling denominato `my-asg`. Specifica la proprietà `TagFilters` come origine dell'applicazione e abilita il dimensionamento predittivo e il dimensionamento dinamico.

```
aws autoscaling-plans create-scaling-plan \
    --scaling-plan-name my-scaling-plan \
```

```
    --cli-input-json file://~/config.json
```

Contenuto del file `config.json`:

```
{
    "ApplicationSource": {
        "TagFilters": [
            {
                "Key": "purpose",
                "Values": [
                    "my-application"
                ]
            }
        ]
    },
    "ScalingInstructions": [
        {
            "ServiceNamespace": "autoscaling",
            "ResourceId": "autoScalingGroup/my-asg",
            "ScalableDimension": "autoscaling:autoScalingGroup:DesiredCapacity",
            "ScheduledActionBufferTime": 300,
            "PredictiveScalingMaxCapacityBehavior":
 "SetForecastCapacityToMaxCapacity",
            "PredictiveScalingMode": "ForecastAndScale",
            "PredefinedLoadMetricSpecification": {
                "PredefinedLoadMetricType": "ASGTotalCPUUtilization"
            },
            "ScalingPolicyUpdateBehavior": "ReplaceExternalPolicies",
            "MinCapacity": 1,
            "MaxCapacity": 4,
            "TargetTrackingConfigurations": [
                {
                    "PredefinedScalingMetricSpecification": {
                        "PredefinedScalingMetricType": "ASGAverageCPUUtilization"
                    },
                    "TargetValue": 50
                }
            ]
        }
    ]
}
```

Output:

```
{
"ScalingPlanVersion": 1
}
```

Per ulteriori informazioni, consulta la Guida per l'utente di AWS Auto Scaling.

- Per i dettagli sull'API, consulta CreateScalingPlan AWS CLICommand Reference.

`delete-scaling-plan`

Il seguente esempio di codice mostra come utilizzare`delete-scaling-plan`.

AWS CLI

Come eliminare un piano di dimensionamento

L'esempio `delete-scaling-plan` seguente elimina il piano di dimensionamento specificato.

```
aws autoscaling-plans delete-scaling-plan \
    --scaling-plan-name my-scaling-plan \
    --scaling-plan-version 1
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta la Guida per l'utente di AWS Auto Scaling.

- Per i dettagli sull'API, consulta DeleteScalingPlan AWS CLICommand Reference.

`describe-scaling-plan-resources`

Il seguente esempio di codice mostra come utilizzare`describe-scaling-plan-resources`.

AWS CLI

Come descrivere le risorse dimensionabili per un piano di dimensionamento

L'esempio `describe-scaling-plan-resources` seguente visualizza i dettagli della singola risorsa dimensionabile (un gruppo Auto Scaling) associata al piano di dimensionamento specificato.

```
aws autoscaling-plans describe-scaling-plan-resources \
    --scaling-plan-name my-scaling-plan \
```

```
    --scaling-plan-version 1
```

Output:

```
{
    "ScalingPlanResources": [
        {
            "ScalableDimension": "autoscaling:autoScalingGroup:DesiredCapacity",
            "ScalingPlanVersion": 1,
            "ResourceId": "autoScalingGroup/my-asg",
            "ScalingStatusCode": "Active",
            "ScalingStatusMessage": "Target tracking scaling policies have been
 applied to the resource.",
            "ScalingPolicies": [
                {
                    "PolicyName": "AutoScaling-my-asg-b1ab65ae-4be3-4634-bd64-
c7471662b251",
                    "PolicyType": "TargetTrackingScaling",
                    "TargetTrackingConfiguration": {
                        "PredefinedScalingMetricSpecification": {
                            "PredefinedScalingMetricType":
 "ALBRequestCountPerTarget",
                            "ResourceLabel": "app/my-alb/f37c06a68c1748aa/
targetgroup/my-target-group/6d4ea56ca2d6a18d"
                        },
                        "TargetValue": 40.0
                    }
                }
            ],
            "ServiceNamespace": "autoscaling",
            "ScalingPlanName": "my-scaling-plan"
        }
    ]
}
```

Per ulteriori informazioni, consulta Che cos'è l' AWS Auto Scaling? nella Guida per l'utente di AWS Auto Scaling.

- Per i dettagli sull'API, consulta DescribeScalingPlanResources AWS CLICommand Reference.

`describe-scaling-plans`

Il seguente esempio di codice mostra come utilizzare`describe-scaling-plans`.

AWS CLI

Come descrivere un piano di dimensionamento

L'esempio `describe-scaling-plans` seguente mostra i dettagli del piano di dimensionamento specificato.

```
aws autoscaling-plans describe-scaling-plans \
    --scaling-plan-names scaling-plan-with-asg-and-ddb
```

Output:

```
{
    "ScalingPlans": [
        {
            "LastMutatingRequestTime": 1565388443.963,
            "ScalingPlanVersion": 1,
            "CreationTime": 1565388443.963,
            "ScalingInstructions": [
                {
                    "ScalingPolicyUpdateBehavior": "ReplaceExternalPolicies",
                    "ScalableDimension":
 "autoscaling:autoScalingGroup:DesiredCapacity",
                    "TargetTrackingConfigurations": [
                        {
                            "PredefinedScalingMetricSpecification": {
                                "PredefinedScalingMetricType":
 "ASGAverageCPUUtilization"
                            },
                            "TargetValue": 50.0,
                            "EstimatedInstanceWarmup": 300,
                            "DisableScaleIn": false
                        }
                    ],
                    "ResourceId": "autoScalingGroup/my-asg",
                    "DisableDynamicScaling": false,
                    "MinCapacity": 1,
                    "ServiceNamespace": "autoscaling",
                    "MaxCapacity": 10
                },
                {
                    "ScalingPolicyUpdateBehavior": "ReplaceExternalPolicies",
                    "ScalableDimension": "dynamodb:table:ReadCapacityUnits",
```

```
                    "TargetTrackingConfigurations": [
                        {
                            "PredefinedScalingMetricSpecification": {
                                "PredefinedScalingMetricType":
"DynamoDBReadCapacityUtilization"
                            },
                            "TargetValue": 50.0,
                            "ScaleInCooldown": 60,
                            "DisableScaleIn": false,
                            "ScaleOutCooldown": 60
                        }
                    ],
                    "ResourceId": "table/my-table",
                    "DisableDynamicScaling": false,
                    "MinCapacity": 5,
                    "ServiceNamespace": "dynamodb",
                    "MaxCapacity": 10000
                },
                {
                    "ScalingPolicyUpdateBehavior": "ReplaceExternalPolicies",
                    "ScalableDimension": "dynamodb:table:WriteCapacityUnits",
                    "TargetTrackingConfigurations": [
                        {
                            "PredefinedScalingMetricSpecification": {
                                "PredefinedScalingMetricType":
"DynamoDBWriteCapacityUtilization"
                            },
                            "TargetValue": 50.0,
                            "ScaleInCooldown": 60,
                            "DisableScaleIn": false,
                            "ScaleOutCooldown": 60
                        }
                    ],
                    "ResourceId": "table/my-table",
                    "DisableDynamicScaling": false,
                    "MinCapacity": 5,
                    "ServiceNamespace": "dynamodb",
                    "MaxCapacity": 10000
                }
            ],
            "ApplicationSource": {
                "TagFilters": [
                    {
                        "Values": [
```

```
                            "my-application-id"
                        ],
                        "Key": "application"
                    }
                ]
            },
            "StatusStartTime": 1565388455.836,
            "ScalingPlanName": "scaling-plan-with-asg-and-ddb",
            "StatusMessage": "Scaling plan has been created and applied to all
 resources.",
            "StatusCode": "Active"
        }
    ]
}
```

Per ulteriori informazioni, consulta Che cos'è l' AWS Auto Scaling? nella Guida per l'utente di AWS Auto Scaling.

- Per i dettagli sull'API, consulta DescribeScalingPlans AWS CLICommand Reference.

`get-scaling-plan-resource-forecast-data`

Il seguente esempio di codice mostra come utilizzare`get-scaling-plan-resource-forecast-data`.

AWS CLI

Come recuperare i dati di previsione del carico

Questo esempio recupera i dati di previsione del carico per una risorsa dimensionabile (un gruppo Auto Scaling) associata al piano di dimensionamento specificato.

```
aws autoscaling-plans get-scaling-plan-resource-forecast-data \
    --scaling-plan-name my-scaling-plan \
    --scaling-plan-version 1 \
    --service-namespace "autoscaling" \
    --resource-id autoScalingGroup/my-asg \
    --scalable-dimension "autoscaling:autoScalingGroup:DesiredCapacity" \
    --forecast-data-type "LoadForecast" \
    --start-time "2019-08-30T00:00:00Z" \
    --end-time "2019-09-06T00:00:00Z"
```

Output:

```
{
    "Datapoints": [...]
}
```

Per ulteriori informazioni, consulta Cos'è l' AWS Auto Scaling nella Guida per l'utente AWS di Auto Scaling.

- Per i dettagli sull'API, consulta AWS CLI Command GetScalingPlanResourceForecastDataReference.

`update-scaling-plan`

Il seguente esempio di codice mostra come utilizzareu`pdate-scaling-plan`.

AWS CLI

Come aggiornare un piano di dimensionamento

L'esempio `update-scaling-plan` seguente modifica la metrica di dimensionamento per un gruppo Auto Scaling nel piano di dimensionamento specificato.

```
aws autoscaling-plans update-scaling-plan \
    --scaling-plan-name my-scaling-plan \
    --scaling-plan-version 1 \
    --scaling-instructions
 '{"ScalableDimension":"autoscaling:autoScalingGroup:DesiredCapacity","ResourceId":"autoScal
my-asg","ServiceNamespace":"autoscaling","TargetTrackingConfigurations":
[{"PredefinedScalingMetricSpecification":
 {"PredefinedScalingMetricType":"ALBRequestCountPerTarget","ResourceLabel":"app/my-
alb/f37c06a68c1748aa/targetgroup/my-target-
group/6d4ea56ca2d6a18d"},"TargetValue":40.0}],"MinCapacity": 1,"MaxCapacity": 10}'
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Che cos'è l' AWS Auto Scaling? nella Guida per l'utente di AWS Auto Scaling.

- Per i dettagli sull'API, consulta UpdateScalingPlan AWS CLICommand Reference.

AWS Backup esempi utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with AWS Backup.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`create-backup-plan`

Il seguente esempio di codice mostra come utilizzare`create-backup-plan`.

AWS CLI

Come creare un piano di backup

L'esempio `create-backup-plan` seguente crea il piano di backup specificato con una conservazione di 35 giorni.

```
aws backup create-backup-plan \
--backup-plan "{\"BackupPlanName\":\"Example-Backup-Plan\",\"Rules\":[{\"RuleName\":
\"DailyBackups\",\"ScheduleExpression\":\"cron(0 5 ? * * *)\",\"StartWindowMinutes
\":480,\"TargetBackupVaultName\":\"Default\",\"Lifecycle\":{\"DeleteAfterDays
\":35}}]}"
```

Output:

```
{
    "BackupPlanId": "1fa3895c-a7f5-484a-a371-2dd6a1a9f729",
    "BackupPlanArn": "arn:aws:backup:us-west-2:123456789012:backup-plan:1fa3895c-
a7f5-484a-a371-2dd6a1a9f729",
    "CreationDate": 1568928754.747,
```

```
    "VersionId": "ZjQ2ZTI5YWQtZDg5Yi00MzYzLWJmZTAtMDI1MzhlMDhjYjEz"
}
```

Per ulteriori informazioni, consulta Creazione di un piano di backup nella Guida per sviluppatori di AWS .

- Per i dettagli sull'API, consulta CreateBackupPlan AWS CLICommand Reference.

`create-backup-vault`

Il seguente esempio di codice mostra come utilizzare`create-backup-vault`.

AWS CLI

Come creare un vault di backup

L'esempio `create-backup-vault` seguente crea una vault di backup con il nome specificato.

```
aws backup create-backup-vault
    --backup-vault-name sample-vault
```

Questo comando non produce alcun output. Output:

```
{
    "BackupVaultName": "sample-vault",
    "BackupVaultArn": "arn:aws:backup:us-west-2:123456789012:backup-vault:sample-
vault",
    "CreationDate": 1568928338.385
}
```

Per ulteriori informazioni, consulta Creazione di un vault di backup nella Guida per sviluppatori di AWS .

- Per i dettagli sull'API, consulta CreateBackupVault AWS CLICommand Reference.

`get-backup-plan-from-template`

Il seguente esempio di codice mostra come utilizzare`get-backup-plan-from-template`.

AWS CLI

Come ottenere un piano di backup esistente da un modello

L'esempio `get-backup-plan-from-template` seguente ottiene un piano di backup esistente da un modello che specifica un backup giornaliero con una conservazione di 35 giorni.

```
aws backup get-backup-plan-from-template \
    --backup-plan-template-id "87c0c1ef-254d-4180-8fef-2e76a2c38aaa"
```

Output:

```
{
    "BackupPlanDocument": {
        "Rules": [
            {
                "RuleName": "DailyBackups",
                "ScheduleExpression": "cron(0 5 ? * * *)",
                "StartWindowMinutes": 480,
                "Lifecycle": {
                    "DeleteAfterDays": 35
                }
            }
        ]
    }
}
```

Per ulteriori informazioni, consulta Creazione di un piano di backup nella Guida per sviluppatori di AWS .

- Per i dettagli sull'API, consulta GetBackupPlanFromTemplate AWS CLICommand Reference.

`get-backup-plan`

Il seguente esempio di codice mostra come utilizzare`get-backup-plan`.

AWS CLI

Come ottenere i dettagli di un piano di backup

L'esempio `get-backup-plan` seguente mostra i dettagli del piano di backup specificato.

```
aws backup get-backup-plan \
    --backup-plan-id "fcbf5d8f-bd77-4f3a-9c97-f24fb3d373a5"
```

Output:

```
{
    "BackupPlan": {
        "BackupPlanName": "Example-Backup-Plan",
        "Rules": [
            {
                "RuleName": "DailyBackups",
                "TargetBackupVaultName": "Default",
                "ScheduleExpression": "cron(0 5 ? * * *)",
                "StartWindowMinutes": 480,
                "CompletionWindowMinutes": 10080,
                "Lifecycle": {
                    "DeleteAfterDays": 35
                },
                "RuleId": "70e0ccdc-e9df-4e83-82ad-c1e5a9471cc3"
            }
        ]
    },
    "BackupPlanId": "fcbf5d8f-bd77-4f3a-9c97-f24fb3d373a5",
    "BackupPlanArn": "arn:aws:backup:us-west-2:123456789012:backup-plan:fcbf5d8f-
bd77-4f3a-9c97-f24fb3d373a5",
    "VersionId": "NjQ2ZTZkODktMGVhNy00MmQ0LWE4YjktZTkwNTQ3OTkyYTcw",
    "CreationDate": 1568926091.57
}
```

Per ulteriori informazioni, consulta Creazione di un piano di backup nella Guida per sviluppatori di AWS .

- Per i dettagli sull'API, consulta GetBackupPlan AWS CLICommand Reference.

`list-backup-jobs`

Il seguente esempio di codice mostra come utilizzare`list-backup-jobs`.

AWS CLI

Esempio 1: come elencare tutti i processi di backup

L'`list-backup-jobs`esempio seguente restituisce i metadati relativi ai processi di backup nel tuo AWS account.

```
aws backup list-backup-jobs
```

Output:

```
{
    "BackupJobs": [
        {
            "BackupJobId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
            "BackupVaultName": "Default",
            "BackupVaultArn": "arn:aws:backup:us-west-2:123456789012:backup-
vault:Default",
            "ResourceArn": "arn:aws:ec2:us-west-2:123456789012:instance/
i-12345678901234567",
            "CreationDate": 1600721892.929,
            "State": "CREATED",
            "PercentDone": "0.0",
            "IamRoleArn": "arn:aws:iam::123456789012:role/service-role/
AWSBackupDefaultServiceRole",
            "StartBy": 1600725492.929,
            "ResourceType": "EC2"
        },
        {
            "BackupJobId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
            "BackupVaultName": "Default",
            "BackupVaultArn": "arn:aws:backup:us-west-2:123456789012:backup-
vault:Default",
            "RecoveryPointArn": "arn:aws:backup:us-west-2:123456789012:recovery-
point:a1b2c3d4-5678-90ab-cdef-EXAMPLE33333",
            "ResourceArn": "arn:aws:elasticfilesystem:us-west-2:123456789012:file-
system/fs-12345678",
            "CreationDate": 1600721724.77,
            "CompletionDate": 1600721744.488,
            "State": "COMPLETED",
            "PercentDone": "100.0",
            "BackupSizeInBytes": 71,
            "IamRoleArn": "arn:aws:iam::123456789012:role/service-role/
AWSBackupDefaultServiceRole",
            "StartBy": 1600725324.77,
            "ResourceType": "EFS"
        }
    ]
}
```

Per ulteriori informazioni, consulta Creazione di un backup nella Guida per sviluppatori di AWS .

Esempio 2: come elencare i processi di backup completati

L'`list-backup-jobs`esempio seguente restituisce i metadati relativi ai processi di backup completati nell'account AWS .

```
aws backup list-backup-jobs \
    --by-state COMPLETED
```

Output:

```
{
    "BackupJobs": [
        {
            "BackupJobId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
            "BackupVaultName": "Default",
            "BackupVaultArn": "arn:aws:backup:us-west-2:123456789012:backup-
vault:Default",
            "RecoveryPointArn": "arn:aws:backup:us-west-2:123456789012:recovery-
point:a1b2c3d4-5678-90ab-cdef-EXAMPLE33333",
            "ResourceArn": "arn:aws:elasticfilesystem:us-west-2:123456789012:file-
system/fs-12345678",
            "CreationDate": 1600721724.77,
            "CompletionDate": 1600721744.488,
            "State": "COMPLETED",
            "PercentDone": "100.0",
            "BackupSizeInBytes": 71,
            "IamRoleArn": "arn:aws:iam::123456789012:role/service-role/
AWSBackupDefaultServiceRole",
            "StartBy": 1600725324.77,
            "ResourceType": "EFS"
        }
    ]
}
```

Per ulteriori informazioni, consulta Creazione di un backup nella Guida per sviluppatori di AWS .

- Per i dettagli sull'API, consulta ListBackupJobs AWS CLICommand Reference.

AWS Batch esempi utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with AWS Batch.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`cancel-job`

Il seguente esempio di codice mostra come utilizzare`cancel-job`.

AWS CLI

Come annullare un processo

Questo esempio annulla un processo con l'ID del processo specificato.

Comando:

```
aws batch cancel-job --job-id bcf0b186-a532-4122-842e-2ccab8d54efb --
reason "Cancelling job."
```

- Per i dettagli sull'API, consulta CancelJob AWS CLICommand Reference.

`create-compute-environment`

Il seguente esempio di codice mostra come utilizzare`create-compute-environment`.

AWS CLI

Come creare un ambiente di calcolo gestito con istanze on demand

Questo esempio crea un ambiente di calcolo gestito con istanze di tipo C4 specifiche che vengono avviate on demand. L'ambiente di calcolo si chiama OnDemand C4.

Comando:

```
aws batch create-compute-environment --cli-input-json file://<path_to_json_file>/
C4OnDemand.json
```

Formato del file JSON:

```
{
  "computeEnvironmentName": "C4OnDemand",
  "type": "MANAGED",
  "state": "ENABLED",
  "computeResources": {
    "type": "EC2",
    "minvCpus": 0,
    "maxvCpus": 128,
    "desiredvCpus": 48,
    "instanceTypes": [
      "c4.large",
      "c4.xlarge",
      "c4.2xlarge",
      "c4.4xlarge",
      "c4.8xlarge"
    ],
    "subnets": [
      "subnet-220c0e0a",
      "subnet-1a95556d",
      "subnet-978f6dce"
    ],
    "securityGroupIds": [
      "sg-cf5093b2"
    ],
    "ec2KeyPair": "id_rsa",
    "instanceRole": "ecsInstanceRole",
    "tags": {
      "Name": "Batch Instance - C4OnDemand"
    }
  },
  "serviceRole": "arn:aws:iam::012345678910:role/AWSBatchServiceRole"
}
```

Output:

```
{
    "computeEnvironmentName": "C4OnDemand",
```

```
    "computeEnvironmentArn": "arn:aws:batch:us-east-1:012345678910:compute-
environment/C4OnDemand"
}
```

Come creare un ambiente di calcolo gestito con istanze spot

Questo esempio crea un ambiente di calcolo gestito con l'istanza di tipo M4 che viene avviata quando il prezzo dell'offerta spot è pari o inferiore al 20% del prezzo on demand per il tipo di istanza. L'ambiente di calcolo è denominato M4Spot.

Comando:

```
aws batch create-compute-environment --cli-input-json file://<path_to_json_file>/
M4Spot.json
```

Formato del file JSON:

```
{
  "computeEnvironmentName": "M4Spot",
  "type": "MANAGED",
  "state": "ENABLED",
  "computeResources": {
    "type": "SPOT",
    "spotIamFleetRole": "arn:aws:iam::012345678910:role/aws-ec2-spot-fleet-role",
    "minvCpus": 0,
    "maxvCpus": 128,
    "desiredvCpus": 4,
    "instanceTypes": [
      "m4"
    ],
    "bidPercentage": 20,
    "subnets": [
      "subnet-220c0e0a",
      "subnet-1a95556d",
      "subnet-978f6dce"
    ],
    "securityGroupIds": [
      "sg-cf5093b2"
    ],
    "ec2KeyPair": "id_rsa",
    "instanceRole": "ecsInstanceRole",
    "tags": {
```

```
      "Name": "Batch Instance - M4Spot"
    }
  },
  "serviceRole": "arn:aws:iam::012345678910:role/AWSBatchServiceRole"
}
```

Output:

```
{
    "computeEnvironmentName": "M4Spot",
    "computeEnvironmentArn": "arn:aws:batch:us-east-1:012345678910:compute-
environment/M4Spot"
}
```

- Per i dettagli sull'API, consulta AWS CLI Command CreateComputeEnvironmentReference.

`create-job-queue`

Il seguente esempio di codice mostra come utilizzare`create-job-queue`.

AWS CLI

Come creare una coda di processi a bassa priorità con un unico ambiente di calcolo

Questo esempio crea una coda di lavoro chiamata LowPriority che utilizza l'ambiente di calcolo M4Spot.

Comando:

```
aws batch create-job-queue --cli-input-json file://<path_to_json_file>/
LowPriority.json
```

Formato del file JSON:

```
{
  "jobQueueName": "LowPriority",
  "state": "ENABLED",
  "priority": 10,
  "computeEnvironmentOrder": [
    {
```

```
      "order": 1,
      "computeEnvironment": "M4Spot"
    }
  ]
}
```

Output:

```
{
    "jobQueueArn": "arn:aws:batch:us-east-1:012345678910:job-queue/LowPriority",
    "jobQueueName": "LowPriority"
}
```

Come creare una coda di processi ad alta priorità con due ambienti di calcolo

Questo esempio crea una coda di lavoro chiamata HighPriority che utilizza l'ambiente di calcolo C4 con un ordine di 1 e l'ambiente di OnDemand calcolo M4Spot con un ordine di 2. Lo scheduler tenterà innanzitutto di inserire i lavori nell'ambiente di calcolo C4. OnDemand

Comando:

```
aws batch create-job-queue --cli-input-json file://<path_to_json_file>/
HighPriority.json
```

Formato del file JSON:

```
{
  "jobQueueName": "HighPriority",
  "state": "ENABLED",
  "priority": 1,
  "computeEnvironmentOrder": [
    {
      "order": 1,
      "computeEnvironment": "C4OnDemand"
    },
    {
      "order": 2,
      "computeEnvironment": "M4Spot"
    }
  ]
}
```

Output:

```
{
    "jobQueueArn": "arn:aws:batch:us-east-1:012345678910:job-queue/HighPriority",
    "jobQueueName": "HighPriority"
}
```

- Per i dettagli sull'API, consulta Command CreateJobQueueReference AWS CLI .

`delete-compute-environment`

Il seguente esempio di codice mostra come utilizzare`delete-compute-environment`.

AWS CLI

Come eliminare un ambiente di calcolo

Questo esempio elimina l'ambiente di OnDemand calcolo P2.

Comando:

```
aws batch delete-compute-environment --compute-environment P2OnDemand
```

- Per i dettagli sull'API, consulta Command DeleteComputeEnvironmentReference AWS CLI .

`delete-job-queue`

Il seguente esempio di codice mostra come utilizzare`delete-job-queue`.

AWS CLI

Come eliminare una coda di processi

Questo esempio elimina la coda dei processi GPGPU.

Comando:

```
aws batch delete-job-queue --job-queue GPGPU
```

- Per i dettagli sull'API, consulta DeleteJobQueue AWS CLICommand Reference.

`deregister-job-definition`

Il seguente esempio di codice mostra come utilizzare`deregister-job-definition`.

AWS CLI

Come annullare la registrazione di una definizione di processo

Questo esempio annulla la registrazione di una definizione di processo denominata sleep10.

Comando:

```
aws batch deregister-job-definition --job-definition sleep10
```

- Per i dettagli sull'API, consulta DeregisterJobDefinition AWS CLICommand Reference.

`describe-compute-environments`

Il seguente esempio di codice mostra come utilizzare`describe-compute-environments`.

AWS CLI

Come eliminare un ambiente di calcolo

Questo esempio descrive l'ambiente di OnDemand calcolo P2.

Comando:

```
aws batch describe-compute-environments --compute-environments P2OnDemand
```

Output:

```
{
    "computeEnvironments": [
        {
            "status": "VALID",
            "serviceRole": "arn:aws:iam::012345678910:role/AWSBatchServiceRole",
            "computeEnvironmentArn": "arn:aws:batch:us-east-1:012345678910:compute-
environment/P2OnDemand",
            "computeResources": {
                "subnets": [
```

```
                    "subnet-220c0e0a",
                    "subnet-1a95556d",
                    "subnet-978f6dce"
                ],
                "tags": {
                    "Name": "Batch Instance - P2OnDemand"
                },
                "desiredvCpus": 48,
                "minvCpus": 0,
                "instanceTypes": [
                    "p2"
                ],
                "securityGroupIds": [
                    "sg-cf5093b2"
                ],
                "instanceRole": "ecsInstanceRole",
                "maxvCpus": 128,
                "type": "EC2",
                "ec2KeyPair": "id_rsa"
            },
            "statusReason": "ComputeEnvironment Healthy",
            "ecsClusterArn": "arn:aws:ecs:us-east-1:012345678910:cluster/
P2OnDemand_Batch_2c06f29d-d1fe-3a49-879d-42394c86effc",
            "state": "ENABLED",
            "computeEnvironmentName": "P2OnDemand",
            "type": "MANAGED"
        }
    ]
}
```

- Per i dettagli sull'API, consulta AWS CLI Command DescribeComputeEnvironmentsReference.

`describe-job-definitions`

Il seguente esempio di codice mostra come utilizzare`describe-job-definitions`.

AWS CLI

Come descrivere le definizioni dei processi attivi

In questo esempio vengono descritte le definizioni di tutti i processi attivi.

Comando:

```
aws batch describe-job-definitions --status ACTIVE
```

Output:

```
{
    "jobDefinitions": [
        {
            "status": "ACTIVE",
            "jobDefinitionArn": "arn:aws:batch:us-east-1:012345678910:job-
definition/sleep60:1",
            "containerProperties": {
                "mountPoints": [],
                "parameters": {},
                "image": "busybox",
                "environment": {},
                "vcpus": 1,
                "command": [
                    "sleep",
                    "60"
                ],
                "volumes": [],
                "memory": 128,
                "ulimits": []
            },
            "type": "container",
            "jobDefinitionName": "sleep60",
            "revision": 1
        }
    ]
}
```

- Per i dettagli sull'API, consulta DescribeJobDefinitions AWS CLICommand Reference.

`describe-job-queues`

Il seguente esempio di codice mostra come utilizzare`describe-job-queues`.

AWS CLI

Come descrivere una coda di processi

Questo esempio descrive la coda dei HighPriority lavori.

Comando:

```
aws batch describe-job-queues --job-queues HighPriority
```

Output:

```
{
    "jobQueues": [
        {
            "status": "VALID",
            "jobQueueArn": "arn:aws:batch:us-east-1:012345678910:job-queue/
HighPriority",
            "computeEnvironmentOrder": [
                {
                    "computeEnvironment": "arn:aws:batch:us-
east-1:012345678910:compute-environment/C4OnDemand",
                    "order": 1
                }
            ],
            "statusReason": "JobQueue Healthy",
            "priority": 1,
            "state": "ENABLED",
            "jobQueueName": "HighPriority"
        }
    ]
}
```

- Per i dettagli sull'API, vedere DescribeJobQueuesin AWS CLI Command Reference.

`describe-jobs`

Il seguente esempio di codice mostra come utilizzare`describe-jobs`.

AWS CLI

Come descrivere un processo

L'esempio `describe-jobs` seguente descrive un processo con l'ID processo specificato.

```
aws batch describe-jobs \
    --jobs bcf0b186-a532-4122-842e-2ccab8d54efb
```

Output:

```
{
    "jobs": [
        {
            "status": "SUBMITTED",
            "container": {
                "mountPoints": [],
                "image": "busybox",
                "environment": [],
                "vcpus": 1,
                "command": [
                    "sleep",
                    "60"
                ],
                "volumes": [],
                "memory": 128,
                "ulimits": []
            },
            "parameters": {},
            "jobDefinition": "arn:aws:batch:us-east-1:012345678910:job-definition/
sleep60:1",
            "jobQueue": "arn:aws:batch:us-east-1:012345678910:job-queue/
HighPriority",
            "jobId": "bcf0b186-a532-4122-842e-2ccab8d54efb",
            "dependsOn": [],
            "jobName": "example",
            "createdAt": 1480483387803
        }
    ]
}
```

- Per i dettagli sull'API, consulta DescribeJobs AWS CLICommand Reference.

`list-jobs`

Il seguente esempio di codice mostra come utilizzare`list-jobs`.

AWS CLI

Come elencare i processi in esecuzione

Questo esempio elenca i lavori in esecuzione nella coda dei HighPriority processi.

Comando:

```
aws batch list-jobs --job-queue HighPriority
```

Output:

```
{
    "jobSummaryList": [
        {
            "jobName": "example",
            "jobId": "e66ff5fd-a1ff-4640-b1a2-0b0a142f49bb"
        }
    ]
}
```

Come elencare i processi inviati

Questo esempio elenca i lavori nella coda dei HighPriority lavori che si trovano nello stato di processo INVIATO.

Comando:

```
aws batch list-jobs --job-queue HighPriority --job-status SUBMITTED
```

Output:

```
{
    "jobSummaryList": [
        {
            "jobName": "example",
            "jobId": "68f0c163-fbd4-44e6-9fd1-25b14a434786"
        }
    ]
}
```

- Per i dettagli sull'API, consulta ListJobs AWS CLICommand Reference.

`register-job-definition`

Il seguente esempio di codice mostra come utilizzare`register-job-definition`.

AWS CLI

Come registrare una definizione di processo

Questo esempio registra una definizione di processo per un processo container semplice.

Comando:

```
aws batch register-job-definition --job-definition-name sleep30 --type container --
container-properties '{ "image": "busybox", "vcpus": 1, "memory": 128, "command":
 [ "sleep", "30"]}'
```

Output:

```
{
    "jobDefinitionArn": "arn:aws:batch:us-east-1:012345678910:job-definition/
sleep30:1",
    "jobDefinitionName": "sleep30",
    "revision": 1
}
```

- Per i dettagli sull'API, consulta RegisterJobDefinition AWS CLICommand Reference.

`submit-job`

Il seguente esempio di codice mostra come utilizzare`submit-job`.

AWS CLI

Come inviare un processo

Questo esempio invia un semplice processo contenitore chiamato example alla coda dei HighPriority lavori.

Comando:

```
aws batch submit-job --job-name example --job-queue HighPriority  --job-
definition sleep60
```

Output:

```
{
    "jobName": "example",
    "jobId": "876da822-4198-45f2-a252-6cea32512ea8"
}
```

- Per i dettagli sull'API, consulta AWS CLI Command SubmitJobReference.

`terminate-job`

Il seguente esempio di codice mostra come utilizzare`terminate-job`.

AWS CLI

Come terminare un processo

Questo esempio termina un processo con l'ID del processo specificato.

Comando:

```
aws batch terminate-job --job-id 61e743ed-35e4-48da-b2de-5c8333821c84 --
reason "Terminating job."
```

- Per i dettagli sull'API, consulta TerminateJob AWS CLICommand Reference.

`update-compute-environment`

Il seguente esempio di codice mostra come utilizzare`update-compute-environment`.

AWS CLI

Come aggiornare un ambiente di calcolo

Questo esempio disabilita l'ambiente di OnDemand calcolo P2 in modo che possa essere eliminato.

Comando:

```
aws batch update-compute-environment --compute-environment P2OnDemand --
state DISABLED
```

Output:

```
{
    "computeEnvironmentName": "P2OnDemand",
    "computeEnvironmentArn": "arn:aws:batch:us-east-1:012345678910:compute-
environment/P2OnDemand"
}
```

- Per i dettagli sull'API, consulta Command UpdateComputeEnvironmentReference AWS CLI .

`update-job-queue`

Il seguente esempio di codice mostra come utilizzareu`pdate-job-queue`.

AWS CLI

Come creare una coda di processi

Questo esempio disabilita una coda di processi in modo che possa essere eliminata.

Comando:

```
aws batch update-job-queue --job-queue GPGPU --state DISABLED
```

Output:

```
{
    "jobQueueArn": "arn:aws:batch:us-east-1:012345678910:job-queue/GPGPU",
    "jobQueueName": "GPGPU"
}
```

- Per i dettagli sull'API, consulta UpdateJobQueue AWS CLICommand Reference.

Budget AWS esempi utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with Budget AWS.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`create-budget`

Il seguente esempio di codice mostra come utilizzare`create-budget`.

AWS CLI

Come creare un budget di utilizzo e costi

Il comando `create-budget` seguente crea un budget di utilizzo e costi.

```
aws budgets create-budget \
    --account-id 111122223333 \
    --budget file://budget.json \
    --notifications-with-subscribers file://notifications-with-subscribers.json
```

Contenuto di `budget.json`:

```
{
    "BudgetLimit": {
        "Amount": "100",
        "Unit": "USD"
    },
    "BudgetName": "Example Tag Budget",
    "BudgetType": "COST",
    "CostFilters": {
        "TagKeyValue": [
            "user:Key$value1",
            "user:Key$value2"
        ]
    },
    "CostTypes": {
        "IncludeCredit": true,
        "IncludeDiscount": true,
        "IncludeOtherSubscription": true,
```

```
        "IncludeRecurring": true,
        "IncludeRefund": true,
        "IncludeSubscription": true,
        "IncludeSupport": true,
        "IncludeTax": true,
        "IncludeUpfront": true,
        "UseBlended": false
    },
    "TimePeriod": {
        "Start": 1477958399,
        "End": 3706473600
    },
    "TimeUnit": "MONTHLY"
}
```

Contenuto di `notifications-with-subscribers.json`:

```
[
    {
        "Notification": {
            "ComparisonOperator": "GREATER_THAN",
            "NotificationType": "ACTUAL",
            "Threshold": 80,
            "ThresholdType": "PERCENTAGE"
        },
        "Subscribers": [
            {
                "Address": "example@example.com",
                "SubscriptionType": "EMAIL"
            }
        ]
    }
]
```

- Per i dettagli sull'API, consulta CreateBudget AWS CLICommand Reference.

`create-notification`

Il seguente esempio di codice mostra come utilizzare`create-notification`.

AWS CLI

Come creare una notifica per il budget di costi e utilizzo specificato

Questo esempio crea una notifica per il budget di costi e utilizzo specificato.

Comando:

```
aws budgets create-notification --account-id 111122223333 --budget-name "Example
 Budget" --
notification NotificationType=ACTUAL,ComparisonOperator=GREATER_THAN,Threshold=80,ThresholdT
 --subscriber SubscriptionType=EMAIL,Address=example@example.com
```

- Per i dettagli sull'API, consulta CreateNotification AWS CLICommand Reference.

`create-subscriber`

Il seguente esempio di codice mostra come utilizzare`create-subscriber`.

AWS CLI

Come creare un abbonato per una notifica associata a un budget di costi e utilizzo

Questo esempio crea un abbonato per la notifica specificata.

Comando:

```
aws budgets create-subscriber --account-id 111122223333 --budget-name "Example
 Budget" --
notification NotificationType=ACTUAL,ComparisonOperator=GREATER_THAN,Threshold=80,ThresholdT
 --subscriber SubscriptionType=EMAIL,Address=example@example.com
```

- Per i dettagli sull'API, consulta CreateSubscriber AWS CLICommand Reference.

`delete-budget`

Il seguente esempio di codice mostra come utilizzare`delete-budget`.

AWS CLI

Come eliminare un budget di costi e utilizzo

Questo esempio elimina il budget di costi e utilizzo specificato.

Comando:

```
aws budgets delete-budget --account-id 111122223333 --budget-name "Example Budget"
```

- Per i dettagli sull'API, consulta DeleteBudget AWS CLICommand Reference.

`delete-notification`

Il seguente esempio di codice mostra come utilizzare`delete-notification`.

AWS CLI

Come eliminare una notifica da un budget

In questo esempio viene eliminata la notifica specificata dal budget specificato.

Comando:

```
aws budgets delete-notification --account-id 111122223333 --budget-name "Example
 Budget" --
notification NotificationType=ACTUAL,ComparisonOperator=GREATER_THAN,Threshold=80,ThresholdT
```

- Per i dettagli sull'API, consulta DeleteNotification AWS CLICommand Reference.

`delete-subscriber`

Il seguente esempio di codice mostra come utilizzare`delete-subscriber`.

AWS CLI

Come eliminare un abbonato da una notifica

In questo esempio viene eliminato l'abbonato specificato dalla notifica specificata.

Comando:

```
aws budgets delete-subscriber --account-id 111122223333 --budget-name "Example
 Budget" --
notification NotificationType=ACTUAL,ComparisonOperator=GREATER_THAN,Threshold=80,ThresholdT
 --subscriber SubscriptionType=EMAIL,Address=example@example.com
```

- Per i dettagli sull'API, consulta DeleteSubscriber AWS CLICommand Reference.

`describe-budget`

Il seguente esempio di codice mostra come utilizzare`describe-budget`.

AWS CLI

Come recuperare un budget associato a un account

Questo esempio recupera il budget di costi e utilizzo specificato.

Comando:

```
aws budgets describe-budget --account-id 111122223333 --budget-name "Example Budget"
```

Output:

```
{
   "Budget": {
       "CalculatedSpend": {
           "ForecastedSpend": {
               "Amount": "2641.548000000000229192664846777915954589843 75",
               "Unit": "USD"
           },
           "ActualSpend": {
               "Amount": "604.456000000000017280399333685636520385742 1875",
               "Unit": "USD"
           }
       },
       "BudgetType": "COST",
       "BudgetLimit": {
           "Amount": "100",
           "Unit": "USD"
       },
       "BudgetName": "Example Budget",
       "CostTypes": {
           "IncludeOtherSubscription": true,
           "IncludeUpfront": true,
           "IncludeRefund": true,
           "UseBlended": false,
           "IncludeDiscount": true,
           "UseAmortized": false,
           "IncludeTax": true,
           "IncludeCredit": true,
```

```
            "IncludeSupport": true,
            "IncludeRecurring": true,
            "IncludeSubscription": true
        },
        "TimeUnit": "MONTHLY",
        "TimePeriod": {
            "Start": 1477958399.0,
            "End": 3706473600.0
        },
        "CostFilters": {
            "AZ": [
                "us-east-1"
            ]
        }
    }
}
```

- Per i dettagli sull'API, consulta DescribeBudget AWS CLICommand Reference.

`describe-budgets`

Il seguente esempio di codice mostra come utilizzare`describe-budgets`.

AWS CLI

Come recuperare i budget associati a un account

Questo esempio recupera i budget di costo e utilizzo per un account.

Comando:

```
aws budgets describe-budgets --account-id 111122223333 --max-results 20
```

Output:

```
{
    "Budgets": [
        {
            "CalculatedSpend": {
                "ForecastedSpend": {
                    "Amount": "2641.548000000000229192664846777915954589843 75",
                    "Unit": "USD"
                },
```

```
            "ActualSpend": {
                "Amount": "604.456000000000017280399333685636520385742187 5",
                "Unit": "USD"
            }
        },
        "BudgetType": "COST",
        "BudgetLimit": {
            "Amount": "100",
            "Unit": "USD"
        },
        "BudgetName": "Example Budget",
        "CostTypes": {
            "IncludeOtherSubscription": true,
            "IncludeUpfront": true,
            "IncludeRefund": true,
            "UseBlended": false,
            "IncludeDiscount": true,
            "UseAmortized": false,
            "IncludeTax": true,
            "IncludeCredit": true,
            "IncludeSupport": true,
            "IncludeRecurring": true,
            "IncludeSubscription": true
        },
        "TimeUnit": "MONTHLY",
        "TimePeriod": {
            "Start": 1477958399.0,
            "End": 3706473600.0
        },
        "CostFilters": {
            "AZ": [
                "us-east-1"
            ]
        }
    }
  ]
}
```

- Per i dettagli sull'API, consulta DescribeBudgets AWS CLICommand Reference.

`describe-notifications-for-budget`

Il seguente esempio di codice mostra come utilizzare`describe-notifications-for-budget`.

AWS CLI

Come recuperare le notifiche relative a un budget

Questo esempio recupera le notifiche relative a un budget di costi e utilizzo.

Comando:

```
aws budgets describe-notifications-for-budget --account-id 111122223333 --budget-
name "Example Budget" --max-results 5
```

Output:

```
{
   "Notifications": [
       {
           "Threshold": 80.0,
           "ComparisonOperator": "GREATER_THAN",
           "NotificationType": "ACTUAL"
       }
   ]
}
```

- Per i dettagli sull'API, consulta DescribeNotificationsForBudget AWS CLICommand Reference.

`describe-subscribers-for-notification`

Il seguente esempio di codice mostra come utilizzare`describe-subscribers-for-notification`.

AWS CLI

Come recuperare gli abbonati per una notifica relativa a un budget

Questo esempio recupera gli abbonati per una notifica relativa al budget di costi e utilizzo.

Comando:

```
aws budgets describe-subscribers-for-notification --
account-id 111122223333 --budget-name "Example Budget" --
notification NotificationType=ACTUAL,ComparisonOperator=GREATER_THAN,Threshold=80,ThresholdT
 --max-results 5
```

Output:

```
{
   "Subscribers": [
       {
           "SubscriptionType": "EMAIL",
           "Address": "example2@example.com"
       },
       {
           "SubscriptionType": "EMAIL",
           "Address": "example@example.com"
       }
   ]
}
```

- Per i dettagli sull'API, consulta DescribeSubscribersForNotification AWS CLICommand Reference.

`update-budget`

Il seguente esempio di codice mostra come utilizzareup`date-budget`.

AWS CLI

Come sostituire un budget con un budget di costi e utilizzo

Questo esempio sostituisce un budget di costi e utilizzo con un nuovo budget.

Comando:

```
aws budgets update-budget --account-id 111122223333 --new-budget file://new-
budget.json
```

new-budget.json:

```
{
    "BudgetLimit": {
       "Amount": "100",
       "Unit": "USD"
    },
    "BudgetName": "Example Budget",
```

```
    "BudgetType": "COST",
    "CostFilters": {
       "AZ" : [ "us-east-1" ]
    },
    "CostTypes": {
       "IncludeCredit": false,
       "IncludeDiscount": true,
       "IncludeOtherSubscription": true,
       "IncludeRecurring": true,
       "IncludeRefund": true,
       "IncludeSubscription": true,
       "IncludeSupport": true,
       "IncludeTax": true,
       "IncludeUpfront": true,
       "UseBlended": false,
       "UseAmortized": true
    },
    "TimePeriod": {
       "Start": 1477958399,
       "End": 3706473600
    },
    "TimeUnit": "MONTHLY"
}
```

- Per i dettagli sull'API, consulta UpdateBudget AWS CLICommand Reference.

`update-notification`

Il seguente esempio di codice mostra come utilizzareupdate-notification.

AWS CLI

Come sostituire una notifica per un budget di costi e utilizzo

Questo esempio sostituisce una notifica dell'80% per un budget di costi e utilizzo con una notifica del 90%.

Comando:

```
aws budgets update-notification --account-id 111122223333 --budget-name "Example
 Budget" --old-notification
  NotificationType=ACTUAL,ComparisonOperator=GREATER_THAN,Threshold=80,ThresholdType=PERCENT
```

```
--new-notification
 NotificationType=ACTUAL,ComparisonOperator=GREATER_THAN,Threshold=90,ThresholdType=PERCENT
```

- Per i dettagli sull'API, consulta UpdateNotification AWS CLICommand Reference.

`update-subscriber`

Il seguente esempio di codice mostra come utilizzare`update-subscriber`.

AWS CLI

Come sostituire un abbonato con un budget di costi e utilizzo

Questo esempio sostituisce l'abbonato con un budget di costi e utilizzo.

Comando:

```
aws budgets update-subscriber --account-id 111122223333 --budget-name "Example
 Budget" --
notification NotificationType=ACTUAL,ComparisonOperator=GREATER_THAN,Threshold=80,ThresholdT
 --old-subscriber SubscriptionType=EMAIL,Address=example@example.com --new-
subscriber SubscriptionType=EMAIL,Address=example2@example.com
```

- Per i dettagli sull'API, consulta UpdateSubscriber AWS CLICommand Reference.

Esempi di utilizzo di Amazon Chime AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando Amazon Chime. AWS Command Line Interface

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`associate-phone-number-with-user`

Il seguente esempio di codice mostra come usare`associate-phone-number-with-user`.

AWS CLI

Come associare un numero di telefono a un utente

L'esempio `associate-phone-number-with-user` seguente associa il numero di telefono specificato a un utente.

```
aws chime associate-phone-number-with-user \
    --account-id 12a3456b-7c89-012d-3456-78901e23fg45 \
    --user-id 1ab2345c-67de-8901-f23g-45h678901j2k \
    --e164-phone-number "+12065550100"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Gestione dei numeri di telefono degli utenti nella Guida all'amministrazione di Amazon Chime.

- Per i dettagli sull'API, consulta AssociatePhoneNumberWithUser AWS CLICommand Reference.

`associate-signin-delegate-groups-with-account`

Il seguente esempio di codice mostra come utilizzare`associate-signin-delegate-groups-with-account`.

AWS CLI

Come associare gruppi di delegati di accesso

L'esempio `associate-signin-delegate-groups-with-account` seguente associa il gruppo di delegati di accesso specificato all'account Amazon Chime specificato.

```
aws chime associate-signin-delegate-groups-with-account \
    --account-id 12a3456b-7c89-012d-3456-78901e23fg45 \
```

```
    --signin-delegate-groups GroupName=my_users
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Gestione dell'accesso e delle autorizzazioni degli utenti nella Guida all'amministrazione di Amazon Chime.

- Per i dettagli sull'API, consulta AssociateSigninDelegateGroupsWithAccount AWS CLICommand Reference.

`batch-create-room-membership`

Il seguente esempio di codice mostra come utilizzare`batch-create-room-membership`.

AWS CLI

Come creare più appartenenze alla chat room

L'esempio `batch-create-room-membership` seguente aggiunge più utenti a una chat room come membri della chat room. Inoltre, assegna agli utenti i ruoli di amministratore e membro.

```
aws chime batch-create-room-membership \
    --account-id 12a3456b-7c89-012d-3456-78901e23fg45 \
    --room-id abcd1e2d-3e45-6789-01f2-3g45h67i890j \
    --membership-item-list "MemberId=1ab2345c-67de-8901-
f23g-45h678901j2k,Role=Administrator" "MemberId=2ab2345c-67de-8901-
f23g-45h678901j2k,Role=Member"
```

Output:

```
{
    "ResponseMetadata": {
        "RequestId": "169ba401-d886-475f-8b3f-e01eac6fadfb",
        "HTTPStatusCode": 201,
        "HTTPHeaders": {
            "x-amzn-requestid": "169ba401-d886-475f-8b3f-e01eac6fadfb",
            "content-type": "application/json",
            "content-length": "13",
            "date": "Mon, 02 Dec 2019 22:46:58 GMT",
            "connection": "keep-alive"
        },
```

```
        "RetryAttempts": 0
    },
    "Errors": []
}
```

Per ulteriori informazioni, consulta Creazione di una chat room nella Guida per l'utente di Amazon Chime.

- Per i dettagli sull'API, consulta BatchCreateRoomMembership AWS CLICommand Reference.

`batch-delete-phone-number`

Il seguente esempio di codice mostra come utilizzare`batch-delete-phone-number`.

AWS CLI

Come eliminare più numeri di telefono

L'esempio `batch-delete-phone-number` seguente elimina tutti i numeri di telefono specificati.

```
aws chime batch-delete-phone-number \
    --phone-number-ids "%2B12065550100" "%2B12065550101"
```

Questo comando non produce alcun output. Output:

```
{
    "PhoneNumberErrors": []
}
```

Per ulteriori informazioni, consulta Utilizzo dei numeri di telefono nella Guida di amministrazione di Amazon Chime.

- Per i dettagli sull'API, consulta BatchDeletePhoneNumber AWS CLICommand Reference.

`batch-suspend-user`

Il seguente esempio di codice mostra come utilizzare`batch-suspend-user`.

AWS CLI

Come sospendere più utenti

L'esempio `batch-suspend-user` seguente sospende gli utenti elencati dall'account Amazon Chime specificato.

```
aws chime batch-suspend-user \
    --account-id a1b2c3d4-5678-90ab-cdef-11111EXAMPLE \
    --user-id-list "a1b2c3d4-5678-90ab-cdef-22222EXAMPLE" "a1b2c3d4-5678-90ab-
cdef-33333EXAMPLE" "a1b2c3d4-5678-90ab-cdef-44444EXAMPLE"
```

Output:

```
{
    "UserErrors": []
}
```

- Per i dettagli sull'API, consulta BatchSuspendUser AWS CLICommand Reference.

`batch-unsuspend-user`

Il seguente esempio di codice mostra come utilizzare`batch-unsuspend-user`.

AWS CLI

Come annullare la sospensione di più utenti

L'esempio `batch-unsuspend-user` seguente rimuove qualsiasi sospensione precedente per gli utenti elencati dall'account Amazon Chime specificato.

```
aws chime batch-unsuspend-user \
    --account-id a1b2c3d4-5678-90ab-cdef-11111EXAMPLE \
    --user-id-list "a1b2c3d4-5678-90ab-cdef-22222EXAMPLE" "a1b2c3d4-5678-90ab-
cdef-33333EXAMPLE" "a1b2c3d4-5678-90ab-cdef-44444EXAMPLE"
```

Output:

```
{
    "UserErrors": []
}
```

- Per i dettagli sull'API, consulta BatchUnsuspendUser AWS CLICommand Reference.

`batch-update-phone-number`

Il seguente esempio di codice mostra come utilizzare`batch-update-phone-number`.

AWS CLI

Come aggiornare contemporaneamente più tipi di prodotto per i numeri di telefono

L'esempio `batch-update-phone-number` seguente aggiorna i tipi di prodotto per tutti i numeri di telefono specificati.

```
aws chime batch-update-phone-number \
    --update-phone-number-request-items PhoneNumberId=
%2B12065550100,ProductType=BusinessCalling PhoneNumberId=
%2B12065550101,ProductType=BusinessCalling
```

Output:

```
{
    "PhoneNumberErrors": []
}
```

Come aggiornare contemporaneamente più nomi di chiamanti di numeri di telefono

L'esempio `batch-update-phone-number` seguente aggiorna i nomi di chiamanti per tutti i numeri di telefono specificati.

```
aws chime batch-update-phone-number \
    --update-phone-number-request-items PhoneNumberId=
%2B14013143874,CallingName=phonenumber1 PhoneNumberId=
%2B14013144061,CallingName=phonenumber2
```

Output:

```
{
    "PhoneNumberErrors": []
}
```

Per ulteriori informazioni, consulta Utilizzo dei numeri di telefono nella Guida di amministrazione di Amazon Chime.

- Per i dettagli sull'API, consulta BatchUpdatePhoneNumber AWS CLICommand Reference.

`batch-update-user`

Il seguente esempio di codice mostra come utilizzare`batch-update-user`.

AWS CLI

Come aggiornare più utenti con un singolo comando

L'esempio `batch-update-user` seguente aggiorna `LicenseType` per ciascuno degli utenti elencati nell'account Amazon Chime specificato.

```
aws chime batch-update-user \
    --account-id a1b2c3d4-5678-90ab-cdef-11111EXAMPLE
    --update-user-request-items "UserId=a1b2c3d4-5678-90ab-
cdef-22222EXAMPLE,LicenseType=Basic" "UserId=a1b2c3d4-5678-90ab-
cdef-33333EXAMPLE,LicenseType=Basic"
```

Output:

```
{
    "UserErrors": []
}
```

- Per i dettagli sull'API, consulta BatchUpdateUser AWS CLICommand Reference.

`create-account`

Il seguente esempio di codice mostra come utilizzare`create-account`.

AWS CLI

Come creare un account

L'`create-account`esempio seguente crea un account Amazon Chime con l'account dell' AWS amministratore.

```
aws chime create-account \
    --name MyChimeAccount
```

Output:

```
{
    "Account": {
        "AwsAccountId": "111122223333",
        "AccountId": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
        "Name": "MyChimeAccount",
        "AccountType": "Team",
        "CreatedTimestamp": "2019-01-04T17:11:22.003Z",
        "DefaultLicense": "Pro",
        "SupportedLicenses": [
            "Basic",
            "Pro"
        ],
        "SigninDelegateGroups": [
            {
                "GroupName": "myGroup"
            },
        ]
    }
}
```

Per ulteriori informazioni, consulta Nozioni di base nella Guida di amministrazione di Amazon Chime.

- Per i dettagli sull'API, consulta CreateAccount AWS CLICommand Reference.

`create-bot`

Il seguente esempio di codice mostra come utilizzare`create-bot`.

AWS CLI

Come creare un bot Amazon Chime

L'esempio `create-bot` seguente crea un bot per l'account Amazon Chime Enterprise specificato.

```
aws chime create-bot \
    --account-id 12a3456b-7c89-012d-3456-78901e23fg45 \
    --display-name "myBot" \
    --domain "example.com"
```

Output:

```
{
    "Bot": {
        "BotId": "123abcd4-5ef6-789g-0h12-34j56789012k",
        "UserId": "123abcd4-5ef6-789g-0h12-34j56789012k",
        "DisplayName": "myBot (Bot)",
        "BotType": "ChatBot",
        "Disabled": false,
        "CreatedTimestamp": "2019-09-09T18:05:56.749Z",
        "UpdatedTimestamp": "2019-09-09T18:05:56.749Z",
        "BotEmail": "myBot-chimebot@example.com",
        "SecurityToken": "wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY"
    }
}
```

Per ulteriori informazioni, consulta Integrare un chat bot con Amazon Chime nella Guida per gli sviluppatori di Amazon Chime.

- Per i dettagli sull'API, consulta CreateBot AWS CLICommand Reference.

`create-phone-number-order`

Il seguente esempio di codice mostra come utilizzare`create-phone-number-order`.

AWS CLI

Come creare un ordine di numeri di telefono

L'esempio `create-phone-number-order` seguente crea un ordine di numeri di telefono per i numeri di telefono specificati.

```
aws chime create-phone-number-order \
    --product-type VoiceConnector \
    --e164-phone-numbers "+12065550100" "+12065550101" "+12065550102"
```

Output:

```
{
    "PhoneNumberOrder": {
        "PhoneNumberOrderId": "abc12345-de67-89f0-123g-h45i678j9012",
        "ProductType": "VoiceConnector",
```

```
        "Status": "Processing",
        "OrderedPhoneNumbers": [
            {
                "E164PhoneNumber": "+12065550100",
                "Status": "Processing"
            },
            {
               "E164PhoneNumber": "+12065550101",
               "Status": "Processing"
            },
            {
              "E164PhoneNumber": "+12065550102",
              "Status": "Processing"
            }
        ],
        "CreatedTimestamp": "2019-08-09T21:35:21.427Z",
        "UpdatedTimestamp": "2019-08-09T21:35:22.408Z"
    }
}
```

Per ulteriori informazioni, consulta Utilizzo dei numeri di telefono nella Guida di amministrazione di Amazon Chime.

- Per i dettagli sull'API, consulta CreatePhoneNumberOrder AWS CLICommand Reference.

`create-room-membership`

Il seguente esempio di codice mostra come utilizzare`create-room-membership`.

AWS CLI

Come creare un'appartenenza a una chat room

L'esempio `create-room-membership` seguente aggiunge l'utente specificato alla chat room come membro della chat room.

```
aws chime create-room-membership \
    --account-id 12a3456b-7c89-012d-3456-78901e23fg45 \
    --room-id abcd1e2d-3e45-6789-01f2-3g45h67i890j \
    --member-id 1ab2345c-67de-8901-f23g-45h678901j2k
```

Output:

```
{
    "RoomMembership": {
        "RoomId": "abcd1e2d-3e45-6789-01f2-3g45h67i890j",
        "Member": {
            "MemberId": "1ab2345c-67de-8901-f23g-45h678901j2k",
            "MemberType": "User",
            "Email": "janed@example.com",
            "FullName": "Jane Doe",
            "AccountId": "12a3456b-7c89-012d-3456-78901e23fg45"
        },
        "Role": "Member",
        "InvitedBy": "arn:aws:iam::111122223333:user/alejandro",
        "UpdatedTimestamp": "2019-12-02T22:36:41.969Z"
    }
}
```

Per ulteriori informazioni, consulta Creazione di una chat room nella Guida per l'utente di Amazon Chime.

- Per i dettagli sull'API, consulta CreateRoomMembership AWS CLICommand Reference.

`create-room`

Il seguente esempio di codice mostra come utilizzare`create-room`.

AWS CLI

Come creare una chat room

L'esempio `create-room` seguente crea una chat room per l'account Amazon Chime specificato.

```
aws chime create-room \
    --account-id 12a3456b-7c89-012d-3456-78901e23fg45 \
    --name chatRoom
```

Output:

```
{
    "Room": {
        "RoomId": "abcd1e2d-3e45-6789-01f2-3g45h67i890j",
        "Name": "chatRoom",
        "AccountId": "12a3456b-7c89-012d-3456-78901e23fg45",
```

```
        "CreatedBy": "arn:aws:iam::111122223333:user/alejandro",
        "CreatedTimestamp": "2019-12-02T22:29:31.549Z",
        "UpdatedTimestamp": "2019-12-02T22:29:31.549Z"
    }
}
```

Per ulteriori informazioni, consulta Creazione di una chat room nella Guida per l'utente di Amazon Chime.

- Per i dettagli sull'API, consulta CreateRoom AWS CLICommand Reference.

`create-user`

Il seguente esempio di codice mostra come utilizzare`create-user`.

AWS CLI

Come creare un profilo utente per un dispositivo condiviso

L'esempio `create-user` seguente crea un profilo di dispositivo condiviso per l'indirizzo e-mail specificato.

```
aws chime create-user \
    --account-id 12a3456b-7c89-012d-3456-78901e23fg45 \
    --email roomdevice@example.com \
    --user-type SharedDevice
```

Output:

```
{
    "User": {
        "UserId": "1ab2345c-67de-8901-f23g-45h678901j2k",
        "AccountId": "12a3456b-7c89-012d-3456-78901e23fg45",
        "PrimaryEmail": "roomdevice@example.com",
        "DisplayName": "Room Device",
        "LicenseType": "Pro",
        "UserType": "SharedDevice",
        "UserRegistrationStatus": "Registered",
        "RegisteredOn": "2020-01-15T22:38:09.806Z",
        "AlexaForBusinessMetadata": {
            "IsAlexaForBusinessEnabled": false
        }
```

```
    }
}
```

Per ulteriori informazioni, consulta Preparazione della configurazione nella Guida di amministrazione di Amazon Chime.

- Per i dettagli sull'API, consulta CreateUser AWS CLICommand Reference.

`delete-account`

Il seguente esempio di codice mostra come utilizzare`delete-account`.

AWS CLI

Come eliminare un account

L'esempio `delete-account` seguente elimina l'account specificato.

```
aws chime delete-account --account-id a1b2c3d4-5678-90ab-cdef-11111EXAMPLE
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Eliminazione dell'account nella Guida di amministrazione di Amazon Chime.

- Per i dettagli sull'API, consulta DeleteAccount AWS CLICommand Reference.

`delete-phone-number`

Il seguente esempio di codice mostra come utilizzare`delete-phone-number`.

AWS CLI

Come eliminare un numero di telefono

L'esempio `delete-phone-number` seguente sposta il numero di telefono specificato nella coda di eliminazione.

```
aws chime delete-phone-number \
    --phone-number-id "+12065550100"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Utilizzo dei numeri di telefono nella Guida di amministrazione di Amazon Chime.

- Per i dettagli sull'API, consulta DeletePhoneNumber AWS CLICommand Reference.

`delete-room-membership`

Il seguente esempio di codice mostra come utilizzare`delete-room-membership`.

AWS CLI

Come rimuovere un utente membro di una chat room

L'esempio `delete-room-membership` seguente rimuove il membro specificato dalla chat room specificata.

```
aws chime delete-room-membership \
    --account-id 12a3456b-7c89-012d-3456-78901e23fg45 \
    --room-id abcd1e2d-3e45-6789-01f2-3g45h67i890j \
    --member-id 1ab2345c-67de-8901-f23g-45h678901j2k
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Creazione di una chat room nella Guida per l'utente di Amazon Chime.

- Per i dettagli sull'API, consulta DeleteRoomMembership AWS CLICommand Reference.

`delete-room`

Il seguente esempio di codice mostra come utilizzare`delete-room`.

AWS CLI

Come eliminare una chat room

L'esempio `delete-room` seguente elimina la chat room specificata e rimuove le appartenenze alla chat room.

```
aws chime delete-room \
    --account-id 12a3456b-7c89-012d-3456-78901e23fg45 \
    --room-id abcd1e2d-3e45-6789-01f2-3g45h67i890j
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Creazione di una chat room nella Guida per l'utente di Amazon Chime.

- Per i dettagli sull'API, consulta DeleteRoom AWS CLICommand Reference.

`disassociate-phone-number-from-user`

Il seguente esempio di codice mostra come utilizzare`disassociate-phone-number-from-user`.

AWS CLI

Come annullare l'associazione di un numero di telefono a un utente

L'esempio `disassociate-phone-number-from-user` seguente annulla l'associazione di un numero di telefono all'utente specificato.

```
aws chime disassociate-phone-number-from-user \
    --account-id 12a3456b-7c89-012d-3456-78901e23fg45 \
    --user-id 1ab2345c-67de-8901-f23g-45h678901j2k
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Gestione dei numeri di telefono degli utenti nella Guida all'amministrazione di Amazon Chime.

- Per i dettagli sull'API, consulta DisassociatePhoneNumberFromUser AWS CLICommand Reference.

`disassociate-signin-delegate-groups-from-account`

Il seguente esempio di codice mostra come utilizzare`disassociate-signin-delegate-groups-from-account`.

AWS CLI

Come annullare l'associazione di gruppi di delegati di accesso

L'esempio `disassociate-signin-delegate-groups-from-account` seguente annulla l'associazione del gruppo di delegati di accesso specificato all'account Amazon Chime specificato.

```
aws chime disassociate-signin-delegate-groups-from-account \
    --account-id 12a3456b-7c89-012d-3456-78901e23fg45 \
    --group-names "my_users"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Gestione dell'accesso e delle autorizzazioni degli utenti nella Guida all'amministrazione di Amazon Chime.

- Per i dettagli sull'API, consulta DisassociateSigninDelegateGroupsFromAccount AWS CLICommand Reference.

`get-account-settings`

Il seguente esempio di codice mostra come utilizzare`get-account-settings`.

AWS CLI

Come recuperare le impostazioni di un account

L'esempio `get-account-settings` seguente recupera le impostazioni dell'account specificato.

```
aws chime get-account-settings --account-id a1b2c3d4-5678-90ab-cdef-11111EXAMPLE
```

Output:

```
{
    "AccountSettings": {
        "DisableRemoteControl": false,
        "EnableDialOut": false
    }
}
```

Per ulteriori informazioni, consulta Gestione degli account Amazon Chime nella Guida di amministrazione di Amazon Chime.

- Per i dettagli sull'API, consulta GetAccountSettings AWS CLICommand Reference.

`get-account`

Il seguente esempio di codice mostra come utilizzare`get-account`.

AWS CLI

Come recuperare i dettagli di un account

L'esempio `get-account` seguente recupera i dettagli dell'account Amazon Chime specificato.

```
aws chime get-account \
    --account-id a1b2c3d4-5678-90ab-cdef-11111EXAMPLE
```

Output:

```
{
    "Account": {
        "AwsAccountId": "111122223333",
        "AccountId": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
        "Name": "EnterpriseDirectory",
        "AccountType": "EnterpriseDirectory",
        "CreatedTimestamp": "2018-12-20T18:38:02.181Z",
        "DefaultLicense": "Pro",
        "SupportedLicenses": [
            "Basic",
            "Pro"
        ],
        "SigninDelegateGroups": [
            {
                "GroupName": "myGroup"
            },
        ]
    }
}
```

Per ulteriori informazioni, consulta Gestione degli account Amazon Chime nella Guida di amministrazione di Amazon Chime.

- Per i dettagli sull'API, consulta GetAccount AWS CLICommand Reference.

`get-bot`

Il seguente esempio di codice mostra come utilizzare`get-bot`.

AWS CLI

Come recuperare i dettagli su un bot

L'esempio `get-bot` seguente visualizza i dettagli del bot specificato.

```
aws chime get-bot \
    --account-id 12a3456b-7c89-012d-3456-78901e23fg45 \
    --bot-id 123abcd4-5ef6-789g-0h12-34j56789012k
```

Output:

```
{
    "Bot": {
        "BotId": "123abcd4-5ef6-789g-0h12-34j56789012k",
        "UserId": "123abcd4-5ef6-789g-0h12-34j56789012k",
        "DisplayName": "myBot (Bot)",
        "BotType": "ChatBot",
        "Disabled": false,
        "CreatedTimestamp": "2019-09-09T18:05:56.749Z",
        "UpdatedTimestamp": "2019-09-09T18:05:56.749Z",
        "BotEmail": "myBot-chimebot@example.com",
        "SecurityToken": "wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY"
    }
}
```

Per ulteriori informazioni, consulta Aggiornare i chat bot nella Guida per gli sviluppatori di Amazon Chime.

- Per i dettagli sull'API, consulta GetBot AWS CLICommand Reference.

`get-global-settings`

Il seguente esempio di codice mostra come utilizzare`get-global-settings`.

AWS CLI

Come ottenere le impostazioni globali

L'`get-global-settings`esempio seguente recupera i nomi dei bucket S3 utilizzati per archiviare i record dei dettagli delle chiamate per Amazon Chime Business Calling e Amazon Chime Voice Connectors associati all'account dell'amministratore. AWS

```
aws chime get-global-settings
```

Output:

```
{
    "BusinessCalling": {
        "CdrBucket": "s3bucket"
    },
    "VoiceConnector": {
        "CdrBucket": "s3bucket"
    }
}
```

Per ulteriori informazioni, consulta Gestione delle impostazioni globali nella Guida di amministrazione di Amazon Chime.

- Per i dettagli sull'API, consulta Command Reference. GetGlobalSettingsAWS CLI

`get-phone-number-order`

Il seguente esempio di codice mostra come utilizzare`get-phone-number-order`.

AWS CLI

Come ottenere i dettagli dell'ordine di numeri di telefono

L'esempio `get-phone-number-order` seguente visualizza i dettagli dell'ordine di numeri di telefono specificato.

```
aws chime get-phone-number-order \
    --phone-number-order-id abc12345-de67-89f0-123g-h45i678j9012
```

Output:

```
{
    "PhoneNumberOrder": {
        "PhoneNumberOrderId": "abc12345-de67-89f0-123g-h45i678j9012",
        "ProductType": "VoiceConnector",
        "Status": "Partial",
        "OrderedPhoneNumbers": [
            {
              "E164PhoneNumber": "+12065550100",
              "Status": "Acquired"
            },
```

```
            {
                "E164PhoneNumber": "+12065550101",
                "Status": "Acquired"
            },
            {
                "E164PhoneNumber": "+12065550102",
                "Status": "Failed"
            }
        ],
        "CreatedTimestamp": "2019-08-09T21:35:21.427Z",
        "UpdatedTimestamp": "2019-08-09T21:35:31.926Z"
    }
}
```

Per ulteriori informazioni, consulta Utilizzo dei numeri di telefono nella Guida di amministrazione di Amazon Chime.

- Per i dettagli sull'API, consulta GetPhoneNumberOrder AWS CLICommand Reference.

`get-phone-number-settings`

Il seguente esempio di codice mostra come utilizzare`get-phone-number-settings`.

AWS CLI

Come recuperare il nome di una chiamata in uscita

L'`get-phone-number-settings`esempio seguente recupera il nome di chiamata in uscita predefinito per l'account dell' AWS utente chiamante.

```
aws chime get-phone-number-settings
```

Questo comando non produce alcun output. Output:

```
{
    "CallingName": "myName",
    "CallingNameUpdatedTimestamp": "2019-10-28T18:56:42.911Z"
}
```

Per ulteriori informazioni, consulta Utilizzo dei numeri di telefono nella Guida di amministrazione di Amazon Chime.

- Per i dettagli sull'API, vedere GetPhoneNumberSettingsin AWS CLI Command Reference.

`get-phone-number`

Il seguente esempio di codice mostra come utilizzare`get-phone-number`.

AWS CLI

Come ottenere i dettagli del numero di telefono

L'esempio `get-phone-number` seguente visualizza i dettagli del numero di telefono specificato.

```
aws chime get-phone-number \
    --phone-number-id +12065550100
```

Output:

```
{
    "PhoneNumber": {
        "PhoneNumberId": "%2B12065550100",
        "E164PhoneNumber": "+12065550100",
        "Type": "Local",
        "ProductType": "VoiceConnector",
        "Status": "Unassigned",
        "Capabilities": {
            "InboundCall": true,
            "OutboundCall": true,
            "InboundSMS": true,
            "OutboundSMS": true,
            "InboundMMS": true,
            "OutboundMMS": true
        },
       "Associations": [
            {
                "Value": "abcdef1ghij2klmno3pqr4",
                "Name": "VoiceConnectorId",
                "AssociatedTimestamp": "2019-10-28T18:40:37.453Z"
            }
        ],
        "CallingNameStatus": "UpdateInProgress",
        "CreatedTimestamp": "2019-08-09T21:35:21.445Z",
        "UpdatedTimestamp": "2019-08-09T21:35:31.745Z"
    }
```

```
}
```

Per ulteriori informazioni, consulta Utilizzo dei numeri di telefono nella Guida di amministrazione di Amazon Chime.

- Per i dettagli sull'API, consulta GetPhoneNumber AWS CLICommand Reference.

`get-room`

Il seguente esempio di codice mostra come utilizzare`get-room`.

AWS CLI

Come ottenere i dettagli di una chat room

L'esempio `get-room` seguente visualizza i dettagli della chat room specificata.

```
aws chime get-room \
    --account-id 12a3456b-7c89-012d-3456-78901e23fg45 \
    --room-id abcd1e2d-3e45-6789-01f2-3g45h67i890j
```

Output:

```
{
    "Room": {
        "RoomId": "abcd1e2d-3e45-6789-01f2-3g45h67i890j",
        "Name": "chatRoom",
        "AccountId": "12a3456b-7c89-012d-3456-78901e23fg45",
        "CreatedBy": "arn:aws:iam::111122223333:user/alejandro",
        "CreatedTimestamp": "2019-12-02T22:29:31.549Z",
        "UpdatedTimestamp": "2019-12-02T22:29:31.549Z"
    }
}
```

Per ulteriori informazioni, consulta Creazione di una chat room nella Guida per l'utente di Amazon Chime.

- Per i dettagli sull'API, consulta GetRoom AWS CLICommand Reference.

`get-user-settings`

Il seguente esempio di codice mostra come utilizzare`get-user-settings`.

AWS CLI

Come recuperare le impostazioni dell'utente

L'esempio `get-user-settings` seguente visualizza le impostazioni dell'utente specificate.

```
aws chime get-user-settings \
    --account-id 12a3456b-7c89-012d-3456-78901e23fg45 \
    --user-id 1ab2345c-67de-8901-f23g-45h678901j2k
```

Output:

```
{
    "UserSettings": {
        "Telephony": {
            "InboundCalling": true,
            "OutboundCalling": true,
            "SMS": true
        }
    }
}
```

Per ulteriori informazioni, consulta Gestione dei numeri di telefono degli utenti nella Guida all'amministrazione di Amazon Chime.

- Per i dettagli sull'API, consulta GetUserSettings AWS CLICommand Reference.

`get-user`

Il seguente esempio di codice mostra come utilizzare`get-user`.

AWS CLI

Come ottenere i dettagli di un utente

L'esempio `get-user` seguente recupera i dettagli dell'utente specificato.

```
aws chime get-user \
    --account-id a1b2c3d4-5678-90ab-cdef-11111EXAMPLE \
    --user-id a1b2c3d4-5678-90ab-cdef-22222EXAMPLE
```

Output:

```
{
    "User": {
        "UserId": "a1b2c3d4-5678-90ab-cdef-22222EXAMPLE",
        "AccountId": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
        "PrimaryEmail": "marthar@example.com",
        "DisplayName": "Martha Rivera",
        "LicenseType": "Pro",
        "UserRegistrationStatus": "Registered",
        "RegisteredOn": "2018-12-20T18:45:25.231Z",
        "InvitedOn": "2018-12-20T18:45:25.231Z",
        "AlexaForBusinessMetadata": {
            "IsAlexaForBusinessEnabled": False,
            "AlexaForBusinessRoomArn": "null"
        },
        "PersonalPIN": "XXXXXXXXXX"
    }
}
```

Per ulteriori informazioni, consulta Gestione degli utenti nella Guida di amministrazione di Amazon Chime.

- Per i dettagli sull'API, consulta GetUser AWS CLICommand Reference.

`invite-users`

Il seguente esempio di codice mostra come utilizzare`invite-users`.

AWS CLI

Come invitare gli utenti ad Amazon Chime

L'esempio `invite-users` seguente invia un'e-mail per invitare un utente nell'account Amazon Chime specificato.

```
aws chime invite-users \
    --account-id a1b2c3d4-5678-90ab-cdef-11111EXAMPLE \
    --user-email-list "alejandror@example.com" "janed@example.com"
```

Output:

```
{
    "Invites": [
        {
            "InviteId": "a1b2c3d4-5678-90ab-cdef-22222EXAMPLE",
            "Status": "Pending",
            "EmailAddress": "alejandror@example.com",
            "EmailStatus": "Sent"
        }
        {
            "InviteId": "a1b2c3d4-5678-90ab-cdef-33333EXAMPLE",
            "Status": "Pending",
            "EmailAddress": "janed@example.com",
            "EmailStatus": "Sent"
        }
    ]
}
```

Per ulteriori informazioni, consulta Invito e sospensione degli utenti nella Guida di amministrazione di Amazon Chime.

- Per i dettagli sull'API, consulta InviteUsers AWS CLICommand Reference.

`list-accounts`

Il seguente esempio di codice mostra come utilizzare`list-accounts`.

AWS CLI

Come ottenere un elenco di account

L'`list-accounts`esempio seguente recupera un elenco degli account Amazon Chime nell'account dell'amministratore AWS .

```
aws chime list-accounts
```

Output:

```
{
    "Accounts": [
        {
            "AwsAccountId": "111122223333",
```

```
            "AccountId": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
            "Name": "First Chime Account",
            "AccountType": "EnterpriseDirectory",
            "CreatedTimestamp": "2018-12-20T18:38:02.181Z",
            "DefaultLicense": "Pro",
            "SupportedLicenses": [
                "Basic",
                "Pro"
            ],
            "SigninDelegateGroups": [
                {
                    "GroupName": "myGroup"
                },
            ]
        },
        {
            "AwsAccountId": "111122223333",
            "AccountId": "a1b2c3d4-5678-90ab-cdef-22222EXAMPLE",
            "Name": "Second Chime Account",
            "AccountType": "Team",
            "CreatedTimestamp": "2018-09-04T21:44:22.292Z",
            "DefaultLicense": "Pro",
            "SupportedLicenses": [
                "Basic",
                "Pro"
            ],
            "SigninDelegateGroups": [
                {
                    "GroupName": "myGroup"
                },
            ]
        }
    ]
}
```

Per ulteriori informazioni, consulta Gestione degli account Amazon Chime nella Guida di amministrazione di Amazon Chime.

- Per i dettagli sull'API, consulta AWS CLI Command ListAccountsReference.

`list-bots`

Il seguente esempio di codice mostra come utilizzare`list-bots`.

AWS CLI

Come recuperare un elenco di bot

L'esempio `list-bots` seguente elenca i bot associati all'account Amazon Chime Enterprise specificato.

```
aws chime list-bots \
    --account-id 12a3456b-7c89-012d-3456-78901e23fg45
```

Output:

```
{
    "Bot": {
        "BotId": "123abcd4-5ef6-789g-0h12-34j56789012k",
        "UserId": "123abcd4-5ef6-789g-0h12-34j56789012k",
        "DisplayName": "myBot (Bot)",
        "BotType": "ChatBot",
        "Disabled": false,
        "CreatedTimestamp": "2019-09-09T18:05:56.749Z",
        "UpdatedTimestamp": "2019-09-09T18:05:56.749Z",
        "BotEmail": "myBot-chimebot@example.com",
        "SecurityToken": "wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY"
    }
}
```

Per ulteriori informazioni, consulta Utilizzare chat bot con Amazon Chime nella Guida per gli sviluppatori di Amazon Chime.

- Per i dettagli sull'API, consulta ListBots AWS CLICommand Reference.

`list-phone-number-orders`

Il seguente esempio di codice mostra come utilizzare`list-phone-number-orders`.

AWS CLI

Come elencare gli ordini di numeri di telefono

L'esempio `list-phone-number-orders` seguente elenca gli ordini di numeri di telefono associati all'account dell'amministratore Amazon Chime.

```
aws chime list-phone-number-orders
```

Output:

```
{
    "PhoneNumberOrders": [
        {
            "PhoneNumberOrderId": "abc12345-de67-89f0-123g-h45i678j9012",
            "ProductType": "VoiceConnector",
            "Status": "Partial",
            "OrderedPhoneNumbers": [
                {
                    "E164PhoneNumber": "+12065550100",
                    "Status": "Acquired"
                },
                {
                    "E164PhoneNumber": "+12065550101",
                    "Status": "Acquired"
                },
                {
                    "E164PhoneNumber": "+12065550102",
                    "Status": "Failed"
                }
            ],
            "CreatedTimestamp": "2019-08-09T21:35:21.427Z",
            "UpdatedTimestamp": "2019-08-09T21:35:31.926Z"
        }
        {
            "PhoneNumberOrderId": "cba54321-ed76-09f5-321g-h54i876j2109",
            "ProductType": "BusinessCalling",
            "Status": "Partial",
            "OrderedPhoneNumbers": [
                {
                    "E164PhoneNumber": "+12065550103",
                    "Status": "Acquired"
                },
                {
                    "E164PhoneNumber": "+12065550104",
                    "Status": "Acquired"
                },
                {
                    "E164PhoneNumber": "+12065550105",
                    "Status": "Failed"
```

```
                }
            ],
            "CreatedTimestamp": "2019-08-09T21:35:21.427Z",
            "UpdatedTimestamp": "2019-08-09T21:35:31.926Z"
        }
    ]
}
```

Per ulteriori informazioni, consulta Utilizzo dei numeri di telefono nella Guida di amministrazione di Amazon Chime.

- Per i dettagli sull'API, consulta ListPhoneNumberOrders AWS CLICommand Reference.

`list-phone-numbers`

Il seguente esempio di codice mostra come utilizzare`list-phone-numbers`.

AWS CLI

Come elencare i numeri di telefono di un account Amazon Chime

L'esempio `list-phone-numbers` seguente elenca i numeri di telefono associati all'account Amazon Chime dell'amministratore.

```
aws chime list-phone-numbers
```

Questo comando non produce alcun output. Output:

```
{
    "PhoneNumbers": [
        {
            "PhoneNumberId": "%2B12065550100",
            "E164PhoneNumber": "+12065550100",
            "Type": "Local",
            "ProductType": "VoiceConnector",
            "Status": "Assigned",
            "Capabilities": {
                "InboundCall": true,
                "OutboundCall": true,
                "InboundSMS": true,
                "OutboundSMS": true,
                "InboundMMS": true,
                "OutboundMMS": true
```

```
            },
            "Associations": [
                {
                    "Value": "abcdef1ghij2klmno3pqr4",
                    "Name": "VoiceConnectorId",
                    "AssociatedTimestamp": "2019-10-28T18:40:37.453Z"
                }
            ],
            "CallingNameStatus": "UpdateInProgress",
            "CreatedTimestamp": "2019-08-12T22:10:20.521Z",
            "UpdatedTimestamp": "2019-10-28T18:42:07.964Z"
        },
        {
            "PhoneNumberId": "%2B12065550101",
            "E164PhoneNumber": "+12065550101",
            "Type": "Local",
            "ProductType": "VoiceConnector",
            "Status": "Assigned",
            "Capabilities": {
                "InboundCall": true,
                "OutboundCall": true,
                "InboundSMS": true,
                "OutboundSMS": true,
                "InboundMMS": true,
                "OutboundMMS": true
            },
            "Associations": [
                {
                    "Value": "abcdef1ghij2klmno3pqr4",
                    "Name": "VoiceConnectorId",
                    "AssociatedTimestamp": "2019-10-28T18:40:37.511Z"
                }
            ],
            "CallingNameStatus": "UpdateInProgress",
            "CreatedTimestamp": "2019-08-12T22:10:20.521Z",
            "UpdatedTimestamp": "2019-10-28T18:42:07.960Z"
        }
    ]
}
```

Per ulteriori informazioni, consulta Utilizzo dei numeri di telefono nella Guida di amministrazione di Amazon Chime.

- Per i dettagli sull'API, consulta ListPhoneNumbers AWS CLICommand Reference.

`list-room-memberships`

Il seguente esempio di codice mostra come utilizzare`list-room-memberships`.

AWS CLI

Come elencare le appartenenze a una chat room

L'esempio `list-room-memberships` seguente visualizza un elenco dei dettagli di appartenenza alla chat room specificata.

```
aws chime list-room-memberships \
    --account-id 12a3456b-7c89-012d-3456-78901e23fg45 \
    --room-id abcd1e2d-3e45-6789-01f2-3g45h67i890j
```

Output:

```
{
    "RoomMemberships": [
        {
            "RoomId": "abcd1e2d-3e45-6789-01f2-3g45h67i890j",
            "Member": {
                "MemberId": "2ab2345c-67de-8901-f23g-45h678901j2k",
                "MemberType": "User",
                "Email": "zhangw@example.com",
                "FullName": "Zhang Wei",
                "AccountId": "12a3456b-7c89-012d-3456-78901e23fg45"
            },
            "Role": "Member",
            "InvitedBy": "arn:aws:iam::111122223333:user/alejandro",
            "UpdatedTimestamp": "2019-12-02T22:46:58.532Z"
        },
        {
            "RoomId": "abcd1e2d-3e45-6789-01f2-3g45h67i890j",
            "Member": {
                "MemberId": "1ab2345c-67de-8901-f23g-45h678901j2k",
                "MemberType": "User",
                "Email": "janed@example.com",
                "FullName": "Jane Doe",
                "AccountId": "12a3456b-7c89-012d-3456-78901e23fg45"
            },
            "Role": "Administrator",
            "InvitedBy": "arn:aws:iam::111122223333:user/alejandro",
```

```
            "UpdatedTimestamp": "2019-12-02T22:46:58.532Z"
        }
    ]
}
```

Per ulteriori informazioni, consulta Creazione di una chat room nella Guida per l'utente di Amazon Chime.

- Per i dettagli sull'API, consulta ListRoomMemberships AWS CLICommand Reference.

`list-rooms`

Il seguente esempio di codice mostra come utilizzare`list-rooms`.

AWS CLI

Come elencare le chat room

L'esempio `list-rooms` seguente visualizza un elenco di chat room nell'account specificato. L'elenco viene filtrato solo in base alle chat room a cui appartiene il membro specificato.

```
aws chime list-rooms \
    --account-id 12a3456b-7c89-012d-3456-78901e23fg45 \
    --member-id 1ab2345c-67de-8901-f23g-45h678901j2k
```

Output:

```
{
    "Room": {
        "RoomId": "abcd1e2d-3e45-6789-01f2-3g45h67i890j",
        "Name": "teamRoom",
        "AccountId": "12a3456b-7c89-012d-3456-78901e23fg45",
        "CreatedBy": "arn:aws:iam::111122223333:user/alejandro",
        "CreatedTimestamp": "2019-12-02T22:29:31.549Z",
        "UpdatedTimestamp": "2019-12-02T22:33:19.310Z"
    }
}
```

Per ulteriori informazioni, consulta Creazione di una chat room nella Guida per l'utente di Amazon Chime.

- Per i dettagli sull'API, consulta ListRooms AWS CLICommand Reference.

`list-users`

Il seguente esempio di codice mostra come utilizzare`list-users`.

AWS CLI

Come elencare gli utenti di un account

L'esempio `list-users` seguente elenca gli utenti dell'account Amazon Chime specificato.

```
aws chime list-users --account-id a1b2c3d4-5678-90ab-cdef-11111EXAMPLE
```

Output:

```
{
    "Users": [
        {
            "UserId": "a1b2c3d4-5678-90ab-cdef-22222EXAMPLE",
            "AccountId": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
            "PrimaryEmail": "mariag@example.com",
            "DisplayName": "Maria Garcia",
            "LicenseType": "Pro",
            "UserType": "PrivateUser",
            "UserRegistrationStatus": "Registered",
            "RegisteredOn": "2018-12-20T18:45:25.231Z"
            "AlexaForBusinessMetadata": {
                "IsAlexaForBusinessEnabled": false
            }
        },
        {
            "UserId": "a1b2c3d4-5678-90ab-cdef-33333EXAMPLE",
            "AccountId": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
            "PrimaryEmail": "richardr@example.com",
            "DisplayName": "Richard Roe",
            "LicenseType": "Pro",
            "UserType": "PrivateUser",
            "UserRegistrationStatus": "Registered",
            "RegisteredOn": "2018-12-20T18:45:45.415Z"
            "AlexaForBusinessMetadata": {
                "IsAlexaForBusinessEnabled": false
            }
        },
        {
            "UserId": "a1b2c3d4-5678-90ab-cdef-44444EXAMPLE",
```

```
            "AccountId": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
            "PrimaryEmail": "saanvis@example.com",
            "DisplayName": "Saanvi Sarkar",
            "LicenseType": "Basic",
            "UserType": "PrivateUser",
            "UserRegistrationStatus": "Registered",
            "RegisteredOn": "2018-12-20T18:46:57.747Z"
            "AlexaForBusinessMetadata": {
                "IsAlexaForBusinessEnabled": false
            }
        },
        {
            "UserId": "a1b2c3d4-5678-90ab-cdef-55555EXAMPLE",
            "AccountId": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
            "PrimaryEmail": "wxiulan@example.com",
            "DisplayName": "Wang Xiulan",
            "LicenseType": "Basic",
            "UserType": "PrivateUser",
            "UserRegistrationStatus": "Registered",
            "RegisteredOn": "2018-12-20T18:47:15.390Z"
            "AlexaForBusinessMetadata": {
                "IsAlexaForBusinessEnabled": false
            }
        }
    ]
}
```

Per ulteriori informazioni, consulta Gestione degli utenti nella Guida di amministrazione di Amazon Chime.

- Per i dettagli sull'API, consulta ListUsers AWS CLICommand Reference.

`logout-user`

Il seguente esempio di codice mostra come utilizzare`logout-user`.

AWS CLI

Come disconnettere un utente

L'esempio `logout-user` seguente disconnette l'utente specificato.

```
aws chime logout-user \
```

```
    --account-id a1b2c3d4-5678-90ab-cdef-11111EXAMPLE \
    --user-id a1b2c3d4-5678-90ab-cdef-22222EXAMPLE
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta LogoutUser AWS CLICommand Reference.

`regenerate-security-token`

Il seguente esempio di codice mostra come utilizzare`regenerate-security-token`.

AWS CLI

Come rigenerare un token di sicurezza

L'esempio `regenerate-security-token` seguente rigenera il token di sicurezza per il bot specificato.

```
aws chime regenerate-security-token \
    --account-id 12a3456b-7c89-012d-3456-78901e23fg45 \
    --bot-id 123abcd4-5ef6-789g-0h12-34j56789012k
```

Output:

```
{
    "Bot": {
        "BotId": "123abcd4-5ef6-789g-0h12-34j56789012k",
        "UserId": "123abcd4-5ef6-789g-0h12-34j56789012k",
        "DisplayName": "myBot (Bot)",
        "BotType": "ChatBot",
        "Disabled": false,
        "CreatedTimestamp": "2019-09-09T18:05:56.749Z",
        "UpdatedTimestamp": "2019-09-09T18:05:56.749Z",
        "BotEmail": "myBot-chimebot@example.com",
        "SecurityToken": "je7MtGbClwBF/2Zp9Utk/h3yCo8nvbEXAMPLEKEY"
    }
}
```

Per ulteriori informazioni, consulta Autenticare le richieste dei chat bot nella Guida per gli sviluppatori di Amazon Chime.

- Per i dettagli sull'API, consulta RegenerateSecurityToken AWS CLICommand Reference.

`reset-personal-pin`

Il seguente esempio di codice mostra come utilizzare`reset-personal-pin`.

AWS CLI

Come reimpostare il PIN della riunione personale di un utente

L'esempio `reset-personal-pin` seguente reimposta il PIN della riunione personale dell'utente specificato.

```
aws chime reset-personal-pin \
    --account-id a1b2c3d4-5678-90ab-cdef-11111EXAMPLE
    --user-id a1b2c3d4-5678-90ab-cdef-22222EXAMPLE
```

Output:

```
{
    "User": {
        "UserId": "a1b2c3d4-5678-90ab-cdef-22222EXAMPLE",
        "AccountId": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
        "PrimaryEmail": "mateo@example.com",
        "DisplayName": "Mateo Jackson",
        "LicenseType": "Pro",
        "UserType": "PrivateUser",
        "UserRegistrationStatus": "Registered",
        "RegisteredOn": "2018-12-20T18:45:25.231Z",
        "AlexaForBusinessMetadata": {
            "IsAlexaForBusinessEnabled": False,
            "AlexaForBusinessRoomArn": "null"
        },
        "PersonalPIN": "XXXXXXXXXX"
    }
}
```

Per ulteriori informazioni, consulta Changing Personal Meeting PINs nella Amazon Chime Administration Guide.

- Per i dettagli sull'API, consulta ResetPersonalPin AWS CLICommand Reference.

`restore-phone-number`

Il seguente esempio di codice mostra come utilizzare`restore-phone-number`.

AWS CLI

Come ripristinare un numero di telefono

L'esempio `restore-phone-number` seguente ripristina il numero di telefono specificato dalla coda di eliminazione.

```
aws chime restore-phone-number \
    --phone-number-id "+12065550100"
```

Output:

```
{
    "PhoneNumber": {
        "PhoneNumberId": "%2B12065550100",
        "E164PhoneNumber": "+12065550100",
        "Type": "Local",
        "ProductType": "BusinessCalling",
        "Status": "Unassigned",
        "Capabilities": {
            "InboundCall": true,
            "OutboundCall": true,
            "InboundSMS": true,
            "OutboundSMS": true,
            "InboundMMS": true,
            "OutboundMMS": true
        },
        "Associations": [],
        "CreatedTimestamp": "2019-08-09T21:35:21.445Z",
        "UpdatedTimestamp": "2019-08-12T22:06:36.355Z"
    }
}
```

Per ulteriori informazioni, consulta Utilizzo dei numeri di telefono nella Guida di amministrazione di Amazon Chime.

- Per i dettagli sull'API, consulta RestorePhoneNumber AWS CLICommand Reference.

`search-available-phone-numbers`

Il seguente esempio di codice mostra come utilizzare`search-available-phone-numbers`.

AWS CLI

Come cercare i numeri di telefono disponibili

L'esempio `search-available-phone-numbers` seguente cerca i numeri di telefono disponibili in base al prefisso.

```
aws chime search-available-phone-numbers \
    --area-code "206"
```

Output:

```
{
    "E164PhoneNumbers": [
        "+12065550100",
        "+12065550101",
        "+12065550102",
        "+12065550103",
        "+12065550104",
        "+12065550105",
        "+12065550106",
        "+12065550107",
        "+12065550108",
        "+12065550109",
    ]
}
```

Per ulteriori informazioni, consulta Utilizzo dei numeri di telefono nella Guida di amministrazione di Amazon Chime.

- Per i dettagli sull'API, consulta SearchAvailablePhoneNumbers AWS CLICommand Reference.

`update-account-settings`

Il seguente esempio di codice mostra come utilizzare`update-account-settings`.

AWS CLI

Come aggiornare le impostazioni dell'account

L'esempio `update-account-settings` seguente disabilita il controllo remoto delle schermate condivise per l'account Amazon Chime specificato.

```
aws chime update-account-settings \
    --account-id a1b2c3d4-5678-90ab-cdef-11111EXAMPLE \
    --account-settings DisableRemoteControl=true
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta UpdateAccountSettings AWS CLICommand Reference.

`update-account`

Il seguente esempio di codice mostra come utilizzare`update-account`.

AWS CLI

Come aggiornare un account

L'esempio `update-account` seguente aggiorna il nome dell'account specificato.

```
aws chime update-account \
    --account-id a1b2c3d4-5678-90ab-cdef-11111EXAMPLE \
    --name MyAccountName
```

Output:

```
{
    "Account": {
        "AwsAccountId": "111122223333",
        "AccountId": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
        "Name": "MyAccountName",
        "AccountType": "Team",
        "CreatedTimestamp": "2018-09-04T21:44:22.292Z",
        "DefaultLicense": "Pro",
        "SupportedLicenses": [
            "Basic",
            "Pro"
        ],
        "SigninDelegateGroups": [
            {
                "GroupName": "myGroup"
            },
        ]
    }
```

```
}
```

Per ulteriori informazioni, consulta Rinominare l'account nella Guida di amministrazione di Amazon Chime.

- Per i dettagli sull'API, consulta UpdateAccount AWS CLICommand Reference.

`update-bot`

Il seguente esempio di codice mostra come utilizzare`update-bot`.

AWS CLI

Come aggiornare un bot

L'esempio `update-bot` seguente aggiorna lo stato del bot specificato per impedirne l'esecuzione.

```
aws chime update-bot \
    --account-id 12a3456b-7c89-012d-3456-78901e23fg45 \
    --bot-id 123abcd4-5ef6-789g-0h12-34j56789012k \
    --disabled
```

Output:

```
{
    "Bot": {
        "BotId": "123abcd4-5ef6-789g-0h12-34j56789012k",
        "UserId": "123abcd4-5ef6-789g-0h12-34j56789012k",
        "DisplayName": "myBot (Bot)",
        "BotType": "ChatBot",
        "Disabled": true,
        "CreatedTimestamp": "2019-09-09T18:05:56.749Z",
        "UpdatedTimestamp": "2019-09-09T18:05:56.749Z",
        "BotEmail": "myBot-chimebot@example.com",
        "SecurityToken": "je7MtGbClwBF/2Zp9Utk/h3yCo8nvbEXAMPLEKEY"
    }
}
```

Per ulteriori informazioni, consulta Aggiornare i chat bot nella Guida per gli sviluppatori di Amazon Chime.

- Per i dettagli sull'API, consulta UpdateBot AWS CLICommand Reference.

`update-global-settings`

Il seguente esempio di codice mostra come utilizzare`update-global-settings`.

AWS CLI

Come aggiornare le impostazioni globali

L'`update-global-settings`esempio seguente aggiorna il bucket S3 utilizzato per archiviare i record dei dettagli delle chiamate per Amazon Chime Business Calling e Amazon Chime Voice Connectors associati all'account dell'amministratore. AWS

```
aws chime update-global-settings \
    --business-calling CdrBucket="s3bucket" \
    --voice-connector CdrBucket="s3bucket"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Gestione delle impostazioni globali nella Guida di amministrazione di Amazon Chime.

- Per i dettagli sull'API, consulta UpdateGlobalSettingsCommand Reference.AWS CLI

`update-phone-number-settings`

Il seguente esempio di codice mostra come utilizzare`update-phone-number-settings`.

AWS CLI

Come aggiornare il nome di una chiamata in uscita

L'`update-phone-number-settings`esempio seguente aggiorna il nome predefinito per le chiamate in uscita per l' AWS account dell'amministratore.

```
aws chime update-phone-number-settings \
    --calling-name "myName"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Utilizzo dei numeri di telefono nella Guida di amministrazione di Amazon Chime.

- Per i dettagli sull'API, consulta UpdatePhoneNumberSettings AWS CLICommand Reference.

`update-phone-number`

Il seguente esempio di codice mostra come utilizzare`update-phone-number`.

AWS CLI

Esempio 1: come aggiornare il tipo di prodotto per un numero di telefono

L'esempio `update-phone-number` seguente aggiorna il tipo di prodotto del numero di telefono specificato.

```
aws chime update-phone-number \
    --phone-number-id "+12065550100" \
    --product-type "BusinessCalling"
```

Output:

```
{
    "PhoneNumber": {
        "PhoneNumberId": "%2B12065550100",
        "E164PhoneNumber": "+12065550100",
        "Type": "Local",
        "ProductType": "BusinessCalling",
        "Status": "Unassigned",
        "Capabilities": {
            "InboundCall": true,
            "OutboundCall": true,
            "InboundSMS": true,
            "OutboundSMS": true,
            "InboundMMS": true,
            "OutboundMMS": true
        },
        "Associations": [],
        "CallingName": "phonenumber1",
        "CreatedTimestamp": "2019-08-09T21:35:21.445Z",
        "UpdatedTimestamp": "2019-08-12T21:44:07.591Z"
    }
```

```
}
```

Esempio 2: come aggiornare il nome della chiamata in uscita per un numero di telefono

L'esempio `update-phone-number` seguente aggiorna il nome della chiamata in uscita per il numero di telefono specificato.

aws chime update-phone-number - phone-number-id «+12065550100" --chiamando «phonenumber2"

Output:

```
{
    "PhoneNumber": {
        "PhoneNumberId": "%2B12065550100",
        "E164PhoneNumber": "+12065550100",
        "Type": "Local",
        "ProductType": "BusinessCalling",
        "Status": "Unassigned",
        "Capabilities": {
            "InboundCall": true,
            "OutboundCall": true,
            "InboundSMS": true,
            "OutboundSMS": true,
            "InboundMMS": true,
            "OutboundMMS": true
        },
        "Associations": [],
        "CallingName": "phonenumber2",
        "CreatedTimestamp": "2019-08-09T21:35:21.445Z",
        "UpdatedTimestamp": "2019-08-12T21:44:07.591Z"
    }
}
```

Per ulteriori informazioni, consulta Utilizzo dei numeri di telefono nella Guida di amministrazione di Amazon Chime.

- Per i dettagli sull'API UpdatePhoneNumber, consulta AWS CLI Command Reference.

`update-room-membership`

Il seguente esempio di codice mostra come utilizzare`update-room-membership`.

AWS CLI

Come aggiornare un'appartenenza a una chat room

L'esempio `update-room-membership` seguente modifica il ruolo del membro della chat room specificato in `Administrator`.

```
aws chime update-room-membership \
    --account-id 12a3456b-7c89-012d-3456-78901e23fg45 \
    --room-id abcd1e2d-3e45-6789-01f2-3g45h67i890j \
    --member-id 1ab2345c-67de-8901-f23g-45h678901j2k \
    --role Administrator
```

Output:

```
{
    "RoomMembership": {
        "RoomId": "abcd1e2d-3e45-6789-01f2-3g45h67i890j",
        "Member": {
            "MemberId": "1ab2345c-67de-8901-f23g-45h678901j2k",
            "MemberType": "User",
            "Email": "sofiamartinez@example.com",
            "FullName": "Sofia Martinez",
            "AccountId": "12a3456b-7c89-012d-3456-78901e23fg45"
        },
        "Role": "Administrator",
        "InvitedBy": "arn:aws:iam::111122223333:user/admin",
        "UpdatedTimestamp": "2019-12-02T22:40:22.931Z"
    }
}
```

Per ulteriori informazioni, consulta Creazione di una chat room nella Guida per l'utente di Amazon Chime.

- Per i dettagli sull'API, consulta UpdateRoomMembership AWS CLICommand Reference.

`update-room`

Il seguente esempio di codice mostra come utilizzare`update-room`.

AWS CLI

Come aggiornare una chat room

L'esempio `update-room` seguente modifica il nome della chat room specificata.

```
aws chime update-room \
    --account-id 12a3456b-7c89-012d-3456-78901e23fg45 \
    --room-id abcd1e2d-3e45-6789-01f2-3g45h67i890j \
    --name teamRoom
```

Output:

```
{
    "Room": {
        "RoomId": "abcd1e2d-3e45-6789-01f2-3g45h67i890j",
        "Name": "teamRoom",
        "AccountId": "12a3456b-7c89-012d-3456-78901e23fg45",
        "CreatedBy": "arn:aws:iam::111122223333:user/alejandro",
        "CreatedTimestamp": "2019-12-02T22:29:31.549Z",
        "UpdatedTimestamp": "2019-12-02T22:33:19.310Z"
    }
}
```

Per ulteriori informazioni, consulta Creazione di una chat room nella Guida per l'utente di Amazon Chime.

- Per i dettagli sull'API, consulta UpdateRoom AWS CLICommand Reference.

`update-user-settings`

Il seguente esempio di codice mostra come utilizzare`update-user-settings`.

AWS CLI

Come aggiornare le impostazioni dell'utente

L'esempio `update-user-settings` seguente consente all'utente specificato di effettuare chiamate in entrata e in uscita e inviare e ricevere messaggi SMS.

```
aws chime update-user-settings \
    --account-id 12a3456b-7c89-012d-3456-78901e23fg45 \
    --user-id 1ab2345c-67de-8901-f23g-45h678901j2k \
    --user-settings "Telephony={InboundCalling=true,OutboundCalling=true,SMS=true}"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Gestione dei numeri di telefono degli utenti nella Guida all'amministrazione di Amazon Chime.

- Per i dettagli sull'API, consulta UpdateUserSettings AWS CLICommand Reference.

`update-user`

Il seguente esempio di codice mostra come utilizzare`update-user`.

AWS CLI

Come aggiornare i dettagli dell'utente

Questo esempio aggiorna i dettagli specificati dell'utente specificato.

Comando:

```
aws chime update-user \
    --account-id a1b2c3d4-5678-90ab-cdef-11111EXAMPLE \
    --user-id a1b2c3d4-5678-90ab-cdef-22222EXAMPLE \
    --license-type "Basic"
```

Output:

```
{
    "User": {
        "UserId": "a1b2c3d4-5678-90ab-cdef-22222EXAMPLE"
    }
}
```

- Per i dettagli sull'API, consulta UpdateUser AWS CLICommand Reference.

Esempi di API Cloud Control che utilizzano AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando l'API AWS Command Line Interface with Cloud Control.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`create-resource`

Il seguente esempio di codice mostra come utilizzare`create-resource`.

AWS CLI

Come creare una risorsa

L'`create-resource`esempio seguente crea una risorsa AWS: :Kinesis: :Stream, denominata ResourceExample, con un periodo di conservazione di 168 ore e un numero di frammenti pari a tre.

```
aws cloudcontrol create-resource \
    --type-name AWS::Kinesis::Stream \
    --desired-state "{\"Name\": \"ResourceExample\",\"RetentionPeriodHours\":168,
 \"ShardCount\":3}"
```

Output:

```
{
    "ProgressEvent": {
        "EventTime": 1632506656.706,
        "TypeName": "AWS::Kinesis::Stream",
        "OperationStatus": "IN_PROGRESS",
        "Operation": "CREATE",
        "Identifier": "ResourceExample",
        "RequestToken": "20999d87-e304-4725-ad84-832dcbfd7fc5"
    }
}
```

Per ulteriori informazioni, consulta Creazione di una risorsa nella Guida per l'utente di Cloud Control API.

- Per i dettagli sull'API, consulta CreateResourceCommand Reference.AWS CLI

`delete-resource`

Il seguente esempio di codice mostra come utilizzare`delete-resource`.

AWS CLI

Come eliminare una risorsa

L'`delete-resource`esempio seguente elimina una risorsa AWS: :Kinesis: :Stream con l'identificatore ResourceExample dal tuo account. AWS

```
aws cloudcontrol delete-resource \
    --type-name AWS::Kinesis::Stream \
    --identifier ResourceExample
```

Output:

```
{
    "ProgressEvent": {
        "TypeName": "AWS::Kinesis::Stream",
        "Identifier": "ResourceExample",
        "RequestToken": "e48f26ff-d0f9-4ab8-a878-120db1edf111",
        "Operation": "DELETE",
        "OperationStatus": "IN_PROGRESS",
        "EventTime": 1632950300.14
    }
}
```

Per ulteriori informazioni, consulta Eliminazione di una risorsa nella Guida per l'utente di Cloud Control API.

- Per i dettagli sull'API, consulta Command Reference. DeleteResourceAWS CLI

`get-resource-request-status`

Il seguente esempio di codice mostra come utilizzare`get-resource-request-status`.

AWS CLI

Come ottenere informazioni sullo stato di una richiesta di risorse

L'esempio `get-resource-request-status` seguente restituisce informazioni sullo stato della richiesta di risorse specificata.

```
aws cloudcontrol get-resource-request-status \
    --request-token "e1a6b86e-46bd-41ac-bfba-001234567890"
```

Output:

```
{
    "ProgressEvent": {
        "TypeName": "AWS::Kinesis::Stream",
        "Identifier": "Demo",
        "RequestToken": "e1a6b86e-46bd-41ac-bfba-001234567890",
        "Operation": "CREATE",
        "OperationStatus": "FAILED",
        "EventTime": 1632950268.481,
        "StatusMessage": "Resource of type 'AWS::Kinesis::Stream' with identifier
 'Demo' already exists.",
        "ErrorCode": "AlreadyExists"
    }
}
```

Per ulteriori informazioni, consulta Gestione delle richieste di operazioni sulle risorse nella Guida per l'utente di Cloud Control API.

- Per i dettagli sull'API, consulta GetResourceRequestStatus AWS CLICommand Reference.

`get-resource`

Il seguente esempio di codice mostra come utilizzare`get-resource`.

AWS CLI

Come ottenere lo stato attuale di una risorsa

L'`get-resource`esempio seguente restituisce lo stato corrente della risorsa AWS: :Kinesis: :Stream denominata. ResourceExample

```
aws cloudcontrol get-resource \
    --type-name AWS::Kinesis::Stream \
    --identifier ResourceExample
```

Output:

```
{
    "TypeName": "AWS::Kinesis::Stream",
    "ResourceDescription": {
        "Identifier": "ResourceExample",
        "Properties": "{\"Arn\":\"arn:aws:kinesis:us-west-2:099908667365:stream/
ResourceExample\",\"RetentionPeriodHours\":168,\"Name\":\"ResourceExample\",
\"ShardCount\":3}"
    }
}
```

Per ulteriori informazioni, consulta Lettura dello stato corrente di una risorsa nella Guida per l'utente di Cloud Control API.

- Per i dettagli sull'API, consulta Command GetResourceReference AWS CLI .

`list-resource-requests`

Il seguente esempio di codice mostra come utilizzare`list-resource-requests`.

AWS CLI

Come elencare le richieste di operazioni per le risorse attive

L'`list-resource-requests`esempio seguente elenca le richieste di risorse per le operazioni CREATE e UPDATE che non sono riuscite nell' AWS account.

```
aws cloudcontrol list-resource-requests \
    --resource-request-status-filter Operations=CREATE,OperationStatuses=FAILED
```

Output:

```
{
    "ResourceRequestStatusSummaries": [
        {
            "TypeName": "AWS::Kinesis::Stream",
            "Identifier": "Demo",
            "RequestToken": "e1a6b86e-46bd-41ac-bfba-633abcdfdbd7",
            "Operation": "CREATE",
            "OperationStatus": "FAILED",
```

```
            "EventTime": 1632950268.481,
            "StatusMessage": "Resource of type 'AWS::Kinesis::Stream' with
 identifier 'Demo' already exists.",
            "ErrorCode": "AlreadyExists"
        }
    ]
}
```

Per ulteriori informazioni, consulta Gestione delle richieste di operazioni sulle risorse nella Guida per l'utente di Cloud Control API.

- Per i dettagli sull'API, consulta ListResourceRequests AWS CLICommand Reference.

`list-resources`

Il seguente esempio di codice mostra come utilizzare`list-resources`.

AWS CLI

Come elencare le risorse di un determinato tipo

L'`list-resources`esempio seguente elenca le risorse AWS: :Kinesis: :Stream fornite nel tuo account. AWS

```
aws cloudcontrol list-resources \
    --type-name AWS::Kinesis::Stream
```

Output:

```
{
    "TypeName": "AWS::Kinesis::Stream",
    "ResourceDescriptions": [
        {
            "Identifier": "MyKinesisStream",
            "Properties": "{\"Name\":\"MyKinesisStream\"}"
        },
        {
            "Identifier": "AnotherStream",
            "Properties": "{\"Name\":\"AnotherStream\"}"
        }
    ]
```

```
}
```

Per ulteriori informazioni, consulta Rilevamento delle risorse nella Guida per l'utente di Cloud Control API.

- Per i dettagli sull'API, consulta ListResourcesCommand Reference.AWS CLI

`update-resource`

Il seguente esempio di codice mostra come utilizzare`update-resource`.

AWS CLI

Come aggiornare le proprietà di una risorsa esistente

L'`update-resource`esempio seguente aggiorna la politica di conservazione di una LogGroup risorsa AWS: :Logs:: denominata ExampleLogGroup a 90 giorni.

```
aws cloudcontrol update-resource \
    --type-name AWS::Logs::LogGroup \
    --identifier ExampleLogGroup \
    --patch-document "[{\"op\":\"replace\",\"path\":\"/RetentionInDays\",\"value
\":90}]"
```

Output:

```
{
    "ProgressEvent": {
        "EventTime": "2021-08-09T18:17:15.219Z",
        "TypeName": "AWS::Logs::LogGroup",
        "OperationStatus": "IN_PROGRESS",
        "Operation": "UPDATE",
        "Identifier": "ExampleLogGroup",
        "RequestToken": "5f40c577-3534-4b20-9599-0b0123456789"
    }
}
```

Per ulteriori informazioni, consulta Aggiornamento di una risorsa nella Guida per l'utente di Cloud Control API.

- Per i dettagli sull'API, consulta AWS CLI Command UpdateResourceReference.

AWS Cloud Map esempi utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with AWS Cloud Map.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`create-http-namespace`

Il seguente esempio di codice mostra come utilizzare`create-http-namespace`.

AWS CLI

Come creare un namespace HTTP

L'esempio `create-http-namespace` seguente crea un namespace HTTP denominato `example.com`.

```
aws servicediscovery create-http-namespace \
    --name example.com \
    --creator-request-id example-request-id
```

Output:

```
{
    "OperationId": "gv4g5meo7ndmeh4fqskygvk23d2fijwa-k9302yzd"
}
```

Per confermare che l'operazione è riuscita, puoi eseguire il comando `get-operation`. Per ulteriori informazioni, consulta get-operation.

Per ulteriori informazioni sulla creazione di uno spazio dei nomi, vedere Creating an AWS Cloud Map namespace per raggruppare i servizi applicativi nella AWS Cloud Map Developer Guide.

- Per i dettagli sull'API, consulta CreateHttpNamespaceCommand Reference.AWS CLI

`create-private-dns-namespace`

Il seguente esempio di codice mostra come utilizzare`create-private-dns-namespace`.

AWS CLI

Come creare un namespace DNS privato

L'esempio `create-private-dns-namespace` seguente crea un namespace DNS privato.

```
aws servicediscovery create-private-dns-namespace \
    --name example.com \
    --vpc vpc-1c56417b
```

Output:

```
{
    "OperationId": "gv4g5meo7ndmeh4fqskygvk23d2fijwa-k9302yzd"
}
```

Per confermare che l'operazione è riuscita, puoi eseguire il comando `get-operation`. Per ulteriori informazioni, consulta get-operation.

Per ulteriori informazioni, consulta Creazione degli spazi dei nomi nella Guida per gli sviluppatori di AWS Cloud Map.

- Per i dettagli sull'API, consulta CreatePrivateDnsNamespace AWS CLICommand Reference.

`create-public-dns-namespace`

Il seguente esempio di codice mostra come utilizzare`create-public-dns-namespace`.

AWS CLI

Come creare un namespace DNS pubblico

L'esempio `create-public-dns-namespace` seguente crea un namespace DNS pubblico denominato `example.com`.

```
aws servicediscovery create-public-dns-namespace \
    --name example-public-dns.com \
    --creator-request-id example-public-request-id \
    --properties DnsProperties={SOA={TTL=60}}
```

Output:

```
{
    "OperationId": "gv4g5meo7ndmeh4fqskygvk23d2fijwa-k9302yzd"
}
```

Per confermare che l'operazione è riuscita, puoi eseguire il comando `get-operation`.

Per ulteriori informazioni sulla creazione di uno spazio dei nomi, vedere Creating an AWS Cloud Map namespace per raggruppare i servizi applicativi nella AWS Cloud Map Developer Guide.

- Per i dettagli sull'API, consulta CreatePublicDnsNamespaceCommand Reference.AWS CLI

`create-service`

Il seguente esempio di codice mostra come utilizzare`create-service`.

AWS CLI

Esempio 1: creare un servizio utilizzando l'ID dello spazio dei nomi

L'esempio `create-service` seguente crea un servizio.

```
aws servicediscovery create-service \
    --name myservice \
    --namespace-id  ns-ylexjili4cdxy3xm \
    --dns-config "RoutingPolicy=MULTIVALUE,DnsRecords=[{Type=A,TTL=60}]"
```

Output:

```
{
    "Service": {
        "Id": "srv-abcd1234xmpl5678",
```

```
        "Arn": "arn:aws:servicediscovery:us-west-2:123456789012:service/srv-
abcd1234xmpl5678",
        "ResourceOwner": "123456789012",
        "Name": "myservice",
        "NamespaceId": "ns-abcd1234xmpl5678",
        "DnsConfig": {
            "NamespaceId": "ns-abcd1234xmpl5678",
            "RoutingPolicy": "MULTIVALUE",
            "DnsRecords": [
                {
                    "Type": "A",
                    "TTL": 60
                }
            ]
        },
        "Type": "DNS_HTTP",
        "CreateDate": "2025-08-18T13:45:31.023000-05:00",
        "CreatorRequestId": "abcd1234-5678-90ab-cdef-xmpl12345678",
        "CreatedByAccount": "123456789012"
    }
}
```

Per ulteriori informazioni, consulta Creazione di un servizio AWS Cloud Map per un componente applicativo nella AWS Cloud Map Developer Guide.

Esempio 2: creare un servizio utilizzando lo spazio dei nomi ARN

L'`create-service`esempio seguente crea un servizio utilizzando uno spazio dei nomi ARN anziché un ID dello spazio dei nomi. La specificazione di un namespace ARN è necessaria quando si crea un servizio in uno spazio dei nomi condiviso.

```
aws servicediscovery create-service \
    --name myservice-arn \
    --namespace-id arn:aws:servicediscovery:us-west-2:123456789012:namespace/ns-
abcd1234xmpl5678 \
    --dns-config "RoutingPolicy=MULTIVALUE,DnsRecords=[{Type=A,TTL=60}]"
```

Output:

```
{
    "Service": {
        "Id": "srv-abcd1234xmpl5678",
```

```
        "Arn": "arn:aws:servicediscovery:us-west-2:123456789012:service/srv-
abcd1234xmpl5678",
        "ResourceOwner": "123456789012",
        "Name": "myservice-arn",
        "NamespaceId": "ns-abcd1234xmpl5678",
        "DnsConfig": {
            "NamespaceId": "ns-abcd1234xmpl5678",
            "RoutingPolicy": "MULTIVALUE",
            "DnsRecords": [
                {
                    "Type": "A",
                    "TTL": 60
                }
            ]
        },
        "Type": "DNS_HTTP",
        "CreateDate": "2025-08-18T13:45:31.023000-05:00",
        "CreatorRequestId": "abcd1234-5678-90ab-cdef-xmpl12345678",
        "CreatedByAccount": "123456789012"
    }
}
```

Per ulteriori informazioni, consulta Shared AWS Cloud Map namespace nella AWS Cloud Map Developer Guide.

- Per i dettagli sull'API, consulta Command CreateServiceReference AWS CLI .

`delete-namespace`

Il seguente esempio di codice mostra come utilizzare`delete-namespace`.

AWS CLI

Esempio 1: Per eliminare un namespace

L'esempio `delete-namespace` seguente elimina un namespace.

```
aws servicediscovery delete-namespace \
    --id ns-abcd1234xmpl5678
```

Output:

```
{
```

```
    "OperationId": "abcd1234-5678-90ab-cdef-xmpl12345678"
}
```

Per confermare che l'operazione è riuscita, puoi eseguire il comando `get-operation`. Per ulteriori informazioni, consulta get-operation.

Per ulteriori informazioni, consulta Eliminazione di uno spazio dei nomi AWS Cloud Map nella AWS Cloud Map Developer Guide.

Esempio 2: Per eliminare uno spazio dei nomi utilizzando lo spazio dei nomi ARN

L'`delete-namespace`esempio seguente elimina uno spazio dei nomi utilizzando il relativo ARN.

```
aws servicediscovery delete-namespace \
    --id arn:aws:servicediscovery:us-west-2:123456789012:namespace/ns-
abcd1234xmpl5678
```

Output:

```
{
    "OperationId": "abcd1234-5678-90ab-cdef-xmpl12345678"
}
```

Per ulteriori informazioni, consulta Eliminazione di uno spazio dei nomi AWS Cloud Map nella AWS Cloud Map Developer Guide.

- Per i dettagli sull'API, consulta Command DeleteNamespaceReference AWS CLI .

`delete-service-attributes`

Il seguente esempio di codice mostra come utilizzare`delete-service-attributes`.

AWS CLI

Esempio 1: Per eliminare un attributo di servizio

L'esempio `delete-service-attributes` seguente elimina un attributo di servizio con la chiave `Port` associata al servizio specificato.

```
aws servicediscovery delete-service-attributes \
    --service-id srv-abcd1234xmpl5678 \
```

```
    --attributes Port
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Servizi AWS Cloud Map nella Guida per gli sviluppatori di AWS Cloud Map.

Esempio 2: Per eliminare un attributo di servizio utilizzando ARN

L'`delete-service-attributes`esempio seguente elimina un attributo di servizio utilizzando il servizio ARN. La specificazione dell'ARN è necessaria per eliminare gli attributi associati ai servizi creati nei namespace condivisi con l'account.

```
aws servicediscovery delete-service-attributes \
    --service-id arn:aws:servicediscovery:us-west-2:123456789012:service/srv-abcd1234xmpl5678 \
    --attributes Port
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta AWS Cloud Map services e Shared AWS Cloud Map namespace nella AWS Cloud Map Developer Guide.

- Per i dettagli sull'API, consulta Command DeleteServiceAttributesReference AWS CLI .

`delete-service`

Il seguente esempio di codice mostra come utilizzare`delete-service`.

AWS CLI

Esempio 1: Per eliminare un servizio

L'esempio `delete-service` seguente elimina un segreto.

```
aws servicediscovery delete-service \
    --id srv-abcd1234xmpl5678
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Eliminazione di un servizio AWS Cloud Map nella AWS Cloud Map Developer Guide.

Esempio 2: Per eliminare un servizio utilizzando ARN

L'`delete-service`esempio seguente elimina un servizio utilizzando il relativo ARN.

```
aws servicediscovery delete-service \
    --id arn:aws:servicediscovery:us-west-2:123456789012:service/srv-
abcd1234xmpl5678
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Eliminazione di un servizio AWS Cloud Map nella AWS Cloud Map Developer Guide.

- Per i dettagli sull'API, consulta DeleteService AWS CLICommand Reference.

`deregister-instance`

Il seguente esempio di codice mostra come utilizzare`deregister-instance`.

AWS CLI

Esempio 1: annullare la registrazione di un'istanza del servizio

L'esempio `deregister-instance` seguente annulla la registrazione di un'istanza del servizio.

```
aws servicediscovery deregister-instance \
    --service-id srv-p5zdwlg5uvvzjita \
    --instance-id myservice-53
```

Output:

```
{
    "OperationId": "4yejorelbukcjzpnr6tlmrghsjwpngf4-k98rnaiq"
}
```

Per confermare che l'operazione è riuscita, puoi eseguire il comando `get-operation`. Per ulteriori informazioni, vedere get-operation.

Per ulteriori informazioni, consulta Annullare la registrazione delle istanze del servizio nella Guida per gli sviluppatori di AWS Cloud Map.

Esempio 2: annullare la registrazione di un'istanza di servizio utilizzando il servizio ARN per gli spazi dei nomi condivisi

L'`deregister-instance`esempio seguente annulla la registrazione di un'istanza di servizio utilizzando un ARN del servizio anziché l'ID del servizio. È necessario specificare un ARN quando si annullano le istanze dai servizi creati in namespace condivisi con l'account.

```
aws servicediscovery deregister-instance \
    --service-id arn:aws:servicediscovery:us-west-2:123456789012:service/srv-p5zdwlg5uvvzjita \
    --instance-id web-server-01
```

Output:

```
{
    "OperationId": "gv4g5meo7ndmkqjrhpn39wk42xmpl"
}
```

Per ulteriori informazioni, consulta Shared AWS Cloud Map namespace e Annullamento della registrazione di un'istanza del servizio Cloud AWS Map nella Cloud Map Developer Guide AWS .

- Per i dettagli sull'API, consulta Command Reference. DeregisterInstanceAWS CLI

`discover-instances-revision`

Il seguente esempio di codice mostra come utilizzare`discover-instances-revision`.

AWS CLI

Esempio 1: scoprire la revisione di un'istanza

L'esempio `discover-instances-revision` seguente rileva la revisione crescente di un'istanza.

```
aws servicediscovery discover-instances-revision \
    --namespace-name example.com \
    --service-name myservice
```

Output:

```
{
```

```
    "InstancesRevision": 123456
}
```

Per ulteriori informazioni, consulta Istanze del servizio AWS Cloud Map nella Guida per gli sviluppatori di AWS Cloud Map.

Esempio 2: per scoprire la revisione delle istanze da uno specifico account proprietario

L'`discover-instances-revision`esempio seguente rileva la revisione delle istanze di uno specifico account proprietario. Il parametro owner-account è necessario per le istanze nei namespace condivisi con l'account.

```
aws servicediscovery discover-instances-revision \
    --namespace-name shared-namespace \
    --service-name shared-service \
    --owner-account 123456789111
```

Output:

```
{
    "InstancesRevision": 1234567890
}
```

Per ulteriori informazioni, consulta gli spazi dei nomi Shared AWS Cloud Map e le istanze del servizio AWS Cloud Map nella AWS Cloud Map Developer Guide.

- Per i dettagli sull'API, consulta DiscoverInstancesRevisionCommand Reference.AWS CLI

`discover-instances`

Il seguente esempio di codice mostra come utilizzare`discover-instances`.

AWS CLI

Esempio 1: per scoprire le istanze registrate

L'esempio `discover-instances` seguente rileva le istanze registrate.

```
aws servicediscovery discover-instances \
    --namespace-name example.com \
    --service-name myservice \
```

```
    --max-results 10 \
    --health-status ALL
```

Output:

```
{
    "Instances": [
        {
            "InstanceId": "myservice-53",
            "NamespaceName": "example.com",
            "ServiceName": "myservice",
            "HealthStatus": "UNKNOWN",
            "Attributes": {
                "AWS_INSTANCE_IPV4": "172.2.1.3",
                "AWS_INSTANCE_PORT": "808"
            }
        }
    ],
    "InstancesRevision": 85648075627387284
}
```

Per ulteriori informazioni, consulta Istanze del servizio AWS Cloud Map nella Guida per gli sviluppatori di AWS Cloud Map.

Esempio 2: per scoprire le istanze da uno specifico account proprietario

L'`discover-instances`esempio seguente scopre le istanze registrate da uno specifico account proprietario. Questo parametro è necessario per individuare le istanze nei namespace condivisi con l'account.

```
aws servicediscovery discover-instances \
    --namespace-name shared-namespace \
    --service-name shared-service \
    --owner-account 123456789111
```

Output:

```
{
    "Instances": [
        {
            "InstanceId": "shared-instance-1234",
            "NamespaceName": "shared-namespace",
```

```
            "ServiceName": "shared-service",
            "HealthStatus": "HEALTHY",
            "Attributes": {
                "AWS_INSTANCE_IPV4": "203.0.113.75",
                "AWS_INSTANCE_PORT": "80"
            }
        }
    ],
    "InstancesRevision": 1234567890
}
```

Per ulteriori informazioni, consulta gli spazi dei nomi Shared AWS Cloud Map e le istanze del servizio AWS Cloud Map nella AWS Cloud Map Developer Guide.

- Per i dettagli sull'API, consulta DiscoverInstancesCommand Reference.AWS CLI

`get-instance`

Il seguente esempio di codice mostra come utilizzare`get-instance`.

AWS CLI

Esempio 1: per ottenere i dettagli di un'istanza

L'esempio `get-instance` seguente ottiene gli attributi di un servizio.

```
aws servicediscovery get-instance \
    --service-id srv-e4anhexample0004
    --instance-id i-abcd1234
```

Output:

```
{
    "ResourceOwner": "123456789012",
    "Instance": {
        "Id": "arn:aws:servicediscovery:us-west-2:111122223333;:service/srv-
e4anhexample0004",
        "Attributes": {
            "AWS_INSTANCE_IPV4": "192.0.2.44",
            "AWS_INSTANCE_PORT": "80",
            "color": "green",
            "region": "us-west-2",
```

```
            "stage": "beta"
        },
        "CreatedByAccount": "123456789012"
    }
}
```

Per ulteriori informazioni, consulta Istanze del servizio AWS Cloud Map nella Guida per gli sviluppatori di AWS Cloud Map.

Esempio 2: per ottenere i dettagli di un'istanza utilizzando il servizio ARN per gli spazi dei nomi condivisi

L'`get-instance`esempio seguente ottiene gli attributi di un'istanza utilizzando un servizio ARN anziché un ID di servizio. È necessario specificare un ARN per ottenere i dettagli delle istanze associate ai namespace condivisi con l'account. L'istanza restituita in questo esempio è stata registrata per account `123456789111` in uno spazio dei nomi di proprietà dell'account. `123456789012`

```
aws servicediscovery get-instance \
    --service-id arn:aws:servicediscovery:us-west-2:123456789012:service/srv-p5zdwlg5uvvzjita \
    --instance-id web-server-01
```

Output:

```
{
    "ResourceOwner": "123456789012",
    "Instance": {
        "Id": "web-server-01",
        "Attributes": {
            "AWS_INSTANCE_IPV4": "203.0.113.15",
            "AWS_INSTANCE_PORT": "80"
        },
        "CreatedByAccount": "123456789111"
    }
}
```

Per ulteriori informazioni sulla condivisione dello spazio dei nomi tra account, consulta Shared AWS Cloud Map namespaces nella Cloud Map Developer Guide.AWS

- Per i dettagli sull'API, consulta Command Reference. GetInstanceAWS CLI

`get-instances-health-status`

Il seguente esempio di codice mostra come utilizzare`get-instances-health-status`.

AWS CLI

Esempio 1: per ottenere lo stato di integrità delle istanze associate a un servizio

L'esempio `get-instances-health-status` seguente ottiene lo stato dell'integrità delle istanze associate al servizio specificato.

```
aws servicediscovery get-instances-health-status \
    --service-id srv-e4anhexample0004
```

Output:

```
{
    "Status": {
        "i-abcd1234": "HEALTHY",
        "i-abcd1235": "UNHEALTHY"
    }
}
```

Per ulteriori informazioni, consulta Istanze del servizio AWS Cloud Map nella Guida per gli sviluppatori di AWS Cloud Map.

Esempio 2: per ottenere lo stato di integrità delle istanze utilizzando il servizio ARN per gli spazi dei nomi condivisi

L'`get-instances-health-status`esempio seguente ottiene lo stato di integrità delle istanze che utilizzano un ARN del servizio anziché l'ID del servizio. È necessario specificare un ARN per ottenere lo stato di integrità delle istanze associate ai namespace condivisi con l'account del richiedente.

```
aws servicediscovery get-instances-health-status \
    --service-id arn:aws:servicediscovery:us-west-2:123456789012:service/srv-
p5zdwlg5uvvzjita
```

Output:

```
{
```

```
    "Status": {
        "web-server-01": "HEALTHY",
        "web-server-02": "UNHEALTHY"
    }
}
```

Per ulteriori informazioni, consulta le istanze del servizio AWS Cloud Map e gli spazi dei nomi Shared AWS Cloud Map nella AWS Cloud Map Developer Guide.

- Per i dettagli sull'API, consulta GetInstancesHealthStatusCommand Reference.AWS CLI

`get-namespace`

Il seguente esempio di codice mostra come utilizzare`get-namespace`.

AWS CLI

Esempio 1: Per ottenere i dettagli di un namespace

L'esempio `get-namespace` seguente recupera le informazioni relative al namespace specificato.

```
aws servicediscovery get-namespace \
    --id ns-abcd1234xmpl5678
```

Output:

```
{
    "Namespace": {
        "Id": "ns-abcd1234xmpl5678",
        "Arn": "arn:aws:servicediscovery:us-west-2:123456789012:namespace/ns-
abcd1234xmpl5678",
        "ResourceOwner": "123456789012",
        "Name": "example-http.com",
        "Type": "HTTP",
        "Description": "Example.com AWS Cloud Map HTTP Namespace",
        "Properties": {
            "DnsProperties": {},
            "HttpProperties": {
                "HttpName": "example-http.com"
            }
        },
        "CreateDate": "2024-02-23T13:35:21.874000-06:00",
```

```
        "CreatorRequestId": "abcd1234-5678-90ab-cdef-xmpl12345678"
    }
}
```

Per ulteriori informazioni, consulta Namespace AWS Cloud Map nella Guida per gli sviluppatori di AWS Cloud Map.

Esempio 2: Per ottenere i dettagli di un namespace utilizzando ARN

L'`get-namespace`esempio seguente recupera le informazioni sullo spazio dei nomi specificato utilizzando il relativo ARN. La specificazione dell'ARN è necessaria per recuperare i dettagli di uno spazio dei nomi condiviso con l'account.

```
aws servicediscovery get-namespace \
    --id arn:aws:servicediscovery:us-west-2:123456789012:namespace/ns-
abcd1234xmpl5678
```

Output:

```
{
    "Namespace": {
        "Id": "ns-abcd1234xmpl5678",
        "Arn": "arn:aws:servicediscovery:us-west-2:123456789012:namespace/ns-
abcd1234xmpl5678",
        "ResourceOwner": "123456789012",
        "Name": "example-http.com",
        "Type": "HTTP",
        "Description": "Example.com AWS Cloud Map HTTP Namespace",
        "Properties": {
            "DnsProperties": {},
            "HttpProperties": {
                "HttpName": "example-http.com"
            }
        },
        "CreateDate": "2024-02-23T13:35:21.874000-06:00",
        "CreatorRequestId": "abcd1234-5678-90ab-cdef-xmpl12345678"
    }
}
```

Per ulteriori informazioni, consulta Shared AWS Cloud Map namespace nella AWS Cloud Map Developer Guide.

- Per i dettagli sull'API, consulta Command GetNamespaceReference AWS CLI .

`get-operation`

Il seguente esempio di codice mostra come utilizzare`get-operation`.

AWS CLI

Esempio 1: Per ottenere il risultato di un'operazione

L'esempio `get-operation` seguente ottiene il risultato di un'operazione di creazione del namespace.

```
aws servicediscovery get-operation \
    --operation-id abcd1234xmpl5678abcd1234xmpl5678-abcd1234
```

Output:

```
{
    "Operation": {
        "Id": "abcd1234xmpl5678abcd1234xmpl5678-abcd1234",
        "Type": "CREATE_NAMESPACE",
        "Status": "SUCCESS",
        "CreateDate": "2025-01-13T13:35:21.874000-06:00",
        "UpdateDate": "2025-01-13T13:36:02.469000-06:00",
        "Targets": {
            "NAMESPACE": "ns-abcd1234xmpl5678"
        }
    }
}
```

Per ulteriori informazioni, consulta Creazione di uno spazio dei nomi AWS Cloud Map per raggruppare i servizi applicativi nella AWS Cloud Map Developer Guide.

Esempio 2: ottenere un'operazione da un account proprietario specifico

L'`get-operation`esempio seguente ottiene il risultato di un'operazione associata a un account proprietario del namespace specifico. Questo parametro è necessario per ottenere il risultato delle operazioni associate ai namespace condivisi con l'account.

```
aws servicediscovery get-operation \
    --operation-id abcd1234xmpl5678abcd1234xmpl5678-abcd1234 \
```

```
    --owner-account 123456789111
```

Output:

```
{
    "Operation": {
        "Id": "abcd1234xmpl5678abcd1234xmpl5678-abcd1234",
        "OwnerAccount": "123456789111",
        "Type": "CREATE_NAMESPACE",
        "Status": "SUCCESS",
        "CreateDate": "2025-01-13T13:35:21.874000-06:00",
        "UpdateDate": "2025-01-13T13:36:02.469000-06:00",
        "Targets": {
            "NAMESPACE": "ns-abcd1234xmpl5678"
        }
    }
}
```

Per ulteriori informazioni, consulta Shared AWS Cloud Map namespace nella AWS Cloud Map Developer Guide.

- Per i dettagli sull'API, consulta Command GetOperationReference AWS CLI .

`get-service-attributes`

Il seguente esempio di codice mostra come utilizzare`get-service-attributes`.

AWS CLI

Esempio 1: Per ottenere gli attributi di un servizio

L'esempio `get-service-attributes` seguente ottiene gli attributi di un servizio.

```
aws servicediscovery get-service-attributes \
    --service-id srv-abcd1234xmpl5678
```

Output:

```
{
    "ServiceAttributes": {
        "ServiceArn": "arn:aws:servicediscovery:us-west-2:123456789012:service/srv-
abcd1234xmpl5678",
```

```
        "ResourceOwner": "123456789012",
        "Attributes": {
            "Port": "80"
        }
    }
}
```

Per ulteriori informazioni, consulta Servizi AWS Cloud Map nella Guida per gli sviluppatori di AWS Cloud Map.

Esempio 2: Per ottenere gli attributi di un servizio utilizzando ARN

L'`get-service-attributes`esempio seguente ottiene gli attributi di un servizio utilizzando il relativo ARN. La specificazione di un ARN è necessaria per ottenere gli attributi di un servizio creato in uno spazio dei nomi condiviso con l'account.

```
aws servicediscovery get-service-attributes \
    --service-id arn:aws:servicediscovery:us-west-2:123456789012:service/srv-abcd1234xmpl5678
```

Output:

```
{
    "ServiceAttributes": {
        "ServiceArn": "arn:aws:servicediscovery:us-west-2:123456789012:service/srv-abcd1234xmpl5678",
        "ResourceOwner": "123456789012",
        "Attributes": {
            "Port": "80"
        }
    }
}
```

Per ulteriori informazioni, consulta AWS Cloud Map services e Shared AWS Cloud Map namespace nella AWS Cloud Map Developer Guide.

- Per i dettagli sull'API, consulta Command GetServiceAttributesReference AWS CLI .

`get-service`

Il seguente esempio di codice mostra come utilizzare`get-service`.

AWS CLI

Esempio 1: Per ottenere le impostazioni di un servizio

L'esempio `get-service` seguente ottiene le impostazioni di un servizio specificato.

```
aws servicediscovery get-service \
    --id srv-abcd1234xmpl5678
```

Output:

```
{
    "Service": {
        "Id": "srv-abcd1234xmpl5678",
        "Arn": "arn:aws:servicediscovery:us-west-2:123456789012:service/srv-
abcd1234xmpl5678",
        "ResourceOwner": "123456789012",
        "Name": "test-service",
        "NamespaceId": "ns-abcd1234xmpl5678",
        "DnsConfig": {},
        "Type": "HTTP",
        "CreateDate": "2025-08-18T13:53:02.775000-05:00",
        "CreatorRequestId": "abcd1234-5678-90ab-cdef-xmpl12345678",
        "CreatedByAccount": "123456789012"
    }
}
```

Per ulteriori informazioni, consulta Servizi AWS Cloud Map nella Guida per gli sviluppatori di AWS Cloud Map.

Esempio 2: Per ottenere le impostazioni di un servizio utilizzando ARN

L'`get-service`esempio seguente ottiene le impostazioni di un servizio specificato utilizzando il relativo ARN. La specificazione dell'ARN è necessaria per recuperare informazioni su un servizio creato in uno spazio dei nomi condiviso con l'account. L'account chiamante 123456789111 ha creato il servizio in uno spazio dei nomi condiviso dall'account. 123456789012

```
aws servicediscovery get-service \
    --id arn:aws:servicediscovery:us-west-2:123456789012:service/srv-
abcd1234xmpl5678
```

Output:

```
{
    "Service": {
        "Id": "srv-abcd1234xmpl5678",
        "Arn": "arn:aws:servicediscovery:us-west-2:123456789012:service/srv-
abcd1234xmpl5678",
        "ResourceOwner": "123456789012",
        "Name": "test-service",
        "NamespaceId": "ns-abcd1234xmpl5678",
        "DnsConfig": {},
        "Type": "HTTP",
        "CreateDate": "2025-08-18T13:53:02.775000-05:00",
        "CreatorRequestId": "abcd1234-5678-90ab-cdef-xmpl12345678",
        "CreatedByAccount": "123456789111"
    }
}
```

Per ulteriori informazioni, consulta Creazione di un servizio AWS Cloud Map per un componente applicativo e Shared AWS Cloud Map namespace nella AWS Cloud Map Developer Guide.

- Per i dettagli sull'API, consulta Command GetServiceReference AWS CLI .

`list-instances`

Il seguente esempio di codice mostra come utilizzare`list-instances`.

AWS CLI

Esempio 1: per elencare le istanze del servizio

L'esempio `list-instances` seguente elenca le istanze del servizio.

```
aws servicediscovery list-instances \
    --service-id srv-qzpwvt2tfqcegapy
```

Output:

```
{
    "Instances": [
        {
            "Id": "i-06bdabbae60f65a4e",
            "Attributes": {
```

```
                "AWS_INSTANCE_IPV4": "172.2.1.3",
                "AWS_INSTANCE_PORT": "808"
            },
            "CreatedByAccount": "123456789012"
        }
    ],
    "ResourceOwner": "123456789012"
}
```

Per ulteriori informazioni, consulta Elencare le istanze del servizio AWS Cloud Map nella AWS Cloud Map Developer Guide.

Esempio 2: elencare le istanze del servizio utilizzando il servizio ARN

L'`list-instances`esempio seguente elenca le istanze di servizio che utilizzano un ARN del servizio anziché l'ID del servizio. È necessario specificare un ARN quando si elencano le istanze associate ai namespace condivisi con l'account.

```
aws servicediscovery list-instances \
    --service-id arn:aws:servicediscovery:us-west-2:123456789012:service/srv-p5zdwlg5uvvzjita
```

Output:

```
{
    "ResourceOwner": "123456789012",
    "Instances": [
        {
            "Id": "web-server-01",
            "Attributes": {
                "AWS_INSTANCE_IPV4": "203.0.113.15",
                "AWS_INSTANCE_PORT": "80"
            },
            "CreatedByAccount": "123456789012"
        },
        {
            "Id": "web-server-02",
            "Attributes": {
                "AWS_INSTANCE_IPV4": "203.0.113.16",
                "AWS_INSTANCE_PORT": "80"
            },
            "CreatedByAccount": "123456789012"
```

```
        }
    ]
}
```

Per ulteriori informazioni sulla condivisione dello spazio dei nomi tra account, consulta Shared AWS Cloud Map namespaces e Elenco delle istanze del servizio Cloud AWS Map nella Cloud Map Developer Guide.AWS

- Per i dettagli sull'API, consulta Command Reference. ListInstancesAWS CLI

`list-namespaces`

Il seguente esempio di codice mostra come utilizzare`list-namespaces`.

AWS CLI

Esempio 1: Per elencare i namespace

L'esempio `list-namespaces` seguente elenca i namespace.

```
aws servicediscovery list-namespaces
```

Output:

```
{
    "Namespaces": [
        {
            "Id": "ns-abcd1234xmpl5678",
            "Arn": "arn:aws:servicediscovery:us-west-2:123456789012:namespace/ns-
abcd1234xmpl5678",
            "ResourceOwner": "123456789012",
            "Name": "local",
            "Type": "DNS_PRIVATE",
            "Properties": {
                "DnsProperties": {
                    "HostedZoneId": "Z06752353VBUDTC32S84S",
                    "SOA": {}
                },
                "HttpProperties": {
                    "HttpName": "local"
                 }
            },
```

```
            "CreateDate": "2023-07-17T13:37:27.872000-05:00"
        },
        {
            "Id": "ns-abcd1234xmpl9012",
            "Arn": "arn:aws:servicediscovery:us-west-2:123456789012:namespace/ns-
abcd1234xmpl9012",
            "ResourceOwner": "123456789012",
            "Name": "My-second-namespace",
            "Type": "HTTP",
            "Description": "My second namespace",
            "Properties": {
                "DnsProperties": {
                    "SOA": {}
                },
                "HttpProperties": {
                    "HttpName": "My-second-namespace"
                }
            },
            "CreateDate": "2023-11-14T10:35:47.840000-06:00"
        }
    ]
}
```

Per ulteriori informazioni, consulta Elencare gli spazi dei nomi di AWS Cloud Map nella AWS Cloud Map Developer Guide.

Esempio 2: per elencare i namespace condivisi da altri account

L'`list-namespaces`esempio seguente elenca gli spazi dei nomi condivisi con l'account chiamante da altri account utilizzando il filtro. AWS `RESOURCE_OWNER`

```
aws servicediscovery list-namespaces \
    --filters Name=RESOURCE_OWNER,Values=OTHER_ACCOUNTS,Condition=EQ
```

Output:

```
{
    "Namespaces": [
        {
            "Id": "ns-abcd1234xmpl5678",
            "Arn": "arn:aws:servicediscovery:us-west-2:123456789111:namespace/ns-
abcd1234xmpl5678",
```

```
            "ResourceOwner": "123456789111",
            "Name": "shared-namespace",
            "Type": "HTTP",
            "Description": "Namespace shared from another account",
            "Properties": {
                "DnsProperties": {
                    "SOA": {}
                },
                "HttpProperties": {
                    "HttpName": "shared-namespace"
                }
            },
            "CreateDate": "2025-01-13T13:35:21.874000-06:00"
        }
    ]
}
```

Per ulteriori informazioni, consulta Shared AWS Cloud Map namespace nella AWS Cloud Map Developer Guide.

- Per i dettagli sull'API, consulta Command ListNamespacesReference AWS CLI .

`list-operations`

Il seguente esempio di codice mostra come utilizzare`list-operations`.

AWS CLI

Come elencare le operazioni che soddisfano i criteri specificati

L'esempio `list-operations` seguente elenca le operazioni il cui stato è `PENDING` o `SUCCESS`.

```
aws servicediscovery list-operations \
    --service-id srv-e4anhexample0004 \
    --filters Name=STATUS,Condition=IN,Values=PENDING,SUCCESS
```

Output:

```
{
    "Operations": [
        {
            "Id": "76yy8ovhpdz0plmjzbsnqgnrqvpv2qdt-kexample",
```

```
            "Status": "SUCCESS"
        },
        {
            "Id": "prysnyzpji3u2ciy45nke83x2zanl7yk-dexample",
            "Status": "SUCCESS"
        },
        {
            "Id": "ko4ekftir7kzlbechsh7xvcdgcpk66gh-7example",
            "Status": "PENDING"
        }
    ]
}
```

Per ulteriori informazioni, consulta Cos'è AWS Cloud Map? nella Guida per gli sviluppatori di AWS Cloud Map.

- Per i dettagli sull'API, consulta ListOperations AWS CLICommand Reference.

`list-services`

Il seguente esempio di codice mostra come utilizzare`list-services`.

AWS CLI

Esempio 1: Per elencare i servizi

L'esempio `list-services` seguente elenca i servizi.

```
aws servicediscovery list-services
```

Output:

```
{
    "Services": [
        {
            "Id": "srv-p5zdwlg5uvvzjita",
            "Arn": "arn:aws:servicediscovery:us-west-2:123456789012:service/srv-
p5zdwlg5uvvzjita",
            "Name": "myservice",
            "DnsConfig": {
                "RoutingPolicy": "MULTIVALUE",
                "DnsRecords": [
```

```
                    {
                        "Type": "A",
                        "TTL": 60
                    }
                ]
            },
            "CreateDate": 1587081768.334
        }
    ]
}
```

Per ulteriori informazioni, consulta Elencare i servizi AWS Cloud Map in un namespace nella AWS Cloud Map Developer Guide.

Esempio 2: per elencare i servizi creati in namespace condivisi

L'`list-services`esempio seguente elenca i servizi creati in namespace condivisi con l'account chiamante da altri account `123456789012` che utilizzano il filtro. AWS `RESOURCE_OWNER`

```
aws servicediscovery list-services \
    --filters Name=RESOURCE_OWNER,Values=OTHER_ACCOUNTS,Condition=EQ
```

Output:

```
{
    "Services": [
        {
            "Id": "srv-abcd1234xmpl5678",
            "Arn": "arn:aws:servicediscovery:us-west-2:123456789111:service/srv-
abcd1234xmpl5678",
            "ResourceOwner": "123456789111",
            "Name": "shared-service",
            "NamespaceId": "ns-abcd1234xmpl5678",
            "Type": "HTTP",
            "Description": "Service in shared namespace",
            "DnsConfig": {},
            "CreateDate": "2025-01-13T13:35:21.874000-06:00",
            "CreatorRequestId": "abcd1234-5678-90ab-cdef-xmpl12345678",
            "CreatedByAccount": "123456789012"
        }
    ]
}
```

Per ulteriori informazioni, consulta Shared AWS Cloud Map namespaces e Listing AWS Cloud Map services in un namespace nella Cloud Map AWS Developer Guide.

- Per i dettagli sull'API, consulta Command Reference. ListServicesAWS CLI

`list-tags-for-resource`

Il seguente esempio di codice mostra come utilizzare`list-tags-for-resource`.

AWS CLI

Come elencare i tag associati alle risorse specificate

L'esempio `list-tags-for-resource` seguente elenca i tag della risorsa specificata.

```
aws servicediscovery list-tags-for-resource \
    --resource-arn arn:aws:servicediscovery:us-west-2:123456789012:namespace/ns-e4anhexample0004
```

Output:

```
{
    "Tags": [
        {
            "Key": "Project",
            "Value": "Zeta"
        },
        {
            "Key": "Department",
            "Value": "Engineering"
        }
    ]
}
```

Per ulteriori informazioni, consulta Taggare le risorse di AWS Cloud Map nella AWS Cloud Map Developer Guide.

- Per i dettagli sull'API, consulta ListTagsForResource AWS CLICommand Reference.

`register-instance`

Il seguente esempio di codice mostra come utilizzare`register-instance`.

AWS CLI

Esempio 1: registrare un'istanza di servizio utilizzando l'ID del servizio

L'esempio `register-instance` seguente registra un'istanza del servizio.

```
aws servicediscovery register-instance \
    --service-id srv-p5zdwlg5uvvzjita \
    --instance-id myservice-53 \
    --attributes=AWS_INSTANCE_IPV4=172.2.1.3,AWS_INSTANCE_PORT=808
```

Output:

```
{
    "OperationId": "4yejorelbukcjzpnr6tlmrghsjwpngf4-k95yg2u7"
}
```

Per confermare che l'operazione è riuscita, puoi eseguire il comando `get-operation`. Per ulteriori informazioni, consulta get-operation.

Per ulteriori informazioni sulla registrazione di un'istanza, consulta Registrazione di una risorsa come istanza del servizio AWS Cloud Map nella AWS Cloud Map Developer Guide.

Esempio 2: registrare un'istanza di servizio utilizzando il servizio ARN

L'`register-instance`esempio seguente registra un'istanza di servizio utilizzando un servizio ARN. È necessario specificare l'ARN quando si registrano istanze nei servizi condivisi con l'account.

```
aws servicediscovery register-instance \
    --service-id arn:aws:servicediscovery:us-west-2:123456789012:service/srv-
p5zdwlg5uvvzjita \
    --instance-id web-server-01 \
    --attributes=AWS_INSTANCE_IPV4=203.0.113.15,AWS_INSTANCE_PORT=80
```

Output:

```
{
    "OperationId": "gv4g5meo7ndmkqjrhpn39wk42xmpl"
}
```

Per ulteriori informazioni sulla condivisione dello spazio dei nomi tra account, consulta Shared AWS Cloud Map namespaces nella Cloud Map Developer Guide.AWS

- Per i dettagli sull'API, consulta Command Reference. RegisterInstanceAWS CLI

`tag-resource`

Il seguente esempio di codice mostra come utilizzare`tag-resource`.

AWS CLI

Come associare i tag alle risorse specificate

L'esempio `tag-resource` seguente associa un tag `Department` al valore `Engineering` del namespace specificato.

```
aws servicediscovery tag-resource \
    --resource-arn arn:aws:servicediscovery:us-west-2:123456789012:namespace/ns-e4anhexample0004 \
    --tags Key=Department, Value=Engineering
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Taggare le risorse di AWS Cloud Map nella AWS Cloud Map Developer Guide.

- Per i dettagli sull'API, consulta TagResource AWS CLICommand Reference.

`untag-resource`

Il seguente esempio di codice mostra come utilizzare`untag-resource`.

AWS CLI

Come rimuovere i tag dalla risorsa specificata

L'esempio `untag-resource` seguente rimuove un tag `Department` dal namespace specificato.

```
aws servicediscovery untag-resource \
    --resource-arn arn:aws:servicediscovery:us-west-2:123456789012:namespace/ns-e4anhexample0004 \
```

```
    --tags Key=Department, Value=Engineering
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Taggare le risorse di AWS Cloud Map nella AWS Cloud Map Developer Guide.

- Per i dettagli sull'API, consulta UntagResource AWS CLICommand Reference.

`update-http-namespace`

Il seguente esempio di codice mostra come utilizzare`update-http-namespace`.

AWS CLI

Esempio 1: aggiornare uno spazio dei nomi HTTP

L'esempio `update-http-namespace` seguente aggiorna la descrizione del namespace HTTP specificato.

```
aws servicediscovery update-http-namespace \
    --id ns-abcd1234xmpl5678 \
    --updater-request-id abcd1234-5678-90ab-cdef-xmpl12345678 \
    --namespace Description="The updated namespace description."
```

Output:

```
{
    "OperationId": "abcd1234-5678-90ab-cdef-xmpl12345678"
}
```

Per confermare che l'operazione è riuscita, puoi eseguire il comando `get-operation`. Per ulteriori informazioni, consulta get-operation.

Per ulteriori informazioni, consulta Namespace AWS Cloud Map nella Guida per gli sviluppatori di AWS Cloud Map.

Esempio 2: aggiornare uno spazio dei nomi HTTP utilizzando ARN

L'`update-http-namespace`esempio seguente aggiorna lo spazio dei nomi HTTP specificato utilizzando il relativo ARN.

```
aws servicediscovery update-http-namespace \
    --id arn:aws:servicediscovery:us-west-2:123456789012:namespace/ns-
abcd1234xmpl5678 \
    --updater-request-id abcd1234-5678-90ab-cdef-xmpl12345678 \
    --namespace Description="The updated namespace description."
```

Output:

```
{
    "OperationId": "abcd1234-5678-90ab-cdef-xmpl12345678"
}
```

Per ulteriori informazioni, consulta Namespace AWS Cloud Map nella Guida per gli sviluppatori di AWS Cloud Map.

- Per i dettagli sull'API, vedere UpdateHttpNamespacein AWS CLI Command Reference.

`update-instance-custom-health-status`

Il seguente esempio di codice mostra come utilizzareupdate-instance-custom-health-status.

AWS CLI

Esempio 1: Per aggiornare un controllo sanitario personalizzato

L'esempio update-instance-custom-health-status seguente aggiorna lo stato del controllo dell'integrità personalizzato per il servizio specificato e l'istanza del servizio di esempio impostandolo su HEALTHY.

```
aws servicediscovery update-instance-custom-health-status \
    --service-id srv-e4anhexample0004 \
    --instance-id example \
    --status HEALTHY
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Configurazione del controllo dell'integrità del servizio AWS Cloud Map nella Guida per gli sviluppatori di AWS Cloud Map.

Esempio 2: Aggiornare un controllo sanitario personalizzato utilizzando il servizio ARN

L'`update-instance-custom-health-status`esempio seguente aggiorna lo stato del controllo sanitario personalizzato utilizzando un servizio ARN. L'ARN è necessario per aggiornare lo stato di integrità delle istanze associate ai namespace condivisi con l'account.

```
aws servicediscovery update-instance-custom-health-status \
    --service-id arn:aws:servicediscovery:us-west-2:123456789012:service/srv-
p5zdwlg5uvvzjita \
    --instance-id web-server-01 \
    --status HEALTHY
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta la configurazione del controllo dello stato del servizio AWS Cloud Map e la condivisione dello spazio dei nomi AWS Cloud Map tra account nella AWS Cloud Map Developer Guide.

- Per i dettagli sull'API, consulta Command UpdateInstanceCustomHealthStatusReference AWS CLI .

`update-private-dns-namespace`

Il seguente esempio di codice mostra come utilizzare`update-private-dns-namespace`.

AWS CLI

Esempio 1: aggiornare uno spazio dei nomi DNS privato utilizzando ID

L'`update-private-dns-namespace`esempio seguente aggiorna la descrizione di uno spazio dei nomi DNS privato utilizzando l'ID dello spazio dei nomi.

```
aws servicediscovery update-private-dns-namespace \
    --id ns-abcd1234xmpl5678 \
    --updater-request-id abcd1234-5678-90ab-cdef-xmpl12345678 \
    --namespace Description="The updated namespace description."
```

Output:

```
{
    "OperationId": "abcd1234-5678-90ab-cdef-xmpl12345678"
}
```

Per confermare che l'operazione è riuscita, puoi eseguire il comando `get-operation`.

Per ulteriori informazioni, consulta Namespace AWS Cloud Map nella Guida per gli sviluppatori di AWS Cloud Map.

Esempio 2: per aggiornare uno spazio dei nomi DNS privato utilizzando ARN

L'`update-private-dns-namespace`esempio seguente aggiorna uno spazio dei nomi DNS privato utilizzando il relativo ARN.

```
aws servicediscovery update-private-dns-namespace \
    --id arn:aws:servicediscovery:us-west-2:123456789012:namespace/ns-
abcd1234xmpl5678 \
    --updater-request-id abcd1234-5678-90ab-cdef-xmpl12345678 \
    --namespace Description="The updated namespace description."
```

Output:

```
{
    "OperationId": "abcd1234-5678-90ab-cdef-xmpl12345678"
}
```

Per ulteriori informazioni, consulta Namespace AWS Cloud Map nella Guida per gli sviluppatori di AWS Cloud Map.

- Per i dettagli sull'API, vedere UpdatePrivateDnsNamespacein Command Reference.AWS CLI

`update-public-dns-namespace`

Il seguente esempio di codice mostra come utilizzareu`pdate-public-dns-namespace`.

AWS CLI

Esempio 1: aggiornare uno spazio dei nomi DNS pubblico utilizzando ID

L'`update-public-dns-namespace`esempio seguente aggiorna la descrizione di uno spazio dei nomi DNS pubblico utilizzando il relativo ID.

```
aws servicediscovery update-public-dns-namespace \
    --id ns-abcd1234xmpl5678 \
    --updater-request-id abcd1234-5678-90ab-cdef-xmpl12345678 \
```

```
    --namespace Description="The updated namespace description."
```

Output:

```
{
    "OperationId": "abcd1234-5678-90ab-cdef-xmpl12345678"
}
```

Per confermare che l'operazione è riuscita, puoi eseguire il comando `get-operation`.

Per ulteriori informazioni, consulta Namespace AWS Cloud Map nella Guida per gli sviluppatori di AWS Cloud Map.

Esempio 2: aggiornare uno spazio dei nomi DNS pubblico utilizzando ARN

L'`update-public-dns-namespace`esempio seguente aggiorna uno spazio dei nomi DNS pubblico utilizzando il relativo ARN.

```
aws servicediscovery update-public-dns-namespace \
    --id arn:aws:servicediscovery:us-west-2:123456789012:namespace/ns-abcd1234xmpl5678 \
    --updater-request-id abcd1234-5678-90ab-cdef-xmpl12345678 \
    --namespace Description="The updated namespace description."
```

Output:

```
{
    "OperationId": "abcd1234-5678-90ab-cdef-xmpl12345678"
}
```

Per ulteriori informazioni, consulta Namespace AWS Cloud Map nella Guida per gli sviluppatori di AWS Cloud Map.

- Per i dettagli sull'API, vedere UpdatePublicDnsNamespacein Command Reference.AWS CLI

`update-service-attributes`

Il seguente esempio di codice mostra come utilizzareu`pdate-service-attributes`.

AWS CLI

Esempio 1: aggiornare un servizio per aggiungere un attributo

L'esempio `update-service-attributes` seguente aggiorna il servizio specificato in modo che venga aggiunto un attributo di servizio con una chiave `Port` e un valore `80`.

```
aws servicediscovery update-service-attributes \
    --service-id srv-abcd1234xmpl5678 \
    --attributes Port=80
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Servizi AWS Cloud Map nella Guida per gli sviluppatori di AWS Cloud Map.

Esempio 2: aggiornare gli attributi di un servizio utilizzando ARN

L'`update-service-attributes`esempio seguente aggiorna un servizio utilizzando il relativo ARN per aggiungere un attributo di servizio. La specificazione dell'ARN è necessaria per aggiungere attributi ai servizi creati nei namespace condivisi con l'account.

```
aws servicediscovery update-service-attributes \
    --service-id arn:aws:servicediscovery:us-west-2:123456789012:service/srv-
abcd1234xmpl5678 \
    --attributes Port=80
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta AWS Cloud Map services e Shared AWS Cloud Map namespace nella AWS Cloud Map Developer Guide.

- Per i dettagli sull'API, consulta Command UpdateServiceAttributesReference AWS CLI .

`update-service`

Il seguente esempio di codice mostra come utilizzare`update-service`.

AWS CLI

Esempio 1: aggiornare un servizio

L'esempio `update-service` seguente aggiorna un servizio in modo che vengano aggiornate le impostazioni `DnsConfig` e `HealthCheckConfig`.

```
aws servicediscovery update-service \
```

```
    --id srv-abcd1234xmpl5678 \
    --
service "DnsConfig={DnsRecords=[{Type=A,TTL=60}]},HealthCheckConfig={Type=HTTP,ResourcePath=
```

Output:

```
{
    "OperationId": "abcd1234-5678-90ab-cdef-xmpl12345678"
}
```

Per confermare che l'operazione è riuscita, puoi eseguire il comando `get-operation`.

Per ulteriori informazioni sull'aggiornamento di un servizio, consulta Aggiornamento di un servizio AWS Cloud Map nella AWS Cloud Map Developer Guide.

Esempio 2: Aggiornare un servizio utilizzando ARN

L'`update-service`esempio seguente aggiorna un servizio utilizzando il relativo ARN. La specificazione di un ARN è necessaria per i servizi creati in namespace condivisi con l'account.

```
aws servicediscovery update-service \
    --id arn:aws:servicediscovery:us-west-2:123456789012:service/srv-
abcd1234xmpl5678 \
    --
service "DnsConfig={DnsRecords=[{Type=A,TTL=60}]},HealthCheckConfig={Type=HTTP,ResourcePath=
```

Output:

```
{
    "OperationId": "abcd1234-5678-90ab-cdef-xmpl12345678"
}
```

Per ulteriori informazioni sull'aggiornamento di un servizio, consulta Updating an AWS Cloud Map service e Shared AWS Cloud Map namespace nella AWS Cloud Map Developer Guide.

- Per i dettagli sull'API, consulta Command UpdateServiceReference AWS CLI .

AWS Cloud9 esempi utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with AWS Cloud9.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`create-environment-ec2`

Il seguente esempio di codice mostra come utilizzare`create-environment-ec2`.

AWS CLI

Per creare un ambiente di AWS sviluppo EC2 Cloud9

L'`create-environment-ec2`esempio seguente crea un AWS ambiente di sviluppo Cloud9 con le impostazioni specificate, avvia un'istanza Amazon Elastic Compute Cloud (EC2Amazon) e quindi si connette dall'istanza all'ambiente.

```
aws cloud9 create-environment-ec2 \
    --name my-demo-env \
    --description "My demonstration development environment." \
    --instance-type t2.micro --image-id amazonlinux-2023-x86_64 \
    --subnet-id subnet-1fab8aEX \
    --automatic-stop-time-minutes 60 \
    --owner-arn arn:aws:iam::123456789012:user/MyDemoUser
```

Output:

```
{
    "environmentId": "8a34f51ce1e04a08882f1e811bd706EX"
}
```

Per ulteriori informazioni, consulta Creazione di un EC2 ambiente nella Guida per l'utente di AWS Cloud9.

- Per i dettagli sull'API, consulta CreateEnvironmentEc2 in AWS CLI Command Reference.

`create-environment-membership`

Il seguente esempio di codice mostra come utilizzare`create-environment-membership`.

AWS CLI

Per aggiungere un membro dell'ambiente a un ambiente di sviluppo AWS Cloud9

Questo esempio aggiunge il membro dell'ambiente specificato all'ambiente di sviluppo AWS Cloud9 specificato.

Comando:

```
aws cloud9 create-environment-membership --environment-
id 8a34f51ce1e04a08882f1e811bd706EX --user-arn arn:aws:iam::123456789012:user/
AnotherDemoUser --permissions read-write
```

Output:

```
{
  "membership": {
    "environmentId": "8a34f51ce1e04a08882f1e811bd706EX",
    "userId": "AIDAJ3LOROMOUXTBSU6EX",
    "userArn": "arn:aws:iam::123456789012:user/AnotherDemoUser",
    "permissions": "read-write"
  }
}
```

- Per i dettagli sull'API, consulta AWS CLI Command CreateEnvironmentMembershipReference.

`delete-environment-membership`

Il seguente esempio di codice mostra come utilizzare`delete-environment-membership`.

AWS CLI

Per eliminare un membro dell'ambiente da un ambiente di sviluppo AWS Cloud9

Questo esempio elimina il membro dell'ambiente specificato dall'ambiente di sviluppo AWS Cloud9 specificato.

Comando:

```
aws cloud9 delete-environment-membership --environment-
id 8a34f51ce1e04a08882f1e811bd706EX --user-arn arn:aws:iam::123456789012:user/
AnotherDemoUser
```

Output:

```
None.
```

- Per i dettagli sull'API, consulta Command DeleteEnvironmentMembershipReference AWS CLI .

`delete-environment`

Il seguente esempio di codice mostra come utilizzare`delete-environment`.

AWS CLI

Per eliminare un ambiente di AWS sviluppo Cloud9

Questo esempio elimina l'ambiente di sviluppo AWS Cloud9 specificato. Se un' EC2 istanza Amazon è connessa all'ambiente, interrompe anche l'istanza.

Comando:

```
aws cloud9 delete-environment --environment-id 8a34f51ce1e04a08882f1e811bd706EX
```

Output:

```
None.
```

- Per i dettagli sull'API, consulta DeleteEnvironment AWS CLICommand Reference.

`describe-environment-memberships`

Il seguente esempio di codice mostra come utilizzare`describe-environment-memberships`.

AWS CLI

Per ottenere informazioni sui membri dell'ambiente per un ambiente di sviluppo AWS Cloud9

Questo esempio ottiene informazioni sui membri dell'ambiente per l'ambiente di sviluppo AWS Cloud9 specificato.

Comando:

```
aws cloud9 describe-environment-memberships --environment-
id 8a34f51ce1e04a08882f1e811bd706EX
```

Output:

```
{
  "memberships": [
    {
      "environmentId": "8a34f51ce1e04a08882f1e811bd706EX",
      "userId": "AIDAJ3LOROMOUXTBSU6EX",
      "userArn": "arn:aws:iam::123456789012:user/AnotherDemoUser",
      "permissions": "read-write"
    },
    {
      "environmentId": "8a34f51ce1e04a08882f1e811bd706EX",
      "userId": "AIDAJNUEDQAQWFELJDLEX",
      "userArn": "arn:aws:iam::123456789012:user/MyDemoUser",
      "permissions": "owner"
    }
  ]
}
```

Per ottenere informazioni sul proprietario di un ambiente di sviluppo AWS Cloud9

Questo esempio ottiene informazioni sul proprietario dell'ambiente di sviluppo AWS Cloud9 specificato.

Comando:

```
aws cloud9 describe-environment-memberships --environment-
id 8a34f51ce1e04a08882f1e811bd706EX --permissions owner
```

Output:

```
{
  "memberships": [
```

```
    {
      "environmentId": "8a34f51ce1e04a08882f1e811bd706EX",
      "userId": "AIDAJNUEDQAQWFELJDLEX",
      "userArn": "arn:aws:iam::123456789012:user/MyDemoUser",
      "permissions": "owner"
    }
  ]
}
```

Per ottenere informazioni su un membro dell'ambiente per più ambienti di sviluppo AWS Cloud9

Questo esempio ottiene informazioni sul membro dell'ambiente specificato per più ambienti di sviluppo AWS Cloud9.

Comando:

```
aws cloud9 describe-environment-memberships --user-
arn arn:aws:iam::123456789012:user/MyDemoUser
```

Output:

```
{
  "memberships": [
    {
      "environmentId": "10a75714bd494714929e7f5ec4125aEX",
      "lastAccess": 1516213427.0,
      "userId": "AIDAJNUEDQAQWFELJDLEX",
      "userArn": "arn:aws:iam::123456789012:user/MyDemoUser",
      "permissions": "owner"
    },
    {
      "environmentId": "1980b80e5f584920801c09086667f0EX",
      "lastAccess": 1516144884.0,
      "userId": "AIDAJNUEDQAQWFELJDLEX",
      "userArn": "arn:aws:iam::123456789012:user/MyDemoUser",
      "permissions": "owner"
    }
  ]
}
```

- Per i dettagli sull'API, consulta AWS CLI Command DescribeEnvironmentMembershipsReference.

`describe-environment-status`

Il seguente esempio di codice mostra come utilizzare`describe-environment-status`.

AWS CLI

Per ottenere informazioni sullo stato di un ambiente di sviluppo AWS Cloud9

Questo esempio ottiene informazioni sullo stato per l'ambiente di sviluppo AWS Cloud9 specificato.

Comando:

```
aws cloud9 describe-environment-status --environment-
id 685f892f431b45c2b28cb69eadcdb0EX
```

Output:

```
{
  "status": "ready",
  "message": "Environment is ready to use"
}
```

- Per i dettagli sull'API, consulta AWS CLI Command DescribeEnvironmentStatusReference.

`describe-environments`

Il seguente esempio di codice mostra come utilizzare`describe-environments`.

AWS CLI

Per ottenere informazioni sugli ambienti di sviluppo AWS Cloud9

Questo esempio ottiene informazioni sugli ambienti di sviluppo AWS Cloud9 specificati.

Comando:

```
aws cloud9 describe-environments --environment-
ids 685f892f431b45c2b28cb69eadcdb0EX 349c86d4579e4e7298d500ff57a6b2EX
```

Output:

```
{
  "environments": [
    {
      "id": "685f892f431b45c2b28cb69eadcdb0EX",
      "name": "my-demo-ec2-env",
      "description": "Created from CodeStar.",
      "type": "ec2",
      "arn": "arn:aws:cloud9:us-
east-1:123456789012:environment:685f892f431b45c2b28cb69eadcdb0EX",
      "ownerArn": "arn:aws:iam::123456789012:user/MyDemoUser",
      "lifecycle": {
        "status": "CREATED"
      }
    },
    {
      "id": "349c86d4579e4e7298d500ff57a6b2EX",
      "name": my-demo-ssh-env",
      "description": "",
      "type": "ssh",
      "arn": "arn:aws:cloud9:us-
east-1:123456789012:environment:349c86d4579e4e7298d500ff57a6b2EX",
      "ownerArn": "arn:aws:iam::123456789012:user/MyDemoUser",
      "lifecycle": {
        "status": "CREATED"
      }
    }
  ]
}
```

- Per i dettagli sull'API, consulta AWS CLI Command DescribeEnvironmentsReference.

`list-environments`

Il seguente esempio di codice mostra come utilizzare`list-environments`.

AWS CLI

Per ottenere un elenco degli identificatori dell'ambiente di AWS sviluppo Cloud9 disponibili

Questo esempio ottiene un elenco di identificatori dell'ambiente di AWS sviluppo Cloud9 disponibili.

Comando:

```
aws cloud9 list-environments
```

Output:

```
{
  "environmentIds": [
    "685f892f431b45c2b28cb69eadcdb0EX",
    "1980b80e5f584920801c09086667f0EX"
  ]
}
```

- Per i dettagli sull'API, consulta Command ListEnvironmentsReference AWS CLI .

`update-environment-membership`

Il seguente esempio di codice mostra come utilizzare`update-environment-membership`.

AWS CLI

Per modificare le impostazioni di un membro dell'ambiente esistente per un ambiente di sviluppo AWS Cloud9

Questo esempio modifica le impostazioni del membro di ambiente esistente specificato per l'ambiente di sviluppo AWS Cloud9 specificato.

Comando:

```
aws cloud9 update-environment-membership --environment-
id 8a34f51ce1e04a08882f1e811bd706EX --user-arn arn:aws:iam::123456789012:user/
AnotherDemoUser --permissions read-only
```

Output:

```
{
  "membership": {
    "environmentId": "8a34f51ce1e04a08882f1e811bd706EX",
    "userId": "AIDAJ3LOROMOUXTBSU6EX",
    "userArn": "arn:aws:iam::123456789012:user/AnotherDemoUser",
    "permissions": "read-only"
  }
```

```
}
```

- Per i dettagli sull'API, consulta AWS CLI Command UpdateEnvironmentMembershipReference.

`update-environment`

Il seguente esempio di codice mostra come utilizzare`update-environment`.

AWS CLI

Per modificare le impostazioni di un ambiente di sviluppo AWS Cloud9 esistente

Questo esempio modifica le impostazioni specificate dell'ambiente di sviluppo AWS Cloud9 esistente specificato.

Comando:

```
aws cloud9 update-environment --environment-id 8a34f51ce1e04a08882f1e811bd706EX
 --name my-changed-demo-env --description "My changed demonstration development
 environment."
```

Output:

```
None.
```

- Per i dettagli sull'API, consulta AWS CLI Command UpdateEnvironmentReference.

CloudFormation esempi utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with CloudFormation.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`activate-type`

Il seguente esempio di codice mostra come utilizzare`activate-type`.

AWS CLI

Come attivare un tipo

L'esempio `activate-type` seguente attiva un'estensione pubblica di terze parti, rendendola disponibile per l'uso nei modelli di stack.

```
aws cloudformation activate-type \
    --region us-west-2 \
    --type RESOURCE \
    --type-name Example::Test::1234567890abcdef0 \
    --type-name-alias Example::Test::Alias
```

Output:

```
{
    "Arn": "arn:aws:cloudformation:us-west-2:123456789012:type/resource/Example-
Test-Alias"
}
```

Per ulteriori informazioni, vedere Using the AWS CloudFormation registry in the AWS CloudFormation User Guide.

- Per i dettagli sull'API, vedere ActivateTypein AWS CLI Command Reference.

`batch-describe-type-configurations`

Il seguente esempio di codice mostra come utilizzare`batch-describe-type-configurations`.

AWS CLI

Come descrivere in batch una configurazione del tipo

L'esempio `batch-describe-type-configurations` seguente configura i dati per il tipo.

```
aws cloudformation batch-describe-type-configurations \
    --region us-west-2 \
    --type-configuration-identifiers TypeArn="arn:aws:cloudformation:us-
west-2:123456789012:type/resource/Example-Test-
Type,TypeConfigurationAlias=MyConfiguration"
```

Output:

```
{
    "Errors": [],
    "UnprocessedTypeConfigurations": [],
    "TypeConfigurations": [
        {
            "Arn": "arn:aws:cloudformation:us-west-2:123456789012:type/resource/
Example-Test-Type",
            "Alias": "MyConfiguration",
            "Configuration": "{\n        \"Example\": {\n            \"ApiKey\":
 \"examplekey\",\n            \"ApplicationKey\": \"examplekey1\",\n
 \"ApiURL\": \"exampleurl\"\n            }\n}",
            "LastUpdated": "2021-10-01T15:25:46.210000+00:00",
            "TypeArn": "arn:aws:cloudformation:us-east-1:123456789012:type/resource/
Example-Test-Type"
        }
    ]
}
```

Per ulteriori informazioni, vedere Using the AWS CloudFormation registry in the AWS CloudFormation User Guide.

- Per i dettagli sull'API, vedere BatchDescribeTypeConfigurationsin AWS CLI Command Reference.

`cancel-update-stack`

Il seguente esempio di codice mostra come utilizzare`cancel-update-stack`.

AWS CLI

Come annullare un aggiornamento di stack in corso

Il comando `cancel-update-stack` seguente annulla un aggiornamento di stack nello stack `myteststack`.

```
aws cloudformation cancel-update-stack --stack-name myteststack
```

- Per i dettagli sull'API, consulta CancelUpdateStack AWS CLICommand Reference.

`continue-update-rollback`

Il seguente esempio di codice mostra come utilizzare`continue-update-rollback`.

AWS CLI

Come riprovare il rollback dell'aggiornamento

L'esempio `continue-update-rollback` seguente riprende un'operazione di rollback da un aggiornamento dello stack precedentemente non riuscito.

```
aws cloudformation continue-update-rollback \
    --stack-name my-stack
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta ContinueUpdateRollback AWS CLICommand Reference.

`create-change-set`

Il seguente esempio di codice mostra come utilizzare`create-change-set`.

AWS CLI

Come creare un set di modifiche

L'esempio `create-change-set` seguente crea un set di modifiche con la capacità `CAPABILITY_IAM`. Il file `template.yaml` è un AWS CloudFormation modello nella cartella corrente che definisce uno stack che include risorse IAM.

```
aws cloudformation create-change-set \
    --stack-name my-application \
    --change-set-name my-change-set \
    --template-body file://template.yaml \
    --capabilities CAPABILITY_IAM
```

Output:

```
{
    "Id": "arn:aws:cloudformation:us-west-2:123456789012:changeSet/my-change-set/
bc9555ba-a949-xmpl-bfb8-f41d04ec5784",
    "StackId": "arn:aws:cloudformation:us-west-2:123456789012:stack/my-application/
d0a825a0-e4cd-xmpl-b9fb-061c69e99204"
}
```

- Per i dettagli sull'API, consulta CreateChangeSet AWS CLICommand Reference.

`create-generated-template`

Il seguente esempio di codice mostra come utilizzare`create-generated-template`.

AWS CLI

Per creare un modello generato da risorse scansionate

L'`create-generated-template`esempio seguente crea un modello generato denominato `MyTemplate` dalle risorse scansionate.

```
aws cloudformation create-generated-template \
    --generated-template-name MyTemplate \
    --resources file://resources.json
```

Contenuto di `resources.json`:

```
[
    {
        "ResourceType": "AWS::EKS::Cluster",
        "LogicalResourceId":"MyCluster",
        "ResourceIdentifier": {
            "ClusterName": "MyAppClusterName"
        }
    },
    {
        "ResourceType": "AWS::AutoScaling::AutoScalingGroup",
        "LogicalResourceId":"MyASG",
        "ResourceIdentifier": {
            "AutoScalingGroupName": "MyAppASGName"
```

```
        }
    },
    {
        "ResourceType": "AWS::EKS::Nodegroup",
        "LogicalResourceId":"MyNodegroup",
        "ResourceIdentifier": {
            "NodegroupName": "MyAppNodegroupName"
        }
    },
    {
        "ResourceType": "AWS::IAM::Role",
        "LogicalResourceId":"MyRole",
        "ResourceIdentifier": {
            "RoleId": "arn:aws::iam::123456789012:role/MyAppIAMRole"
        }
    }
]
```

Output:

```
{
  "Arn":
    "arn:aws:cloudformation:us-east-1:123456789012:generatedtemplate/7fc8512c-
d8cb-4e02-b266-d39c48344e48",
  "Name": "MyTemplate"
}
```

Per ulteriori informazioni, consulta Creare un CloudFormation modello da risorse scansionate con il generatore IAc nella Guida per l'AWS CloudFormation utente.

- Per i dettagli sull'API, consulta AWS CLI Command CreateGeneratedTemplateReference.

`create-stack-instances`

Il seguente esempio di codice mostra come utilizzare`create-stack-instances`.

AWS CLI

Come creare istanze di stack

L'esempio `create-stack-instances` seguente crea istanze di uno stack impostato in due account e quattro Regioni. L'impostazione della tolleranza ai guasti garantisce che

l'aggiornamento venga tentato in tutti gli account e le Regioni, anche se non è possibile creare alcuni stack.

```
aws cloudformation create-stack-instances \
    --stack-set-name my-stack-set \
    --accounts 123456789012 223456789012 \
    --regions us-east-1 us-east-2 us-west-1 us-west-2 \
    --operation-preferences FailureToleranceCount=7
```

Output:

```
{
    "OperationId": "d7995c31-83c2-xmpl-a3d4-e9ca2811563f"
}
```

Per creare uno stack, utilizza il comando `create-stack-set`.

- Per i dettagli sull'API, consulta CreateStackInstances AWS CLICommand Reference.

`create-stack-refactor`

Il seguente esempio di codice mostra come utilizzare`create-stack-refactor`.

AWS CLI

Per creare la definizione dello stack per un'operazione di rifattorizzazione dello stack

L'`create-stack-refactor`esempio seguente crea la definizione dello stack per il refactoring dello stack.

```
aws cloudformation create-stack-refactor \
    --stack-definitions \
      StackName=Stack1,TemplateBody@=file://template1-updated.yaml \
      StackName=Stack2,TemplateBody@=file://template2-updated.yaml
```

Output:

```
{
    "StackRefactorId": "9c384f70-4e07-4ed7-a65d-fee5eb430841"
}
```

Per ulteriori informazioni, consulta Stack refactoring nella Guida per l'utente.AWS CloudFormation

- Per i dettagli sull'API, consulta Command CreateStackRefactorReference AWS CLI .

`create-stack-set`

Il seguente esempio di codice mostra come utilizzare`create-stack-set`.

AWS CLI

Come creare un set di stack

L'`create-stack-set`esempio seguente crea un set di stack utilizzando il modello di file YAML specificato. `template.yaml`è un AWS CloudFormation modello nella cartella corrente che definisce uno stack.

```
aws cloudformation create-stack-set \
    --stack-set-name my-stack-set \
    --template-body file://template.yaml \
    --description "SNS topic"
```

Output:

```
{
    "StackSetId": "my-stack-set:8d0f160b-d157-xmpl-a8e6-c0ce8e5d8cc1"
}
```

Per aggiungere istanze di stack al set di stack, utilizza il comando `create-stack-instances`.

- Per i dettagli sull'API, vedere CreateStackSetin AWS CLI Command Reference.

`create-stack`

Il seguente esempio di codice mostra come utilizzare`create-stack`.

AWS CLI

Per creare una AWS CloudFormation pila

Il comando `create-stacks` seguente crea uno stack denominato `myteststack` utilizzando il modello `sampletemplate.json`:

```
aws cloudformation create-stack --stack-name myteststack --template-body file://
sampletemplate.json --parameters ParameterKey=KeyPairName,ParameterValue=TestKey
 ParameterKey=SubnetIDs,ParameterValue=SubnetID1\\,SubnetID2
```

Output:

```
{
    "StackId": "arn:aws:cloudformation:us-east-1:123456789012:stack/
myteststack/466df9e0-0dff-08e3-8e2f-5088487c4896"
}
```

Per ulteriori informazioni, consulta Stacks nella Guida per l'AWS CloudFormation utente.

- Per i dettagli sull'API, consulta CreateStack AWS CLICommand Reference.

`deactivate-type`

Il seguente esempio di codice mostra come utilizzare`deactivate-type`.

AWS CLI

Come disattivare un tipo

L'esempio `deactivate-type` seguente disattiva un'estensione pubblica precedentemente attivata in questo account e Regione.

```
aws cloudformation deactivate-type \
    --region us-west-2 \
    --type MODULE \
    --type-name Example::Test::Type::MODULE
```

Questo comando non produce alcun output.

Per ulteriori informazioni, vedere Using the AWS CloudFormation registry in the AWS CloudFormation User Guide.

- Per i dettagli sull'API, vedere DeactivateTypein AWS CLI Command Reference.

`delete-change-set`

Il seguente esempio di codice mostra come utilizzare`delete-change-set`.

AWS CLI

Come eliminare un set di modifiche

L'esempio `delete-change-set` seguente elimina un set di modifiche specificando il nome del set di modifiche e il nome dello stack.

```
aws cloudformation delete-change-set \
    --stack-name my-stack \
    --change-set-name my-change-set
```

Questo comando non produce alcun output.

L'esempio `delete-change-set` seguente elimina un set di modifiche specificando l'ARN completo del set di modifiche.

```
aws cloudformation delete-change-set \
    --change-set-name arn:aws:cloudformation:us-east-2:123456789012:changeSet/my-change-set/4eca1a01-e285-xmpl-8026-9a1967bfb4b0
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta DeleteChangeSet AWS CLICommand Reference.

`delete-generated-template`

Il seguente esempio di codice mostra come utilizzare`delete-generated-template`.

AWS CLI

Per eliminare un modello generato

L'`delete-generated-template`esempio seguente elimina il modello specificato.

```
aws cloudformation delete-generated-template \
    --generated-template-name MyTemplate
```

Questo comando non produce alcun output.

Per ulteriori informazioni, vedere Generazione di modelli da risorse esistenti nella Guida per l'AWS CloudFormation utente.

- Per i dettagli sull'API, consulta DeleteGeneratedTemplate AWS CLICommand Reference.

`delete-stack-instances`

Il seguente esempio di codice mostra come utilizzare`delete-stack-instances`.

AWS CLI

Come eliminare istanze di stack

L'esempio `delete-stack-instances` seguente elimina le istanze di uno stack impostato in due account in due Regioni e termina gli stack.

```
aws cloudformation delete-stack-instances \
    --stack-set-name my-stack-set \
    --accounts 123456789012 567890123456 \
    --regions us-east-1 us-west-1 \
    --no-retain-stacks
```

Output:

```
{
    "OperationId": "ad49f10c-fd1d-413f-a20a-8de6e2fa8f27"
}
```

Per eliminare un set di stack vuoto, utilizza il comando `delete-stack-set`.

- Per i dettagli sull'API, consulta DeleteStackInstances AWS CLICommand Reference.

`delete-stack-set`

Il seguente esempio di codice mostra come utilizzare`delete-stack-set`.

AWS CLI

Come eliminare un set di stack

Il comando seguente elimina il set di stack vuoto specificato. Il set di stack deve essere vuoto.

```
aws cloudformation delete-stack-set \
    --stack-set-name my-stack-set
```

Questo comando non produce alcun output.

Per eliminare le istanze dal set di stack, utilizza il comando `delete-stack-instances`.

- Per i dettagli sull'API, consulta DeleteStackSet AWS CLICommand Reference.

`delete-stack`

Il seguente esempio di codice mostra come utilizzare`delete-stack`.

AWS CLI

Come eliminare uno stack

L'esempio `delete-stack` seguente elimina lo stack specificato.

```
aws cloudformation delete-stack \
    --stack-name my-stack
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta DeleteStack AWS CLICommand Reference.

`deploy`

Il seguente esempio di codice mostra come utilizzare`deploy`.

AWS CLI

Il comando seguente implementa il modello denominato `template.json` in uno stack denominato `my-new-stack`:

```
aws cloudformation deploy --template-file /path_to_template/template.json
 --stack-name my-new-stack --parameter-overrides Key1=Value1 Key2=Value2 --
tags Key1=Value1 Key2=Value2
```

- Per informazioni dettagliate sull'API, consulta Deploy in AWS CLI Command Reference.

`deregister-type`

Il seguente esempio di codice mostra come utilizzare`deregister-type`.

AWS CLI

Come annullare la registrazione di una versione del tipo

L'`deregister-type`esempio seguente rimuove la versione del tipo specificato dall'uso attivo nel CloudFormation registro, in modo che non possa più essere utilizzata nelle CloudFormation operazioni.

```
aws cloudformation deregister-type \
    --type RESOURCE \
    --type-name My::Logs::LogGroup \
    --version-id 00000002
```

Questo comando non produce alcun output.

Per ulteriori informazioni, vedere Using the CloudFormation Registry in the AWS CloudFormation Users Guide.

- Per i dettagli sull'API, consulta DeregisterType AWS CLICommand Reference.

`describe-account-limits`

Il seguente esempio di codice mostra come utilizzare`describe-account-limits`.

AWS CLI

Come ottenere informazioni sui limiti dell'account

Il comando seguente recupera un elenco di limiti regionali per l'account corrente.

```
aws cloudformation describe-account-limits
```

Output:

```
{
    "AccountLimits": [
        {
            "Name": "StackLimit",
            "Value": 200
        },
        {
```

```
            "Name": "StackOutputsLimit",
            "Value": 60
        },
        {
            "Name": "ConcurrentResourcesLimit",
            "Value": 2500
        }
    ]
}
```

- Per i dettagli sull'API, consulta DescribeAccountLimits AWS CLICommand Reference.

`describe-change-set`

Il seguente esempio di codice mostra come utilizzare`describe-change-set`.

AWS CLI

Come ottenere informazioni su un set di modifiche

L'esempio `describe-change-set` seguente visualizza i dettagli del set di modifiche specificato tramite il nome del set di modifiche e il nome dello stack.

```
aws cloudformation describe-change-set \
    --change-set-name my-change-set \
    --stack-name my-stack
```

L'esempio `describe-change-set` seguente visualizza i dettagli del set di modifiche specificato tramite l'ARN completo del set di modifiche:

```
aws cloudformation describe-change-set \
    --change-set-name arn:aws:cloudformation:us-west-2:123456789012:changeSet/my-change-set/bc9555ba-a949-xmpl-bfb8-f41d04ec5784
```

Output:

```
{
    "Changes": [
        {
            "Type": "Resource",
```

```
            "ResourceChange": {
                "Action": "Modify",
                "LogicalResourceId": "function",
                "PhysicalResourceId": "my-function-SEZV4XMPL4S5",
                "ResourceType": "AWS::Lambda::Function",
                "Replacement": "False",
                "Scope": [
                    "Properties"
                ],
                "Details": [
                    {
                        "Target": {
                            "Attribute": "Properties",
                            "Name": "Timeout",
                            "RequiresRecreation": "Never"
                        },
                        "Evaluation": "Static",
                        "ChangeSource": "DirectModification"
                    }
                ]
            }
        }
    ],
    "ChangeSetName": "my-change-set",
    "ChangeSetId": "arn:aws:cloudformation:us-west-2:123456789012:changeSet/my-
change-set/4eca1a01-e285-xmpl-8026-9a1967bfb4b0",
    "StackId": "arn:aws:cloudformation:us-west-2:123456789012:stack/my-stack/
d0a825a0-e4cd-xmpl-b9fb-061c69e99204",
    "StackName": "my-stack",
    "Description": null,
    "Parameters": null,
    "CreationTime": "2019-10-02T05:20:56.651Z",
    "ExecutionStatus": "AVAILABLE",
    "Status": "CREATE_COMPLETE",
    "StatusReason": null,
    "NotificationARNs": [],
    "RollbackConfiguration": {},
    "Capabilities": [
        "CAPABILITY_IAM"
    ],
    "Tags": null
}
```

- Per i dettagli sull'API, consulta DescribeChangeSet AWS CLICommand Reference.

describe-generated-template

Il seguente esempio di codice mostra come utilizzaredescribe-generated-template.

AWS CLI

Per descrivere un modello generato

L'describe-generated-templateesempio seguente descrive il modello specificato.

```
aws cloudformation describe-generated-template \
    --generated-template-name MyTemplate
```

Output:

```
{
    "GeneratedTemplateId": "arn:aws:cloudformation:us-
east-1:123456789012:generatedTemplate/7d881acf-f307-4ded-910e-f8fb49b96894",
    "GeneratedTemplateName": "MyTemplate",
    "Resources": [
        {
            "ResourceType": "AWS::EC2::SecurityGroup",
            "LogicalResourceId": "EC2SecurityGroup",
            "ResourceIdentifier": {
                "Id": "sg-1234567890abcdef0"
            },
            "ResourceStatus": "COMPLETE",
            "ResourceStatusReason": "Resource Template complete",
            "Warnings": []
        },
        {
            "ResourceType": "AWS::EC2::Instance",
            "LogicalResourceId": "EC2Instance",
            "ResourceIdentifier": {
                "InstanceId": "i-1234567890abcdef0"
            },
            "ResourceStatus": "COMPLETE",
            "ResourceStatusReason": "Resource Template complete",
            "Warnings": []
        },
        {
            "ResourceType": "AWS::EC2::KeyPair",
            "LogicalResourceId": "EC2KeyPairSshkeypair",
```

```
            "ResourceIdentifier": {
                "KeyName": "sshkeypair"
            },
            "ResourceStatus": "COMPLETE",
            "ResourceStatusReason": "Resource Template complete",
            "Warnings": []
        }
    ],
    "Status": "COMPLETE",
    "StatusReason": "All resources complete",
    "CreationTime": "2025-09-23T19:38:06.435000+00:00",
    "LastUpdatedTime": "2025-09-23T19:38:10.798000+00:00",
    "Progress": {
        "ResourcesSucceeded": 3,
        "ResourcesFailed": 0,
        "ResourcesProcessing": 0,
        "ResourcesPending": 0
    },
    "TemplateConfiguration": {
        "DeletionPolicy": "RETAIN",
        "UpdateReplacePolicy": "RETAIN"
    },
    "TotalWarnings": 0
}
```

Per ulteriori informazioni, vedere Generazione di modelli da risorse esistenti nella Guida AWS CloudFormation per l'utente.

- Per i dettagli sull'API, consulta DescribeGeneratedTemplate AWS CLICommand Reference.

`describe-publisher`

Il seguente esempio di codice mostra come utilizzare`describe-publisher`.

AWS CLI

Come descrivere un publisher

L'esempio `describe-publisher` seguente configura le informazioni per un publisher.

```
aws cloudformation describe-publisher \
    --region us-west-2 \
    --publisher-id 000q6TfUovXsEMmgKowxDZLlwqr2QUsh
```

Output:

```
{
    "PublisherId": "000q6TfUovXsEMmgKowxDZLlwqr2QUshd2e75c8c",
    "PublisherStatus": "VERIFIED",
    "IdentityProvider": "AWS_Marketplace",
    "PublisherProfile": "https://aws.amazon.com/marketplace/seller-profile?
id=2c5dc1f0-17cd-4259-8e46-822a83gdtegd"
}
```

Per ulteriori informazioni, vedere Using the AWS CloudFormation registry in the AWS CloudFormation User Guide.

- Per i dettagli sull'API, vedere DescribePublisherin AWS CLI Command Reference.

`describe-resource-scan`

Il seguente esempio di codice mostra come utilizzare`describe-resource-scan`.

AWS CLI

Per descrivere una scansione delle risorse

L'`describe-resource-scan`esempio seguente descrive la scansione delle risorse con l'ID di scansione specificato.

```
aws cloudformation describe-resource-scan --region \
    --resource-scan-id arn:aws:cloudformation:us-
east-1:123456789012:resourceScan/0a699f15-489c-43ca-a3ef-3e6ecfa5da60
```

Output:

```
{
    "ResourceScanId": "arn:aws:cloudformation:us-
east-1:123456789012:resourceScan/0a699f15-489c-43ca-a3ef-3e6ecfa5da60",
    "Status": "COMPLETE",
    "StartTime": "2025-08-21T03:10:38.485000+00:00",
    "EndTime": "2025-08-21T03:20:28.485000+00:00",
    "PercentageCompleted": 100.0,
    "ResourceTypes": [
        "AWS::CloudFront::CachePolicy",
```

```
        "AWS::CloudFront::OriginRequestPolicy",
        "AWS::EC2::DHCPOptions",
        "AWS::EC2::InternetGateway",
        "AWS::EC2::KeyPair",
        "AWS::EC2::NetworkAcl",
        "AWS::EC2::NetworkInsightsPath",
        "AWS::EC2::NetworkInterface",
        "AWS::EC2::PlacementGroup",
        "AWS::EC2::Route",
        "AWS::EC2::RouteTable",
        "AWS::EC2::SecurityGroup",
        "AWS::EC2::Subnet",
        "AWS::EC2::SubnetCidrBlock",
        "AWS::EC2::SubnetNetworkAclAssociation",
        "AWS::EC2::SubnetRouteTableAssociation",
        ...
    ],
    "ResourcesRead": 676
}
```

Per ulteriori informazioni, consulta Generazione di modelli da risorse esistenti nella Guida AWS CloudFormation per l'utente.

- Per i dettagli sull'API, consulta DescribeResourceScan AWS CLICommand Reference.

`describe-stack-drift-detection-status`

Il seguente esempio di codice mostra come utilizzare`describe-stack-drift-detection-status`.

AWS CLI

Come controllare lo stato di un'operazione di rilevamento delle deviazioni

L'esempio `describe-stack-drift-detection-status` seguente mostra lo stato di un'operazione di rilevamento delle deviazioni. Per ottenere l'ID, esegui il comando `detect-stack-drift`.

```
aws cloudformation describe-stack-drift-detection-status \
    --stack-drift-detection-id 1a229160-e4d9-xmpl-ab67-0a4f93df83d4
```

Output:

```
{
    "StackId": "arn:aws:cloudformation:us-west-2:123456789012:stack/my-stack/
d0a825a0-e4cd-xmpl-b9fb-061c69e99204",
    "StackDriftDetectionId": "1a229160-e4d9-xmpl-ab67-0a4f93df83d4",
    "StackDriftStatus": "DRIFTED",
    "DetectionStatus": "DETECTION_COMPLETE",
    "DriftedStackResourceCount": 1,
    "Timestamp": "2019-10-02T05:54:30.902Z"
}
```

- Per i dettagli sull'API, consulta DescribeStackDriftDetectionStatus AWS CLICommand Reference.

`describe-stack-events`

Il seguente esempio di codice mostra come utilizzare`describe-stack-events`.

AWS CLI

Come descrivere gli eventi dello stack

L'esempio `describe-stack-events` seguente visualizza i 2 eventi più recenti per lo stack specificato.

```
aws cloudformation describe-stack-events \
    --stack-name my-stack \
    --max-items 2

{
    "StackEvents": [
        {
            "StackId": "arn:aws:cloudformation:us-west-2:123456789012:stack/my-
stack/d0a825a0-e4cd-xmpl-b9fb-061c69e99204",
            "EventId": "4e1516d0-e4d6-xmpl-b94f-0a51958a168c",
            "StackName": "my-stack",
            "LogicalResourceId": "my-stack",
            "PhysicalResourceId": "arn:aws:cloudformation:us-
west-2:123456789012:stack/my-stack/d0a825a0-e4cd-xmpl-b9fb-061c69e99204",
            "ResourceType": "AWS::CloudFormation::Stack",
            "Timestamp": "2019-10-02T05:34:29.556Z",
            "ResourceStatus": "UPDATE_COMPLETE"
        },
```

```
        {
            "StackId": "arn:aws:cloudformation:us-west-2:123456789012:stack/my-
stack/d0a825a0-e4cd-xmpl-b9fb-061c69e99204",
            "EventId": "4dd3c810-e4d6-xmpl-bade-0aaf8b31ab7a",
            "StackName": "my-stack",
            "LogicalResourceId": "my-stack",
            "PhysicalResourceId": "arn:aws:cloudformation:us-
west-2:123456789012:stack/my-stack/d0a825a0-e4cd-xmpl-b9fb-061c69e99204",
            "ResourceType": "AWS::CloudFormation::Stack",
            "Timestamp": "2019-10-02T05:34:29.127Z",
            "ResourceStatus": "UPDATE_COMPLETE_CLEANUP_IN_PROGRESS"
        }
    ],

 "NextToken": "eyJOZXh0VG9XMPLiOiBudWxsLCAiYm90b190cnVuY2F0ZV9hbW91bnQiOiAyfQ=="
}
```

- Per i dettagli sull'API, consulta DescribeStackEvents AWS CLICommand Reference.

`describe-stack-instance`

Il seguente esempio di codice mostra come utilizzare`describe-stack-instance`.

AWS CLI

Come descrivere un'istanza di stack

Il comando seguente descrive un'istanza dello stack specificato impostato nell'account e nella Regione specificati. Il set di stack si trova nella Regione e nell'account correnti e l'istanza si trova nella Regione `us-west-2` dell'account `123456789012`:

```
aws cloudformation describe-stack-instance \
    --stack-set-name my-stack-set \
    --stack-instance-account 123456789012 \
    --stack-instance-region us-west-2
```

Output:

```
{
    "StackInstance": {
        "StackSetId": "enable-config:296a3360-xmpl-40af-be78-9341e95bf743",
        "Region": "us-west-2",
```

```
        "Account": "123456789012",
        "StackId": "arn:aws:cloudformation:us-west-2:123456789012:stack/
StackSet-enable-config-e6cac20f-xmpl-46e9-8314-53e0d4591532/4287f9a0-e615-
xmpl-894a-12b31d3117be",
        "ParameterOverrides": [],
        "Status": "OUTDATED",
        "StatusReason": "ResourceLogicalId:ConfigBucket,
 ResourceType:AWS::S3::Bucket, ResourceStatusReason:You have attempted to create
 more buckets than allowed (Service: Amazon S3; Status Code: 400; Error Code:
 TooManyBuckets; Request ID: F7F21CXMPL580224; S3 Extended Request ID: egd/
Fdt89BXMPLyiqbMNljVk55Yqqvi3NYW2nKLUVWhUGEhNfCmZdyj967lhriaG/dWMobSO40o=)."
    }
}
```

- Per i dettagli sull'API, consulta DescribeStackInstance AWS CLICommand Reference.

`describe-stack-refactor`

Il seguente esempio di codice mostra come utilizzare`describe-stack-refactor`.

AWS CLI

Per descrivere un'operazione di rifattorizzazione dello stack

L'`describe-stack-refactor`esempio seguente descrive l'operazione di rifattorizzazione dello stack con l'ID di rifattore dello stack specificato.

```
aws cloudformation describe-stack-refactor \
    --stack-refactor-id 9c384f70-4e07-4ed7-a65d-fee5eb430841
```

Output:

```
{
    "StackRefactorId": "9c384f70-4e07-4ed7-a65d-fee5eb430841",
    "StackIds": [
        "arn:aws:cloudformation:us-east-1:123456789012:stack/
Stack1/3e6a1ff0-94b1-11f0-aa6f-0a88d2e03acf",
        "arn:aws:cloudformation:us-east-1:123456789012:stack/
Stack2/5da91650-94b1-11f0-81cf-0a23500e151b"
    ],
    "ExecutionStatus": "AVAILABLE",
    "Status": "CREATE_COMPLETE"
```

```
}
```

Per ulteriori informazioni, consulta Stack refactoring nella Guida per l'utente.AWS CloudFormation

- Per i dettagli sull'API, consulta Command DescribeStackRefactorReference AWS CLI .

`describe-stack-resource-drifts`

Il seguente esempio di codice mostra come utilizzare`describe-stack-resource-drifts`.

AWS CLI

Come ottenere informazioni sulle risorse con deviazioni rispetto alla definizione dello stack

Il comando seguente visualizza informazioni sulle risorse con deviazioni per lo stack specificato. Per avviare il rilevamento delle deviazioni, utilizza il comando `detect-stack-drift`:

```
aws cloudformation describe-stack-resource-drifts \
    --stack-name my-stack
```

L'output mostra una funzione AWS Lambda che è stata modificata: out-of-band

```
{
    "StackResourceDrifts": [
        {
            "StackId": "arn:aws:cloudformation:us-west-2:123456789012:stack/my-
stack/d0a825a0-e4cd-xmpl-b9fb-061c69e99204",
            "LogicalResourceId": "function",
            "PhysicalResourceId": "my-function-SEZV4XMPL4S5",
            "ResourceType": "AWS::Lambda::Function",
            "ExpectedProperties": "{\"Description\":\"Write a file to S3.\",
\"Environment\":{\"Variables\":{\"bucket\":\"my-stack-bucket-1vc62xmplgguf
\"}},\"Handler\":\"index.handler\",\"MemorySize\":128,\"Role\":
\"arn:aws:iam::123456789012:role/my-functionRole-HIZXMPLEOM9E\",\"Runtime\":
\"nodejs10.x\",\"Tags\":[{\"Key\":\"lambda:createdBy\",\"Value\":\"SAM\"}],\"Timeout
\":900,\"TracingConfig\":{\"Mode\":\"Active\"}}",
            "ActualProperties": "{\"Description\":\"Write a file to S3.\",
\"Environment\":{\"Variables\":{\"bucket\":\"my-stack-bucket-1vc62xmplgguf
\"}},\"Handler\":\"index.handler\",\"MemorySize\":256,\"Role\":
\"arn:aws:iam::123456789012:role/my-functionRole-HIZXMPLEOM9E\",\"Runtime\":
\"nodejs10.x\",\"Tags\":[{\"Key\":\"lambda:createdBy\",\"Value\":\"SAM\"}],\"Timeout
\":22,\"TracingConfig\":{\"Mode\":\"Active\"}}",
```

```
            "PropertyDifferences": [
                {
                    "PropertyPath": "/MemorySize",
                    "ExpectedValue": "128",
                    "ActualValue": "256",
                    "DifferenceType": "NOT_EQUAL"
                },
                {
                    "PropertyPath": "/Timeout",
                    "ExpectedValue": "900",
                    "ActualValue": "22",
                    "DifferenceType": "NOT_EQUAL"
                }
            ],
            "StackResourceDriftStatus": "MODIFIED",
            "Timestamp": "2019-10-02T05:54:44.064Z"
        }
    ]
}
```

- Per i dettagli sull'API, consulta DescribeStackResourceDrifts AWS CLICommand Reference.

`describe-stack-resource`

Il seguente esempio di codice mostra come utilizzare`describe-stack-resource`.

AWS CLI

Come ottenere informazioni su una risorsa dello stack

L'esempio `describe-stack-resource` seguente visualizza i dettagli della risorsa denominata `MyFunction` nello stack specificato.

```
aws cloudformation describe-stack-resource \
    --stack-name MyStack \
    --logical-resource-id MyFunction
```

Output:

```
{
    "StackResourceDetail": {
        "StackName": "MyStack",
```

```
        "StackId": "arn:aws:cloudformation:us-east-2:123456789012:stack/MyStack/
d0a825a0-e4cd-xmpl-b9fb-061c69e99204",
        "LogicalResourceId": "MyFunction",
        "PhysicalResourceId": "my-function-SEZV4XMPL4S5",
        "ResourceType": "AWS::Lambda::Function",
        "LastUpdatedTimestamp": "2019-10-02T05:34:27.989Z",
        "ResourceStatus": "UPDATE_COMPLETE",
        "Metadata": "{}",
        "DriftInformation": {
            "StackResourceDriftStatus": "IN_SYNC"
        }
    }
}
```

- Per i dettagli sull'API, consulta DescribeStackResource AWS CLICommand Reference.

`describe-stack-resources`

Il seguente esempio di codice mostra come utilizzare`describe-stack-resources`.

AWS CLI

Come ottenere informazioni su una risorsa dello stack

L'esempio `describe-stack-resources` seguente visualizza i dettagli delle risorse nello stack specificato.

```
aws cloudformation describe-stack-resources \
    --stack-name my-stack
```

Output:

```
{
    "StackResources": [
        {
            "StackName": "my-stack",
            "StackId": "arn:aws:cloudformation:us-west-2:123456789012:stack/my-
stack/d0a825a0-e4cd-xmpl-b9fb-061c69e99204",
            "LogicalResourceId": "bucket",
            "PhysicalResourceId": "my-stack-bucket-1vc62xmplgguf",
            "ResourceType": "AWS::S3::Bucket",
            "Timestamp": "2019-10-02T04:34:11.345Z",
```

```
            "ResourceStatus": "CREATE_COMPLETE",
            "DriftInformation": {
                "StackResourceDriftStatus": "IN_SYNC"
            }
        },
        {
            "StackName": "my-stack",
            "StackId": "arn:aws:cloudformation:us-west-2:123456789012:stack/my-
stack/d0a825a0-e4cd-xmpl-b9fb-061c69e99204",
            "LogicalResourceId": "function",
            "PhysicalResourceId": "my-function-SEZV4XMPL4S5",
            "ResourceType": "AWS::Lambda::Function",
            "Timestamp": "2019-10-02T05:34:27.989Z",
            "ResourceStatus": "UPDATE_COMPLETE",
            "DriftInformation": {
                "StackResourceDriftStatus": "IN_SYNC"
            }
        },
        {
            "StackName": "my-stack",
            "StackId": "arn:aws:cloudformation:us-west-2:123456789012:stack/my-
stack/d0a825a0-e4cd-xmpl-b9fb-061c69e99204",
            "LogicalResourceId": "functionRole",
            "PhysicalResourceId": "my-functionRole-HIZXMPLEOM9E",
            "ResourceType": "AWS::IAM::Role",
            "Timestamp": "2019-10-02T04:34:06.350Z",
            "ResourceStatus": "CREATE_COMPLETE",
            "DriftInformation": {
                "StackResourceDriftStatus": "IN_SYNC"
            }
        }
    ]
}
```

- Per i dettagli sull'API, consulta DescribeStackResources AWS CLICommand Reference.

`describe-stack-set-operation`

Il seguente esempio di codice mostra come utilizzare`describe-stack-set-operation`.

AWS CLI

Come ottenere informazioni su un'operazione di set di stack

Il seguente describe-stack-set-operation `esempio mostra i dettagli di un'operazione di aggiornamento sul set di stack specificato.

```
aws cloudformation describe-stack-set-operation \
    --stack-set-name enable-config \
    --operation-id 35d45ebc-ed88-xmpl-ab59-0197a1fc83a0
```

Output:

```
{
    "StackSetOperation": {
        "OperationId": "35d45ebc-ed88-xmpl-ab59-0197a1fc83a0",
        "StackSetId": "enable-config:296a3360-xmpl-40af-be78-9341e95bf743",
        "Action": "UPDATE",
        "Status": "SUCCEEDED",
        "OperationPreferences": {
            "RegionOrder": [
                "us-east-1",
                "us-west-2",
                "eu-west-1",
                "us-west-1"
            ],
            "FailureToleranceCount": 7,
            "MaxConcurrentCount": 2
        },
        "AdministrationRoleARN": "arn:aws:iam::123456789012:role/
AWSCloudFormationStackSetAdministrationRole",
        "ExecutionRoleName": "AWSCloudFormationStackSetExecutionRole",
        "CreationTimestamp": "2019-10-03T16:28:44.377Z",
        "EndTimestamp": "2019-10-03T16:42:08.607Z"
    }
}
```

- Per i dettagli sull'API, vedere DescribeStackSetOperationin AWS CLI Command Reference.

`describe-stack-set`

Il seguente esempio di codice mostra come utilizzare`describe-stack-set`.

AWS CLI

Come ottenere informazioni su un set di stack

Il seguente describe-stack-set `esempio mostra i dettagli sul set di stack specificato.

```
aws cloudformation describe-stack-set \
    --stack-set-name my-stack-set
```

Output:

```
{
    "StackSet": {
        "StackSetName": "my-stack-set",
        "StackSetId": "my-stack-set:296a3360-xmpl-40af-be78-9341e95bf743",
        "Description": "Create an Amazon SNS topic",
        "Status": "ACTIVE",
        "TemplateBody": "AWSTemplateFormatVersion: '2010-09-09'\nDescription: An AWS
 SNS topic\nResources:\n  topic:\n    Type: AWS::SNS::Topic",
        "Parameters": [],
        "Capabilities": [],
        "Tags": [],
        "StackSetARN": "arn:aws:cloudformation:us-west-2:123456789012:stackset/
enable-config:296a3360-xmpl-40af-be78-9341e95bf743",
        "AdministrationRoleARN": "arn:aws:iam::123456789012:role/
AWSCloudFormationStackSetAdministrationRole",
        "ExecutionRoleName": "AWSCloudFormationStackSetExecutionRole"
    }
}
```

- Per i dettagli sull'API, vedere DescribeStackSetin AWS CLI Command Reference.

`describe-stacks`

Il seguente esempio di codice mostra come utilizzare`describe-stacks`.

AWS CLI

Per descrivere le AWS CloudFormation pile

Il comando `describe-stacks` seguente mostra le informazioni di riepilogo per lo stack `myteststack`:

```
aws cloudformation describe-stacks --stack-name myteststack
```

Output:

```
{
    "Stacks":  [
        {
            "StackId": "arn:aws:cloudformation:us-east-1:123456789012:stack/
myteststack/466df9e0-0dff-08e3-8e2f-5088487c4896",
            "Description": "AWS CloudFormation Sample Template S3_Bucket: Sample
 template showing how to create a publicly accessible S3 bucket. **WARNING** This
 template creates an S3 bucket. You will be billed for the AWS resources used if you
 create a stack from this template.",
            "Tags": [],
            "Outputs": [
                {
                    "Description": "Name of S3 bucket to hold website content",
                    "OutputKey": "BucketName",
                    "OutputValue": "myteststack-s3bucket-jssofi1zie2w"
                }
            ],
            "StackStatusReason": null,
            "CreationTime": "2013-08-23T01:02:15.422Z",
            "Capabilities": [],
            "StackName": "myteststack",
            "StackStatus": "CREATE_COMPLETE",
            "DisableRollback": false
        }
    ]
}
```

Per ulteriori informazioni, consulta Stacks nella Guida per l'AWS CloudFormation utente.

- Per i dettagli sull'API, consulta DescribeStacks AWS CLICommand Reference.

`describe-type-registration`

Il seguente esempio di codice mostra come utilizzare`describe-type-registration`.

AWS CLI

Come visualizzare le informazioni di registrazione del tipo

L'esempio `describe-type-registration` seguente visualizza informazioni sulla registrazione del tipo specificato, inclusi lo stato, il tipo e la versione correnti del tipo.

```
aws cloudformation describe-type-registration \
```

```
    --registration-token a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Output:

```
{
    "ProgressStatus": "COMPLETE",
    "TypeArn": "arn:aws:cloudformation:us-west-2:123456789012:type/resource/My-Logs-LogGroup",
    "Description": "Deployment is currently in DEPLOY_STAGE of status COMPLETED; ",
    "TypeVersionArn": "arn:aws:cloudformation:us-west-2:123456789012:type/resource/My-Logs-LogGroup/00000001"
}
```

Per ulteriori informazioni, vedere Using the CloudFormation Registry in the AWS CloudFormation Users Guide.

- Per i dettagli sull'API, consulta DescribeTypeRegistration AWS CLICommand Reference.

`describe-type`

Il seguente esempio di codice mostra come utilizzare`describe-type`.

AWS CLI

Come visualizzare le informazioni sul tipo

L'esempio `describe-type` seguente visualizza le informazioni sul tipo specificato.

```
aws cloudformation describe-type \
    --type-name My::Logs::LogGroup \
    --type RESOURCE
```

Output:

```
{
    "SourceUrl": "https://github.com/aws-cloudformation/aws-cloudformation-resource-providers-logs.git",
    "Description": "Customized resource derived from AWS::Logs::LogGroup",
    "TimeCreated": "2019-12-03T23:29:33.321Z",
    "Visibility": "PRIVATE",
    "TypeName": "My::Logs::LogGroup",
    "LastUpdated": "2019-12-03T23:29:33.321Z",
```

```
    "DeprecatedStatus": "LIVE",
    "ProvisioningType": "FULLY_MUTABLE",
    "Type": "RESOURCE",
    "Arn": "arn:aws:cloudformation:us-west-2:123456789012:type/resource/My-Logs-
LogGroup/00000001",
    "Schema": "[details omitted]"
}
```

Per ulteriori informazioni, vedere Using the CloudFormation Registry in the AWS CloudFormation Users Guide.

- Per i dettagli sull'API, consulta DescribeType AWS CLICommand Reference.

`detect-stack-drift`

Il seguente esempio di codice mostra come utilizzare`detect-stack-drift`.

AWS CLI

Come rilevare risorse con deviazioni

L'esempio `detect-stack-drift` seguente avvia il rilevamento delle deviazioni per lo stack specificato.

```
aws cloudformation detect-stack-drift \
    --stack-name my-stack
```

Output:

```
{
    "StackDriftDetectionId": "1a229160-e4d9-xmpl-ab67-0a4f93df83d4"
}
```

È quindi possibile utilizzare questo ID con il comando `describe-stack-resource-drifts` per descrivere le risorse con deviazioni.

- Per i dettagli sull'API, consulta DetectStackDrift AWS CLICommand Reference.

`detect-stack-resource-drift`

Il seguente esempio di codice mostra come utilizzare`detect-stack-resource-drift`.

AWS CLI

Come rilevare la deviazione di una risorsa

L'esempio `detect-stack-resource-drift` seguente verifica la presenza di deviazione per una risorsa denominata `MyFunction` in uno stack denominato `MyStack`:

```
aws cloudformation detect-stack-resource-drift \
   --stack-name MyStack \
   --logical-resource-id MyFunction
```

L'output mostra una funzione AWS Lambda che è stata modificata: out-of-band

```
{
    "StackResourceDrift": {
        "StackId": "arn:aws:cloudformation:us-west-2:123456789012:stack/MyStack/
d0a825a0-e4cd-xmpl-b9fb-061c69e99204",
        "LogicalResourceId": "MyFunction",
        "PhysicalResourceId": "my-function-SEZV4XMPL4S5",
        "ResourceType": "AWS::Lambda::Function",
        "ExpectedProperties": "{\"Description\":\"Write a file to S3.\",
\"Environment\":{\"Variables\":{\"bucket\":\"my-stack-bucket-1vc62xmplgguf
\"}},\"Handler\":\"index.handler\",\"MemorySize\":128,\"Role\":
\"arn:aws:iam::123456789012:role/my-functionRole-HIZXMPLEOM9E\",\"Runtime\":
\"nodejs10.x\",\"Tags\":[{\"Key\":\"lambda:createdBy\",\"Value\":\"SAM\"}],\"Timeout
\":900,\"TracingConfig\":{\"Mode\":\"Active\"}}",
        "ActualProperties": "{\"Description\":\"Write a file to S3.\",\"Environment
\":{\"Variables\":{\"bucket\":\"my-stack-bucket-1vc62xmplgguf\"}},\"Handler\":
\"index.handler\",\"MemorySize\":256,\"Role\":\"arn:aws:iam::123456789012:role/
my-functionRole-HIZXMPLEOM9E\",\"Runtime\":\"nodejs10.x\",\"Tags\":[{\"Key\":
\"lambda:createdBy\",\"Value\":\"SAM\"}],\"Timeout\":22,\"TracingConfig\":{\"Mode\":
\"Active\"}}",
        "PropertyDifferences": [
            {
                "PropertyPath": "/MemorySize",
                "ExpectedValue": "128",
                "ActualValue": "256",
                "DifferenceType": "NOT_EQUAL"
            },
            {
                "PropertyPath": "/Timeout",
                "ExpectedValue": "900",
                "ActualValue": "22",
```

```
                "DifferenceType": "NOT_EQUAL"
            }
        ],
        "StackResourceDriftStatus": "MODIFIED",
        "Timestamp": "2019-10-02T05:58:47.433Z"
    }
}
```

- Per i dettagli sull'API, consulta DetectStackResourceDrift AWS CLICommand Reference.

`detect-stack-set-drift`

Il seguente esempio di codice mostra come utilizzare`detect-stack-set-drift`.

AWS CLI

Come rilevare la deviazione su un set di stack e su tutte le istanze di stack associate

L'esempio `detect-stack-set-drift` seguente avvia le operazioni di rilevamento delle deviazioni sul set di stack specificato, incluse tutte le istanze di stack associate al set di stack, e restituisce un ID di operazione che può essere utilizzato per tenere traccia dello stato dell'operazione di deviazione.

```
aws cloudformation detect-stack-set-drift \
    --stack-set-name stack-set-drift-example
```

Output:

```
{
    "OperationId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
}
```

Per ulteriori informazioni, vedere Rilevamento delle modifiche di configurazione non gestite negli stack set nella Guida per l'AWS CloudFormation utente.

- Per i dettagli sull'API, consulta Command DetectStackSetDriftReference AWS CLI .

`estimate-template-cost`

Il seguente esempio di codice mostra come utilizzare`estimate-template-cost`.

AWS CLI

Come stimare il costo del modello

L'esempio `estimate-template-cost` seguente genera una stima dei costi per un modello denominato `template.yaml` nella cartella corrente.

```
aws cloudformation estimate-template-cost \
    --template-body file://template.yaml
```

Output:

```
{
    "Url": "http://calculator.s3.amazonaws.com/calc5.html?
key=cloudformation/7870825a-xmpl-4def-92e7-c4f8dd360cca"
}
```

- Per i dettagli sull'API, consulta EstimateTemplateCost AWS CLICommand Reference.

`execute-change-set`

Il seguente esempio di codice mostra come utilizzare`execute-change-set`.

AWS CLI

Come eseguire un set di modifiche

L'esempio `execute-change-set` seguente esegue un set di modifiche specificato tramite il nome del set di modifiche e il nome dello stack.

```
aws cloudformation execute-change-set \
    --change-set-name my-change-set \
    --stack-name my-stack
```

L'esempio `execute-change-set` seguente esegue un set di modifiche specificato tramite l'ARN completo del set di modifiche.

```
aws cloudformation execute-change-set \
    --change-set-name arn:aws:cloudformation:us-west-2:123456789012:changeSet/my-
change-set/bc9555ba-a949-xmpl-bfb8-f41d04ec5784
```

- Per i dettagli sull'API, consulta ExecuteChangeSet AWS CLICommand Reference.

`execute-stack-refactor`

Il seguente esempio di codice mostra come utilizzare`execute-stack-refactor`.

AWS CLI

Per completare un'operazione di rifattorizzazione dello stack

L'`execute-stack-refactor`esempio seguente completa l'operazione di rifattorizzazione dello stack con l'ID di rifattore dello stack specificato.

```
aws cloudformation execute-stack-refactor \
    --stack-refactor-id 9c384f70-4e07-4ed7-a65d-fee5eb430841
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Stack refactoring nella Guida per l'utente.AWS CloudFormation

- Per i dettagli sull'API, consulta Command ExecuteStackRefactorReference AWS CLI .

`get-stack-policy`

Il seguente esempio di codice mostra come utilizzare`get-stack-policy`.

AWS CLI

Come visualizzare una policy di stack

L'esempio `get-stack-policy` seguente visualizza la policy di stack per lo stack specificato. Per collegare una policy a uno stack, utilizza il comando `set-stack-policy`.

```
aws cloudformation get-stack-policy \
    --stack-name my-stack
```

Output:

```
{
    "StackPolicyBody": "{\n  \"Statement\" : [\n    {\n      \"Effect\" :
 \"Allow\",\n      \"Action\" : \"Update:*\",\n      \"Principal\": \"*\",\n
```

```
     \"Resource\" : \"*\"\n    },\n    {\n      \"Effect\" : \"Deny\",\n
 \"Action\" : \"Update:*\",\n      \"Principal\": \"*\",\n      \"Resource\" :
 \"LogicalResourceId/bucket\"\n    }\n  ]\n}\n"
}
```

- Per i dettagli sull'API, consulta GetStackPolicy AWS CLICommand Reference.

`get-template-summary`

Il seguente esempio di codice mostra come utilizzare`get-template-summary`.

AWS CLI

Come visualizzare un riepilogo del modello

Il comando seguente visualizza informazioni di riepilogo sulle risorse e sui metadati per il file modello specificato.

```
aws cloudformation get-template-summary \
   --template-body file://template.yaml
```

Output:

```
{
    "Parameters": [],
    "Description": "A VPC and subnets.",
    "ResourceTypes": [
        "AWS::EC2::VPC",
        "AWS::EC2::Subnet",
        "AWS::EC2::Subnet",
        "AWS::EC2::RouteTable",
        "AWS::EC2::VPCEndpoint",
        "AWS::EC2::SubnetRouteTableAssociation",
        "AWS::EC2::SubnetRouteTableAssociation",
        "AWS::EC2::VPCEndpoint"
    ],
    "Version": "2010-09-09"
}
```

- Per i dettagli sull'API, consulta GetTemplateSummary AWS CLICommand Reference.

`get-template`

Il seguente esempio di codice mostra come utilizzare`get-template`.

AWS CLI

Per visualizzare il corpo del modello per uno AWS CloudFormation stack

Il comando `get-template` seguente mostra il modello per lo stack `myteststack`:

```
aws cloudformation get-template --stack-name myteststack
```

Output:

```
{
    "TemplateBody": {
        "AWSTemplateFormatVersion": "2010-09-09",
        "Outputs": {
            "BucketName": {
                "Description": "Name of S3 bucket to hold website content",
                "Value": {
                    "Ref": "S3Bucket"
                }
            }
        },
        "Description": "AWS CloudFormation Sample Template S3_Bucket: Sample
 template showing how to create a publicly accessible S3 bucket. **WARNING** This
 template creates an S3 bucket. You will be billed for the AWS resources used if you
 create a stack from this template.",
        "Resources": {
            "S3Bucket": {
                "Type": "AWS::S3::Bucket",
                "Properties": {
                    "AccessControl": "PublicRead"
                }
            }
        }
    }
}
```

- Per i dettagli sull'API, consulta GetTemplate AWS CLICommand Reference.

`list-change-sets`

Il seguente esempio di codice mostra come utilizzare`list-change-sets`.

AWS CLI

Come elencare i set di modifiche

L'esempio `list-change-sets` seguente visualizza un elenco dei set di modifiche in sospeso per lo stack specificato.

```
aws cloudformation list-change-sets \
   --stack-name my-stack
```

Output:

```
{
    "Summaries": [
        {
            "StackId": "arn:aws:cloudformation:us-west-2:123456789012:stack/my-
stack/d0a825a0-e4cd-xmpl-b9fb-061c69e99204",
            "StackName": "my-stack",
            "ChangeSetId": "arn:aws:cloudformation:us-west-2:123456789012:changeSet/
my-change-set/70160340-7914-xmpl-bcbf-128a1fa78b5d",
            "ChangeSetName": "my-change-set",
            "ExecutionStatus": "AVAILABLE",
            "Status": "CREATE_COMPLETE",
            "CreationTime": "2019-10-02T05:38:54.297Z"
        }
    ]
}
```

- Per i dettagli sull'API, consulta ListChangeSets AWS CLICommand Reference.

`list-exports`

Il seguente esempio di codice mostra come utilizzare`list-exports`.

AWS CLI

Come elencare le esportazioni

L'esempio `list-exports` seguente visualizza un elenco delle esportazioni dagli stack nella Regione corrente.

```
aws cloudformation list-exports
```

Output:

```
{
    "Exports": [
        {
            "ExportingStackId": "arn:aws:cloudformation:us-
west-2:123456789012:stack/private-vpc/99764070-b56c-xmpl-bee8-062a88d1d800",
            "Name": "private-vpc-subnet-a",
            "Value": "subnet-07b410xmplddcfa03"
        },
        {
            "ExportingStackId": "arn:aws:cloudformation:us-
west-2:123456789012:stack/private-vpc/99764070-b56c-xmpl-bee8-062a88d1d800",
            "Name": "private-vpc-subnet-b",
            "Value": "subnet-075ed3xmplebd2fb1"
        },
        {
            "ExportingStackId": "arn:aws:cloudformation:us-
west-2:123456789012:stack/private-vpc/99764070-b56c-xmpl-bee8-062a88d1d800",
            "Name": "private-vpc-vpcid",
            "Value": "vpc-011d7xmpl100e9841"
        }
    ]
}
```

- Per i dettagli sull'API, consulta ListExports AWS CLICommand Reference.

`list-generated-templates`

Il seguente esempio di codice mostra come utilizzare`list-generated-templates`.

AWS CLI

Per elencare i modelli generati

L'`list-generated-templates`esempio seguente elenca tutti i modelli generati.

```
aws cloudformation list-generated-templates
```

Output:

```
{
    "Summaries": [
        {
            "GeneratedTemplateId": "arn:aws:cloudformation:us-
east-1:123456789012:generatedtemplate/7fc8512c-d8cb-4e02-b266-d39c48344e48",
            "GeneratedTemplateName": "MyTemplate",
            "Status": "COMPLETE",
            "StatusReason": "All resources complete",
            "CreationTime": "2025-09-23T20:13:24.283000+00:00",
            "LastUpdatedTime": "2025-09-23T20:13:28.610000+00:00",
            "NumberOfResources": 4
        },
        {
            "GeneratedTemplateId": "arn:aws:cloudformation:us-
east-1:123456789012:generatedTemplate/f10dd1c4-edc6-4823-8153-ab6112b8d051",
            "GeneratedTemplateName": "MyEC2InstanceTemplate",
            "Status": "COMPLETE",
            "StatusReason": "All resources complete",
            "CreationTime": "2024-08-08T19:35:49.790000+00:00",
            "LastUpdatedTime": "2024-08-08T19:35:52.207000+00:00",
            "NumberOfResources": 3
        },
        {
            "GeneratedTemplateId": "arn:aws:cloudformation:us-
east-1:123456789012:generatedTemplate/e5a1c89f-7ce2-41bd-9bdf-75b7c852e3ca",
            "GeneratedTemplateName": "MyEKSNodeGroupTemplate",
            "Status": "COMPLETE",
            "StatusReason": "All resources complete",
            "CreationTime": "2024-07-16T20:39:27.883000+00:00",
            "LastUpdatedTime": "2024-07-16T20:39:35.766000+00:00",
            "NumberOfResources": 4
        }
    ]
}
```

Per ulteriori informazioni, vedere Generazione di modelli da risorse esistenti nella Guida AWS CloudFormation per l'utente.

- Per i dettagli sull'API, consulta ListGeneratedTemplates AWS CLICommand Reference.

`list-imports`

Il seguente esempio di codice mostra come utilizzare`list-imports`.

AWS CLI

Come elencare le importazioni

L'esempio `list-imports` seguente elenca gli stack che importano l'esportazione specificata. Per ottenere l'elenco delle esportazioni disponibili, utilizza il comando `list-exports`.

```
aws cloudformation list-imports \
    --export-name private-vpc-vpcid
```

Output:

```
{
    "Imports": [
        "my-database-stack"
    ]
}
```

- Per i dettagli sull'API, consulta ListImports AWS CLICommand Reference.

`list-resource-scan-related-resources`

Il seguente esempio di codice mostra come utilizzare`list-resource-scan-related-resources`.

AWS CLI

Per elencare le risorse correlate da una scansione delle risorse

L'`list-resource-scan-related-resources`esempio seguente elenca le risorse della scansione delle risorse specificata che sono correlate alle risorse in`resources.json`.

```
aws cloudformation list-resource-scan-related-resources \
    --resource-scan-id arn:aws:cloudformation:us-
east-1:123456789012:resourceScan/0a699f15-489c-43ca-a3ef-3e6ecfa5da60 \
    --resources file://resources.json
```

Contenuto di `resources.json`:

```
[
    {
        "ResourceType": "AWS::EKS::Cluster",
        "ResourceIdentifier": {
            "ClusterName": "MyAppClusterName"
        }
    },
    {
        "ResourceType": "AWS::AutoScaling::AutoScalingGroup",
        "ResourceIdentifier": {
            "AutoScalingGroupName": "MyAppASGName"
        }
    }
]
```

Output:

```
{
    "RelatedResources": [
        {
            "ResourceType": "AWS::EKS::Nodegroup",
            "ResourceIdentifier": {
                "NodegroupName": "MyAppNodegroupName"
            },
            "ManagedByStack": false
        },
        {
            "ResourceType": "AWS::IAM::Role",
            "ResourceIdentifier": {
                "RoleId": "arn:aws::iam::123456789012:role/MyAppIAMRole"
            },
            "ManagedByStack": false
        }
    ]
}
```

Per ulteriori informazioni, consulta Creare un CloudFormation modello da risorse scansionate con il generatore IAc nella Guida per l'AWS CloudFormation utente.

- Per i dettagli sull'API, consulta AWS CLI Command ListResourceScanRelatedResourcesReference.

`list-resource-scan-resources`

Il seguente esempio di codice mostra come utilizzare`list-resource-scan-resources`.

AWS CLI

Per elencare le risorse da una scansione delle risorse

L'`list-resource-scan-resources`esempio seguente elenca le risorse della scansione delle risorse specificata, filtrate per identificatore di risorsa.

```
aws cloudformation list-resource-scan-resources \
    --resource-scan-id arn:aws:cloudformation:us-
east-1:123456789012:resourceScan/0a699f15-489c-43ca-a3ef-3e6ecfa5da60 \
    --resource-identifier MyApp
```

Output:

```
{
    "Resources": [
        {
            "ResourceType": "AWS::EKS::Cluster",
            "ResourceIdentifier": {
                "ClusterName": "MyAppClusterName"
            },
            "ManagedByStack": false
        },
        {
            "ResourceType": "AWS::AutoScaling::AutoScalingGroup",
            "ResourceIdentifier": {
                "AutoScalingGroupName": "MyAppASGName"
            },
            "ManagedByStack": false
        }
    ]
}
```

Per ulteriori informazioni, consulta Creare un CloudFormation modello da risorse scansionate con il generatore IAc nella Guida per l'utente.AWS CloudFormation

- Per i dettagli sull'API, consulta AWS CLI Command ListResourceScanResourcesReference.

`list-stack-instances`

Il seguente esempio di codice mostra come utilizzare`list-stack-instances`.

AWS CLI

Come elencare le istanze di uno stack

L'esempio `list-stack-instances` seguente elenca le istanze create dal set di stack specificato.

```
aws cloudformation list-stack-instances \
    --stack-set-name enable-config
```

L'output di esempio include i dettagli su uno stack che non è stato aggiornato a causa di un errore:

```
{
    "Summaries": [
        {
            "StackSetId": "enable-config:296a3360-xmpl-40af-be78-9341e95bf743",
            "Region": "us-west-2",
            "Account": "123456789012",
            "StackId": "arn:aws:cloudformation:ap-northeast-1:123456789012:stack/
StackSet-enable-config-35a6ac50-d9f8-4084-86e4-7da34d5de4c4/a1631cd0-e5fb-xmpl-
b474-0aa20f14f06e",
            "Status": "CURRENT"
        },
        {
            "StackSetId": "enable-config:296a3360-xmpl-40af-be78-9341e95bf743",
            "Region": "us-west-2",
            "Account": "123456789012",
            "StackId": "arn:aws:cloudformation:us-west-2:123456789012:stack/
StackSet-enable-config-e6cac20f-xmpl-46e9-8314-53e0d4591532/eab53680-e5fa-xmpl-
ba14-0a522351f81e",
            "Status": "OUTDATED",
            "StatusReason": "ResourceLogicalId:ConfigDeliveryChannel,
 ResourceType:AWS::Config::DeliveryChannel, ResourceStatusReason:Failed to put
 delivery channel 'StackSet-enable-config-e6cac20f-xmpl-46e9-8314-53e0d4591532-
ConfigDeliveryChannel-1OJWJ7XD59WR0' because the maximum number of delivery
 channels: 1 is reached. (Service: AmazonConfig; Status Code: 400; Error Code:
 MaxNumberOfDeliveryChannelsExceededException; Request ID: d14b34a0-ef7c-xmpl-
acf8-8a864370ae56)."
```

```
        }
    ]
}
```

- Per i dettagli sull'API, consulta ListStackInstances AWS CLICommand Reference.

`list-stack-refactor-actions`

Il seguente esempio di codice mostra come utilizzare`list-stack-refactor-actions`.

AWS CLI

Per elencare le azioni per un'operazione di rifattorizzazione dello stack

L'`list-stack-refactor-actions`esempio seguente elenca le azioni per l'operazione di rifattorizzazione dello stack con l'ID di rifattore dello stack specificato.

```
aws cloudformation list-stack-refactor-actions \
    --stack-refactor-id 9c384f70-4e07-4ed7-a65d-fee5eb430841
```

Output:

```
{
    "StackRefactorActions": [
        {
            "Action": "MOVE",
            "Entity": "RESOURCE",
            "PhysicalResourceId": "MyTestLambdaRole",
            "Description": "No configuration changes detected.",
            "Detection": "AUTO",
            "TagResources": [],
            "UntagResources": [],
            "ResourceMapping": {
                "Source": {
                    "StackName": "arn:aws:cloudformation:us-
east-1:123456789012:stack/Stack1/3e6a1ff0-94b1-11f0-aa6f-0a88d2e03acf",
                    "LogicalResourceId": "MyLambdaRole"
                },
                "Destination": {
                    "StackName": "arn:aws:cloudformation:us-
east-1:123456789012:stack/Stack2/5da91650-94b1-11f0-81cf-0a23500e151b",
                    "LogicalResourceId": "MyLambdaRole"
```

```
                }
            }
        },
        {
            "Action": "MOVE",
            "Entity": "RESOURCE",
            "PhysicalResourceId": "MyTestFunction",
            "Description": "Resource configuration changes will be validated during
 refactor execution.",
            "Detection": "AUTO",
            "TagResources": [
                {
                    "Key": "aws:cloudformation:stack-name",
                    "Value": "Stack2"
                },
                {
                    "Key": "aws:cloudformation:logical-id",
                    "Value": "MyFunction"
                },
                {
                    "Key": "aws:cloudformation:stack-id",
                    "Value": "arn:aws:cloudformation:us-east-1:123456789012:stack/
Stack2/5da91650-94b1-11f0-81cf-0a23500e151b"
                }
            ],
            "UntagResources": [
                "aws:cloudformation:stack-name",
                "aws:cloudformation:logical-id",
                "aws:cloudformation:stack-id"
            ],
            "ResourceMapping": {
                "Source": {
                    "StackName": "arn:aws:cloudformation:us-
east-1:123456789012:stack/Stack1/3e6a1ff0-94b1-11f0-aa6f-0a88d2e03acf",
                    "LogicalResourceId": "MyFunction"
                },
                "Destination": {
                    "StackName": "arn:aws:cloudformation:us-
east-1:123456789012:stack/Stack2/5da91650-94b1-11f0-81cf-0a23500e151b",
                    "LogicalResourceId": "MyFunction"
                }
            }
        }
    ]
```

```
}
```

Per ulteriori informazioni, consulta Stack refactoring nella Guida per l'utente.AWS CloudFormation

- Per i dettagli sull'API, consulta Command ListStackRefactorActionsReference AWS CLI .

`list-stack-resources`

Il seguente esempio di codice mostra come utilizzare`list-stack-resources`.

AWS CLI

Come elencare le risorse in uno stack

Il comando seguente visualizza l'elenco delle risorse nello stack specificato.

```
aws cloudformation list-stack-resources \
    --stack-name my-stack
```

Output:

```
{
    "StackResourceSummaries": [
        {
            "LogicalResourceId": "bucket",
            "PhysicalResourceId": "my-stack-bucket-1vc62xmplgguf",
            "ResourceType": "AWS::S3::Bucket",
            "LastUpdatedTimestamp": "2019-10-02T04:34:11.345Z",
            "ResourceStatus": "CREATE_COMPLETE",
            "DriftInformation": {
                "StackResourceDriftStatus": "IN_SYNC"
            }
        },
        {
            "LogicalResourceId": "function",
            "PhysicalResourceId": "my-function-SEZV4XMPL4S5",
            "ResourceType": "AWS::Lambda::Function",
            "LastUpdatedTimestamp": "2019-10-02T05:34:27.989Z",
            "ResourceStatus": "UPDATE_COMPLETE",
            "DriftInformation": {
                "StackResourceDriftStatus": "IN_SYNC"
            }
        },
```

```
        {
            "LogicalResourceId": "functionRole",
            "PhysicalResourceId": "my-functionRole-HIZXMPLEOM9E",
            "ResourceType": "AWS::IAM::Role",
            "LastUpdatedTimestamp": "2019-10-02T04:34:06.350Z",
            "ResourceStatus": "CREATE_COMPLETE",
            "DriftInformation": {
                "StackResourceDriftStatus": "IN_SYNC"
            }
        }
    ]
}
```

- Per i dettagli sull'API, consulta ListStackResources AWS CLICommand Reference.

`list-stack-set-operation-results`

Il seguente esempio di codice mostra come utilizzare`list-stack-set-operation-results`.

AWS CLI

Come elencare i risultati dell'operazione sul set di stack

Il comando seguente visualizza i risultati di un'operazione di aggiornamento sulle istanze del set di stack specificato.

```
aws cloudformation list-stack-set-operation-results \
    --stack-set-name enable-config \
    --operation-id 35d45ebc-ed88-xmpl-ab59-0197a1fc83a0
```

Output:

```
{
    "Summaries": [
        {
            "Account": "223456789012",
            "Region": "us-west-2",
            "Status": "SUCCEEDED",
            "AccountGateResult": {
                "Status": "SKIPPED",
                "StatusReason": "Function not found: arn:aws:lambda:eu-
west-1:223456789012:function:AWSCloudFormationStackSetAccountGate"
```

```
            }
        },
        {
            "Account": "223456789012",
            "Region": "ap-south-1",
            "Status": "CANCELLED",
            "StatusReason": "Cancelled since failure tolerance has exceeded"
        }
    ]
}
```

Nota: lo stato `SKIPPED` di `AccountGateResult` è previsto per le operazioni con esito positivo, a meno che non si crei una funzione di controllo dell'account.

- Per i dettagli sull'API, consulta ListStackSetOperationResults AWS CLICommand Reference.

`list-stack-set-operations`

Il seguente esempio di codice mostra come utilizzare`list-stack-set-operations`.

AWS CLI

Come elencare le operazioni sul set di stack

L'esempio `list-stack-set-operations` seguente visualizza l'elenco delle operazioni più recenti sul set di stack specificato.

```
aws cloudformation list-stack-set-operations \
    --stack-set-name my-stack-set
```

Output:

```
{
    "Summaries": [
        {
            "OperationId": "35d45ebc-ed88-xmpl-ab59-0197a1fc83a0",
            "Action": "UPDATE",
            "Status": "SUCCEEDED",
            "CreationTimestamp": "2019-10-03T16:28:44.377Z",
            "EndTimestamp": "2019-10-03T16:42:08.607Z"
        },
        {
            "OperationId": "891aa98f-7118-xmpl-00b2-00954d1dd0d6",
```

```
            "Action": "UPDATE",
            "Status": "FAILED",
            "CreationTimestamp": "2019-10-03T15:43:53.916Z",
            "EndTimestamp": "2019-10-03T15:45:58.925Z"
        }
    ]
}
```

- Per i dettagli sull'API, consulta ListStackSetOperations AWS CLICommand Reference.

`list-stack-sets`

Il seguente esempio di codice mostra come utilizzare`list-stack-sets`.

AWS CLI

Come elencare i set di stack

L'esempio `list-stack-sets` seguente visualizza l'elenco dei set di stack nella Regione e nell'account correnti.

```
aws cloudformation list-stack-sets
```

Output:

```
{
    "Summaries": [
        {
            "StackSetName": "enable-config",
            "StackSetId": "enable-config:296a3360-xmpl-40af-be78-9341e95bf743",
            "Description": "Enable AWS Config",
            "Status": "ACTIVE"
        }
    ]
}
```

- Per i dettagli sull'API, consulta ListStackSets AWS CLICommand Reference.

`list-stacks`

Il seguente esempio di codice mostra come utilizzare`list-stacks`.

AWS CLI

Per elencare le AWS CloudFormation pile

Il comando `list-stacks` seguente mostra un riepilogo di tutti gli stack con stato `CREATE_COMPLETE`:

```
aws cloudformation list-stacks --stack-status-filter CREATE_COMPLETE
```

Output:

```
[
    {
        "StackId": "arn:aws:cloudformation:us-east-1:123456789012:stack/
myteststack/466df9e0-0dff-08e3-8e2f-5088487c4896",
        "TemplateDescription": "AWS CloudFormation Sample Template S3_Bucket: Sample
 template showing how to create a publicly accessible S3 bucket. **WARNING** This
 template creates an S3 bucket. You will be billed for the AWS resources used if you
 create a stack from this template.",
        "StackStatusReason": null,
        "CreationTime": "2013-08-26T03:27:10.190Z",
        "StackName": "myteststack",
        "StackStatus": "CREATE_COMPLETE"
    }
]
```

- Per i dettagli sull'API, consulta ListStacks AWS CLICommand Reference.

`list-type-registrations`

Il seguente esempio di codice mostra come utilizzare`list-type-registrations`.

AWS CLI

Come elencare le registrazioni completate di un tipo

L'esempio `list-type-registrations` seguente visualizza un elenco delle registrazioni di tipo completate per il tipo specificato.

```
aws cloudformation list-type-registrations \
    --type RESOURCE \
```

```
    --type-name My::Logs::LogGroup \
    --registration-status-filter COMPLETE
```

Output:

```
{
    "RegistrationTokenList": [
        "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
        "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
        "a1b2c3d4-5678-90ab-cdef-EXAMPLE33333"
    ]
}
```

Per ulteriori informazioni, vedere Using the CloudFormation Registry in the AWS CloudFormation Users Guide.

- Per i dettagli sull'API, consulta ListTypeRegistrations AWS CLICommand Reference.

`list-type-versions`

Il seguente esempio di codice mostra come utilizzare`list-type-versions`.

AWS CLI

Come elencare la versione di un'estensione

L'esempio `list-type-versions` seguente restituisce informazioni di riepilogo sulle versioni di un'estensione.

```
aws cloudformation list-type-versions \
  --endpoint https://example.com \
  --region us-west-2 \
  --type RESOURCE \
  --type-name My::Resource::Example \
  --publisher-id 123456789012
```

Questo comando non produce alcun output.

Per ulteriori informazioni, vedere Using the AWS CloudFormation registry in the AWS CloudFormation User Guide.

- Per i dettagli sull'API, vedere ListTypeVersionsin AWS CLI Command Reference.

`list-types`

Il seguente esempio di codice mostra come utilizzare`list-types`.

AWS CLI

Come elencare i tipi di risorsa privata di un account

L'`list-types`esempio seguente visualizza un elenco dei tipi di risorse private attualmente registrati nell' AWS account corrente.

```
aws cloudformation list-types
```

Output:

```
{
    "TypeSummaries": [
        {
            "Description": "WordPress blog resource for internal use",
            "LastUpdated": "2019-12-04T18:28:15.059Z",
            "TypeName": "My::WordPress::BlogExample",
            "TypeArn": "arn:aws:cloudformation:us-west-2:123456789012:type/resource/
My-WordPress-BlogExample",
            "DefaultVersionId": "00000005",
            "Type": "RESOURCE"
        },
        {
            "Description": "Customized resource derived from AWS::Logs::LogGroup",
            "LastUpdated": "2019-12-04T18:28:15.059Z",
            "TypeName": "My::Logs::LogGroup",
            "TypeArn": "arn:aws:cloudformation:us-west-2:123456789012:type/resource/
My-Logs-LogGroup",
            "DefaultVersionId": "00000003",
            "Type": "RESOURCE"
        }
    ]
}
```

Per ulteriori informazioni, vedere Using the CloudFormation Registry nella AWS CloudFormation Users Guide.

- Per i dettagli sull'API, consulta ListTypes AWS CLICommand Reference.

`package`

Il seguente esempio di codice mostra come utilizzare`package`.

AWS CLI

Il comando seguente esporta un modello denominato `template.json` caricando gli artefatti locali nel bucket S3 `bucket-name` e scrive il modello esportato in `packaged-template.json`.

```
aws cloudformation package --template-file /path_to_template/template.json --s3-
bucket bucket-name --output-template-file packaged-template.json --use-json
```

- Per informazioni dettagliate sull'API, consulta Package in AWS CLI Command Reference.

`publish-type`

Il seguente esempio di codice mostra come utilizzare`publish-type`.

AWS CLI

Come pubblicare un'estensione

L'`publish-type`esempio seguente pubblica l'estensione specificata nel CloudFormation registro come estensione pubblica in questa regione.

```
aws cloudformation publish-type \
  --region us-west-2 \
  --type RESOURCE \
  --type-name Example::Test::1234567890abcdef0
```

Output:

```
{
    "PublicTypeArn":"arn:aws:cloudformation:us-west-2::type/
resource/000q6TfUovXsEMmgKowxDZLlwqr2QUshd2e75c8c/Example-
Test-1234567890abcdef0/1.0.0"
}
```

Per ulteriori informazioni, vedere Using the AWS CloudFormation registry in the AWS CloudFormation User Guide.

- Per i dettagli sull'API, vedere PublishTypein AWS CLI Command Reference.

`register-publisher`

Il seguente esempio di codice mostra come utilizzare`register-publisher`.

AWS CLI

Come registrare un publisher

L'esempio `register-publisher` seguente registra un publisher e accetta il parametro dei termini e delle condizioni.

```
aws cloudformation register-publisher \
  --region us-west-2 \
  --accept-terms-and-conditions
```

Output:

```
{
    "PublisherId": "000q6TfUovXsEMmgKowxDZLlwqr2QUshd2e75c8c"
}
```

Per ulteriori informazioni, vedere Using the AWS CloudFormation registry in the AWS CloudFormation User Guide.

- Per i dettagli sull'API, vedere RegisterPublisherin AWS CLI Command Reference.

`register-type`

Il seguente esempio di codice mostra come utilizzare`register-type`.

AWS CLI

Come registrare un tipo di risorsa

L'esempio `register-type` seguente registra il tipo di risorsa specificato come tipo di risorsa privata nell'account dell'utente.

```
aws cloudformation register-type \
```

```
    --type-name My::Organization::ResourceName \
    --schema-handler-package s3://bucket_name/my-organization-resource_name.zip \
    --type RESOURCE
```

Output:

```
{
    "RegistrationToken": "f5525280-104e-4d35-bef5-8f1f1example"
}
```

Per ulteriori informazioni, vedere Registrazione dei provider di risorse nella Guida utente dell'interfaccia a riga di CloudFormation comando per lo sviluppo dei tipi.

- Per i dettagli sull'API, consulta RegisterType AWS CLICommand Reference.

`set-stack-policy`

Il seguente esempio di codice mostra come utilizzare`set-stack-policy`.

AWS CLI

Come applicare una policy di stack

L'esempio `set-stack-policy` seguente disabilita gli aggiornamenti per la risorsa specificata nello stack specificato. `stack-policy.json` è un documento JSON che definisce le operazioni consentite sulle risorse nello stack.

```
aws cloudformation set-stack-policy \
    --stack-name my-stack \
    --stack-policy-body file://stack-policy.json
```

Output:

```
{
  "Statement" : [
    {
      "Effect" : "Allow",
      "Action" : "Update:*",
      "Principal": "*",
      "Resource" : "*"
```

```
    },
    {
      "Effect" : "Deny",
      "Action" : "Update:*",
      "Principal": "*",
      "Resource" : "LogicalResourceId/bucket"
    }
  ]
}
```

- Per i dettagli sull'API, consulta SetStackPolicy AWS CLICommand Reference.

`set-type-configuration`

Il seguente esempio di codice mostra come utilizzare`set-type-configuration`.

AWS CLI

Come configurare i dati

L'`set-type-configuration`esempio seguente specifica i dati di configurazione per un' CloudFormation estensione registrata, nell'account e nella regione specificati.

```
aws cloudformation set-type-configuration \
    --region us-west-2 \
    --type RESOURCE \
    --type-name Example::Test::Type \
    --configuration-alias default \
    --configuration "{\"CredentialKey\": \"testUserCredential\"}"
```

Output:

```
{
    "ConfigurationArn": "arn:aws:cloudformation:us-west-2:123456789012:type-
configuration/resource/Example-Test-Type/default"
}
```

Per ulteriori informazioni, vedere Uso del AWS CloudFormation registro nella Guida per l'AWS CloudFormation utente.

- Per i dettagli sull'API, vedere SetTypeConfigurationin AWS CLI Command Reference.

`set-type-default-version`

Il seguente esempio di codice mostra come utilizzare`set-type-default-version`.

AWS CLI

Come impostare una versione predefinita del tipo

L'esempio `set-type-default-version` seguente imposta la versione del tipo specificata da utilizzare come predefinita per questo tipo.

```
aws cloudformation set-type-default-version \
    --type RESOURCE \
    --type-name My::Logs::LogGroup \
    --version-id 00000003
```

Questo comando non produce alcun output.

Per ulteriori informazioni, vedere Using the CloudFormation Registry in the AWS CloudFormation Users Guide.

- Per i dettagli sull'API, consulta SetTypeDefaultVersion AWS CLICommand Reference.

`signal-resource`

Il seguente esempio di codice mostra come utilizzare`signal-resource`.

AWS CLI

Come segnalare una risorsa

L'esempio `signal-resource` seguente segnala `success` per soddisfare la condizione di attesa denominata `MyWaitCondition` nello stack denominato `my-stack`.

```
aws cloudformation signal-resource \
    --stack-name my-stack \
    --logical-resource-id MyWaitCondition \
    --unique-id 1234 \
    --status SUCCESS
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta SignalResource AWS CLICommand Reference.

`start-resource-scan`

Il seguente esempio di codice mostra come utilizzare`start-resource-scan`.

AWS CLI

Per avviare una scansione delle risorse

L'`start-resource-scan`esempio seguente avvia una scansione delle risorse che analizza tutte le risorse esistenti nell'account e nella regione correnti.

```
aws cloudformation start-resource-scan
```

Output:

```
{
    "ResourceScanId":
      "arn:aws:cloudformation:us-
east-1:123456789012:resourceScan/0a699f15-489c-43ca-a3ef-3e6ecfa5da60"
}
```

Per ulteriori informazioni, vedere Avvio di una scansione delle risorse con il generatore CloudFormation IAc nella Guida per l'AWS CloudFormation utente.

- Per i dettagli sull'API, consulta StartResourceScan AWS CLICommand Reference.

`stop-stack-set-operation`

Il seguente esempio di codice mostra come utilizzare`stop-stack-set-operation`.

AWS CLI

Come arrestare un'operazione del set di stack

L'esempio `stop-stack-set-operation` seguente arresta un'operazione di aggiornamento in corso sul set di stack specificato.

```
aws cloudformation stop-stack-set-operation \
    --stack-set-name my-stack-set \
    --operation-id 1261cd27-490b-xmpl-ab42-793a896c69e6
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta StopStackSetOperation AWS CLICommand Reference.

`test-type`

Il seguente esempio di codice mostra come utilizzare`test-type`.

AWS CLI

Come testare un'estensione

L'`test-type`esempio seguente verifica un'estensione registrata per assicurarsi che soddisfi tutti i requisiti necessari per la pubblicazione nel CloudFormation registro.

```
aws cloudformation test-type \
    --arn arn:aws:cloudformation:us-west-2:123456789012:type/resource/Sample-Test-Resource123/00000001
```

Output:

```
{
    "TypeVersionArn": "arn:aws:cloudformation:us-west-2:123456789012:type/resource/Sample-Test-Resource123/00000001"
}
```

Per ulteriori informazioni, vedere Uso del AWS CloudFormation registro nella Guida per l'AWS CloudFormation utente.

- Per i dettagli sull'API, vedere TestTypein AWS CLI Command Reference.

`update-stack-instances`

Il seguente esempio di codice mostra come utilizzare`update-stack-instances`.

AWS CLI

Come aggiornare le istanze di stack

L'esempio `update-stack-instances` seguente riprova a eseguire un aggiornamento sulle istanze di stack in due account presenti in due Regioni con le impostazioni più recenti.

L'impostazione della tolleranza ai guasti specificata garantisce che l'aggiornamento venga tentato in tutti gli account e le Regioni, anche se non è possibile aggiornare alcuni stack.

```
aws cloudformation update-stack-instances \
    --stack-set-name my-stack-set \
    --accounts 123456789012 567890123456 \
    --regions us-east-1 us-west-2 \
    --operation-preferences FailureToleranceCount=3
```

Output:

```
{
    "OperationId": "103ebdf2-21ea-xmpl-8892-de5e30733132"
}
```

- Per i dettagli sull'API, consulta UpdateStackInstances AWS CLICommand Reference.

`update-stack-set`

Il seguente esempio di codice mostra come utilizzare`update-stack-set`.

AWS CLI

Come aggiornare un set di stack

L'esempio `update-stack-set` seguente aggiunge un tag con il nome della chiave `Owner` e il valore `IT` alle istanze di stack nel set di stack specificato.

```
aws cloudformation update-stack-set \
    --stack-set-name my-stack-set \
    --use-previous-template \
    --tags Key=Owner,Value=IT
```

Output:

```
{
    "OperationId": "e2b60321-6cab-xmpl-bde7-530c6f47950e"
}
```

- Per i dettagli sull'API, consulta UpdateStackSet AWS CLICommand Reference.

update-stack

Il seguente esempio di codice mostra come utilizzareupdate-stack.

AWS CLI

Per aggiornare gli AWS CloudFormation stack

Il comando update-stack seguente aggiorna il modello e i parametri di input per lo stack mystack:

```
aws cloudformation update-stack --stack-name mystack --
template-url https://s3.amazonaws.com/sample/updated.template --
parameters ParameterKey=KeyPairName,ParameterValue=SampleKeyPair
 ParameterKey=SubnetIDs,ParameterValue=SampleSubnetID1\\,SampleSubnetID2
```

Il comando update-stack seguente aggiorna solo il valore del parametro SubnetIDs per lo stack mystack: Se non specifichi un valore di parametro, viene utilizzato il valore predefinito specificato nel modello:

```
aws cloudformation update-stack --stack-name mystack --
template-url https://s3.amazonaws.com/sample/updated.template
 --parameters ParameterKey=KeyPairName,UsePreviousValue=true
 ParameterKey=SubnetIDs,ParameterValue=SampleSubnetID1\\,UpdatedSampleSubnetID2
```

Il comando update-stack seguente aggiunge due argomenti di notifica per stack allo stack mystack:

```
aws cloudformation update-stack --stack-name mystack --use-previous-template --
notification-arns "arn:aws:sns:use-east-1:123456789012:mytopic1" "arn:aws:sns:us-
east-1:123456789012:mytopic2"
```

Per ulteriori informazioni, consulta gli aggiornamenti AWS CloudFormation dello stack nella Guida per l'AWS CloudFormation utente.

- Per i dettagli sull'API, consulta UpdateStack AWS CLICommand Reference.

update-termination-protection

Il seguente esempio di codice mostra come utilizzareupdate-termination-protection.

AWS CLI

Come abilitare la protezione da terminazione

L'esempio `update-termination-protection` seguente abilita la protezione da terminazione nello stack specificato.

```
aws cloudformation update-termination-protection \
    --stack-name my-stack \
    --enable-termination-protection
```

Output:

```
{
    "StackId": "arn:aws:cloudformation:us-west-2:123456789012:stack/my-stack/
d0a825a0-e4cd-xmpl-b9fb-061c69e99204"
}
```

- Per i dettagli sull'API, consulta UpdateTerminationProtection AWS CLICommand Reference.

`validate-template`

Il seguente esempio di codice mostra come utilizzare`validate-template`.

AWS CLI

Per convalidare un modello AWS CloudFormation

Il comando `validate-template` seguente convalida il modello `sampletemplate.json`.

```
aws cloudformation validate-template --template-body file://sampletemplate.json
```

Output:

```
{
    "Description": "AWS CloudFormation Sample Template S3_Bucket: Sample template
 showing how to create a publicly accessible S3 bucket. **WARNING** This template
 creates an S3 bucket. You will be billed for the AWS resources used if you create a
 stack from this template.",
    "Parameters": [],
```

```
    "Capabilities": []
}
```

Per ulteriori informazioni, consulta Lavorare con i AWS CloudFormation modelli nella Guida per l'AWS CloudFormation utente.

- Per i dettagli sull'API, consulta ValidateTemplate AWS CLICommand Reference.

CloudFront esempi utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with CloudFront.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`associate-distribution-tenant-web-acl`

Il seguente esempio di codice mostra come utilizzare`associate-distribution-tenant-web-acl`.

AWS CLI

Per associare un ACL Web a un tenant di CloudFront distribuzione

L'`associate-distribution-tenant-web-acl`esempio seguente associa un ACL Web a una distribuzione con. CloudFront ETag `E13V1IB3VIYABC`

```
aws cloudfront associate-distribution-tenant-web-acl \
    --id dt_2wjDZi3hD1ivOXf6rpZJO1AB \
    --if-match E13V1IB3VIYABC \
```

```
    --web-acl-arn arn:aws:wafv2:us-east-1:123456789012:global/webacl/web-global-example/626900da-5f64-418b-ba9b-743f37123ABC
```

Output:

```
{
    "ETag": "E1VC38T7YXBABC",
    "Id": "dt_2wjDZi3hD1ivOXf6rpZJO1AB",
    "WebACLArn": "arn:aws:wafv2:us-east-1:123456789012:global/webacl/web-global-example/626900da-5f64-418b-ba9b-743f37123ABC"
}
```

Per ulteriori informazioni, consulta Usare le protezioni AWS WAF nella Amazon CloudFront Developer Guide.

- Per i dettagli sull'API, consulta AWS CLI Command AssociateDistributionTenantWebAclReference.

`associate-distribution-web-acl`

Il seguente esempio di codice mostra come utilizzare`associate-distribution-web-acl`.

AWS CLI

Per associare un ACL Web a una distribuzione CloudFront

L'`associate-distribution-web-acl`esempio seguente associa un ACL Web a una distribuzione. CloudFront

```
aws cloudfront associate-distribution-web-acl \
    --id E1XNX8R2GOAABC \
    --if-match E2YWS1C2J3OABC \
    --web-acl-arn arn:aws:wafv2:us-east-1:123456789012:global/webacl/web-global-example/626900da-5f64-418b-ba9b-743f3746cABC
```

Output:

```
{
    "ETag": "E3QE7ED60U0ABC",
    "Id": "E1XNX8R2GOAABC",
    "WebACLArn": "arn:aws:wafv2:us-east-1:123456789012:global/webacl/web-global-example/626900da-5f64-418b-ba9b-743f3746cABC"
```

```
}
```

Per ulteriori informazioni, consulta Usare le protezioni AWS WAF nella Amazon CloudFront Developer Guide.

- Per i dettagli sull'API, consulta AWS CLI Command AssociateDistributionWebAclReference.

`create-cloud-front-origin-access-identity`

Il seguente esempio di codice mostra come utilizzare`create-cloud-front-origin-access-identity`.

AWS CLI

Per creare un'identità di accesso all' CloudFront origine

L'esempio seguente crea un'identità di accesso all' CloudFront origine (OAI) fornendo la configurazione OAI come argomento della riga di comando:

```
aws cloudfront create-cloud-front-origin-access-identity \
    --cloud-front-origin-access-identity-config \
        CallerReference="cli-example",Comment="Example OAI"
```

È possibile ottenere lo stesso risultato indicando la configurazione OAI in un file JSON, come mostrato nell'esempio seguente:

```
aws cloudfront create-cloud-front-origin-access-identity \
    --cloud-front-origin-access-identity-config file://OAI-config.json
```

Il file `OAI-config.json` è un documento JSON nella directory corrente che contiene quanto segue:

```
{
    "CallerReference": "cli-example",
    "Comment": "Example OAI"
}
```

Sia che si fornisca la configurazione OAI con un argomento della riga di comando o con un file JSON, l'output è lo stesso:

```
{
```

```
    "Location": "https://cloudfront.amazonaws.com/2019-03-26/origin-access-identity/
cloudfront/E74FTE3AEXAMPLE",
    "ETag": "E2QWRUHEXAMPLE",
    "CloudFrontOriginAccessIdentity": {
        "Id": "E74FTE3AEXAMPLE",
        "S3CanonicalUserId":
 "cd13868f797c227fbea2830611a26fe0a21ba1b826ab4bed9b7771c9aEXAMPLE",
        "CloudFrontOriginAccessIdentityConfig": {
            "CallerReference": "cli-example",
            "Comment": "Example OAI"
        }
    }
}
```

- Per i dettagli sull'API, consulta AWS CLI Command CreateCloudFrontOriginAccessIdentityReference.

`create-connection-group`

Il seguente esempio di codice mostra come utilizzare`create-connection-group`.

AWS CLI

Per creare un gruppo di connessione in CloudFront

L'`create-connection-group`esempio seguente crea un gruppo di connessioni abilitato, specifica un elenco di IP statici Anycast e lo disabilita. IPv6

```
aws cloudfront create-connection-group \
    --name cg-with-anycast-ip-list \
    --no-ipv6-enabled \
    --enabled \
    --anycast-ip-list-id aip_CCkW6gKrDiBD4n78123ABC \
    --tags "Items=[{Key=abc,Value=123}]"
```

Output:

```
{
    "ETag": "E23ZP02F085ABC",
    "ConnectionGroup": {
        "Id": "cg_2yb6uj74B4PCbfhT31WFdiSABC",
        "Name": "cg-with-anycast-ip-list",
```

```
        "Arn": "arn:aws:cloudfront::123456789012:connection-group/
cg_2yb6uj74B4PCbfhT31WFdiSABC",
        "CreatedTime": "2025-06-16T16:25:50.061000+00:00",
        "LastModifiedTime": "2025-06-16T16:25:50.061000+00:00",
        "Tags": {
            "Items": [
                {
                    "Key": "abc",
                    "Value": "123"
                }
            ]
        },
        "Ipv6Enabled": false,
        "RoutingEndpoint": "dj6xusxq65abc.cloudfront.net",
        "AnycastIpListId": "aip_CCkW6gKrDiBD4n78123ABC",
        "Status": "InProgress",
        "Enabled": true,
        "IsDefault": false
    }
}
```

Per ulteriori informazioni, consulta Creare un gruppo di connessione personalizzato (opzionale) nella Amazon CloudFront Developer Guide.

- Per i dettagli sull'API, consulta CreateConnectionGroup AWS CLICommand Reference.

`create-distribution-tenant`

Il seguente esempio di codice mostra come utilizzare`create-distribution-tenant`.

AWS CLI

Esempio 1: creare un tenant di CloudFront distribuzione che utilizza un certificato personalizzato

L'`create-distribution-tenant`esempio seguente crea un tenant di CloudFront distribuzione che specifica le personalizzazioni per disabilitare WAF, aggiungere restrizioni geografiche e utilizza un altro certificato TLS.

```
aws cloudfront create-distribution-tenant \
    --cli-input-json file://tenant.json
```

Contenuto di `tenant.json`:

```
{
    "DistributionId": "E1XNX8R2GOAABC",
    "Domains": [
        {
            "Domain": "example.com"
        }
    ],
    "Parameters": [
        {
            "Name": "testParam",
            "Value": "defaultValue"
        }
    ],
    "ConnectionGroupId": "cg_2whCJoXMYCjHcxaLGrkllvyABC",
    "Enabled": false,
    "Tags": {
        "Items": [
            {
                "Key": "tag",
                "Value": "tagValue"
            }
        ]
    },
    "Name": "new-tenant-customizations",
    "Customizations": {
        "GeoRestrictions": {
            "Locations": ["DE"],
            "RestrictionType": "whitelist"
        },
        "WebAcl": {
            "Action": "disable"
        },
        "Certificate": {
            "Arn": "arn:aws:acm:us-east-1:123456789012:certificate/ec53f564-
ea5a-4e4a-a0a2-e3c989449abc"
        }
    }
}
```

Output:

```
{
    "ETag": "E23ZP02F085ABC",
```

```
    "DistributionTenant": {
        "Id": "dt_2yN5tYwVbPKr7m2IB69M1yp1AB",
        "DistributionId": "E1XNX8R2GOAABC",
        "Name": "new-tenant-customizations",
        "Arn": "arn:aws:cloudfront::123456789012:distribution-tenant/
dt_2yN5tYwVbPKr7m2IB69M1yp1AB",
        "Domains": [
            {
                "Domain": "example.com",
                "Status": "active"
            }
        ],
        "Tags": {
            "Items": [
                {
                    "Key": "tag",
                    "Value": "tagValue"
                }
            ]
        },
        "Customizations": {
            "WebAcl": {
                "Action": "disable"
            },
            "Certificate": {
                "Arn": "arn:aws:acm:us-east-1:123456789012:certificate/ec53f564-
ea5a-4e4a-a0a2-e3c989449abc"
            },
            "GeoRestrictions": {
                "RestrictionType": "whitelist",
                "Locations": [
                    "DE"
                ]
            }
        },
        "Parameters": [
            {
                "Name": "testParam",
                "Value": "defaultValue"
            }
        ],
        "ConnectionGroupId": "cg_2whCJoXMYCjHcxaLGrkllvyABC",
        "CreatedTime": "2025-06-11T17:20:06.432000+00:00",
        "LastModifiedTime": "2025-06-11T17:20:06.432000+00:00",
```

```
        "Enabled": false,
        "Status": "InProgress"
    }
}
```

Esempio 2: come creare un tenant di distribuzione con un certificato ereditato

L'esempio `create-distribution-tenant` seguente crea un tenant di distribuzione e specifica un certificato TLS ereditato dalla distribuzione multi-tenant.

```
aws cloudfront create-distribution-tenant \
    --cli-input-json file://tenant.json
```

Contenuto di `tenant.json`:

```
{
    "DistributionId": "E1HVIAU7U12ABC",
    "Domains": [
        {
            "Domain": "example.com"
        }
    ],
    "Parameters": [
        {
            "Name": "tenantName",
            "Value": "first-tenant"
        }
    ],
    "Enabled": true,
    "Name": "new-tenant-no-cert"
}
```

Output:

```
{
    "ETag": "E23ZP02F0ABC",
    "DistributionTenant": {
        "Id": "dt_2zhRB0vBe0B72LZCVy1mgzI1AB",
        "DistributionId": "E1HVIAU7U12ABC",
        "Name": "new-tenant-no-cert",
        "Arn": "arn:aws:cloudfront::123456789012:distribution-tenant/
dt_2zhRB0vBe0B72LZCVy1mgzI1AB",
```

```
        "Domains": [
            {
                "Domain": "example.com",
                "Status": "active"
            }
        ],
        "Parameters": [
            {
                "Name": "tenantName",
                "Value": "first-tenant"
            }
        ],
        "ConnectionGroupId": "cg_2yQEwpipGFN0hhA0ZemPabOABC",
        "CreatedTime": "2025-07-10T20:59:38.414000+00:00",
        "LastModifiedTime": "2025-07-10T20:59:38.414000+00:00",
        "Enabled": true,
        "Status": "InProgress"
    }
}
```

Esempio 3: creare un tenant di distribuzione utilizzando un CloudFront token di convalida -hosted CloudFront

L'`create-distribution-tenant`esempio seguente crea un tenant di distribuzione e utilizza un token di convalida CloudFront -hosted per il nome di dominio.

```
aws cloudfront create-distribution-tenant \
    --cli-input-json file://tenant.json
```

Contenuto di `tenant.json`:

```
{
    "DistributionId": "E2GJ5J9QN12ABC",
    "Domains": [
        {
            "Domain": "example.com"
        }
    ],
    "Parameters": [
        {
            "Name": "tenantName",
            "Value": "first-tenant"
```

```
        }
    ],
    "ConnectionGroupId": "cg_2yQEwpipGFN0hhA0ZemPabOABC",
    "Enabled": true,
    "Name": "new-tenant-cf-hosted",
    "ManagedCertificateRequest": {
        "ValidationTokenHost": "cloudfront"
    }
}
```

Importante: per eseguire correttamente questo comando, è necessario configurare un record DNS CNAME che indirizzi il nuovo dominio (example.com) all'endpoint di routing del gruppo di connessione associato al tenant di distribuzione. Questo record CNAME deve inoltre essere propagato prima CloudFront di poter completare con successo questa richiesta.

Output:

```
{
    "ETag": "E23ZP02F0ABC",
    "DistributionTenant": {
        "Id": "dt_2zhStKrA524GvvTWJX92Ozl1AB",
        "DistributionId": "E2GJ5J9QN12ABC",
        "Name": "new-tenant-cf-hosted",
        "Arn": "arn:aws:cloudfront::123456789012:distribution-tenant/
dt_2zhStKrA524GvvTWJX92Ozl1AB",
        "Domains": [
            {
                "Domain": "example.com",
                "Status": "inactive"
            }
        ],
        "Parameters": [
            {
                "Name": "tenantName",
                "Value": "first-tenant"
            }
        ],
        "ConnectionGroupId": "cg_2zhSaGatwwXjTjE42nneZzqABC",
        "CreatedTime": "2025-07-10T21:13:46.416000+00:00",
        "LastModifiedTime": "2025-07-10T21:13:46.416000+00:00",
        "Enabled": true,
        "Status": "InProgress"
    }
```

```
}
```

Esempio 4: creare un tenant di CloudFront distribuzione utilizzando un token di convalida ospitato autonomamente

L'`create-distribution-tenant`esempio seguente crea un tenant di CloudFront distribuzione e utilizza un token di convalida ospitato autonomamente.

```
aws cloudfront create-distribution-tenant \
    --cli-input-json file://tenant.json
```

Contenuto di `tenant.json`:

```
{
    "DistributionId": "E2GJ5J9QN12ABC",
    "Domains": [
        {
            "Domain": "example.com"
        }
    ],
    "Parameters": [
        {
            "Name": "tenantName",
            "Value": "first-tenant"
        }
    ],
    "Enabled": true,
    "Name": "new-tenant-self-hosted",
    "ManagedCertificateRequest": {
        "ValidationTokenHost": "self-hosted"
    }
}
```

Output:

```
{
    "ETag": "E23ZP02F0ABC",
    "DistributionTenant": {
        "Id": "dt_2zhTFBV93OfFJJ3YMdNM5BC1AB",
        "DistributionId": "E2GJ5J9QN12ABC",
        "Name": "new-tenant-self-hosted",
```

```
        "Arn": "arn:aws:cloudfront::123456789012:distribution-tenant/
dt_2zhTFBV93OfFJJ3YMdNM5BC1AB",
        "Domains": [
            {
                "Domain": "example.com",
                "Status": "inactive"
            }
        ],
        "Parameters": [
            {
                "Name": "tenantName",
                "Value": "first-tenant"
            }
        ],
        "ConnectionGroupId": "cg_2yQEwpipGFN0hhA0ZemPabOABC",
        "CreatedTime": "2025-07-10T21:16:39.828000+00:00",
        "LastModifiedTime": "2025-07-10T21:16:39.828000+00:00",
        "Enabled": true,
        "Status": "InProgress"
    }
}
```

Importante: dopo l'esecuzione di questo comando, il tenant di distribuzione viene creato senza convalida. Per convalidare la richiesta di certificato gestito e configurare il DNS quando sei pronto per iniziare a ricevere traffico, consulta la sezione Configurazione completa del dominio nella Amazon CloudFront Developer Guide.

Per ulteriori informazioni sulla creazione di tenant di distribuzione, consulta Create a distribution nell'Amazon CloudFront Developer Guide.

- Per i dettagli sull'API, consulta CreateDistributionTenant AWS CLICommand Reference.

`create-distribution-with-tags`

Il seguente esempio di codice mostra come utilizzare`create-distribution-with-tags`.

AWS CLI

Per creare una CloudFront distribuzione con tag

L'esempio `create-distribution-with-tags` seguente crea una distribuzione con due tag fornendo la configurazione di distribuzione e i tag in un file JSON denominato `dist-config-with-tags.json`.

```
aws cloudfront create-distribution-with-tags \
    --distribution-config-with-tags file://dist-config-with-tags.json
```

Il file `dist-config-with-tags.json` è un documento JSON nella cartella corrente. Nota l'oggetto `Tags` nella parte superiore del file, che contiene due tag:

`Name = ExampleDistributionProject = ExampleProject`

Contenuto di `dist-config-with-tags.json`:

```
{
    "Tags": {
        "Items": [
            {
                "Key": "Name",
                "Value": "ExampleDistribution"
            },
            {
                "Key": "Project",
                "Value": "ExampleProject"
            }
        ]
    },
    "DistributionConfig": {
        "CallerReference": "cli-example",
        "Aliases": {
            "Quantity": 0
        },
        "DefaultRootObject": "index.html",
        "Origins": {
            "Quantity": 1,
            "Items": [
                {
                    "Id": "amzn-s3-demo-bucket.s3.amazonaws.com-cli-example",
                    "DomainName": "amzn-s3-demo-bucket.s3.amazonaws.com",
                    "OriginPath": "",
                    "CustomHeaders": {
                        "Quantity": 0
                    },
                    "S3OriginConfig": {
                        "OriginAccessIdentity": ""
                    }
                }
```

```
            ]
        },
        "OriginGroups": {
            "Quantity": 0
        },
        "DefaultCacheBehavior": {
            "TargetOriginId": "amzn-s3-demo-bucket.s3.amazonaws.com-cli-example",
            "ForwardedValues": {
                "QueryString": false,
                "Cookies": {
                    "Forward": "none"
                },
                "Headers": {
                    "Quantity": 0
                },
                "QueryStringCacheKeys": {
                    "Quantity": 0
                }
            },
            "TrustedSigners": {
                "Enabled": false,
                "Quantity": 0
            },
            "ViewerProtocolPolicy": "allow-all",
            "MinTTL": 0,
            "AllowedMethods": {
                "Quantity": 2,
                "Items": [
                    "HEAD",
                    "GET"
                ],
                "CachedMethods": {
                    "Quantity": 2,
                    "Items": [
                        "HEAD",
                        "GET"
                    ]
                }
            },
            "SmoothStreaming": false,
            "DefaultTTL": 86400,
            "MaxTTL": 31536000,
            "Compress": false,
            "LambdaFunctionAssociations": {
```

```
                "Quantity": 0
            },
            "FieldLevelEncryptionId": ""
        },
        "CacheBehaviors": {
            "Quantity": 0
        },
        "CustomErrorResponses": {
            "Quantity": 0
        },
        "Comment": "",
        "Logging": {
            "Enabled": false,
            "IncludeCookies": false,
            "Bucket": "",
            "Prefix": ""
        },
        "PriceClass": "PriceClass_All",
        "Enabled": true,
        "ViewerCertificate": {
            "CloudFrontDefaultCertificate": true,
            "MinimumProtocolVersion": "TLSv1",
            "CertificateSource": "cloudfront"
        },
        "Restrictions": {
            "GeoRestriction": {
                "RestrictionType": "none",
                "Quantity": 0
            }
        },
        "WebACLId": "",
        "HttpVersion": "http2",
        "IsIPV6Enabled": true
    }
}
```

Output:

```
{
    "Location": "https://cloudfront.amazonaws.com/2019-03-26/distribution/
EDFDVBD6EXAMPLE",
    "ETag": "E2QWRUHEXAMPLE",
    "Distribution": {
```

```
        "Id": "EDFDVBD6EXAMPLE",
        "ARN": "arn:aws:cloudfront::123456789012:distribution/EDFDVBD6EXAMPLE",
        "Status": "InProgress",
        "LastModifiedTime": "2019-12-04T23:35:41.433Z",
        "InProgressInvalidationBatches": 0,
        "DomainName": "d111111abcdef8.cloudfront.net",
        "ActiveTrustedSigners": {
            "Enabled": false,
            "Quantity": 0
        },
        "DistributionConfig": {
            "CallerReference": "cli-example",
            "Aliases": {
                "Quantity": 0
            },
            "DefaultRootObject": "index.html",
            "Origins": {
                "Quantity": 1,
                "Items": [
                    {
                        "Id": "amzn-s3-demo-bucket.s3.amazonaws.com-cli-example",
                        "DomainName": "amzn-s3-demo-bucket.s3.amazonaws.com",
                        "OriginPath": "",
                        "CustomHeaders": {
                            "Quantity": 0
                        },
                        "S3OriginConfig": {
                            "OriginAccessIdentity": ""
                        }
                    }
                ]
            },
            "OriginGroups": {
                "Quantity": 0
            },
            "DefaultCacheBehavior": {
                "TargetOriginId": "amzn-s3-demo-bucket.s3.amazonaws.com-cli-
example",
                "ForwardedValues": {
                    "QueryString": false,
                    "Cookies": {
                        "Forward": "none"
                    },
                    "Headers": {
```

```
                    "Quantity": 0
                },
                "QueryStringCacheKeys": {
                    "Quantity": 0
                }
            },
            "TrustedSigners": {
                "Enabled": false,
                "Quantity": 0
            },
            "ViewerProtocolPolicy": "allow-all",
            "MinTTL": 0,
            "AllowedMethods": {
                "Quantity": 2,
                "Items": [
                    "HEAD",
                    "GET"
                ],
                "CachedMethods": {
                    "Quantity": 2,
                    "Items": [
                        "HEAD",
                        "GET"
                    ]
                }
            },
            "SmoothStreaming": false,
            "DefaultTTL": 86400,
            "MaxTTL": 31536000,
            "Compress": false,
            "LambdaFunctionAssociations": {
                "Quantity": 0
            },
            "FieldLevelEncryptionId": ""
        },
        "CacheBehaviors": {
            "Quantity": 0
        },
        "CustomErrorResponses": {
            "Quantity": 0
        },
        "Comment": "",
        "Logging": {
            "Enabled": false,
```

```
                "IncludeCookies": false,
                "Bucket": "",
                "Prefix": ""
            },
            "PriceClass": "PriceClass_All",
            "Enabled": true,
            "ViewerCertificate": {
                "CloudFrontDefaultCertificate": true,
                "MinimumProtocolVersion": "TLSv1",
                "CertificateSource": "cloudfront"
            },
            "Restrictions": {
                "GeoRestriction": {
                    "RestrictionType": "none",
                    "Quantity": 0
                }
            },
            "WebACLId": "",
            "HttpVersion": "http2",
            "IsIPV6Enabled": true
        }
    }
}
```

- Per i dettagli sull'API, consulta CreateDistributionWithTags AWS CLICommand Reference.

`create-distribution`

Il seguente esempio di codice mostra come utilizzare`create-distribution`.

AWS CLI

Esempio 1: creare una CloudFront distribuzione

L'esempio `create-distribution` seguente crea una distribuzione per un bucket S3 denominato `amzn-s3-demo-bucket` e specifica inoltre `index.html` come oggetto root predefinito utilizzando argomenti della riga di comando.

```
aws cloudfront create-distribution \
    --origin-domain-name amzn-s3-demo-bucket.s3.amazonaws.com \
    --default-root-object index.html
```

Output:

```
{
    "Location": "https://cloudfront.amazonaws.com/2019-03-26/distribution/
EMLARXS9EXAMPLE",
    "ETag": "E9LHASXEXAMPLE",
    "Distribution": {
        "Id": "EMLARXS9EXAMPLE",
        "ARN": "arn:aws:cloudfront::123456789012:distribution/EMLARXS9EXAMPLE",
        "Status": "InProgress",
        "LastModifiedTime": "2019-11-22T00:55:15.705Z",
        "InProgressInvalidationBatches": 0,
        "DomainName": "d111111abcdef8.cloudfront.net",
        "ActiveTrustedSigners": {
            "Enabled": false,
            "Quantity": 0
        },
        "DistributionConfig": {
            "CallerReference": "cli-example",
            "Aliases": {
                "Quantity": 0
            },
            "DefaultRootObject": "index.html",
            "Origins": {
                "Quantity": 1,
                "Items": [
                    {
                        "Id": "amzn-s3-demo-bucket.s3.amazonaws.com-cli-example",
                        "DomainName": "amzn-s3-demo-bucket.s3.amazonaws.com",
                        "OriginPath": "",
                        "CustomHeaders": {
                            "Quantity": 0
                        },
                        "S3OriginConfig": {
                            "OriginAccessIdentity": ""
                        }
                    }
                ]
            },
            "OriginGroups": {
                "Quantity": 0
            },
            "DefaultCacheBehavior": {
```

```
                "TargetOriginId": "amzn-s3-demo-bucket.s3.amazonaws.com-cli-
example",
                "ForwardedValues": {
                    "QueryString": false,
                    "Cookies": {
                        "Forward": "none"
                    },
                    "Headers": {
                        "Quantity": 0
                    },
                    "QueryStringCacheKeys": {
                        "Quantity": 0
                    }
                },
                "TrustedSigners": {
                    "Enabled": false,
                    "Quantity": 0
                },
                "ViewerProtocolPolicy": "allow-all",
                "MinTTL": 0,
                "AllowedMethods": {
                    "Quantity": 2,
                    "Items": [
                        "HEAD",
                        "GET"
                    ],
                    "CachedMethods": {
                        "Quantity": 2,
                        "Items": [
                            "HEAD",
                            "GET"
                        ]
                    }
                },
                "SmoothStreaming": false,
                "DefaultTTL": 86400,
                "MaxTTL": 31536000,
                "Compress": false,
                "LambdaFunctionAssociations": {
                    "Quantity": 0
                },
                "FieldLevelEncryptionId": ""
            },
            "CacheBehaviors": {
```

```
                "Quantity": 0
            },
            "CustomErrorResponses": {
                "Quantity": 0
            },
            "Comment": "",
            "Logging": {
                "Enabled": false,
                "IncludeCookies": false,
                "Bucket": "",
                "Prefix": ""
            },
            "PriceClass": "PriceClass_All",
            "Enabled": true,
            "ViewerCertificate": {
                "CloudFrontDefaultCertificate": true,
                "MinimumProtocolVersion": "TLSv1",
                "CertificateSource": "cloudfront"
            },
            "Restrictions": {
                "GeoRestriction": {
                    "RestrictionType": "none",
                    "Quantity": 0
                }
            },
            "WebACLId": "",
            "HttpVersion": "http2",
            "IsIPV6Enabled": true
        }
    }
}
```

Esempio 2: creare una CloudFront distribuzione utilizzando un file JSON

L'esempio `create-distribution` seguente crea una distribuzione per un bucket S3 denominato `amzn-s3-demo-bucket` e specifica inoltre `index.html` come oggetto root predefinito utilizzando un file JSON.

```
aws cloudfront create-distribution \
    --distribution-config file://dist-config.json
```

Contenuto di `dist-config.json`:

```
{
    "CallerReference": "cli-example",
    "Aliases": {
        "Quantity": 0
    },
    "DefaultRootObject": "index.html",
    "Origins": {
        "Quantity": 1,
        "Items": [
            {
                "Id": "amzn-s3-demo-bucket.s3.amazonaws.com-cli-example",
                "DomainName": "amzn-s3-demo-bucket.s3.amazonaws.com",
                "OriginPath": "",
                "CustomHeaders": {
                    "Quantity": 0
                },
                "S3OriginConfig": {
                    "OriginAccessIdentity": ""
                }
            }
        ]
    },
    "OriginGroups": {
        "Quantity": 0
    },
    "DefaultCacheBehavior": {
        "TargetOriginId": "amzn-s3-demo-bucket.s3.amazonaws.com-cli-example",
        "ForwardedValues": {
            "QueryString": false,
            "Cookies": {
                "Forward": "none"
            },
            "Headers": {
                "Quantity": 0
            },
            "QueryStringCacheKeys": {
                "Quantity": 0
            }
        },
        "TrustedSigners": {
            "Enabled": false,
            "Quantity": 0
        },
```

```
        "ViewerProtocolPolicy": "allow-all",
        "MinTTL": 0,
        "AllowedMethods": {
            "Quantity": 2,
            "Items": [
                "HEAD",
                "GET"
            ],
            "CachedMethods": {
                "Quantity": 2,
                "Items": [
                    "HEAD",
                    "GET"
                ]
            }
        },
        "SmoothStreaming": false,
        "DefaultTTL": 86400,
        "MaxTTL": 31536000,
        "Compress": false,
        "LambdaFunctionAssociations": {
            "Quantity": 0
        },
        "FieldLevelEncryptionId": ""
    },
    "CacheBehaviors": {
        "Quantity": 0
    },
    "CustomErrorResponses": {
        "Quantity": 0
    },
    "Comment": "",
    "Logging": {
        "Enabled": false,
        "IncludeCookies": false,
        "Bucket": "",
        "Prefix": ""
    },
    "PriceClass": "PriceClass_All",
    "Enabled": true,
    "ViewerCertificate": {
        "CloudFrontDefaultCertificate": true,
        "MinimumProtocolVersion": "TLSv1",
        "CertificateSource": "cloudfront"
```

```
    },
    "Restrictions": {
        "GeoRestriction": {
            "RestrictionType": "none",
            "Quantity": 0
        }
    },
    "WebACLId": "",
    "HttpVersion": "http2",
    "IsIPV6Enabled": true
}
```

Consulta l'esempio 1 per un output di esempio.

Esempio 3: creare una distribuzione CloudFront multi-tenant con un certificato

L'`create-distribution`esempio seguente crea una CloudFront distribuzione con supporto multi-tenant e specifica un certificato TLS.

```
aws cloudfront create-distribution \
    --distribution-config file://dist-config.json
```

Contenuto di `dist-config.json`:

```
{
    "CallerReference": "cli-example-with-cert",
    "Comment": "CLI example distribution",
    "DefaultRootObject": "index.html",
    "Origins": {
        "Quantity": 1,
        "Items": [
            {
                "Id": "amzn-s3-demo-bucket.s3.us-east-1.amazonaws.com",
                "DomainName": "amzn-s3-demo-bucket.s3.us-east-1.amazonaws.com",
                "OriginPath": "/{{tenantName}}",
                "CustomHeaders": {
                    "Quantity": 0
                },
                "S3OriginConfig": {
                    "OriginAccessIdentity": ""
                }
            }
        ]
```

```
    },
    "DefaultCacheBehavior": {
        "TargetOriginId": "amzn-s3-demo-bucket.s3.us-east-1.amazonaws.com",
        "CachePolicyId": "658327ea-f89d-4fab-a63d-7e88639e5ABC",
        "ViewerProtocolPolicy": "allow-all",
        "AllowedMethods": {
            "Quantity": 2,
            "Items": ["HEAD", "GET"],
            "CachedMethods": {
                "Quantity": 2,
                "Items": ["HEAD", "GET"]
            }
        }
    },
    "Enabled": true,
    "ViewerCertificate": {
        "ACMCertificateArn": "arn:aws:acm:us-
east-1:123456789012:certificate/191306a1-db01-49ca-90ef-fc414ee5dabc",
        "SSLSupportMethod": "sni-only"
    },
    "HttpVersion": "http2",
    "ConnectionMode": "tenant-only",
    "TenantConfig": {
        "ParameterDefinitions": [
            {
                "Name": "tenantName",
                "Definition": {
                    "StringSchema": {
                        "Comment": "tenantName parameter",
                        "DefaultValue": "root",
                        "Required": false
                    }
                }
            }
        ]
    }
}
```

Output:

```
{
    "Location": "https://cloudfront.amazonaws.com/2020-05-31/distribution/
E1HVIAU7UABC",
```

```
    "ETag": "E20LT7R1BABC",
    "Distribution": {
        "Id": "E1HVIAU7U12ABC",
        "ARN": "arn:aws:cloudfront::123456789012:distribution/E1HVIAU7U12ABC",
        "Status": "InProgress",
        "LastModifiedTime": "2025-07-10T20:33:31.117000+00:00",
        "InProgressInvalidationBatches": 0,
        "DomainName": "example.com",
        "ActiveTrustedSigners": {
            "Enabled": false,
            "Quantity": 0
        },
        "ActiveTrustedKeyGroups": {
            "Enabled": false,
            "Quantity": 0
        },
        "DistributionConfig": {
            "CallerReference": "cli-example-with-cert",
            "DefaultRootObject": "index.html",
            "Origins": {
                "Quantity": 1,
                "Items": [
                    {
                        "Id": "amzn-s3-demo-bucket.s3.us-east-1.amazonaws.com",
                        "DomainName": "amzn-s3-demo-bucket.s3.us-
east-1.amazonaws.com",
                        "OriginPath": "/{{tenantName}}",
                        "CustomHeaders": {
                            "Quantity": 0
                        },
                        "S3OriginConfig": {
                            "OriginAccessIdentity": ""
                        },
                        "ConnectionAttempts": 3,
                        "ConnectionTimeout": 10,
                        "OriginShield": {
                            "Enabled": false
                        },
                        "OriginAccessControlId": ""
                    }
                ]
            },
            "OriginGroups": {
                "Quantity": 0
```

```
            },
            "DefaultCacheBehavior": {
                "TargetOriginId": "amzn-s3-demo-bucket.s3.us-east-1.amazonaws.com",
                "TrustedKeyGroups": {
                    "Enabled": false,
                    "Quantity": 0
                },
                "ViewerProtocolPolicy": "allow-all",
                "AllowedMethods": {
                    "Quantity": 2,
                    "Items": ["HEAD", "GET"],
                    "CachedMethods": {
                        "Quantity": 2,
                        "Items": ["HEAD", "GET"]
                    }
                },
                "Compress": false,
                "LambdaFunctionAssociations": {
                    "Quantity": 0
                },
                "FunctionAssociations": {
                    "Quantity": 0
                },
                "FieldLevelEncryptionId": "",
                "CachePolicyId": "658327ea-f89d-4fab-a63d-7e88639e5ABC",
                "GrpcConfig": {
                    "Enabled": false
                }
            },
            "CacheBehaviors": {
                "Quantity": 0
            },
            "CustomErrorResponses": {
                "Quantity": 0
            },
            "Comment": "CLI example distribution",
            "Logging": {
                "Enabled": false,
                "IncludeCookies": false,
                "Bucket": "",
                "Prefix": ""
            },
            "Enabled": true,
            "ViewerCertificate": {
```

```
                "CloudFrontDefaultCertificate": false,
                "ACMCertificateArn": "arn:aws:acm:us-
east-1:123456789012:certificate/1954f095-11b6-4daf-9952-0c308a00abc",
                "SSLSupportMethod": "sni-only",
                "MinimumProtocolVersion": "TLSv1.2_2021",
                "Certificate": "arn:aws:acm:us-
east-1:123456789012:certificate/1954f095-11b6-4daf-9952-0c308a00abc",
                "CertificateSource": "acm"
            },
            "Restrictions": {
                "GeoRestriction": {
                    "RestrictionType": "none",
                    "Quantity": 0
                }
            },
            "WebACLId": "",
            "HttpVersion": "http2",
            "TenantConfig": {
                "ParameterDefinitions": [
                    {
                        "Name": "tenantName",
                        "Definition": {
                            "StringSchema": {
                                "Comment": "tenantName parameter",
                                "DefaultValue": "root",
                                "Required": false
                            }
                        }
                    }
                ]
            },
            "ConnectionMode": "tenant-only"
        }
    }
}
```

Per ulteriori informazioni, consulta Lavorare con le distribuzioni nella Amazon CloudFront Developer Guide.

Esempio 4: creare una distribuzione CloudFront multi-tenant senza certificato

L'`create-distribution`esempio seguente crea una CloudFront distribuzione con supporto multi-tenant ma senza un certificato TLS.

```
aws cloudfront create-distribution \
    --distribution-config file://dist-config.json
```

Contenuto di `dist-config.json`:

```
{
    "CallerReference": "cli-example",
    "Comment": "CLI example distribution",
    "DefaultRootObject": "index.html",
    "Origins": {
        "Quantity": 1,
        "Items": [
            {
                "Id": "amzn-s3-demo-bucket.s3.us-east-1.amazonaws.com",
                "DomainName": "amzn-s3-demo-bucket.s3.us-east-1.amazonaws.com",
                "OriginPath": "/{{tenantName}}",
                "CustomHeaders": {
                    "Quantity": 0
                },
                "S3OriginConfig": {
                    "OriginAccessIdentity": ""
                }
            }
        ]
    },
    "DefaultCacheBehavior": {
        "TargetOriginId": "amzn-s3-demo-bucket.s3.us-east-1.amazonaws.com",
        "CachePolicyId": "658327ea-f89d-4fab-a63d-7e88639e5ABC",
        "ViewerProtocolPolicy": "allow-all",
        "AllowedMethods": {
            "Quantity": 2,
            "Items": [
                "HEAD",
                "GET"
            ],
            "CachedMethods": {
                "Quantity": 2,
                "Items": [
                    "HEAD",
                    "GET"
                ]
            }
        }
```

```
    },
    "Enabled": true,
    "HttpVersion": "http2",
    "ConnectionMode": "tenant-only",
    "TenantConfig": {
        "ParameterDefinitions": [
            {
                "Name": "tenantName",
                "Definition": {
                    "StringSchema": {
                        "Comment": "tenantName parameter",
                        "DefaultValue": "root",
                        "Required": false
                    }
                }
            }
        ]
    }
}
```

Output:

```
{
    "Location": "https://cloudfront.amazonaws.com/2020-05-31/distribution/
E2GJ5J9QN12ABC",
    "ETag": "E37YLVVQIABC",
    "Distribution": {
        "Id": "E2GJ5J9QNABC",
        "ARN": "arn:aws:cloudfront::123456789012:distribution/E2GJ5J9QN12ABC",
        "Status": "InProgress",
        "LastModifiedTime": "2025-07-10T20:35:20.565000+00:00",
        "InProgressInvalidationBatches": 0,
        "DomainName": "example.com",
        "ActiveTrustedSigners": {
            "Enabled": false,
            "Quantity": 0
        },
        "ActiveTrustedKeyGroups": {
            "Enabled": false,
            "Quantity": 0
        },
        "DistributionConfig": {
            "CallerReference": "cli-example-no-cert",
```

```
            "DefaultRootObject": "index.html",
            "Origins": {
                "Quantity": 1,
                "Items": [
                    {
                        "Id": "amzn-s3-demo-bucket.s3.us-east-1.amazonaws.com",
                        "DomainName": "amzn-s3-demo-bucket.s3.us-
east-1.amazonaws.com",
                        "OriginPath": "/{{tenantName}}",
                        "CustomHeaders": {
                            "Quantity": 0
                        },
                        "S3OriginConfig": {
                            "OriginAccessIdentity": ""
                        },
                        "ConnectionAttempts": 3,
                        "ConnectionTimeout": 10,
                        "OriginShield": {
                            "Enabled": false
                        },
                        "OriginAccessControlId": ""
                    }
                ]
            },
            "OriginGroups": {
                "Quantity": 0
            },
            "DefaultCacheBehavior": {
                "TargetOriginId": "amzn-s3-demo-bucket.s3.us-east-1.amazonaws.com",
                "TrustedKeyGroups": {
                    "Enabled": false,
                    "Quantity": 0
                },
                "ViewerProtocolPolicy": "allow-all",
                "AllowedMethods": {
                    "Quantity": 2,
                    "Items": [
                        "HEAD",
                        "GET"
                    ],
                    "CachedMethods": {
                        "Quantity": 2,
                        "Items": [
                            "HEAD",
```

```
                        "GET"
                    ]
                }
            },
            "Compress": false,
            "LambdaFunctionAssociations": {
                "Quantity": 0
            },
            "FunctionAssociations": {
                "Quantity": 0
            },
            "FieldLevelEncryptionId": "",
            "CachePolicyId": "658327ea-f89d-4fab-a63d-7e88639e5ABC",
            "GrpcConfig": {
                "Enabled": false
            }
        },
        "CacheBehaviors": {
            "Quantity": 0
        },
        "CustomErrorResponses": {
            "Quantity": 0
        },
        "Comment": "CLI example distribution",
        "Logging": {
            "Enabled": false,
            "IncludeCookies": false,
            "Bucket": "",
            "Prefix": ""
        },
        "Enabled": true,
        "ViewerCertificate": {
            "CloudFrontDefaultCertificate": true,
            "SSLSupportMethod": "sni-only",
            "MinimumProtocolVersion": "TLSv1",
            "CertificateSource": "cloudfront"
        },
        "Restrictions": {
            "GeoRestriction": {
                "RestrictionType": "none",
                "Quantity": 0
            }
        },
        "WebACLId": "",
```

```
            "HttpVersion": "http2",
            "TenantConfig": {
                "ParameterDefinitions": [
                    {
                        "Name": "tenantName",
                        "Definition": {
                            "StringSchema": {
                                "Comment": "tenantName parameter",
                                "DefaultValue": "root",
                                "Required": false
                            }
                        }
                    }
                ]
            },
            "ConnectionMode": "tenant-only"
        }
    }
}
```

Per ulteriori informazioni, consulta Configurare le distribuzioni nell'Amazon CloudFront Developer Guide.

- Per i dettagli sull'API, consulta CreateDistribution AWS CLICommand Reference.

`create-field-level-encryption-config`

Il seguente esempio di codice mostra come utilizzare`create-field-level-encryption-config`.

AWS CLI

Per creare una configurazione di CloudFront crittografia a livello di campo

L'esempio seguente crea una configurazione di crittografia a livello di campo fornendo i parametri di configurazione in un file JSON denominato `fle-config.json`. Per creare una configurazione di crittografia a livello di campo, è necessario disporre di un profilo di crittografia allo stesso livello. Per creare un profilo, vedere il comando create-field-level-encryption -profile.

Per ulteriori informazioni sulla crittografia a CloudFront livello di campo, consulta Using Field-Level Encryption to Help Protect Sensitive Data nella Amazon Developer Guide. CloudFront

```
aws cloudfront create-field-level-encryption-config \
```

```
    --field-level-encryption-config file://fle-config.json
```

Il file `fle-config.json` è un documento JSON nella cartella corrente che contiene quanto segue:

```
{
    "CallerReference": "cli-example",
    "Comment": "Example FLE configuration",
    "QueryArgProfileConfig": {
        "ForwardWhenQueryArgProfileIsUnknown": true,
        "QueryArgProfiles": {
            "Quantity": 0
        }
    },
    "ContentTypeProfileConfig": {
        "ForwardWhenContentTypeIsUnknown": true,
        "ContentTypeProfiles": {
            "Quantity": 1,
            "Items": [
                {
                    "Format": "URLEncoded",
                    "ProfileId": "P280MFCLSYOCVU",
                    "ContentType": "application/x-www-form-urlencoded"
                }
            ]
        }
    }
}
```

Output:

```
{
    "Location": "https://cloudfront.amazonaws.com/2019-03-26/field-level-encryption/
C3KM2WVD605UAY",
    "ETag": "E2P4Z4VU7TY5SG",
    "FieldLevelEncryption": {
        "Id": "C3KM2WVD605UAY",
        "LastModifiedTime": "2019-12-10T21:30:18.974Z",
        "FieldLevelEncryptionConfig": {
            "CallerReference": "cli-example",
            "Comment": "Example FLE configuration",
            "QueryArgProfileConfig": {
                "ForwardWhenQueryArgProfileIsUnknown": true,
```

```
                "QueryArgProfiles": {
                    "Quantity": 0,
                    "Items": []
                }
            },
            "ContentTypeProfileConfig": {
                "ForwardWhenContentTypeIsUnknown": true,
                "ContentTypeProfiles": {
                    "Quantity": 1,
                    "Items": [
                        {
                            "Format": "URLEncoded",
                            "ProfileId": "P280MFCLSYOCVU",
                            "ContentType": "application/x-www-form-urlencoded"
                        }
                    ]
                }
            }
        }
    }
}
```

- Per i dettagli sull'API, consulta CreateFieldLevelEncryptionConfigCommand Reference.AWS CLI

`create-field-level-encryption-profile`

Il seguente esempio di codice mostra come utilizzare`create-field-level-encryption-profile`.

AWS CLI

Per creare un profilo di CloudFront crittografia a livello di campo

L'esempio seguente crea un profilo di crittografia a livello di campo fornendo i parametri in un file JSON denominato `fle-profile-config.json`. Prima di poter creare un profilo di crittografia a livello di campo, è necessario disporre di una chiave pubblica. CloudFront Per creare una chiave CloudFront pubblica, vedi il create-public-key comando.

Per ulteriori informazioni sulla crittografia a CloudFront livello di campo, consulta Using Field-Level Encryption to Help Protect Sensitive Data nella Amazon Developer Guide. CloudFront

```
aws cloudfront create-field-level-encryption-profile \
    --field-level-encryption-profile-config file://fle-profile-config.json
```

Il file `fle-profile-config.json` è un documento JSON nella cartella corrente che contiene quanto segue:

```
{
    "Name": "ExampleFLEProfile",
    "CallerReference": "cli-example",
    "Comment": "FLE profile for AWS CLI example",
    "EncryptionEntities": {
        "Quantity": 1,
        "Items": [
            {
                "PublicKeyId": "K2K8NC4HVFE3M0",
                "ProviderId": "ExampleFLEProvider",
                "FieldPatterns": {
                    "Quantity": 1,
                    "Items": [
                        "ExampleSensitiveField"
                    ]
                }
            }
        ]
    }
}
```

Output:

```
{
    "Location": "https://cloudfront.amazonaws.com/2019-03-26/field-level-encryption-
profile/PPK0UOSIF5WSV",
    "ETag": "E2QWRUHEXAMPLE",
    "FieldLevelEncryptionProfile": {
        "Id": "PPK0UOSIF5WSV",
        "LastModifiedTime": "2019-12-10T01:03:16.537Z",
        "FieldLevelEncryptionProfileConfig": {
            "Name": "ExampleFLEProfile",
            "CallerReference": "cli-example",
            "Comment": "FLE profile for AWS CLI example",
            "EncryptionEntities": {
                "Quantity": 1,
```

```
                "Items": [
                    {
                        "PublicKeyId": "K2K8NC4HVFE3M0",
                        "ProviderId": "ExampleFLEProvider",
                        "FieldPatterns": {
                            "Quantity": 1,
                            "Items": [
                                "ExampleSensitiveField"
                            ]
                        }
                    }
                ]
            }
        }
    }
}
```

- Per i dettagli sull'API, consulta CreateFieldLevelEncryptionProfileCommand Reference.AWS CLI

`create-invalidation-for-distribution-tenant`

Il seguente esempio di codice mostra come utilizzare`create-invalidation-for-distribution-tenant`.

AWS CLI

Per creare un' CloudFront invalidazione per un tenant di distribuzione

L'`create-invalidation-for-distribution-tenant`esempio seguente crea un'invalidazione per tutti i file in un tenant di distribuzione. CloudFront

```
aws cloudfront create-invalidation-for-distribution-tenant \
    --id dt_2wjDZi3hD1ivOXf6rpZJO1AB \
    --invalidation-batch '{"Paths": {"Quantity": 1, "Items": ["/*"]},
 "CallerReference": "invalidation-$(date +%s)"}'
```

Output:

```
{
    "Location": "https://cloudfront.amazonaws.com/2020-05-31/distribution-tenant/
dt_2wjDZi3hD1ivOXf6rpZJO1AB/invalidation/I2JGL2F1ZAA426PGG0YLLKABC",
```

```
    "Invalidation": {
        "Id": "I2JGL2F1ZAA426PGG0YLLKABC",
        "Status": "InProgress",
        "CreateTime": "2025-05-07T16:59:25.947000+00:00",
        "InvalidationBatch": {
            "Paths": {
                "Quantity": 1,
                "Items": [
                    "/*"
                ]
            },
            "CallerReference": "invalidation-$(date +%s)"
        }
    }
}
```

Per ulteriori informazioni, consulta Invalidare i file per rimuovere contenuti nell'Amazon CloudFront Developer Guide.

- Per i dettagli sull'API, consulta AWS CLI Command CreateInvalidationForDistributionTenantReference.

`create-invalidation`

Il seguente esempio di codice mostra come utilizzare`create-invalidation`.

AWS CLI

Per creare un'invalidazione per una distribuzione CloudFront

L'`create-invalidation`esempio seguente crea un'invalidazione per i file specificati nella distribuzione specificata: CloudFront

```
aws cloudfront create-invalidation \
    --distribution-id EDFDVBD6EXAMPLE \
    --paths "/example-path/example-file.jpg" "/example-path/example-file2.png"
```

Output:

```
{
    "Location": "https://cloudfront.amazonaws.com/2019-03-26/distribution/
EDFDVBD6EXAMPLE/invalidation/I1JLWSDAP8FU89",
    "Invalidation": {
```

```
        "Id": "I1JLWSDAP8FU89",
        "Status": "InProgress",
        "CreateTime": "2019-12-05T18:24:51.407Z",
        "InvalidationBatch": {
            "Paths": {
                "Quantity": 2,
                "Items": [
                    "/example-path/example-file2.png",
                    "/example-path/example-file.jpg"
                ]
            },
            "CallerReference": "cli-1575570291-670203"
        }
    }
}
```

Nell'esempio precedente, la AWS CLI generava automaticamente un risultato casuale. `CallerReference` Per specificare i parametri `CallerReference` o per evitare di passare i parametri di invalidazione come argomenti della riga di comando, è possibile utilizzare un file JSON. L'esempio seguente crea un'invalidazione per due file fornendo i parametri di invalidazione in un file JSON denominato `inv-batch.json`:

```
aws cloudfront create-invalidation \
    --distribution-id EDFDVBD6EXAMPLE \
    --invalidation-batch file://inv-batch.json
```

Contenuto di `inv-batch.json`:

```
{
    "Paths": {
        "Quantity": 2,
        "Items": [
            "/example-path/example-file.jpg",
            "/example-path/example-file2.png"
        ]
    },
    "CallerReference": "cli-example"
}
```

Output:

```
{
```

```
    "Location": "https://cloudfront.amazonaws.com/2019-03-26/distribution/
EDFDVBD6EXAMPLE/invalidation/I2J0I21PCUYOIK",
    "Invalidation": {
        "Id": "I2J0I21PCUYOIK",
        "Status": "InProgress",
        "CreateTime": "2019-12-05T18:40:49.413Z",
        "InvalidationBatch": {
            "Paths": {
                "Quantity": 2,
                "Items": [
                    "/example-path/example-file.jpg",
                    "/example-path/example-file2.png"
                ]
            },
            "CallerReference": "cli-example"
        }
    }
}
```

- Per i dettagli sull'API, consulta CreateInvalidation AWS CLICommand Reference.

`create-public-key`

Il seguente esempio di codice mostra come utilizzare`create-public-key`.

AWS CLI

Per creare una chiave CloudFront pubblica

L'esempio seguente crea una chiave CloudFront pubblica fornendo i parametri in un file JSON denominato`pub-key-config.json`. Prima di poter utilizzare questo comando, è necessario disporre di una chiave pubblica con codifica PEM. Per ulteriori informazioni, consulta Create an RSA Key Pair nella Amazon CloudFront Developer Guide.

```
aws cloudfront create-public-key \
    --public-key-config file://pub-key-config.json
```

Il file `pub-key-config.json` è un documento JSON nella cartella corrente che contiene quanto segue. Nota che la chiave pubblica è codificata in formato PEM.

```
{
```

```
    "CallerReference": "cli-example",
    "Name": "ExampleKey",
    "EncodedKey": "-----BEGIN PUBLIC KEY-----
\nMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAxPMbCA2Ks0lnd7IR+3pw
\nwd3H/7jPGwj8bLUmore7bX+oeGpZ6QmLAe/1UOWcmZX2u70dYcSIzB1ofZtcn4cJ
\nenHBAzO3ohBY/L1tQGJfS2A+omnN6H16VZE1JCK8XSJyfze7MDLcUyHZETdxuvRb
\nA9X343/vMAuQPnhinFJ8Wdy8YBXSPpy7r95ylUQd9LfYTBzVZYG2tSesplcOkjM3\n2Uu
+oMWxQAw1NINnSLPinMVsutJy6ZqlV3McWNWe4T+STGtWhrPNqJEn45sIcCx4\nq
+kGZ2NQ0FyIyT2eiLKOX5Rgb/a36E/aMk4VoDsaenBQgG7WLTnstb9sr7MIhS6A\nrwIDAQAB\n-----END
 PUBLIC KEY-----\n",
    "Comment": "example public key"
}
```

Output:

```
{
    "Location": "https://cloudfront.amazonaws.com/2019-03-26/public-key/
KDFB19YGCR002",
    "ETag": "E2QWRUHEXAMPLE",
    "PublicKey": {
        "Id": "KDFB19YGCR002",
        "CreatedTime": "2019-12-05T18:51:43.781Z",
        "PublicKeyConfig": {
            "CallerReference": "cli-example",
            "Name": "ExampleKey",
            "EncodedKey": "-----BEGIN PUBLIC KEY-----
\nMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAxPMbCA2Ks0lnd7IR+3pw
\nwd3H/7jPGwj8bLUmore7bX+oeGpZ6QmLAe/1UOWcmZX2u70dYcSIzB1ofZtcn4cJ
\nenHBAzO3ohBY/L1tQGJfS2A+omnN6H16VZE1JCK8XSJyfze7MDLcUyHZETdxuvRb
\nA9X343/vMAuQPnhinFJ8Wdy8YBXSPpy7r95ylUQd9LfYTBzVZYG2tSesplcOkjM3\n2Uu
+oMWxQAw1NINnSLPinMVsutJy6ZqlV3McWNWe4T+STGtWhrPNqJEn45sIcCx4\nq
+kGZ2NQ0FyIyT2eiLKOX5Rgb/a36E/aMk4VoDsaenBQgG7WLTnstb9sr7MIhS6A\nrwIDAQAB\n-----END
 PUBLIC KEY-----\n",
            "Comment": "example public key"
        }
    }
}
```

- Per i dettagli sull'API, consulta CreatePublicKey AWS CLICommand Reference.

`delete-cloud-front-origin-access-identity`

Il seguente esempio di codice mostra come utilizzare`delete-cloud-front-origin-access-identity`.

AWS CLI

Per eliminare un'identità di accesso di CloudFront origine

Il seguente esempio elimina l'identità di accesso origine (OAI) con l'ID `E74FTE3AEXAMPLE`. Per eliminare un'identità di accesso origine (OAI), è necessario disporre dei relativi `ETag` e ID. L'ID OAI viene restituito nell'output dei comandi create-cloud-front-origin -access-identity e -access-identities. list-cloud-front-origin Per ottenere il, usa il comando -access-identity o `ETag` -. get-cloud-front-origin get-cloud-front-origin access-identity-config Utilizza l'opzione `--if-match` per fornire gli `ETag` dell'identità di accesso origine (OAI).

```
aws cloudfront delete-cloud-front-origin-access-identity \
    --id E74FTE3AEXAMPLE \
    --if-match E2QWRUHEXAMPLE
```

Se ha esito positivo, questo comando non produce alcun output.

- Per i dettagli sull'API, consulta Command DeleteCloudFrontOriginAccessIdentityReference AWS CLI .

`delete-connection-group`

Il seguente esempio di codice mostra come utilizzare`delete-connection-group`.

AWS CLI

Come eliminare un gruppo di connessioni

L'esempio `delete-connection-group` seguente elimina un gruppi di connessioni. Il gruppo di connessione deve essere disabilitato e non può essere associato a nessuna CloudFront risorsa.

```
aws cloudfront delete-connection-group \
    --id cg_2wjLpjbHkLUdhWAjHllcOeABC \
    --if-match ETVPDKIKX0DABC
```

Se ha esito positivo, questo comando non produce alcun output.

Per ulteriori informazioni sulla gestione dei gruppi di connessione, consulta Creare un gruppo di connessione personalizzato (opzionale) nella Amazon CloudFront Developer Guide.

- Per i dettagli sull'API, consulta DeleteConnectionGroup AWS CLICommand Reference.

`delete-distribution-tenant`

Il seguente esempio di codice mostra come utilizzare`delete-distribution-tenant`.

AWS CLI

Come eliminare un tenant di distribuzione

L'`delete-distribution-tenant`esempio seguente elimina un tenant di distribuzione con. ETag `ETVPDKIKX0DABC` Il tenant di distribuzione deve essere disabilitato e non può essere associato ad alcuna risorsa. CloudFront

```
aws cloudfront delete-distribution-tenant \
    --id dt_2wjMUbg3NHZEQ7OfoalP5zi1AB \
    --if-match ETVPDKIKX0DABC
```

Se ha esito positivo, questo comando non produce alcun output.

Per ulteriori informazioni, consulta Eliminare una distribuzione nella Amazon CloudFront Developer Guide.

- Per i dettagli sull'API, consulta DeleteDistributionTenant AWS CLICommand Reference.

`delete-distribution`

Il seguente esempio di codice mostra come utilizzare`delete-distribution`.

AWS CLI

Per eliminare una CloudFront distribuzione

L'esempio seguente elimina la CloudFront distribuzione con l'ID. `EDFDVBD6EXAMPLE` Prima di eliminare una distribuzione, devi disabilitarla. Per disabilitare una distribuzione, utilizza il comando update-distribution. Per ulteriori informazioni, consulta gli esempi del comando update-distribution.

Quando una distribuzione è disabilitata, è possibile eliminarla. Per eliminare una distribuzione, è necessario utilizzare l'opzione `--if-match` per fornire il tag entità (`ETag`) della distribuzione. Per ottenere il`ETag`, utilizzare il comando get-distribution o. get-distribution-config

```
aws cloudfront delete-distribution \
    --id EDFDVBD6EXAMPLE \
    --if-match E2QWRUHEXAMPLE
```

Se ha esito positivo, questo comando non produce alcun output.

- Per i dettagli sull'API, consulta AWS CLI Command DeleteDistributionReference.

`delete-field-level-encryption-config`

Il seguente esempio di codice mostra come utilizzare`delete-field-level-encryption-config`.

AWS CLI

Per eliminare una configurazione di CloudFront crittografia a livello di campo

L'esempio seguente elimina la configurazione di crittografia a CloudFront livello di campo con l'ID. `C3KM2WVD605UAY` Per eliminare una configurazione di crittografia a livello di campo, è necessario disporre dei relativi `ETag` e ID. L'ID viene restituito nell'output dei comandi create-field-level-encryption -config e -configs. list-field-level-encryption Per ottenere il`ETag`, usa il comando or -config. get-field-level-encryption get-field-level-encryption Utilizza l'opzione `--if-match` per fornire il valore `ETag` della configurazione.

```
aws cloudfront delete-field-level-encryption-config \
    --id C3KM2WVD605UAY \
    --if-match E26M4BIAV81ZF6
```

Se ha esito positivo, questo comando non produce alcun output.

- Per i dettagli sull'API, consulta AWS CLI Command DeleteFieldLevelEncryptionConfigReference.

`delete-field-level-encryption-profile`

Il seguente esempio di codice mostra come utilizzare`delete-field-level-encryption-profile`.

AWS CLI

Per eliminare un profilo di CloudFront crittografia a livello di campo

L'esempio seguente elimina il profilo di crittografia a CloudFront livello di campo con l'ID. `PPK0UOSIF5WSV` Per eliminare un profilo di crittografia a livello di campo, è necessario disporre dei relativi `ETag` e ID. L'ID viene restituito nell'output dei comandi create-field-level-encryption -profile e -profiles. list-field-level-encryption Per ottenere il`ETag`, usa il comando get-field-level-encryption -profile o get-field-level-encryption -profile-config. Utilizza l'opzione `--if-match` per fornire gli `ETag` del profilo.

```
aws cloudfront delete-field-level-encryption-profile \
    --id PPK0UOSIF5WSV \
    --if-match EJETYFJ9CL66D
```

Se ha esito positivo, questo comando non produce alcun output.

- Per i dettagli sull'API, consulta Command Reference. DeleteFieldLevelEncryptionProfileAWS CLI

`delete-public-key`

Il seguente esempio di codice mostra come utilizzare`delete-public-key`.

AWS CLI

Per eliminare una chiave CloudFront pubblica

L'esempio seguente elimina la chiave CloudFront pubblica con l'ID. `KDFB19YGCR002` Per eliminare una chiave pubblica, è necessario disporre dei relativi `ETag` e ID. L'ID viene restituito nell'output dei list-public-keys comandi create-public-key and. Per ottenere il`ETag`, usa il get-public-key-config comando get-public-key or. Utilizza l'opzione `--if-match` per fornire gli `ETag` della chiave pubblica.

```
aws cloudfront delete-public-key \
    --id KDFB19YGCR002 \
    --if-match E2QWRUHEXAMPLE
```

Se ha esito positivo, questo comando non produce alcun output.

- Per i dettagli sull'API, consulta DeletePublicKey AWS CLICommand Reference.

`disassociate-distribution-tenant-web-acl`

Il seguente esempio di codice mostra come utilizzare`disassociate-distribution-tenant-web-acl`.

AWS CLI

Come annullare l'associazione tra una lista di controllo degli accessi (ACL) web e un tenant di distribuzione

L'`disassociate-distribution-tenant-web-acl`esempio seguente dissocia un ACL Web da un tenant di distribuzione con. ETag `E1PA6795UKMABC`

```
aws cloudfront disassociate-distribution-tenant-web-acl \
    --id dt_2wjDZi3hD1ivOXf6rpZJOSNE1AB \
    --if-match E1PA6795UKMABC
```

Output:

```
{
    "ETag": "E13V1IB3VIYABC",
    "Id": "dt_2wjDZi3hD1ivOXf6rpZJOSNE1AB"
}
```

Per ulteriori informazioni, consulta Disabilitare le protezioni di sicurezza AWS WAF nella Amazon CloudFront Developer Guide.

- Per i dettagli sull'API, consulta AWS CLI Command DisassociateDistributionTenantWebAclReference.

`disassociate-distribution-web-acl`

Il seguente esempio di codice mostra come utilizzare`disassociate-distribution-web-acl`.

AWS CLI

Per dissociare un ACL Web da una distribuzione CloudFront

L'`disassociate-distribution-web-acl`esempio seguente rimuove l'associazione tra un ACL Web e una CloudFront distribuzione con. ETag `E13V1IB3VIYABC`

```
aws cloudfront disassociate-distribution-web-acl \
    --id E1XNX8R2G0AABC \
    --if-match EEZQ9Z24VM1ABC
```

Output:

```
{
    "ETag": "E2YWS1C2J30ABC",
    "Id": "E1XNX8R2G0AABC"
}
```

Per ulteriori informazioni, consulta Disabilitare le protezioni di sicurezza AWS WAF nella Amazon CloudFront Developer Guide.

- Per i dettagli sull'API, consulta AWS CLI Command DisassociateDistributionWebAclReference.

`get-cloud-front-origin-access-identity-config`

Il seguente esempio di codice mostra come utilizzare`get-cloud-front-origin-access-identity-config`.

AWS CLI

Per ottenere una configurazione dell'identità di accesso all' CloudFront origine

L'esempio seguente ottiene i metadati sull'identità di accesso all' CloudFront origine (OAI) con l'`IDE74FTE3AEXAMPLE`, incluso il relativo. `ETag` L'ID OAI viene restituito nell'output dei comandi -access-identity e create-cloud-front-origin -access-identities. list-cloud-front-origin

```
aws cloudfront get-cloud-front-origin-access-identity-config --id E74FTE3AEXAMPLE
```

Output:

```
{
    "ETag": "E2QWRUHEXAMPLE",
    "CloudFrontOriginAccessIdentityConfig": {
        "CallerReference": "cli-example",
```

```
        "Comment": "Example OAI"
    }
}
```

- Per i dettagli sull'API, consulta Command Reference. GetCloudFrontOriginAccessIdentityConfigAWS CLI

`get-cloud-front-origin-access-identity`

Il seguente esempio di codice mostra come utilizzare`get-cloud-front-origin-access-identity`.

AWS CLI

Per ottenere un'identità di accesso all' CloudFront origine

L'esempio seguente ottiene l'identità di accesso di CloudFront origine (OAI) con l'IDE74FTE3AEXAMPLE, incluso il relativo ID canonico S3 `ETag` e l'ID canonico S3 associato. L'ID OAI viene restituito nell'output dei comandi -access-identity e -access-identitiescreate-cloud-front-origin. list-cloud-front-origin

```
aws cloudfront get-cloud-front-origin-access-identity --id E74FTE3AEXAMPLE
```

Output:

```
{
    "ETag": "E2QWRUHEXAMPLE",
    "CloudFrontOriginAccessIdentity": {
        "Id": "E74FTE3AEXAMPLE",
        "S3CanonicalUserId":
 "cd13868f797c227fbea2830611a26fe0a21ba1b826ab4bed9b7771c9aEXAMPLE",
        "CloudFrontOriginAccessIdentityConfig": {
            "CallerReference": "cli-example",
            "Comment": "Example OAI"
        }
    }
}
```

- Per i dettagli sull'API, consulta Command Reference. GetCloudFrontOriginAccessIdentityAWS CLI

`get-connection-group-by-routing-endpoint`

Il seguente esempio di codice mostra come utilizzare`get-connection-group-by-routing-endpoint`.

AWS CLI

Come ottenere un gruppo di connessione tramite un endpoint di routing

L'esempio `get-connection-group-by-routing-endpoint` seguente recupera le informazioni su un gruppo di connessioni utilizzando il relativo endpoint di routing.

```
aws cloudfront get-connection-group-by-routing-endpoint \
    --routing-endpoint dvdg9gprgabc.cloudfront.net
```

Output:

```
{
    "ETag": "E23ZP02F085ABC",
    "ConnectionGroup": {
        "Id": "cg_2wjDWTBKTlRB87cAaUQFaakABC",
        "Name": "connection-group-2",
        "Arn": "arn:aws:cloudfront::123456789012:connection-group/
cg_2wjDWTBKTlRB87cAaUQFaakABC",
        "CreatedTime": "2025-05-06T15:42:00.790000+00:00",
        "LastModifiedTime": "2025-05-06T15:42:00.790000+00:00",
        "Ipv6Enabled": true,
        "RoutingEndpoint": "dvdg9gprgabc.cloudfront.net",
        "Status": "Deployed",
        "Enabled": true,
        "IsDefault": false
    }
}
```

Per ulteriori informazioni, consulta Creare un gruppo di connessione personalizzato (opzionale) nella Amazon CloudFront Developer Guide.

- Per i dettagli sull'API, consulta GetConnectionGroupByRoutingEndpoint AWS CLICommand Reference.

`get-connection-group`

Il seguente esempio di codice mostra come utilizzare`get-connection-group`.

AWS CLI

Per creare un gruppo di CloudFront connessione

L'`get-connection-group`esempio seguente recupera informazioni su un gruppo di CloudFront connessione.

```
aws cloudfront get-connection-group \
    --identifier cg_2wjDWTBKTlRB87cAaUQFaakABC
```

Output:

```
{
    "ETag": "E23ZP02F085ABC",
    "ConnectionGroup": {
        "Id": "cg_2wjDWTBKTlRB87cAaUQFaakABC",
        "Name": "connection-group-2",
        "Arn": "arn:aws:cloudfront::123456789012:connection-group/
cg_2wjDWTBKTlRB87cAaUQFaakABC",
        "CreatedTime": "2025-05-06T15:42:00.790000+00:00",
        "LastModifiedTime": "2025-05-06T15:42:00.790000+00:00",
        "Ipv6Enabled": true,
        "RoutingEndpoint": "dvdg9gprgabc.cloudfront.net",
        "Status": "Deployed",
        "Enabled": true,
        "IsDefault": false
    }
}
```

Per ulteriori informazioni, consulta Creare un gruppo di connessione personalizzato (opzionale) nella Amazon CloudFront Developer Guide.

- Per i dettagli sull'API, consulta GetConnectionGroup AWS CLICommand Reference.

`get-distribution-config`

Il seguente esempio di codice mostra come utilizzare`get-distribution-config`.

AWS CLI

Per ottenere una configurazione CloudFront di distribuzione

L'esempio seguente ottiene i metadati sulla CloudFront distribuzione con l'IDEDFDVBD6EXAMPLE, incluso il relativoETag. L'ID distribuzione viene restituito nei comandi create-distribution e list-distribution.

```
aws cloudfront get-distribution-config \
    --id EDFDVBD6EXAMPLE
```

Output:

```
{
    "ETag": "E2QWRUHEXAMPLE",
    "DistributionConfig": {
        "CallerReference": "cli-example",
        "Aliases": {
            "Quantity": 0
        },
        "DefaultRootObject": "index.html",
        "Origins": {
            "Quantity": 1,
            "Items": [
                {
                    "Id": "amzn-s3-demo-bucket.s3.amazonaws.com-cli-example",
                    "DomainName": "amzn-s3-demo-bucket.s3.amazonaws.com",
                    "OriginPath": "",
                    "CustomHeaders": {
                        "Quantity": 0
                    },
                    "S3OriginConfig": {
                        "OriginAccessIdentity": ""
                    }
                }
            ]
        },
        "OriginGroups": {
            "Quantity": 0
        },
        "DefaultCacheBehavior": {
            "TargetOriginId": "amzn-s3-demo-bucket.s3.amazonaws.com-cli-example",
            "ForwardedValues": {
```

```
                "QueryString": false,
                "Cookies": {
                    "Forward": "none"
                },
                "Headers": {
                    "Quantity": 0
                },
                "QueryStringCacheKeys": {
                    "Quantity": 0
                }
            },
            "TrustedSigners": {
                "Enabled": false,
                "Quantity": 0
            },
            "ViewerProtocolPolicy": "allow-all",
            "MinTTL": 0,
            "AllowedMethods": {
                "Quantity": 2,
                "Items": [
                    "HEAD",
                    "GET"
                ],
                "CachedMethods": {
                    "Quantity": 2,
                    "Items": [
                        "HEAD",
                        "GET"
                    ]
                }
            },
            "SmoothStreaming": false,
            "DefaultTTL": 86400,
            "MaxTTL": 31536000,
            "Compress": false,
            "LambdaFunctionAssociations": {
                "Quantity": 0
            },
            "FieldLevelEncryptionId": ""
        },
        "CacheBehaviors": {
            "Quantity": 0
        },
        "CustomErrorResponses": {
```

```
            "Quantity": 0
        },
        "Comment": "",
        "Logging": {
            "Enabled": false,
            "IncludeCookies": false,
            "Bucket": "",
            "Prefix": ""
        },
        "PriceClass": "PriceClass_All",
        "Enabled": true,
        "ViewerCertificate": {
            "CloudFrontDefaultCertificate": true,
            "MinimumProtocolVersion": "TLSv1",
            "CertificateSource": "cloudfront"
        },
        "Restrictions": {
            "GeoRestriction": {
                "RestrictionType": "none",
                "Quantity": 0
            }
        },
        "WebACLId": "",
        "HttpVersion": "http2",
        "IsIPV6Enabled": true
    }
}
```

- Per i dettagli sull'API, consulta GetDistributionConfig AWS CLICommand Reference.

`get-distribution-tenant-by-domain`

Il seguente esempio di codice mostra come utilizzare`get-distribution-tenant-by-domain`.

AWS CLI

Come ottenere informazioni sui tenant di distribuzione per dominio

L'esempio `get-distribution-tenant-by-domain` seguente recupera le informazioni su un tenant di distribuzione utilizzando il dominio specificato.

```
aws cloudfront get-distribution-tenant-by-domain \
```

```
    --domain example.com
```

Output:

```
{
    "ETag": "E23ZP02F085ABC",
    "DistributionTenant": {
        "Id": "dt_2xVInRKCfUzQHgxosDs9hiLk1AB",
        "DistributionId": "E1XNX8R2GOAABC",
        "Name": "example-tenant-4",
        "Arn": "arn:aws:cloudfront::123456789012:distribution-tenant/
dt_2xVInRKCfUzQHgxosDs9hiLk1AB",
        "Domains": [
            {
                "Domain": "example.com",
                "Status": "active"
            }
        ],
        "Parameters": [
            {
                "Name": "testParam",
                "Value": "defaultValue"
            }
        ],
        "ConnectionGroupId": "cg_2whCJoXMYCjHcxaLGrkllvyABC",
        "CreatedTime": "2025-05-23T16:16:20.871000+00:00",
        "LastModifiedTime": "2025-05-23T16:16:20.871000+00:00",
        "Enabled": false,
        "Status": "Deployed"
    }
}
```

Per ulteriori informazioni, consulta Comprendi come funzionano le distribuzioni multi-tenant nella Amazon CloudFront Developer Guide.

- Per i dettagli sull'API, consulta Command GetDistributionTenantByDomainReference AWS CLI .

`get-distribution-tenant`

Il seguente esempio di codice mostra come utilizzare`get-distribution-tenant`.

AWS CLI

Per ottenere dettagli su un tenant CloudFront di distribuzione

L'`get-distribution-tenant`esempio seguente recupera informazioni su un tenant di CloudFront distribuzione.

```
aws cloudfront get-distribution-tenant \
    --id dt_2wjDZi3hD1ivOXf6rpZJOSNE1AB
```

Output:

```
{
    "ETag": "E23ZP02F085ABC",
    "DistributionTenant": {
        "Id": "dt_2wjDZi3hD1ivOXf6rpZJOSNE1AB",
        "DistributionId": "E1XNX8R2GOAABC",
        "Name": "example-tenant-2",
        "Arn": "arn:aws:cloudfront::123456789012:distribution-tenant/
dt_2wjDZi3hD1ivOXf6rpZJOSNE1AB",
        "Domains": [
            {
                "Domain": "example.com",
                "Status": "inactive"
            }
        ],
        "ConnectionGroupId": "cg_2wjDWTBKTlRB87cAaUQFaakABC",
        "CreatedTime": "2025-05-06T15:42:28.542000+00:00",
        "LastModifiedTime": "2025-05-06T15:42:37.724000+00:00",
        "Enabled": true,
        "Status": "InProgress"
    }
}
```

Per ulteriori informazioni, consulta Comprendi come funzionano le distribuzioni multi-tenant nella Amazon CloudFront Developer Guide.

- Per i dettagli sull'API, consulta Command GetDistributionTenantReference AWS CLI .

`get-distribution`

Il seguente esempio di codice mostra come utilizzare`get-distribution`.

AWS CLI

Per ottenere una CloudFront distribuzione

L'`get-distribution`esempio seguente ottiene la CloudFront distribuzione con l'ID`EDFDVBD6EXAMPLE`, incluso il relativo`ETag`. L'ID distribuzione viene restituito nei comandi create-distribution e list-distribution.

```
aws cloudfront get-distribution \
    --id EDFDVBD6EXAMPLE
```

Output:

```
{
    "ETag": "E2QWRUHEXAMPLE",
    "Distribution": {
        "Id": "EDFDVBD6EXAMPLE",
        "ARN": "arn:aws:cloudfront::123456789012:distribution/EDFDVBD6EXAMPLE",
        "Status": "Deployed",
        "LastModifiedTime": "2019-12-04T23:35:41.433Z",
        "InProgressInvalidationBatches": 0,
        "DomainName": "d111111abcdef8.cloudfront.net",
        "ActiveTrustedSigners": {
            "Enabled": false,
            "Quantity": 0
        },
        "DistributionConfig": {
            "CallerReference": "cli-example",
            "Aliases": {
                "Quantity": 0
            },
            "DefaultRootObject": "index.html",
            "Origins": {
                "Quantity": 1,
                "Items": [
                    {
                        "Id": "amzn-s3-demo-bucket.s3.amazonaws.com-cli-example",
                        "DomainName": "amzn-s3-demo-bucket.s3.amazonaws.com",
                        "OriginPath": "",
                        "CustomHeaders": {
                            "Quantity": 0
                        },
                        "S3OriginConfig": {
```

```
                        "OriginAccessIdentity": ""
                    }
                }
            ]
        },
        "OriginGroups": {
            "Quantity": 0
        },
        "DefaultCacheBehavior": {
            "TargetOriginId": "amzn-s3-demo-bucket.s3.amazonaws.com-cli-
example",
            "ForwardedValues": {
                "QueryString": false,
                "Cookies": {
                    "Forward": "none"
                },
                "Headers": {
                    "Quantity": 0
                },
                "QueryStringCacheKeys": {
                    "Quantity": 0
                }
            },
            "TrustedSigners": {
                "Enabled": false,
                "Quantity": 0
            },
            "ViewerProtocolPolicy": "allow-all",
            "MinTTL": 0,
            "AllowedMethods": {
                "Quantity": 2,
                "Items": [
                    "HEAD",
                    "GET"
                ],
                "CachedMethods": {
                    "Quantity": 2,
                    "Items": [
                        "HEAD",
                        "GET"
                    ]
                }
            },
            "SmoothStreaming": false,
```

```
                "DefaultTTL": 86400,
                "MaxTTL": 31536000,
                "Compress": false,
                "LambdaFunctionAssociations": {
                    "Quantity": 0
                },
                "FieldLevelEncryptionId": ""
            },
            "CacheBehaviors": {
                "Quantity": 0
            },
            "CustomErrorResponses": {
                "Quantity": 0
            },
            "Comment": "",
            "Logging": {
                "Enabled": false,
                "IncludeCookies": false,
                "Bucket": "",
                "Prefix": ""
            },
            "PriceClass": "PriceClass_All",
            "Enabled": true,
            "ViewerCertificate": {
                "CloudFrontDefaultCertificate": true,
                "MinimumProtocolVersion": "TLSv1",
                "CertificateSource": "cloudfront"
            },
            "Restrictions": {
                "GeoRestriction": {
                    "RestrictionType": "none",
                    "Quantity": 0
                }
            },
            "WebACLId": "",
            "HttpVersion": "http2",
            "IsIPV6Enabled": true
        }
    }
}
```

- Per i dettagli sull'API, consulta GetDistribution AWS CLICommand Reference.

`get-field-level-encryption-config`

Il seguente esempio di codice mostra come utilizzare`get-field-level-encryption-config`.

AWS CLI

Per ottenere i metadati relativi a una configurazione di crittografia a CloudFront livello di campo

L'esempio seguente ottiene i metadati relativi alla configurazione di crittografia a CloudFront livello di campo con l'ID, inclusi: `C3KM2WVD605UAY` `ETag`

```
aws cloudfront get-field-level-encryption-config --id C3KM2WVD605UAY
```

Output:

```
{
    "ETag": "E2P4Z4VU7TY5SG",
    "FieldLevelEncryptionConfig": {
        "CallerReference": "cli-example",
        "Comment": "Example FLE configuration",
        "QueryArgProfileConfig": {
            "ForwardWhenQueryArgProfileIsUnknown": true,
            "QueryArgProfiles": {
                "Quantity": 0,
                "Items": []
            }
        },
        "ContentTypeProfileConfig": {
            "ForwardWhenContentTypeIsUnknown": true,
            "ContentTypeProfiles": {
                "Quantity": 1,
                "Items": [
                    {
                        "Format": "URLEncoded",
                        "ProfileId": "P280MFCLSYOCVU",
                        "ContentType": "application/x-www-form-urlencoded"
                    }
                ]
            }
        }
    }
}
```

- Per i dettagli sull'API, consulta Command GetFieldLevelEncryptionConfigReference AWS CLI .

`get-field-level-encryption-profile-config`

Il seguente esempio di codice mostra come utilizzare`get-field-level-encryption-profile-config`.

AWS CLI

Per ottenere una configurazione del profilo di crittografia CloudFront a livello di campo

L'esempio seguente ottiene i metadati relativi al profilo di crittografia a CloudFront livello di campo con ID, inclusi: `PPK0UOSIF5WSV` `ETag`

```
aws cloudfront get-field-level-encryption-profile-config --id PPK0UOSIF5WSV
```

Output:

```
{
    "ETag": "E1QQG65FS2L2GC",
    "FieldLevelEncryptionProfileConfig": {
        "Name": "ExampleFLEProfile",
        "CallerReference": "cli-example",
        "Comment": "FLE profile for AWS CLI example",
        "EncryptionEntities": {
            "Quantity": 1,
            "Items": [
                {
                    "PublicKeyId": "K2K8NC4HVFE3M0",
                    "ProviderId": "ExampleFLEProvider",
                    "FieldPatterns": {
                        "Quantity": 1,
                        "Items": [
                            "ExampleSensitiveField"
                        ]
                    }
                }
            ]
        }
    }
}
```

- Per i dettagli sull'API, consulta Command GetFieldLevelEncryptionProfileConfigReference AWS CLI .

`get-field-level-encryption-profile`

Il seguente esempio di codice mostra come utilizzare`get-field-level-encryption-profile`.

AWS CLI

Per ottenere un profilo di CloudFront crittografia a livello di campo

L'esempio seguente ottiene il profilo di crittografia a CloudFront livello di campo con IDPPK0UOSIF5WSV, incluso il relativo: ETag

```
aws cloudfront get-field-level-encryption-profile --id PPK0UOSIF5WSV
```

Output:

```
{
    "ETag": "E1QQG65FS2L2GC",
    "FieldLevelEncryptionProfile": {
        "Id": "PPK0UOSIF5WSV",
        "LastModifiedTime": "2019-12-10T01:03:16.537Z",
        "FieldLevelEncryptionProfileConfig": {
            "Name": "ExampleFLEProfile",
            "CallerReference": "cli-example",
            "Comment": "FLE profile for AWS CLI example",
            "EncryptionEntities": {
                "Quantity": 1,
                "Items": [
                    {
                        "PublicKeyId": "K2K8NC4HVFE3M0",
                        "ProviderId": "ExampleFLEProvider",
                        "FieldPatterns": {
                            "Quantity": 1,
                            "Items": [
                                "ExampleSensitiveField"
                            ]
                        }
                    }
                ]
            }
```

```
        }
    }
}
```

- Per i dettagli sull'API, consulta AWS CLI Command GetFieldLevelEncryptionProfileReference.

`get-field-level-encryption`

Il seguente esempio di codice mostra come utilizzare`get-field-level-encryption`.

AWS CLI

Per ottenere una configurazione di CloudFront crittografia a livello di campo

L'esempio seguente ottiene la configurazione di crittografia a CloudFront livello di campo con l'ID, inclusa la relativa: `C3KM2WVD605UAY` `ETag`

```
aws cloudfront get-field-level-encryption --id C3KM2WVD605UAY
```

Output:

```
{
    "ETag": "E2P4Z4VU7TY5SG",
    "FieldLevelEncryption": {
        "Id": "C3KM2WVD605UAY",
        "LastModifiedTime": "2019-12-10T21:30:18.974Z",
        "FieldLevelEncryptionConfig": {
            "CallerReference": "cli-example",
            "Comment": "Example FLE configuration",
            "QueryArgProfileConfig": {
                "ForwardWhenQueryArgProfileIsUnknown": true,
                "QueryArgProfiles": {
                    "Quantity": 0,
                    "Items": []
                }
            },
            "ContentTypeProfileConfig": {
                "ForwardWhenContentTypeIsUnknown": true,
                "ContentTypeProfiles": {
                    "Quantity": 1,
                    "Items": [
                        {
```

```
                            "Format": "URLEncoded",
                            "ProfileId": "P280MFCLSYOCVU",
                            "ContentType": "application/x-www-form-urlencoded"
                        }
                    ]
                }
            }
        }
    }
}
```

- Per i dettagli sull'API, consulta AWS CLI Command GetFieldLevelEncryptionReference.

`get-invalidation-for-distribution-tenant`

Il seguente esempio di codice mostra come utilizzare`get-invalidation-for-distribution-tenant`.

AWS CLI

Come ottenere un'invalidazione per un tenant di distribuzione

L'esempio `get-invalidation-for-distribution-tenant` seguente ottiene informazioni su un'invalidazione per un tenant di distribuzione.

```
aws cloudfront get-invalidation-for-distribution-tenant \
    --distribution-tenant-id dt_2wjDZi3hD1ivOXf6rpZJOSNE1AB \
    --id I4CU23QAPKMUDUU06F9OFGFABC
```

Output:

```
{
    "Invalidation": {
        "Id": "I4CU23QAPKMUDUU06F9OFGFABC",
        "Status": "Completed",
        "CreateTime": "2025-05-06T15:46:12.824000+00:00",
        "InvalidationBatch": {
            "Paths": {
                "Quantity": 2,
                "Items": [
                    "/example/invalidation",
```

```
                    "/more/invalidations"
                ]
            },
            "CallerReference": "007ee5a6-d0a0-42be-bb61-e7b915969b48"
        }
    }
}
```

Per ulteriori informazioni, consulta Invalidare i file per rimuovere contenuti nell'Amazon CloudFront Developer Guide.

- Per i dettagli sull'API, consulta AWS CLI Command GetInvalidationForDistributionTenantReference.

`get-invalidation`

Il seguente esempio di codice mostra come utilizzare`get-invalidation`.

AWS CLI

Per ottenere un' CloudFront invalidazione

L'esempio seguente ottiene l'invalidazione con l'ID `I2J0I21PCUYOIK` per la CloudFront distribuzione con l'ID: `EDFDVBD6EXAMPLE`

```
aws cloudfront get-invalidation --id I2J0I21PCUYOIK --distribution-
id EDFDVBD6EXAMPLE
```

Output:

```
{
    "Invalidation": {
        "Status": "Completed",
        "InvalidationBatch": {
            "Paths": {
                "Items": [
                    "/example-path/example-file.jpg",
                    "/example-path/example-file-2.jpg"
                ],
                "Quantity": 2
            },
```

```
            "CallerReference": "cli-example"
        },
        "Id": "I2J0I21PCUYOIK",
        "CreateTime": "2019-12-05T18:40:49.413Z"
    }
}
```

- Per i dettagli sull'API, consulta AWS CLI Command GetInvalidationReference.

`get-managed-certificate-details`

Il seguente esempio di codice mostra come utilizzare`get-managed-certificate-details`.

AWS CLI

Come ottenere i dettagli dei certificati gestiti

L'`get-managed-certificate-details`esempio seguente recupera i dettagli di un certificato ACM CloudFront gestito.

```
aws cloudfront get-managed-certificate-details \
    --identifier dt_2wjDZi3hD1ivOXf6rpZJOSNE1AB
```

Output:

```
{
    "ManagedCertificateDetails": {
        "CertificateArn": "arn:aws:acm:us-
east-1:123456789012:certificate/655dc1fe-6d37-451d-a013-c2db3a034abc",
        "CertificateStatus": "pending-validation",
        "ValidationTokenHost": "self-hosted",
        "ValidationTokenDetails": [
            {
                "Domain": "example.com",
                "RedirectTo": "validation.us-east-1.acm-
validations.aws/123456789012/.well-known/pki-validation/
b315c9ae21284e7918bb9f3f422ab1c7.txt",
                "RedirectFrom": "example.com/.well-known/pki-validation/
b315c9ae21284e7918bb9f3f422ac3c7.txt"
            }
        ]
    }
```

```
}
```

Per ulteriori informazioni, consulta Richiedi certificati per il tuo tenant di CloudFront distribuzione nell'Amazon CloudFront Developer Guide.

- Per i dettagli sull'API, consulta GetManagedCertificateDetails AWS CLICommand Reference.

`get-public-key-config`

Il seguente esempio di codice mostra come utilizzare`get-public-key-config`.

AWS CLI

Per ottenere una configurazione a chiave CloudFront pubblica

L'esempio seguente ottiene i metadati sulla chiave CloudFront pubblica con l'ID`KDFB19YGCR002`, incluso il relativo`ETag`. L'ID della chiave pubblica viene restituito nei list-public-keys comandi create-public-key and.

```
aws cloudfront get-public-key-config --id KDFB19YGCR002
```

Output:

```
{
    "ETag": "E2QWRUHEXAMPLE",
    "PublicKeyConfig": {
        "CallerReference": "cli-example",
        "Name": "ExampleKey",
        "EncodedKey": "-----BEGIN PUBLIC KEY-----
\nMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAxPMbCA2Ks0lnd7IR+3pw
\nwd3H/7jPGwj8bLUmore7bX+oeGpZ6QmLAe/1UOWcmZX2u70dYcSIzB1ofZtcn4cJ
\nenHBAzO3ohBY/L1tQGJfS2A+omnN6H16VZE1JCK8XSJyfze7MDLcUyHZETdxuvRb
\nA9X343/vMAuQPnhinFJ8Wdy8YBXSPpy7r95ylUQd9LfYTBzVZYG2tSesplcOkjM3\n2Uu
+oMWxQAw1NINnSLPinMVsutJy6ZqlV3McWNWe4T+STGtWhrPNqJEn45sIcCx4\nq
+kGZ2NQ0FyIyT2eiLKOX5Rgb/a36E/aMk4VoDsaenBQgG7WLTnstb9sr7MIhS6A\nrwIDAQAB\n-----END
 PUBLIC KEY-----\n",
        "Comment": "example public key"
    }
}
```

- Per i dettagli sull'API, consulta GetPublicKeyConfig AWS CLICommand Reference.

`get-public-key`

Il seguente esempio di codice mostra come utilizzare`get-public-key`.

AWS CLI

Per ottenere una chiave CloudFront pubblica

L'esempio seguente ottiene la chiave CloudFront pubblica con l'IDKDFB19YGCR002, incluso il relativoETag. L'ID della chiave pubblica viene restituito nei list-public-keys comandi create-public-key and.

```
aws cloudfront get-public-key --id KDFB19YGCR002
```

Output:

```
{
    "ETag": "E2QWRUHEXAMPLE",
    "PublicKey": {
        "Id": "KDFB19YGCR002",
        "CreatedTime": "2019-12-05T18:51:43.781Z",
        "PublicKeyConfig": {
            "CallerReference": "cli-example",
            "Name": "ExampleKey",
            "EncodedKey": "-----BEGIN PUBLIC KEY-----
\nMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAxPMbCA2Ks0lnd7IR+3pw
\nwd3H/7jPGwj8bLUmore7bX+oeGpZ6QmLAe/1UOWcmZX2u70dYcSIzB1ofZtcn4cJ
\nenHBAzO3ohBY/L1tQGJfS2A+omnN6H16VZE1JCK8XSJyfze7MDLcUyHZETdxuvRb
\nA9X343/vMAuQPnhinFJ8Wdy8YBXSPpy7r95ylUQd9LfYTBzVZYG2tSesplcOkjM3\n2Uu
+oMWxQAw1NINnSLPinMVsutJy6ZqlV3McWNWe4T+STGtWhrPNqJEn45sIcCx4\nq
+kGZ2NQ0FyIyT2eiLKOX5Rgb/a36E/aMk4VoDsaenBQgG7WLTnstb9sr7MIhS6A\nrwIDAQAB\n-----END
 PUBLIC KEY-----\n",
            "Comment": "example public key"
        }
    }
}
```

- Per i dettagli sull'API, consulta GetPublicKey AWS CLICommand Reference.

`list-cloud-front-origin-access-identities`

Il seguente esempio di codice mostra come utilizzare`list-cloud-front-origin-access-identities`.

AWS CLI

Per elencare le identità di accesso all' CloudFront origine

L'esempio seguente ottiene un elenco delle identità di accesso di CloudFront origine (OAIs) presenti nel tuo AWS account:

```
aws cloudfront list-cloud-front-origin-access-identities
```

Output:

```
{
    "CloudFrontOriginAccessIdentityList": {
        "Items": [
            {
                "Id": "E74FTE3AEXAMPLE",
                "S3CanonicalUserId":
 "cd13868f797c227fbea2830611a26fe0a21ba1b826ab4bed9b7771c9aEXAMPLE",
                "Comment": "Example OAI"
            },
            {
                "Id": "EH1HDMBEXAMPLE",
                "S3CanonicalUserId":
 "1489f6f2e6faacaae7ff64c4c3e6956c24f78788abfc1718c3527c263bf7a17EXAMPLE",
                "Comment": "Test OAI"
            },
            {
                "Id": "E2X2C9TEXAMPLE",
                "S3CanonicalUserId":
 "cbfeebb915a64749f9be546a45b3fcfd3a31c779673c13c4dd460911ae402c2EXAMPLE",
                "Comment": "Example OAI #2"
            }
        ]
    }
}
```

- Per i dettagli sull'API, consulta ListCloudFrontOriginAccessIdentities AWS CLICommand Reference.

`list-connection-groups`

Il seguente esempio di codice mostra come utilizzare`list-connection-groups`.

AWS CLI

Come elencare i gruppi di connessioni

L'`list-connection-groups`esempio seguente elenca i gruppi di connessione disponibili nell' AWS account.

```
aws cloudfront list-connection-groups
```

Output:

```
{
    "ConnectionGroups": [
        {
            "Id": "cg_2whCJoXMYCjHcxaLGrkllvyABC",
            "Name": "CreatedByCloudFront-cg_2whCJoXMYCjHcxaLGrkllvyABC",
            "Arn": "arn:aws:cloudfront::123456789012:connection-group/
cg_2whCJoXMYCjHcxaLGrkllvyABC",
            "RoutingEndpoint": "d3sx0pso7m5abc.cloudfront.net",
            "CreatedTime": "2025-05-05T22:32:29.630000+00:00",
            "LastModifiedTime": "2025-05-05T22:32:29.630000+00:00",
            "ETag": "E23ZP02F085ABC",
            "Enabled": true,
            "Status": "Deployed",
            "IsDefault": true
        },
        {
            "Id": "cg_2wjDWTBKTlRB87cAaUQFaakABC",
            "Name": "connection-group-2",
            "Arn": "arn:aws:cloudfront::123456789012:connection-group/
cg_2wjDWTBKTlRB87cAaUQFaakABC",
            "RoutingEndpoint": "dvdg9gprgabc.cloudfront.net",
            "CreatedTime": "2025-05-06T15:42:00.790000+00:00",
            "LastModifiedTime": "2025-05-06T15:42:00.790000+00:00",
            "ETag": "E23ZP02F085ABC",
            "Enabled": true,
            "Status": "Deployed",
            "IsDefault": false
        }
```

```
    ]
}
```

Per ulteriori informazioni, consulta Creare un gruppo di connessione personalizzato (opzionale) nella Amazon CloudFront Developer Guide.

- Per i dettagli sull'API, consulta ListConnectionGroups AWS CLICommand Reference.

`list-distribution-tenants-by-customization`

Il seguente esempio di codice mostra come utilizzare`list-distribution-tenants-by-customization`.

AWS CLI

Come elencare i tenant di distribuzione in base alla personalizzazione

L'esempio `list-distribution-tenants-by-customization` seguente elenca i tenant di distribuzione che utilizzano la lista di controllo degli accessi (ACL) web specificata.

```
aws cloudfront list-distribution-tenants-by-customization \
    --web-acl-arn arn:aws:wafv2:us-east-1:123456789012:global/webacl/
CreatedByCloudFront-0273cd2f/a3c19bce-42b5-48a1-a8d4-b2bb2f28eabc
```

Output:

```
{
    "DistributionTenantList": [
        {
            "Id": "dt_2wjDZi3hD1ivOXf6rpZJOSNE1AB",
            "DistributionId": "E1XNX8R2GOAABC",
            "Name": "example-tenant-2",
            "Arn": "arn:aws:cloudfront::123456789012:distribution-tenant/
dt_2wjDZi3hD1ivOXf6rpZJOSNE1AB",
            "Domains": [
                {
                    "Domain": "example.com",
                    "Status": "inactive"
                }
            ],
            "ConnectionGroupId": "cg_2wjDWTBKTlRB87cAaUQFaakABC",
```

```
            "Customizations": {
                "WebAcl": {
                    "Action": "override",
                    "Arn": "arn:aws:wafv2:us-east-1:123456789012:global/webacl/
CreatedByCloudFront-0273cd2f/a3c19bce-42b5-48a1-a8d4-b2bb2f28eabc"
                },
                "GeoRestrictions": {
                    "RestrictionType": "whitelist",
                    "Locations": [
                        "AL"
                    ]
                }
            },
            "CreatedTime": "2025-05-06T15:42:28.542000+00:00",
            "LastModifiedTime": "2025-05-06T16:14:08.710000+00:00",
            "ETag": "E1F83G8C2ARABC",
            "Enabled": true,
            "Status": "Deployed"
        }
    ]
}
```

Per ulteriori informazioni, consulta Personalizzazioni dei tenant di distribuzione nella Amazon CloudFront Developer Guide.

- Per i dettagli sull'API, consulta AWS CLI Command ListDistributionTenantsByCustomizationReference.

`list-distribution-tenants`

Il seguente esempio di codice mostra come utilizzare`list-distribution-tenants`.

AWS CLI

Per elencare gli inquilini CloudFront di distribuzione

L'`list-distribution-tenants`esempio seguente elenca i 3 tenant di CloudFront distribuzione presenti nell' AWS account in base al gruppo di connessione associato.

```
aws cloudfront list-distribution-tenants \
    --association-filter ConnectionGroupId=cg_2whCJoXMYCjHcxaLGrkllvyABC \
    --max-items 3
```

Output:

```
{
    "DistributionTenantList": [
        {
            "Id": "dt_2yMvQgam3QkJo2z54FDl91dk1AB",
            "DistributionId": "E1XNX8R2GOAABC",
            "Name": "new-tenant-customizations",
            "Arn": "arn:aws:cloudfront::123456789012:distribution-tenant/
dt_2yMvQgam3QkJo2z54FDl91dk1AB",
            "Domains": [
                {
                    "Domain": "example.com",
                    "Status": "active"
                }
            ],
            "ConnectionGroupId": "cg_2whCJoXMYCjHcxaLGrkllvyABC",
            "Customizations": {
                "WebAcl": {
                    "Action": "disable"
                },
                "GeoRestrictions": {
                    "RestrictionType": "whitelist",
                    "Locations": [
                        "DE"
                    ]
                }
            },
            "CreatedTime": "2025-06-11T15:54:02.142000+00:00",
            "LastModifiedTime": "2025-06-11T15:54:02.142000+00:00",
            "ETag": "E23ZP02F085ABC",
            "Enabled": false,
            "Status": "Deployed"
        },
        {
            "Id": "dt_2yMuV7NJuBcAB0cwwxMCBZQ1AB",
            "DistributionId": "E1XNX8R2GOAABC",
            "Name": "new-tenant",
            "Arn": "arn:aws:cloudfront::123456789012:distribution-tenant/
dt_2yMuV7NJuBcAB0cwwxMCBZQ1AB",
            "Domains": [
                {
                    "Domain": "1.example.com",
                    "Status": "active"
```

```
                }
            ],
            "ConnectionGroupId": "cg_2whCJoXMYCjHcxaLGrkllvyABC",
            "Customizations": {
                "GeoRestrictions": {
                    "RestrictionType": "whitelist",
                    "Locations": [
                        "DE"
                    ]
                }
            },
            "CreatedTime": "2025-06-11T15:46:23.466000+00:00",
            "LastModifiedTime": "2025-06-11T15:46:23.466000+00:00",
            "ETag": "E23ZP02F085ABC",
            "Enabled": false,
            "Status": "Deployed"
        },
        {
            "Id": "dt_2xVInRKCfUzQHgxosDs9hiLk1AB",
            "DistributionId": "E1XNX8R2GOAABC",
            "Name": "new-tenant-2",
            "Arn": "arn:aws:cloudfront::123456789012:distribution-tenant/
dt_2xVInRKCfUzQHgxosDs9hiLk1AB",
            "Domains": [
                {
                    "Domain": "2.example.com",
                    "Status": "active"
                }
            ],
            "ConnectionGroupId": "cg_2whCJoXMYCjHcxaLGrkllvyABC",
            "CreatedTime": "2025-05-23T16:16:20.871000+00:00",
            "LastModifiedTime": "2025-05-23T16:16:20.871000+00:00",
            "ETag": "E23ZP02F085ABC",
            "Enabled": false,
            "Status": "Deployed"
        }
    ],
    "NextToken": "eyJNYXJrZXIiOiBudWxsLCAiYm90b190cnVuY2F0ZV9hbW91bnQiOiAzfQ=="
}
```

Per ulteriori informazioni, consulta Comprendi come funzionano le distribuzioni multi-tenant nella Amazon CloudFront Developer Guide.

- Per i dettagli sull'API, consulta Command ListDistributionTenantsReference AWS CLI .

`list-distributions-by-connection-mode`

Il seguente esempio di codice mostra come utilizzare`list-distributions-by-connection-mode`.

AWS CLI

Per elencare CloudFront le distribuzioni in base alla modalità di connessione

L'`list-distributions-by-connection-mode`esempio seguente elenca le CloudFront distribuzioni con la modalità di connessione specificata.

```
aws cloudfront list-distributions-by-connection-mode \
    --connection-mode tenant-only
```

Output:

```
{
    "DistributionList": {
        "Items": [
            {
                "Id": "E1XNX8R2GOAABC",
                "ARN": "arn:aws:cloudfront::123456789012:distribution/
E1XNX8R2GOAABC",
                "ETag": "EPT4JPJQDY1ABC",
                "Status": "Deployed",
                "LastModifiedTime": "2025-05-23T16:16:15.691000+00:00",
                "DomainName": "-",
                "Aliases": {
                    "Quantity": 0
                },
                "Origins": {
                    "Quantity": 1,
                    "Items": [
                        {
                            "Id": "example-cfn-simple-distribution123",
                            "DomainName": "example.com",
                            "OriginPath": "",
                            "CustomHeaders": {
                                "Quantity": 0
                            },
                            "S3OriginConfig": {
                                "OriginAccessIdentity": ""
```

```
                        },
                        "ConnectionAttempts": 3,
                        "ConnectionTimeout": 10,
                        "OriginShield": {
                            "Enabled": false
                        },
                        "OriginAccessControlId": "E2CJRMB5LKEABC"
                    }
                ]
            },
            "OriginGroups": {
                "Quantity": 0
            },
            "DefaultCacheBehavior": {
                "TargetOriginId": "example-cfn-simple-distribution123",
                "TrustedKeyGroups": {
                    "Enabled": false,
                    "Quantity": 0
                },
                "ViewerProtocolPolicy": "allow-all",
                "AllowedMethods": {
                    "Quantity": 2,
                    "Items": [
                        "HEAD",
                        "GET"
                    ],
                    "CachedMethods": {
                        "Quantity": 2,
                        "Items": [
                            "HEAD",
                            "GET"
                        ]
                    }
                },
                "Compress": true,
                "LambdaFunctionAssociations": {
                    "Quantity": 0
                },
                "FunctionAssociations": {
                    "Quantity": 0
                },
                "FieldLevelEncryptionId": "",
                "CachePolicyId": "658327ea-f89d-4fab-a63d-7e88639e5abc",
                "GrpcConfig": {
```

```
                        "Enabled": false
                    }
                },
                "CacheBehaviors": {
                    "Quantity": 0
                },
                "CustomErrorResponses": {
                    "Quantity": 0
                },
                "Comment": "",
                "PriceClass": "PriceClass_All",
                "Enabled": true,
                "ViewerCertificate": {
                    "CloudFrontDefaultCertificate": false,
                    "ACMCertificateArn": "arn:aws:acm:us-
east-1:123456789012:certificate/ec53f564-ea5a-4e4a-a0a2-e3c989449abc",
                    "SSLSupportMethod": "sni-only",
                    "MinimumProtocolVersion": "TLSv1.2_2021",
                    "Certificate": "arn:aws:acm:us-east-1:123456789012:certificate/
ec53f564-ea5a-4e4a-a0a2-e3c989449abc",
                    "CertificateSource": "acm"
                },
                "Restrictions": {
                    "GeoRestriction": {
                        "RestrictionType": "none",
                        "Quantity": 0
                    }
                },
                "WebACLId": "arn:aws:wafv2:us-east-1:123456789012:global/webacl/web-
global-example/626900da-5f64-418b-ba9b-743f3746cabc",
                "HttpVersion": "http2",
                "IsIPV6Enabled": false,
                "Staging": false,
                "ConnectionMode": "tenant-only"
            }
        ]
    }
}
```

Per ulteriori informazioni, consulta Creare un gruppo di connessione personalizzato (opzionale) nella Amazon CloudFront Developer Guide.

- Per i dettagli sull'API, consulta ListDistributionsByConnectionMode AWS CLICommand Reference.

`list-distributions`

Il seguente esempio di codice mostra come utilizzare`list-distributions`.

AWS CLI

Per elencare le CloudFront distribuzioni

L'esempio seguente ottiene un elenco delle CloudFront distribuzioni presenti nel tuo AWS account.

```
aws cloudfront list-distributions
```

Output:

```
{
    "DistributionList": {
        "Items": [
            {
                "Id": "E23YS8OEXAMPLE",
                "ARN": "arn:aws:cloudfront::123456789012:distribution/
E23YS8OEXAMPLE",
                "Status": "Deployed",
                "LastModifiedTime": "2024-08-05T18:23:40.375000+00:00",
                "DomainName": "abcdefgh12ijk.cloudfront.net",
                "Aliases": {
                    "Quantity": 0
                },
                "Origins": {
                    "Quantity": 1,
                    "Items": [
                        {
                            "Id": "amzn-s3-demo-bucket.s3.us-east-1.amazonaws.com",
                            "DomainName": "amzn-s3-demo-bucket.s3.us-
east-1.amazonaws.com",
                            "OriginPath": "",
                            "CustomHeaders": {
                                "Quantity": 0
                            },
                            "S3OriginConfig": {
                                "OriginAccessIdentity": ""
                            },
                            "ConnectionAttempts": 3,
```

```
                            "ConnectionTimeout": 10,
                            "OriginShield": {
                                "Enabled": false
                            },
                            "OriginAccessControlId": "EIAP8PEXAMPLE"
                        }
                    ]
                },
                "OriginGroups": {
                    "Quantity": 0
                },
                "DefaultCacheBehavior": {
                    "TargetOriginId": "amzn-s3-demo-bucket.s3.us-
east-1.amazonaws.com",
                    "TrustedSigners": {
                        "Enabled": false,
                        "Quantity": 0
                    },
                    "TrustedKeyGroups": {
                        "Enabled": false,
                        "Quantity": 0
                    },
                    "ViewerProtocolPolicy": "allow-all",
                    "AllowedMethods": {
                        "Quantity": 2,
                        "Items": [
                            "HEAD",
                            "GET"
                        ],
                        "CachedMethods": {
                            "Quantity": 2,
                            "Items": [
                                "HEAD",
                                "GET"
                            ]
                        }
                    },
                    "SmoothStreaming": false,
                    "Compress": true,
                    "LambdaFunctionAssociations": {
                        "Quantity": 0
                    },
                    "FunctionAssociations": {
                        "Quantity": 0
```

```
                    },
                    "FieldLevelEncryptionId": "",
                    "CachePolicyId": "658327ea-f89d-4fab-a63d-7e886EXAMPLE"
                },
                "CacheBehaviors": {
                    "Quantity": 0
                },
                "CustomErrorResponses": {
                    "Quantity": 0
                },
                "Comment": "",
                "PriceClass": "PriceClass_All",
                "Enabled": true,
                "ViewerCertificate": {
                    "CloudFrontDefaultCertificate": true,
                    "SSLSupportMethod": "vip",
                    "MinimumProtocolVersion": "TLSv1",
                    "CertificateSource": "cloudfront"
                },
                "Restrictions": {
                    "GeoRestriction": {
                        "RestrictionType": "none",
                        "Quantity": 0
                    }
                },
                "WebACLId": "",
                "HttpVersion": "HTTP2",
                "IsIPV6Enabled": true,
                "Staging": false
            }
        ]
    }
}
```

- Per i dettagli sull'API, consulta ListDistributions AWS CLICommand Reference.

`list-domain-conflicts`

Il seguente esempio di codice mostra come utilizzare`list-domain-conflicts`.

AWS CLI

Per elencare i conflitti di dominio per una CloudFront distribuzione

L'`list-domain-conflicts`esempio seguente elenca i conflitti di dominio per una CloudFront distribuzione.

```
aws cloudfront list-domain-conflicts \
    --domain example.com \
    --domain-control-validation-
resource "DistributionTenantId=dt_2x9GhoK0TZRsohWzv1b9It8J1AB"
```

Output:

```
{
    "DomainConflicts": [
        {
            "Domain": "example.com",
            "ResourceType": "distribution-tenant",
            "ResourceId": "***************ohWzv1b9It8J1AB",
            "AccountId": "123456789012"
        }
    ]
}
```

Per ulteriori informazioni, consulta Spostare un nome di dominio alternativo in una distribuzione diversa nella Amazon CloudFront Developer Guide.

- Per i dettagli sull'API, consulta ListDomainConflicts AWS CLICommand Reference.

`list-field-level-encryption-configs`

Il seguente esempio di codice mostra come utilizzare`list-field-level-encryption-configs`.

AWS CLI

Per elencare le configurazioni di CloudFront crittografia a livello di campo

L'esempio seguente ottiene un elenco delle configurazioni di crittografia a CloudFront livello di campo presenti nell'account: AWS

```
aws cloudfront list-field-level-encryption-configs
```

Output:

```
{
```

```
    "FieldLevelEncryptionList": {
        "MaxItems": 100,
        "Quantity": 1,
        "Items": [
            {
                "Id": "C3KM2WVD605UAY",
                "LastModifiedTime": "2019-12-10T21:30:18.974Z",
                "Comment": "Example FLE configuration",
                "QueryArgProfileConfig": {
                    "ForwardWhenQueryArgProfileIsUnknown": true,
                    "QueryArgProfiles": {
                        "Quantity": 0,
                        "Items": []
                    }
                },
                "ContentTypeProfileConfig": {
                    "ForwardWhenContentTypeIsUnknown": true,
                    "ContentTypeProfiles": {
                        "Quantity": 1,
                        "Items": [
                            {
                                "Format": "URLEncoded",
                                "ProfileId": "P280MFCLSYOCVU",
                                "ContentType": "application/x-www-form-urlencoded"
                            }
                        ]
                    }
                }
            }
        ]
    }
}
```

- Per i dettagli sull'API, consulta Command ListFieldLevelEncryptionConfigsReference AWS CLI .

`list-field-level-encryption-profiles`

Il seguente esempio di codice mostra come utilizzare`list-field-level-encryption-profiles`.

AWS CLI

Per elencare i profili di CloudFront crittografia a livello di campo

L'esempio seguente ottiene un elenco dei profili di crittografia a CloudFront livello di campo presenti nell'account: AWS

```
aws cloudfront list-field-level-encryption-profiles
```

Output:

```
{
    "FieldLevelEncryptionProfileList": {
        "MaxItems": 100,
        "Quantity": 2,
        "Items": [
            {
                "Id": "P280MFCLSYOCVU",
                "LastModifiedTime": "2019-12-05T01:05:39.896Z",
                "Name": "ExampleFLEProfile",
                "EncryptionEntities": {
                    "Quantity": 1,
                    "Items": [
                        {
                            "PublicKeyId": "K2K8NC4HVFE3M0",
                            "ProviderId": "ExampleFLEProvider",
                            "FieldPatterns": {
                                "Quantity": 1,
                                "Items": [
                                    "ExampleSensitiveField"
                                ]
                            }
                        }
                    ]
                },
                "Comment": "FLE profile for AWS CLI example"
            },
            {
                "Id": "PPK0UOSIF5WSV",
                "LastModifiedTime": "2019-12-10T01:03:16.537Z",
                "Name": "ExampleFLEProfile2",
                "EncryptionEntities": {
                    "Quantity": 1,
                    "Items": [
                        {
                            "PublicKeyId": "K2ABC10EXAMPLE",
                            "ProviderId": "ExampleFLEProvider2",
```

```
                            "FieldPatterns": {
                                "Quantity": 1,
                                "Items": [
                                    "ExampleSensitiveField2"
                                ]
                            }
                        }
                    ]
                },
                "Comment": "FLE profile #2 for AWS CLI example"
            }
        ]
    }
}
```

- Per i dettagli sull'API, consulta AWS CLI Command ListFieldLevelEncryptionProfilesReference.

`list-invalidations-for-distribution-tenant`

Il seguente esempio di codice mostra come utilizzare`list-invalidations-for-distribution-tenant`.

AWS CLI

Per elencare le invalidazioni per un CloudFront tenant di distribuzione

L'`list-invalidations-for-distribution-tenant`esempio seguente elenca le invalidazioni per un tenant di distribuzione. CloudFront

```
aws cloudfront list-invalidations-for-distribution-tenant \
    --id dt_2wjDZi3hD1ivOXf6rpZJOSNE1AB
```

Output:

```
{
    "InvalidationList": {
        "Items": [
            {
                "Id": "I4CU23QAPKMUDUU06F9OFGFABC",
                "CreateTime": "2025-05-06T15:46:12.824000+00:00",
                "Status": "Completed"
            }
```

```
        ]
    }
}
```

Per ulteriori informazioni, consulta Invalidare i file per rimuovere contenuti nell'Amazon CloudFront Developer Guide.

- Per i dettagli sull'API, consulta AWS CLI Command ListInvalidationsForDistributionTenantReference.

`list-invalidations`

Il seguente esempio di codice mostra come utilizzare`list-invalidations`.

AWS CLI

Per elencare le CloudFront invalidazioni

L'esempio seguente ottiene un elenco delle invalidazioni per la CloudFront distribuzione con l'ID: `EDFDVBD6EXAMPLE`

```
aws cloudfront list-invalidations --distribution-id EDFDVBD6EXAMPLE
```

Output:

```
{
    "InvalidationList": {
        "Marker": "",
        "Items": [
            {
                "Status": "Completed",
                "Id": "YNY2LI2BVJ4NJU",
                "CreateTime": "2019-08-31T21:15:52.042Z"
            }
        ],
        "IsTruncated": false,
        "MaxItems": 100,
        "Quantity": 1
    }
}
```

- Per i dettagli sull'API, consulta AWS CLI Command ListInvalidationsReference.

`list-public-keys`

Il seguente esempio di codice mostra come utilizzare`list-public-keys`.

AWS CLI

Per elencare le chiavi CloudFront pubbliche

L'esempio seguente ottiene un elenco delle chiavi CloudFront pubbliche del tuo AWS account:

```
aws cloudfront list-public-keys
```

Output:

```
{
    "PublicKeyList": {
        "MaxItems": 100,
        "Quantity": 2,
        "Items": [
            {
                "Id": "K2K8NC4HVFE3M0",
                "Name": "ExampleKey",
                "CreatedTime": "2019-12-05T01:04:28.818Z",
                "EncodedKey": "-----BEGIN PUBLIC KEY-----
\nMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAxPMbCA2Ks0lnd7IR+3pw
\nwd3H/7jPGwj8bLUmore7bX+oeGpZ6QmLAe/1UOWcmZX2u70dYcSIzB1ofZtcn4cJ
\nenHBAzO3ohBY/L1tQGJfS2A+omnN6H16VZE1JCK8XSJyfze7MDLcUyHZETdxuvRb
\nA9X343/vMAuQPnhinFJ8Wdy8YBXSPpy7r95ylUQd9LfYTBzVZYG2tSesplcOkjM3\n2Uu
+oMWxQAw1NINnSLPinMVsutJy6ZqlV3McWNWe4T+STGtWhrPNqJEn45sIcCx4\nq
+kGZ2NQ0FyIyT2eiLKOX5Rgb/a36E/aMk4VoDsaenBQgG7WLTnstb9sr7MIhS6A\nrwIDAQAB\n-----END
 PUBLIC KEY-----\n",
                "Comment": "example public key"
            },
            {
                "Id": "K1S0LWQ2L5HTBU",
                "Name": "ExampleKey2",
                "CreatedTime": "2019-12-09T23:28:11.110Z",
                "EncodedKey": "-----BEGIN PUBLIC KEY-----
\nMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEApOCAg88A8+f4dujn9Izt
\n26LxtgAkn2opGgo/NKpMiaisyw5qlg3f1gol7FV6pYNl78iJg3EO8JBbwtlH
+cR9\nLGSf60NDeVhm76Oc39Np/vWgOdsGQcRbi9WmKZeSODqjQGzVZWqPmito3FzWVk6b
\nfVY5N36U/RdbVAJm95Km+qaMYlbIdF40t72bi3IkKYV5hlB2XoDjlQ9F6ajQKyTB
\nMHa3SN8q+3ZjQ4sJJ7D1V6r4wR8jDcFVD5NckWJmmgIVnkOQM37NYeoDnkaOuTpu\nha/
```

```
+3b8tOb2z3LBVHPkp85zJRAOXacSwf5rZtPYKBNFsixTa2n55k2r218mOkMC4\nUwIDAQAB\n-----END
 PUBLIC KEY-----",
                "Comment": "example public key #2"
            }
        ]
    }
}
```

- Per i dettagli sull'API, consulta ListPublicKeys AWS CLICommand Reference.

`list-tags-for-resource`

Il seguente esempio di codice mostra come utilizzare`list-tags-for-resource`.

AWS CLI

Per elencare i tag di una CloudFront distribuzione

L'esempio seguente ottiene un elenco dei tag per una CloudFront distribuzione:

```
aws cloudfront list-tags-for-resource \
    --resource arn:aws:cloudfront::123456789012:distribution/EDFDVBD6EXAMPLE
```

Output:

```
{
    "Tags": {
        "Items": [
            {
                "Key": "DateCreated",
                "Value": "2019-12-04"
            },
            {
                "Key": "Name",
                "Value": "Example name"
            },
            {
                "Key": "Project",
                "Value": "Example project"
            }
        ]
    }
```

```
}
```

- Per i dettagli sull'API, consulta ListTagsForResource AWS CLICommand Reference.

`sign`

Il seguente esempio di codice mostra come utilizzare`sign`.

AWS CLI

Per firmare un CloudFront URL

L'esempio seguente firma un CloudFront URL. Per firmare un URL, è necessario l'ID della coppia di chiavi (chiamato ID chiave di accesso nella console di AWS gestione) e la chiave privata della coppia di CloudFront chiavi del firmatario affidabile. Per ulteriori informazioni su signed URLs, consulta Serving Private Content with Signed URLs and Signed Cookies nella Amazon CloudFront Developer Guide.

```
aws cloudfront sign \
    --url https://d111111abcdef8.cloudfront.net/private-content/private-file.html \
    --key-pair-id APKAEIBAERJR2EXAMPLE \
    --private-key file://cf-signer-priv-key.pem \
    --date-less-than 2020-01-01
```

Output:

```
https://d111111abcdef8.cloudfront.net/private-content/private-
file.html?Expires=1577836800&Signature=nEXK7Kby47XKeZQKVc6pwkif6oZc-
JWSpDkH0UH7EBGGqvgurkecCbgL5VfUAXyLQuJxFwRQWscz-
owcq9KpmewCXrXQbPaJZNi9XSNwf4YKurPDQYaRQawKoeenH0GFteRf9ELK-
Bs3nljTLjtbgzIUt7QJNKXcWr8AuUYikzGdJ4-qzx6WnxXfH~fxg4-
GGl6l2kgCpXUB6Jx6K~Y3kpVOdzUPOIqFLHAnJojbhxqrVejomZZ2XrquDvNUCCIbePGnR3d24UPaLXG4FKOqNEaWDIB
GNvjRJxqWf93uMobeMOiVYahb-e0KItiQewGcm0eLZQ__&Key-Pair-Id=APKAEIBAERJR2EXAMPLE
```

- Per informazioni dettagliate sull'API, consulta Sign in AWS CLI Command Reference.

`tag-resource`

Il seguente esempio di codice mostra come usare`tag-resource`.

AWS CLI

Per etichettare una CloudFront distribuzione

L'`tag-resource`esempio seguente aggiunge due tag alla CloudFront distribuzione specificata.

```
aws cloudfront tag-resource \
    --resource arn:aws:cloudfront::123456789012:distribution/EDFDVBD6EXAMPLE \
    --tags 'Items=[{Key=Name,Value="Example name"},{Key=Project,Value="Example
 project"}]'
```

Anziché utilizzare gli argomenti della riga di comando, è possibile fornire i tag in un file JSON, come mostrato nell'esempio seguente:

```
aws cloudfront tag-resource \
    --resource arn:aws:cloudfront::123456789012:distribution/EDFDVBD6EXAMPLE \
    --tags file://tags.json
```

Contenuto di `tags.json`:

```
{
    "Items": [
        {
            "Key": "Name",
            "Value": "Example name"
        },
        {
            "Key": "Project",
            "Value": "Example project"
        }
    ]
}
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, vedere TagResourcein AWS CLI Command Reference.

`untag-resource`

Il seguente esempio di codice mostra come utilizzare`untag-resource`.

AWS CLI

Per rimuovere i tag da una CloudFront distribuzione

L'esempio seguente rimuove due tag da una CloudFront distribuzione utilizzando argomenti della riga di comando:

```
aws cloudfront untag-resource \
    --resource arn:aws:cloudfront::123456789012:distribution/EDFDVBD6EXAMPLE \
    --tag-keys Items=Name,Project
```

Anziché utilizzare gli argomenti della riga di comando, è possibile fornire le chiavi di tag in un file JSON, come mostrato nell'esempio seguente:

```
aws cloudfront untag-resource \
    --resource arn:aws:cloudfront::123456789012:distribution/EDFDVBD6EXAMPLE \
    --tag-keys file://tag-keys.json
```

Il file `tag-keys.json` è un documento JSON nella cartella corrente che contiene quanto segue:

```
{
    "Items": [
        "Name",
        "Project"
    ]
}
```

Se ha esito positivo, questo comando non produce alcun output.

- Per i dettagli sull'API, consulta UntagResource AWS CLICommand Reference.

`update-cloud-front-origin-access-identity`

Il seguente esempio di codice mostra come utilizzare`update-cloud-front-origin-access-identity`.

AWS CLI

Per aggiornare un'identità di accesso all' CloudFront origine

Il seguente esempio aggiorna l'identità di accesso origine (OAI) con l'ID `E74FTE3AEXAMPLE`. L'unico campo che è possibile aggiornare è il campo `Comment` dell'identità di accesso origine (OAI).

Per aggiornare un'identità di accesso origine (OAI), è necessario disporre dei relativi `ETag` e ID. L'ID OAI viene restituito nell'output dei comandi create-cloud-front-origin -access-identity e -access-identities. list-cloud-front-origin Per ottenere il, usa il comando -access-identity o `ETag` -. get-cloud-front-origin get-cloud-front-origin access-identity-config Utilizza l'opzione `--if-match` per fornire gli `ETag` dell'identità di accesso origine (OAI).

```
aws cloudfront update-cloud-front-origin-access-identity \
    --id E74FTE3AEXAMPLE \
    --if-match E2QWRUHEXAMPLE \
    --cloud-front-origin-access-identity-config \
        CallerReference=cli-example,Comment="Example OAI Updated"
```

È possibile ottenere lo stesso risultato indicando la configurazione OAI in un file JSON, come mostrato nell'esempio seguente:

```
aws cloudfront update-cloud-front-origin-access-identity \
    --id E74FTE3AEXAMPLE \
    --if-match E2QWRUHEXAMPLE \
    --cloud-front-origin-access-identity-config file://OAI-config.json
```

Il file `OAI-config.json` è un documento JSON nella directory corrente che contiene quanto segue:

```
{
    "CallerReference": "cli-example",
    "Comment": "Example OAI Updated"
}
```

Sia che si fornisca la configurazione OAI con un argomento della riga di comando o con un file JSON, l'output è lo stesso:

```
{
    "ETag": "E9LHASXEXAMPLE",
    "CloudFrontOriginAccessIdentity": {
        "Id": "E74FTE3AEXAMPLE",
```

```
        "S3CanonicalUserId":
 "cd13868f797c227fbea2830611a26fe0a21ba1b826ab4bed9b7771c9aEXAMPLE",
        "CloudFrontOriginAccessIdentityConfig": {
            "CallerReference": "cli-example",
            "Comment": "Example OAI Updated"
        }
    }
}
```

- Per i dettagli sull'API, consulta Command [UpdateCloudFrontOriginAccessIdentity](#)Reference AWS CLI .

`update-connection-group`

Il seguente esempio di codice mostra come utilizzare`update-connection-group`.

AWS CLI

Per aggiornare un gruppo di CloudFront connessione

L'`update-connection-group`esempio seguente disattiva e IPv6 disattiva un gruppo di CloudFront connessioni.

```
aws cloudfront update-connection-group \
    --id cg_2yHsDkcPKeUlVkk3aEgLKcjABC \
    --no-ipv6-enabled \
    --no-enabled \
    --if-match E3UN6WX5RRO2ABC
```

Output:

```
{
    "ETag": "E1F83G8C2ARABC",
    "ConnectionGroup": {
        "Id": "cg_2yHsDkcPKeUlVkk3aEgLKcjABC",
        "Name": "cg-example",
        "Arn": "arn:aws:cloudfront::123456789012:connection-group/
cg_2yHsDkcPKeUlVkk3aEgLKcjABC",
        "CreatedTime": "2025-06-09T20:58:35.481000+00:00",
        "LastModifiedTime": "2025-06-11T16:25:54.280000+00:00",
        "Ipv6Enabled": false,
        "RoutingEndpoint": "du9xp1elo1abc.cloudfront.net",
```

```
        "Status": "InProgress",
        "Enabled": false,
        "IsDefault": false
    }
}
```

Per ulteriori informazioni, consulta Creare un gruppo di connessione personalizzato (opzionale) nella Amazon CloudFront Developer Guide.

- Per i dettagli sull'API, consulta UpdateConnectionGroup AWS CLICommand Reference.

`update-distribution-tenant`

Il seguente esempio di codice mostra come utilizzare`update-distribution-tenant`.

AWS CLI

Per aggiornare un CloudFront tenant di distribuzione

L'`update-distribution-tenant`esempio seguente aggiorna un tenant di CloudFront distribuzione con un nuovo valore di parametro e aggiunge un paese alle restrizioni geografiche.

```
aws cloudfront update-distribution-tenant \
    --cli-input-json file://update-tenant.json
```

Contenuto di `update-tenant.json`:

```
{
    "Id": "dt_2yMvQgam3QkJo2z54FDl91dk1AB",
    "IfMatch": "E1F83G8C2ARABC",
    "Parameters": [
        {
            "Name": "testParam",
            "Value": "newParameterValue"
        }
    ],
    "Customizations": {
        "WebAcl": {
            "Action": "disable"
        },
        "GeoRestrictions": {
            "RestrictionType": "whitelist",
```

```
            "Locations": [
                "DE",
                "GB",
                "ES"
            ]
        }
    }
}
```

Output:

```
{
    "ETag": "E1PA6795UKMABC",
    "DistributionTenant": {
        "Id": "dt_2yMvQgam3QkJo2z54FDl91dk1AB",
        "DistributionId": "E1XNX8R2GOAABC",
        "Name": "new-tenant-customizations",
        "Arn": "arn:aws:cloudfront::123456789012:distribution-tenant/
dt_2yMvQgam3QkJo2z54FDl91dk1AB",
        "Domains": [
            {
                "Domain": "example.com",
                "Status": "active"
            }
        ],
        "Customizations": {
            "WebAcl": {
                "Action": "disable"
            },
            "GeoRestrictions": {
                "RestrictionType": "whitelist",
                "Locations": [
                    "DE",
                    "ES",
                    "GB"
                ]
            }
        },
        "Parameters": [
            {
                "Name": "testParam",
                "Value": "newParameterValue"
            }
```

```
        ],
        "ConnectionGroupId": "cg_2whCJoXMYCjHcxaLGrkllvyABC",
        "CreatedTime": "2025-06-11T15:54:02.142000+00:00",
        "LastModifiedTime": "2025-06-11T16:42:45.531000+00:00",
        "Enabled": false,
        "Status": "InProgress"
    }
}
```

Per ulteriori informazioni, consulta Personalizzazioni dei tenant di distribuzione nella Amazon CloudFront Developer Guide.

- Per i dettagli sull'API, consulta AWS CLI Command UpdateDistributionTenantReference.

`update-distribution`

Il seguente esempio di codice mostra come utilizzare`update-distribution`.

AWS CLI

Esempio 1: aggiornare l'oggetto radice predefinito di una CloudFront distribuzione

L'esempio seguente aggiorna l'oggetto radice predefinito `index.html` per la CloudFront distribuzione con l'ID`EDFDVBD6EXAMPLE`.

```
aws cloudfront update-distribution \
    --id EDFDVBD6EXAMPLE \
    --default-root-object index.html
```

Output:

```
{
    "ETag": "E2QWRUHEXAMPLE",
    "Distribution": {
        "Id": "EDFDVBD6EXAMPLE",
        "ARN": "arn:aws:cloudfront::123456789012:distribution/EDFDVBD6EXAMPLE",
        "Status": "InProgress",
        "LastModifiedTime": "2019-12-06T18:55:39.870Z",
        "InProgressInvalidationBatches": 0,
        "DomainName": "d111111abcdef8.cloudfront.net",
        "ActiveTrustedSigners": {
            "Enabled": false,
            "Quantity": 0
```

```
        },
        "DistributionConfig": {
            "CallerReference": "6b10378d-49be-4c4b-a642-419ccaf8f3b5",
            "Aliases": {
                "Quantity": 0
            },
            "DefaultRootObject": "index.html",
            "Origins": {
                "Quantity": 1,
                "Items": [
                    {
                        "Id": "example-website",
                        "DomainName": "www.example.com",
                        "OriginPath": "",
                        "CustomHeaders": {
                            "Quantity": 0
                        },
                        "CustomOriginConfig": {
                            "HTTPPort": 80,
                            "HTTPSPort": 443,
                            "OriginProtocolPolicy": "match-viewer",
                            "OriginSslProtocols": {
                                "Quantity": 2,
                                "Items": [
                                    "SSLv3",
                                    "TLSv1"
                                ]
                            },
                            "OriginReadTimeout": 30,
                            "OriginKeepaliveTimeout": 5
                        }
                    }
                ]
            },
            "OriginGroups": {
                "Quantity": 0
            },
            "DefaultCacheBehavior": {
                "TargetOriginId": "example-website",
                "ForwardedValues": {
                    "QueryString": false,
                    "Cookies": {
                        "Forward": "none"
                    },
```

```
                    "Headers": {
                        "Quantity": 1,
                        "Items": [
                            "*"
                        ]
                    },
                    "QueryStringCacheKeys": {
                        "Quantity": 0
                    }
                },
                "TrustedSigners": {
                    "Enabled": false,
                    "Quantity": 0
                },
                "ViewerProtocolPolicy": "allow-all",
                "MinTTL": 0,
                "AllowedMethods": {
                    "Quantity": 2,
                    "Items": [
                        "HEAD",
                        "GET"
                    ],
                    "CachedMethods": {
                        "Quantity": 2,
                        "Items": [
                            "HEAD",
                            "GET"
                        ]
                    }
                },
                "SmoothStreaming": false,
                "DefaultTTL": 86400,
                "MaxTTL": 31536000,
                "Compress": false,
                "LambdaFunctionAssociations": {
                    "Quantity": 0
                },
                "FieldLevelEncryptionId": ""
            },
            "CacheBehaviors": {
                "Quantity": 0
            },
            "CustomErrorResponses": {
                "Quantity": 0
```

```
            },
            "Comment": "",
            "Logging": {
                "Enabled": false,
                "IncludeCookies": false,
                "Bucket": "",
                "Prefix": ""
            },
            "PriceClass": "PriceClass_All",
            "Enabled": true,
            "ViewerCertificate": {
                "CloudFrontDefaultCertificate": true,
                "MinimumProtocolVersion": "TLSv1",
                "CertificateSource": "cloudfront"
            },
            "Restrictions": {
                "GeoRestriction": {
                    "RestrictionType": "none",
                    "Quantity": 0
                }
            },
            "WebACLId": "",
            "HttpVersion": "http1.1",
            "IsIPV6Enabled": true
        }
    }
}
```

Esempio 2: aggiornare una CloudFront distribuzione

L'esempio seguente disabilita la CloudFront distribuzione con l'ID `EMLARXS9EXAMPLE` fornendo la configurazione della distribuzione in un file JSON denominato. `dist-config-disable.json` Per aggiornare una distribuzione, è necessario utilizzare l'opzione `--if-match` per fornire il tag entità (`ETag`) della distribuzione. Per ottenere il`ETag`, usa il comando get-distribution o. get-distribution-config Nota che il campo `Enabled` è impostato su `false` nel file JSON.

Dopo aver utilizzato l'esempio seguente per disabilitare una distribuzione, puoi utilizzare il comando delete-distribution per eliminarla.

```
aws cloudfront update-distribution \
    --id EMLARXS9EXAMPLE \
    --if-match E2QWRUHEXAMPLE \
```

```
    --distribution-config file://dist-config-disable.json
```

Contenuto di `dist-config-disable.json`:

```
{
    "CallerReference": "cli-1574382155-496510",
    "Aliases": {
        "Quantity": 0
    },
    "DefaultRootObject": "index.html",
    "Origins": {
        "Quantity": 1,
        "Items": [
            {
                "Id": "amzn-s3-demo-bucket.s3.amazonaws.com-1574382155-273939",
                "DomainName": "amzn-s3-demo-bucket.s3.amazonaws.com",
                "OriginPath": "",
                "CustomHeaders": {
                    "Quantity": 0
                },
                "S3OriginConfig": {
                    "OriginAccessIdentity": ""
                }
            }
        ]
    },
    "OriginGroups": {
        "Quantity": 0
    },
    "DefaultCacheBehavior": {
        "TargetOriginId": "amzn-s3-demo-bucket.s3.amazonaws.com-1574382155-273939",
        "ForwardedValues": {
            "QueryString": false,
            "Cookies": {
                "Forward": "none"
            },
            "Headers": {
                "Quantity": 0
            },
            "QueryStringCacheKeys": {
                "Quantity": 0
            }
        },
```

```
        "TrustedSigners": {
            "Enabled": false,
            "Quantity": 0
        },
        "ViewerProtocolPolicy": "allow-all",
        "MinTTL": 0,
        "AllowedMethods": {
            "Quantity": 2,
            "Items": [
                "HEAD",
                "GET"
            ],
            "CachedMethods": {
                "Quantity": 2,
                "Items": [
                    "HEAD",
                    "GET"
                ]
            }
        },
        "SmoothStreaming": false,
        "DefaultTTL": 86400,
        "MaxTTL": 31536000,
        "Compress": false,
        "LambdaFunctionAssociations": {
            "Quantity": 0
        },
        "FieldLevelEncryptionId": ""
    },
    "CacheBehaviors": {
        "Quantity": 0
    },
    "CustomErrorResponses": {
        "Quantity": 0
    },
    "Comment": "",
    "Logging": {
        "Enabled": false,
        "IncludeCookies": false,
        "Bucket": "",
        "Prefix": ""
    },
    "PriceClass": "PriceClass_All",
    "Enabled": false,
```

```
    "ViewerCertificate": {
        "CloudFrontDefaultCertificate": true,
        "MinimumProtocolVersion": "TLSv1",
        "CertificateSource": "cloudfront"
    },
    "Restrictions": {
        "GeoRestriction": {
            "RestrictionType": "none",
            "Quantity": 0
        }
    },
    "WebACLId": "",
    "HttpVersion": "http2",
    "IsIPV6Enabled": true
}
```

Output:

```
{
    "ETag": "E9LHASXEXAMPLE",
    "Distribution": {
        "Id": "EMLARXS9EXAMPLE",
        "ARN": "arn:aws:cloudfront::123456789012:distribution/EMLARXS9EXAMPLE",
        "Status": "InProgress",
        "LastModifiedTime": "2019-12-06T18:32:35.553Z",
        "InProgressInvalidationBatches": 0,
        "DomainName": "d111111abcdef8.cloudfront.net",
        "ActiveTrustedSigners": {
            "Enabled": false,
            "Quantity": 0
        },
        "DistributionConfig": {
            "CallerReference": "cli-1574382155-496510",
            "Aliases": {
                "Quantity": 0
            },
            "DefaultRootObject": "index.html",
            "Origins": {
                "Quantity": 1,
                "Items": [
                    {
                        "Id": "amzn-s3-demo-
bucket.s3.amazonaws.com-1574382155-273939",
```

```
                        "DomainName": "amzn-s3-demo-bucket.s3.amazonaws.com",
                        "OriginPath": "",
                        "CustomHeaders": {
                            "Quantity": 0
                        },
                        "S3OriginConfig": {
                            "OriginAccessIdentity": ""
                        }
                    }
                ]
            },
            "OriginGroups": {
                "Quantity": 0
            },
            "DefaultCacheBehavior": {
                "TargetOriginId": "amzn-s3-demo-
bucket.s3.amazonaws.com-1574382155-273939",
                "ForwardedValues": {
                    "QueryString": false,
                    "Cookies": {
                        "Forward": "none"
                    },
                    "Headers": {
                        "Quantity": 0
                    },
                    "QueryStringCacheKeys": {
                        "Quantity": 0
                    }
                },
                "TrustedSigners": {
                    "Enabled": false,
                    "Quantity": 0
                },
                "ViewerProtocolPolicy": "allow-all",
                "MinTTL": 0,
                "AllowedMethods": {
                    "Quantity": 2,
                    "Items": [
                        "HEAD",
                        "GET"
                    ],
                    "CachedMethods": {
                        "Quantity": 2,
                        "Items": [
```

```
                        "HEAD",
                        "GET"
                    ]
                }
            },
            "SmoothStreaming": false,
            "DefaultTTL": 86400,
            "MaxTTL": 31536000,
            "Compress": false,
            "LambdaFunctionAssociations": {
                "Quantity": 0
            },
            "FieldLevelEncryptionId": ""
        },
        "CacheBehaviors": {
            "Quantity": 0
        },
        "CustomErrorResponses": {
            "Quantity": 0
        },
        "Comment": "",
        "Logging": {
            "Enabled": false,
            "IncludeCookies": false,
            "Bucket": "",
            "Prefix": ""
        },
        "PriceClass": "PriceClass_All",
        "Enabled": false,
        "ViewerCertificate": {
            "CloudFrontDefaultCertificate": true,
            "MinimumProtocolVersion": "TLSv1",
            "CertificateSource": "cloudfront"
        },
        "Restrictions": {
            "GeoRestriction": {
                "RestrictionType": "none",
                "Quantity": 0
            }
        },
        "WebACLId": "",
        "HttpVersion": "http2",
        "IsIPV6Enabled": true
    }
```

```
    }
}
```

- Per i dettagli sull'API, consulta AWS CLI Command UpdateDistributionReference.

`update-domain-association`

Il seguente esempio di codice mostra come utilizzareu`pdate-domain-association`.

AWS CLI

Come aggiornare un'associazione di dominio

L'`update-domain-association`esempio seguente aggiorna un'associazione di dominio per un tenant di distribuzione con ETag `E23ZP02F085ABC`.

```
aws cloudfront update-domain-association \
    --domain example.com \
    --target-resource DistributionTenantId=dt_2x9GhoK0TZRsohWzv1b9It8J1AB \
    --if-match E23ZP02F085ABC
```

Output:

```
{
    "ETag": "ETVPDKIKX0ABC",
    "Domain": "example.com",
    "ResourceId": "dt_2x9GhoK0TZRsohWzv1b9It8J1AB"
}
```

Per ulteriori informazioni, consulta Spostare un nome di dominio alternativo in una distribuzione diversa nella Amazon CloudFront Developer Guide.

- Per i dettagli sull'API, consulta UpdateDomainAssociation AWS CLICommand Reference.

`update-field-level-encryption-config`

Il seguente esempio di codice mostra come utilizzareu`pdate-field-level-encryption-config`.

AWS CLI

Per aggiornare una configurazione di CloudFront crittografia a livello di campo

L'esempio seguente aggiorna il campo `Comment` della configurazione di crittografia a livello di campo con l'ID `C3KM2WVD605UAY` fornendo i parametri in un file JSON.

Per aggiornare una configurazione di crittografia a livello di campo, è necessario disporre dei relativi `ETag` e ID. L'ID viene restituito nell'output dei comandi create-field-level-encryption -config e -configs. list-field-level-encryption Per ottenere il`ETag`, usa il comando or -config. get-field-level-encryption get-field-level-encryption Utilizza l'opzione `--if-match` per fornire il valore `ETag` della configurazione.

```
aws cloudfront update-field-level-encryption-config \
    --id C3KM2WVD605UAY \
    --if-match E2P4Z4VU7TY5SG \
    --field-level-encryption-config file://fle-config.json
```

Il file `fle-config.json` è un documento JSON nella directory corrente che contiene quanto segue:

```
{
    "CallerReference": "cli-example",
    "Comment": "Updated example FLE configuration",
    "QueryArgProfileConfig": {
        "ForwardWhenQueryArgProfileIsUnknown": true,
        "QueryArgProfiles": {
            "Quantity": 0
        }
    },
    "ContentTypeProfileConfig": {
        "ForwardWhenContentTypeIsUnknown": true,
        "ContentTypeProfiles": {
            "Quantity": 1,
            "Items": [
                {
                    "Format": "URLEncoded",
                    "ProfileId": "P280MFCLSYOCVU",
                    "ContentType": "application/x-www-form-urlencoded"
                }
            ]
        }
    }
}
```

Output:

```
{
    "ETag": "E26M4BIAV81ZF6",
    "FieldLevelEncryption": {
        "Id": "C3KM2WVD605UAY",
        "LastModifiedTime": "2019-12-10T22:26:26.170Z",
        "FieldLevelEncryptionConfig": {
            "CallerReference": "cli-example",
            "Comment": "Updated example FLE configuration",
            "QueryArgProfileConfig": {
                "ForwardWhenQueryArgProfileIsUnknown": true,
                "QueryArgProfiles": {
                    "Quantity": 0,
                    "Items": []
                }
            },
            "ContentTypeProfileConfig": {
                "ForwardWhenContentTypeIsUnknown": true,
                "ContentTypeProfiles": {
                    "Quantity": 1,
                    "Items": [
                        {
                            "Format": "URLEncoded",
                            "ProfileId": "P280MFCLSYOCVU",
                            "ContentType": "application/x-www-form-urlencoded"
                        }
                    ]
                }
            }
        }
    }
}
```

- Per i dettagli sull'API, consulta AWS CLI Command UpdateFieldLevelEncryptionConfigReference.

`update-field-level-encryption-profile`

Il seguente esempio di codice mostra come utilizzareu`pdate-field-level-encryption-profile`.

AWS CLI

Per aggiornare un profilo di CloudFront crittografia a livello di campo

L'esempio seguente aggiorna il profilo di crittografia a livello di campo con l'ID `PPK0UOSIF5WSV`. Nell'esempio vengono aggiornati i campi `Name` e `Comment` del profilo e viene aggiunta una seconda voce `FieldPatterns` fornendo i parametri in un file JSON.

Per aggiornare un profilo di crittografia a livello di campo, è necessario disporre dei relativi `ETag` e ID. L'ID viene restituito nell'output dei comandi create-field-level-encryption -profile e list-field-level-encryption -profiles. Per ottenere il`ETag`, usa il comando get-field-level-encryption -profile o get-field-level-encryption -profile-config. Utilizza l'opzione `--if-match` per fornire gli `ETag` del profilo.

```
aws cloudfront update-field-level-encryption-profile \
    --id PPK0UOSIF5WSV \
    --if-match E1QQG65FS2L2GC \
    --field-level-encryption-profile-config file://fle-profile-config.json
```

Il file `fle-profile-config.json` è un documento JSON nella directory corrente che contiene quanto segue:

```
{
    "Name": "ExampleFLEProfileUpdated",
    "CallerReference": "cli-example",
    "Comment": "Updated FLE profile for AWS CLI example",
    "EncryptionEntities": {
        "Quantity": 1,
        "Items": [
            {
                "PublicKeyId": "K2K8NC4HVFE3M0",
                "ProviderId": "ExampleFLEProvider",
                "FieldPatterns": {
                    "Quantity": 2,
                    "Items": [
                        "ExampleSensitiveField",
                        "SecondExampleSensitiveField"
                    ]
                }
            }
        ]
```

```
    }
}
```

Output:

```
{
    "ETag": "EJETYFJ9CL66D",
    "FieldLevelEncryptionProfile": {
        "Id": "PPK0UOSIF5WSV",
        "LastModifiedTime": "2019-12-10T19:05:58.296Z",
        "FieldLevelEncryptionProfileConfig": {
            "Name": "ExampleFLEProfileUpdated",
            "CallerReference": "cli-example",
            "Comment": "Updated FLE profile for AWS CLI example",
            "EncryptionEntities": {
                "Quantity": 1,
                "Items": [
                    {
                        "PublicKeyId": "K2K8NC4HVFE3M0",
                        "ProviderId": "ExampleFLEProvider",
                        "FieldPatterns": {
                            "Quantity": 2,
                            "Items": [
                                "ExampleSensitiveField",
                                "SecondExampleSensitiveField"
                            ]
                        }
                    }
                ]
            }
        }
    }
}
```

- Per i dettagli sull'API, consulta Command Reference. UpdateFieldLevelEncryptionProfileAWS CLI

`verify-dns-configuration`

Il seguente esempio di codice mostra come utilizzare`verify-dns-configuration`.

AWS CLI

Come verificare la configurazione DNS per un dominio

L'esempio `verify-dns-configuration` seguente verifica la configurazione DNS per un dominio.

```
aws cloudfront verify-dns-configuration \
    --domain example.com \
    --identifier dt_2x9GhoK0TZRsohWzv1b9It8J1AB
```

Output:

```
{
    "DnsConfigurationList": [
        {
            "Domain": "example.com",
            "Status": "valid-configuration"
        }
    ]
}
```

Per ulteriori informazioni, consulta Spostare un nome di dominio alternativo in una distribuzione diversa nella Amazon CloudFront Developer Guide.

- Per i dettagli sull'API, consulta VerifyDnsConfiguration AWS CLICommand Reference.

CloudSearch Esempi di utilizzo di Amazon AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando il AWS Command Line Interface con Amazon CloudSearch.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`upload-documents`

Il seguente esempio di codice mostra come usare`upload-documents`.

AWS CLI

Il `upload-documents` comando seguente carica un batch di documenti JSON su un dominio Amazon CloudSearch :

```
aws cloudsearchdomain upload-documents --endpoint-url https://doc-my-domain.us-west-1.cloudsearch.amazonaws.com --content-type application/json --documents document-batch.json
```

Output:

```
{
  "status": "success",
  "adds": 5000,
  "deletes": 0
}
```

- Per i dettagli sull'API, consulta AWS CLI Command UploadDocumentsReference.

CloudTrail esempi utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with CloudTrail.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`add-tags`

Il seguente esempio di codice mostra come utilizzare`add-tags`.

AWS CLI

Come aggiungere tag al trail

Il comando `add-tags` seguente aggiunge tag per `Trail1`:

```
aws cloudtrail add-tags --resource-id arn:aws:cloudtrail:us-
east-1:123456789012:trail/Trail1 --tags-
list Key=name,Value=Alice Key=location,Value=us
```

- Per i dettagli sull'API, consulta AddTags AWS CLICommand Reference.

`create-subscription`

Il seguente esempio di codice mostra come utilizzare`create-subscription`.

AWS CLI

Per creare e configurare AWS risorse per un percorso

Il comando `create-subscription` seguente crea un nuovo bucket S3 e un argomento SNS per `Trail1`.

```
aws cloudtrail create-subscription \
    --name Trail1 \
    --s3-new-bucket amzn-s3-demo-bucket \
    --sns-new-topic my-topic
```

Output:

```
Setting up new S3 bucket amzn-s3-demo-bucket...
Setting up new SNS topic my-topic...
Creating/updating CloudTrail configuration...
CloudTrail configuration:
    {
        "trailList": [
            {
```

```
                "IncludeGlobalServiceEvents": true,
                "Name": "Trail1",
                "TrailARN": "arn:aws:cloudtrail:us-east-1:123456789012:trail/
Trail1",
                "LogFileValidationEnabled": false,
                "IsMultiRegionTrail": false,
                "S3BucketName": "amzn-s3-demo-bucket",
                "SnsTopicName": "my-topic",
                "HomeRegion": "us-east-1"
            }
        ],
        "ResponseMetadata": {
        "HTTPStatusCode": 200,
        "RequestId": "f39e51f6-c615-11e5-85bd-d35ca21ee3e2"
        }
    }
Starting CloudTrail service...
Logs will be delivered to my-bucket
```

- Per i dettagli sull'API, consulta CreateSubscription AWS CLICommand Reference.

`create-trail`

Il seguente esempio di codice mostra come utilizzare`create-trail`.

AWS CLI

Come creare un trail

L'esempio `create-trail` seguente crea un trail multi-Regione denominato `Trail1` e specifica un bucket S3.

```
aws cloudtrail create-trail \
    --name Trail1 \
    --s3-bucket-name amzn-s3-demo-bucket \
    --is-multi-region-trail
```

Output:

```
{
    "IncludeGlobalServiceEvents": true,
    "Name": "Trail1",
```

```
    "TrailARN": "arn:aws:cloudtrail:us-west-2:123456789012:trail/Trail1",
    "LogFileValidationEnabled": false,
    "IsMultiRegionTrail": true,
    "S3BucketName": "amzn-s3-demo-bucket"
}
```

- Per i dettagli sull'API, consulta CreateTrail AWS CLICommand Reference.

`delete-trail`

Il seguente esempio di codice mostra come utilizzare`delete-trail`.

AWS CLI

Come eliminare un trail

Il comando `delete-trail` seguente elimina un trail denominato `Trail1`.

```
aws cloudtrail delete-trail --name Trail1
```

- Per i dettagli sull'API, consulta DeleteTrail AWS CLICommand Reference.

`describe-trails`

Il seguente esempio di codice mostra come utilizzare`describe-trails`.

AWS CLI

Come descrivere un trail

L'esempio `describe-trails` seguente restituisce le impostazioni per `Trail1` e `Trail2`.

```
aws cloudtrail describe-trails \
    --trail-name-list Trail1 Trail2
```

Output:

```
{
    "trailList": [
        {
            "IncludeGlobalServiceEvents": true,
            "Name": "Trail1",
```

```
            "TrailARN": "arn:aws:cloudtrail:us-east-1:123456789012:trail/Trail1",
            "LogFileValidationEnabled": false,
            "IsMultiRegionTrail": false,
            "S3BucketName": "amzn-s3-demo-bucket",
            "CloudWatchLogsRoleArn": "arn:aws:iam::123456789012:role/
CloudTrail_CloudWatchLogs_Role",
            "CloudWatchLogsLogGroupArn": "arn:aws:logs:us-east-1:123456789012:log-
group:CloudTrail:*",
            "SnsTopicName": "my-topic",
            "HomeRegion": "us-east-1"
        },
        {
            "IncludeGlobalServiceEvents": true,
            "Name": "Trail2",
            "S3KeyPrefix": "my-prefix",
            "TrailARN": "arn:aws:cloudtrail:us-east-1:123456789012:trail/Trail2",
            "LogFileValidationEnabled": false,
            "IsMultiRegionTrail": false,
            "S3BucketName": "amzn-s3-demo-bucket2",
            "KmsKeyId": "arn:aws:kms:us-
east-1:123456789012:key/4c5ae5ac-3c13-421e-8335-c7868ef6a769",
            "HomeRegion": "us-east-1"
        }
    ]
}
```

- Per i dettagli sull'API, consulta DescribeTrails AWS CLICommand Reference.

`get-event-selectors`

Il seguente esempio di codice mostra come utilizzare`get-event-selectors`.

AWS CLI

Come visualizzare le impostazioni del selettore di eventi per un trail

Il seguente `get-event-selectors` comando restituisce le impostazioni per `Trail1`:

```
aws cloudtrail get-event-selectors --trail-name Trail1
```

Output:

```
{
```

```
  "EventSelectors": [
      {
          "IncludeManagementEvents": true,
          "DataResources": [],
          "ReadWriteType": "All"
      }
  ],
  "TrailARN": "arn:aws:cloudtrail:us-east-1:123456789012:trail/Trail1"
}
```

- Per i dettagli sull'API, consulta GetEventSelectors AWS CLICommand Reference.

`get-trail-status`

Il seguente esempio di codice mostra come utilizzare`get-trail-status`.

AWS CLI

Come ottenere lo stato di un trail

Il comando `get-trail-status` seguente restituisce i dettagli di distribuzione e registrazione per `Trail1`.

```
aws cloudtrail get-trail-status --name Trail1
```

Output:

```
{
  "LatestNotificationTime": 1454022144.869,
  "LatestNotificationAttemptSucceeded": "2016-01-28T23:02:24Z",
  "LatestDeliveryAttemptTime": "2016-01-28T23:02:24Z",
  "LatestDeliveryTime": 1454022144.869,
  "TimeLoggingStarted": "2015-11-06T18:36:38Z",
  "LatestDeliveryAttemptSucceeded": "2016-01-28T23:02:24Z",
  "IsLogging": true,
  "LatestCloudWatchLogsDeliveryTime": 1454022144.918,
  "StartLoggingTime": 1446834998.695,
  "StopLoggingTime": 1446834996.933,
  "LatestNotificationAttemptTime": "2016-01-28T23:02:24Z",
  "TimeLoggingStopped": "2015-11-06T18:36:36Z"
}
```

- Per i dettagli sull'API, consulta GetTrailStatus AWS CLICommand Reference.

`list-public-keys`

Il seguente esempio di codice mostra come utilizzare`list-public-keys`.

AWS CLI

Come elencare tutte le chiavi pubbliche di un trail

Il comando `list-public-keys` seguente restituisce tutte le chiavi pubbliche le cui chiavi private sono state utilizzate per firmare i file digest compresi nell'intervallo di tempo specificato:

```
aws cloudtrail list-public-keys --start-time 2016-01-01T20:30:00.000Z
```

Output:

```
{
  "PublicKeyList": [
      {
         "ValidityStartTime": 1453076702.0,
         "ValidityEndTime": 1455668702.0,
         "Value": "MIIBCgKCAQEAlSS3cl92HDycr/MTj0moOhas8habjrraXw+KzlWF0axSI2tcF
+3iJ9BKQAVSKxGwxwu3m0wG3J
+kUl1xboEcEPHYoIYMbgfSw7KGnuDKwkLzsQWhUJ0cIbOHASox1vv/5fNXkrHhGbDCHeVXm804c83nvHUEFYThr1PfyP
+4WGDk+BGH5m9iuiAKkipEHWmUl8/P7XpfpWQuk4h8g3pXZOrNXr08lbh4d39svj7UqdhvOXoBISp9t/
EXYuePGEtBdrKD9Dz+VHwyUPtBQvYr9BnkF88qBnaPNhS44rzwIDAQAB",
         "Fingerprint": "7f3f401420072e50a65a141430817ab3"
      }
  ]
}
```

- Per i dettagli sull'API, consulta ListPublicKeys AWS CLICommand Reference.

`list-tags`

Il seguente esempio di codice mostra come utilizzare`list-tags`.

AWS CLI

Come elencare i tag di un trail

Il comando `list-tags` seguente elenca i tag per `Trail1` e`Trail2`:

```
aws cloudtrail list-tags --resource-id-list arn:aws:cloudtrail:us-
east-1:123456789012:trail/Trail1 arn:aws:cloudtrail:us-east-1:123456789012:trail/
Trail2
```

Output:

```
{
 "ResourceTagList": [
     {
         "ResourceId": "arn:aws:cloudtrail:us-east-1:123456789012:trail/Trail1",
         "TagsList": [
             {
                 "Value": "Alice",
                 "Key": "name"
             },
             {
                 "Value": "us",
                 "Key": "location"
             }
         ]
     },
     {
         "ResourceId": "arn:aws:cloudtrail:us-east-1:123456789012:trail/Trail2",
         "TagsList": [
             {
                 "Value": "Bob",
                 "Key": "name"
             }
         ]
     }
  ]
}
```

- Per i dettagli sull'API, consulta ListTags AWS CLICommand Reference.

`lookup-events`

Il seguente esempio di codice mostra come utilizzare`lookup-events`.

AWS CLI

Come cercare gli eventi relativi a un percorso

Il comando `lookup-events` seguente cerca gli eventi delle attività dell'API in base all'attributo `EventName`:

```
aws cloudtrail lookup-events --lookup-
attributes AttributeKey=EventName,AttributeValue=ConsoleLogin
```

Output:

```
{
  "Events": [
     {
         "EventId": "654ccbc0-ba0d-486a-9076-dbf7274677a7",
         "Username": "my-session-name",
         "EventTime": "2021-11-18T09:41:02-08:00",
         "CloudTrailEvent": "{\"eventVersion\":\"1.02\",\"userIdentity\":{\"type\":
\"AssumedRole\",\"principalId\":\"AROAJIKPFTA72SWU4L7T4:my-session-name\",\"arn\":
\"arn:aws:sts::123456789012:assumed-role/my-role/my-session-name\",\"accountId\":
\"123456789012\",\"sessionContext\":{\"attributes\":{\"mfaAuthenticated\":\"false
\",\"creationDate\":\"2016-01-26T21:42:12Z\"},\"sessionIssuer\":{\"type\":\"Role\",
\"principalId\":\"AROAJIKPFTA72SWU4L7T4\",\"arn\":\"arn:aws:iam::123456789012:role/
my-role\",\"accountId\":\"123456789012\",\"userName\":\"my-role\"}}},\"eventTime
\":\"2016-01-26T21:42:12Z\",\"eventSource\":\"signin.amazonaws.com\",\"eventName\":
\"ConsoleLogin\",\"awsRegion\":\"us-east-1\",\"sourceIPAddress\":\"72.21.198.70\",
\"userAgent\":\"Mozilla/5.0 (Macintosh; Intel Mac OS X 10_9_5) AppleWebKit/537.36
 (KHTML, like Gecko) Chrome/47.0.2526.111 Safari/537.36\",\"requestParameters
\":null,\"responseElements\":{\"ConsoleLogin\":\"Success\"},\"additionalEventData\":
{\"MobileVersion\":\"No\",\"MFAUsed\":\"No\"},\"eventID\":\"654ccbc0-ba0d-486a-9076-
dbf7274677a7\",\"eventType\":\"AwsConsoleSignIn\",\"recipientAccountId\":
\"123456789012\"}",
         "EventName": "ConsoleLogin",
         "Resources": []
     }
  ]
}
```

- Per i dettagli sull'API, consulta LookupEvents AWS CLICommand Reference.

`put-event-selectors`

Il seguente esempio di codice mostra come utilizzare`put-event-selectors`.

AWS CLI

Esempio 1: configura un trail per registrare eventi di gestione ed eventi di dati utilizzando i selettori di eventi avanzati

È possibile aggiungere selettori di eventi avanzati e condizioni per i selettori di eventi avanzati, fino a un massimo di 500 valori per tutte le condizioni e i selettori su un trail. È possibile utilizzare selettori di eventi avanzati per registrare tutti i tipi di eventi dati disponibili. È possibile utilizzare selettori di eventi avanzati o selettori di eventi di base, ma non entrambi. Se si applicano selettori di eventi avanzati a un percorso, tutti i selettori di eventi di base esistenti vengono sovrascritti.

L'`put-event-selectors`esempio seguente crea un selettore di eventi avanzato per un percorso denominato `myTrail` per registrare tutti gli eventi di gestione, registrare le chiamate S3 PutObject e DeleteObject API per tutti i bucket S3 tranne uno, registrare le chiamate API dei dati per una funzione Lambda denominata `myFunction` e registrare le chiamate API Publish su un argomento SNS denominato. `myTopic`

```
aws cloudtrail put-event-selectors \
    --trail-name myTrail \
    --advanced-event-selectors '[{"Name": "Log all management events",
 "FieldSelectors": [{ "Field": "eventCategory", "Equals": ["Management"] }] },
{"Name": "Log PutObject and DeleteObject events for all but one
 bucket","FieldSelectors": [{ "Field": "eventCategory", "Equals": ["Data"] },
{ "Field": "resources.type", "Equals": ["AWS::S3::Object"] },{ "Field":
 "eventName", "Equals": ["PutObject","DeleteObject"] },{ "Field": "resources.ARN",
 "NotStartsWith": ["arn:aws:s3:::amzn-s3-demo-bucket/"] }]},{"Name": "Log
 data events for a specific Lambda function","FieldSelectors": [{ "Field":
 "eventCategory", "Equals": ["Data"] },{ "Field": "resources.type",
 "Equals": ["AWS::Lambda::Function"] },{ "Field": "resources.ARN", "Equals":
 ["arn:aws:lambda:us-east-1:123456789012:function:myFunction"] }]},{"Name":
 "Log all Publish API calls on a specific SNS topic","FieldSelectors":
 [{ "Field": "eventCategory", "Equals": ["Data"] },{ "Field": "resources.type",
 "Equals": ["AWS::SNS::Topic"] },{ "Field": "eventName", "Equals":
 ["Publish"] },{ "Field": "resources.ARN", "Equals": ["arn:aws:sns:us-
east-1:123456789012:myTopic.fifo"] }]}]'
```

Output:

```
{
    "TrailARN": "arn:aws:cloudtrail:us-east-1:123456789012:trail/myTrail",
    "AdvancedEventSelectors": [
        {
            "Name": "Log all management events",
            "FieldSelectors": [
                {
                    "Field": "eventCategory",
                    "Equals": [
                        "Management"
                    ]
                }
            ]
        },
        {
            "Name": "Log PutObject and DeleteObject events for all but one bucket",
            "FieldSelectors": [
                {
                    "Field": "eventCategory",
                    "Equals": [
                        "Data"
                    ]
                },
                {
                    "Field": "resources.type",
                    "Equals": [
                        "AWS::S3::Object"
                    ]
                },
                {
                    "Field": "eventName",
                    "Equals": [
                        "PutObject",
                        "DeleteObject"
                    ]
                },
                {
                    "Field": "resources.ARN",
                    "NotStartsWith": [
                        "arn:aws:s3:::amzn-s3-demo-bucket/"
                    ]
                }
            ]
```

```
        },
        {
            "Name": "Log data events for a specific Lambda function",
            "FieldSelectors": [
                {
                    "Field": "eventCategory",
                    "Equals": [
                        "Data"
                    ]
                },
                {
                    "Field": "resources.type",
                    "Equals": [
                        "AWS::Lambda::Function"
                    ]
                },
                {
                    "Field": "resources.ARN",
                    "Equals": [
                        "arn:aws:lambda:us-east-1:123456789012:function:myFunction"
                    ]
                }
            ]
        },
        {
            "Name": "Log all Publish API calls on a specific SNS topic",
            "FieldSelectors": [
                {
                    "Field": "eventCategory",
                    "Equals": [
                        "Data"
                    ]
                },
                {
                    "Field": "resources.type",
                    "Equals": [
                        "AWS::SNS::Topic"
                    ]
                },
                {
                    "Field": "eventName",
                    "Equals": [
                        "Publish"
                    ]
```

```
                },
                {
                    "Field": "resources.ARN",
                    "Equals": [
                        "arn:aws:sns:us-east-1:123456789012:myTopic.fifo"
                    ]
                }
            ]
        }
    ]
}
```

Per ulteriori informazioni, consulta Registrare gli eventi utilizzando selettori di eventi avanzati nella Guida per l'utente.AWS CloudTrail

Esempio 2: configura i selettori di eventi per un trail per registrare tutti gli eventi di gestione e gli eventi relativi ai dati

Per un trail puoi configurare fino a 5 selettori di eventi e un massimo di 250 risorse di dati. I selettori di eventi vengono anche chiamati selettori di eventi di base. Puoi utilizzare i selettori di eventi per registrare eventi di gestione ed eventi di dati per oggetti S3, funzioni DynamoDB e funzioni Lambda. Per registrare eventi di dati per altri tipi di risorsa, è necessario utilizzare selettori di eventi avanzati.

L'`put-event-selectors`esempio seguente crea un selettore di eventi per un percorso denominato `TrailName` per includere tutti gli eventi di gestione, gli eventi di dati per due combinazioni di Amazon bucket/prefix S3 e gli eventi dati per una singola funzione AWS Lambda denominata. `hello-world-python-function`

```
aws cloudtrail put-event-selectors \
    --trail-name TrailName \
    --event-selectors '[{"ReadWriteType": "All","IncludeManagementEvents":
 true,"DataResources": [{"Type":"AWS::S3::Object", "Values": ["arn:aws:s3:::amzn-
s3-demo-bucket/prefix","arn:aws:s3:::amzn-s3-demo-bucket2/prefix2"]},
{"Type": "AWS::Lambda::Function","Values": ["arn:aws:lambda:us-
west-2:999999999999:function:hello-world-python-function"]}]}]'
```

Output:

```
{
    "EventSelectors": [
        {
```

```
            "IncludeManagementEvents": true,
            "DataResources": [
                {
                    "Values": [
                        "arn:aws:s3:::amzn-s3-demo-bucket/prefix",
                        "arn:aws:s3:::amzn-s3-demo-bucket2/prefix2"
                    ],
                    "Type": "AWS::S3::Object"
                },
                {
                    "Values": [
                        "arn:aws:lambda:us-west-2:123456789012:function:hello-world-
python-function"
                    ],
                    "Type": "AWS::Lambda::Function"
                },
            ],
            "ReadWriteType": "All"
        }
    ],
    "TrailARN": "arn:aws:cloudtrail:us-east-2:123456789012:trail/TrailName"
}
```

Per ulteriori informazioni, consulta Registrare gli eventi utilizzando selettori di eventi di base nella Guida per l'utente.AWS CloudTrail

Esempio 3: configura i selettori di eventi per un trail per registrare gli eventi di gestione, tutti gli eventi dei dati S3 sugli oggetti S3 e tutti gli eventi dei dati Lambda sulle funzioni del tuo account

L'`put-event-selectors`esempio seguente crea un selettore di eventi per un percorso denominato `TrailName2` che include tutti gli eventi di gestione e tutti gli eventi di dati per tutti i bucket Amazon S3 AWS e le funzioni Lambda nell'account. AWS

```
aws cloudtrail put-event-selectors \
    --trail-name TrailName2 \
    --event-selectors '[{"ReadWriteType": "All","IncludeManagementEvents":
 true,"DataResources": [{"Type":"AWS::S3::Object", "Values": ["arn:aws:s3"]},
{"Type": "AWS::Lambda::Function","Values": ["arn:aws:lambda"]}]}]'
```

Output:

```
{
```

```
    "EventSelectors": [
        {
            "IncludeManagementEvents": true,
            "DataResources": [
                {
                    "Values": [
                        "arn:aws:s3"
                    ],
                    "Type": "AWS::S3::Object"
                },
                {
                    "Values": [
                        "arn:aws:lambda"
                    ],
                    "Type": "AWS::Lambda::Function"
                },
            ],
            "ReadWriteType": "All"
        }
    ],
    "TrailARN": "arn:aws:cloudtrail:us-east-2:123456789012:trail/TrailName2"
}
```

Per ulteriori informazioni, consulta Registrare gli eventi utilizzando selettori di eventi di base nella Guida per l'utente.AWS CloudTrail

- Per i dettagli sull'API, consulta PutEventSelectors AWS CLICommand Reference.

`remove-tags`

Il seguente esempio di codice mostra come utilizzare`remove-tags`.

AWS CLI

Come rimuovere i tag per un trail

Il comando `remove-tags` seguente rimuove i tag specificati per `Trail1`:

```
aws cloudtrail remove-tags --resource-id arn:aws:cloudtrail:us-
east-1:123456789012:trail/Trail1 --tags-list Key=name Key=location
```

- Per i dettagli sull'API, consulta RemoveTags AWS CLICommand Reference.

`start-logging`

Il seguente esempio di codice mostra come utilizzare`start-logging`.

AWS CLI

Come avviare la registrazione per un trail

Il comando `start-logging` seguente attiva la registrazione dei log per `Trail1`:

```
aws cloudtrail start-logging --name Trail1
```

- Per i dettagli sull'API, consulta StartLogging AWS CLICommand Reference.

`stop-logging`

Il seguente esempio di codice mostra come utilizzare`stop-logging`.

AWS CLI

Come arrestare la registrazione di un trail

Il comando `stop-logging` seguente disattiva la registrazione dei log per `Trail1`:

```
aws cloudtrail stop-logging --name Trail1
```

- Per i dettagli sull'API, consulta StopLogging AWS CLICommand Reference.

`update-subscription`

Il seguente esempio di codice mostra come utilizzareupdate-subscription.

AWS CLI

Come aggiornare le impostazioni di configurazione per un trail

L'esempio `update-subscription` seguente aggiorna il trail in modo che venga specificato un nuovo bucket S3 e un argomento SNS.

```
aws cloudtrail update-subscription \
    --name Trail1 \
```

```
    --s3-new-bucket amzn-s3-demo-bucket \
    --sns-new-topic my-topic-new
```

Output:

```
Setting up new S3 bucket amzn-s3-demo-bucket...
Setting up new SNS topic my-topic-new...
Creating/updating CloudTrail configuration...
CloudTrail configuration:
{
    "trailList": [
        {
            "IncludeGlobalServiceEvents": true,
            "Name": "Trail1",
            "TrailARN": "arn:aws:cloudtrail:us-east-1:123456789012:trail/Trail1",
            "LogFileValidationEnabled": false,
            "IsMultiRegionTrail": false,
            "S3BucketName": "amzn-s3-demo-bucket",
            "SnsTopicName": "my-topic-new",
            "HomeRegion": "us-east-1"
        }
    ],
    "ResponseMetadata": {
        "HTTPStatusCode": 200,
        "RequestId": "31126f8a-c616-11e5-9cc6-2fd637936879"
    }
}
```

- Per i dettagli sull'API, consulta UpdateSubscription AWS CLICommand Reference.

`update-trail`

Il seguente esempio di codice mostra come utilizzare`update-trail`.

AWS CLI

Come aggiornare un trail

L'esempio `update-trail` seguente aggiorna un trail in modo che venga utilizzato un bucket esistente per la distribuzione dei log.

```
aws cloudtrail update-trail \
```

```
    --name Trail1 \
    --s3-bucket-name amzn-s3-demo-bucket
```

Output:

```
{
    "IncludeGlobalServiceEvents": true,
    "Name": "Trail1",
    "TrailARN": "arn:aws:cloudtrail:us-west-2:123456789012:trail/Trail1",
    "LogFileValidationEnabled": false,
    "IsMultiRegionTrail": true,
    "S3BucketName": "amzn-s3-demo-bucket"
}
```

- Per i dettagli sull'API, consulta UpdateTrail AWS CLICommand Reference.

`validate-logs`

Il seguente esempio di codice mostra come utilizzare`validate-logs`.

AWS CLI

Come convalidare un file di log

Il seguente `validate-logs` comando convalida i log per `Trail1`:

```
aws cloudtrail validate-logs --trail-arn arn:aws:cloudtrail:us-
east-1:123456789012:trail/Trail1 --start-time 20160129T19:00:00Z
```

Output:

```
Validating log files for trail arn:aws:cloudtrail:us-east-1:123456789012:trail/
Trail1 between 2016-01-29T19:00:00Z and 2016-01-29T22:15:43Z
Results requested for 2016-01-29T19:00:00Z to 2016-01-29T22:15:43Z
Results found for 2016-01-29T19:24:57Z to 2016-01-29T21:24:57Z:
3/3 digest files valid
15/15 log files valid
```

- Per i dettagli sull'API, consulta ValidateLogs AWS CLICommand Reference.

CloudWatch esempi utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with CloudWatch.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`delete-alarms`

Il seguente esempio di codice mostra come utilizzare`delete-alarms`.

AWS CLI

Come eliminare un allarme

L'esempio seguente utilizza il `delete-alarms` comando per eliminare l' CloudWatch allarme Amazon denominato «myalarm»:

```
aws cloudwatch delete-alarms --alarm-names myalarm
```

Output:

```
This command returns to the prompt if successful.
```

- Per i dettagli sull'API, consulta DeleteAlarms AWS CLICommand Reference.

`delete-anomaly-detector`

Il seguente esempio di codice mostra come utilizzare`delete-anomaly-detector`.

AWS CLI

Come eliminare un modello di rilevamento delle anomalie specificato

L'esempio `delete-anomaly-detector` seguente elimina un modello di rilevatore di anomalie nell'account specificato.

```
aws cloudwatch delete-anomaly-detector \
    --namespace AWS/Logs \
    --metric-name IncomingBytes \
    --stat SampleCount
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Eliminazione di un modello di rilevamento delle anomalie nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta AWS CLI Command DeleteAnomalyDetectorReference.

`delete-dashboards`

Il seguente esempio di codice mostra come utilizzare`delete-dashboards`.

AWS CLI

Come eliminare le dashboard specificate

L'esempio `delete-dashboards` seguente elimina due dashboard denominate `Dashboard-A` e `Dashboard-B` presenti nell'account specificato.

```
aws cloudwatch delete-dashboards \
    --dashboard-names Dashboard-A Dashboard-B
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta i CloudWatch dashboard di Amazon nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta DeleteDashboards AWS CLICommand Reference.

`delete-insight-rules`

Il seguente esempio di codice mostra come utilizzare`delete-insight-rules`.

AWS CLI

Come eliminare le regole di Contributor Insights specificate

L'esempio `delete-insight-rules` seguente elimina due regole di Contributor Insights denominate `Rule-A` e `Rule-B` nell'account specificato.

```
aws cloudwatch delete-insight-rules \
    --rule-names Rule-A Rule-B
```

Output:

```
{
    "Failures": []
}
```

Per ulteriori informazioni, consulta Usa Contributor Insights per analizzare i dati ad alta cardinalità nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta Command DeleteInsightRulesReference AWS CLI .

`delete-metric-stream`

Il seguente esempio di codice mostra come utilizzare`delete-metric-stream`.

AWS CLI

Come eliminare un flusso di metriche specificato

L'esempio `delete-metric-stream` seguente elimina il flusso di metriche denominato `QuickPartial-gSCKvO` nell'account specificato.

```
aws cloudwatch delete-metric-stream \
    --name QuickPartial-gSCKvO
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Use metric Streams nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta AWS CLI Command DeleteMetricStreamReference.

`describe-alarm-history`

Il seguente esempio di codice mostra come utilizzare`describe-alarm-history`.

AWS CLI

Come recuperare la cronologia di un allarme

L'esempio seguente utilizza il `describe-alarm-history` comando per recuperare la cronologia dell'CloudWatch allarme Amazon denominato «myalarm»:

```
aws cloudwatch describe-alarm-history --alarm-name "myalarm" --history-item-
type StateUpdate
```

Output:

```
{
    "AlarmHistoryItems": [
        {
            "Timestamp": "2014-04-09T18:59:06.442Z",
            "HistoryItemType": "StateUpdate",
            "AlarmName": "myalarm",
            "HistoryData": "{\"version\":\"1.0\",\"oldState\":{\"stateValue\":
\"ALARM\",\"stateReason\":\"testing purposes\"},\"newState\":{\"stateValue\":\"OK
\",\"stateReason\":\"Threshold Crossed: 2 datapoints were not greater than the
 threshold (70.0). The most recent datapoints: [38.958, 40.292].\",\"stateReasonData
\":{\"version\":\"1.0\",\"queryDate\":\"2014-04-09T18:59:06.419+0000\",\"startDate
\":\"2014-04-09T18:44:00.000+0000\",\"statistic\":\"Average\",\"period\":300,
\"recentDatapoints\":[38.958,40.292],\"threshold\":70.0}}}",
            "HistorySummary": "Alarm updated from ALARM to OK"
        },
        {
            "Timestamp": "2014-04-09T18:59:05.805Z",
            "HistoryItemType": "StateUpdate",
            "AlarmName": "myalarm",
            "HistoryData": "{\"version\":\"1.0\",\"oldState\":{\"stateValue
\":\"OK\",\"stateReason\":\"Threshold Crossed: 2 datapoints were
 not greater than the threshold (70.0). The most recent datapoints:
 [38.839999999999996, 39.714].\",\"stateReasonData\":{\"version\":
\"1.0\",\"queryDate\":\"2014-03-11T22:45:41.569+0000\",\"startDate\":
\"2014-03-11T22:30:00.000+0000\",\"statistic\":\"Average\",\"period\":300,
\"recentDatapoints\":[38.839999999999996,39.714],\"threshold\":70.0}},\"newState\":
{\"stateValue\":\"ALARM\",\"stateReason\":\"testing purposes\"}}",
```

```
            "HistorySummary": "Alarm updated from OK to ALARM"
        }
    ]
}
```

- Per i dettagli sull'API, consulta AWS CLI Command DescribeAlarmHistoryReference.

`describe-alarms-for-metric`

Il seguente esempio di codice mostra come utilizzare`describe-alarms-for-metric`.

AWS CLI

Come visualizzare informazioni sugli allarmi associati a un parametro

L'esempio seguente utilizza il `describe-alarms-for-metric` comando per visualizzare informazioni su eventuali allarmi associati al EC2 CPUUtilization parametro Amazon e all'istanza con l'ID i-0c986c72. :

```
aws cloudwatch describe-alarms-for-metric --metric-name CPUUtilization --
namespace AWS/EC2 --dimensions Name=InstanceId,Value=i-0c986c72
```

Output:

```
{
    "MetricAlarms": [
        {
            "EvaluationPeriods": 10,
            "AlarmArn": "arn:aws:cloudwatch:us-
east-1:111122223333:alarm:myHighCpuAlarm2",
            "StateUpdatedTimestamp": "2013-10-30T03:03:51.479Z",
            "AlarmConfigurationUpdatedTimestamp": "2013-10-30T03:03:50.865Z",
            "ComparisonOperator": "GreaterThanOrEqualToThreshold",
            "AlarmActions": [
                "arn:aws:sns:us-east-1:111122223333:NotifyMe"
            ],
            "Namespace": "AWS/EC2",
            "AlarmDescription": "CPU usage exceeds 70 percent",
            "StateReasonData": "{\"version\":\"1.0\",\"queryDate\":
\"2013-10-30T03:03:51.479+0000\",\"startDate\":\"2013-10-30T02:08:00.000+0000\",
\"statistic\":\"Average\",\"period\":300,\"recentDatapoints\":
```

```
[40.698,39.612,42.432,39.796,38.816,42.28,42.854,40.088,40.760000000000005,41.316],
\"threshold\":70.0}",
            "Period": 300,
            "StateValue": "OK",
            "Threshold": 70.0,
            "AlarmName": "myHighCpuAlarm2",
            "Dimensions": [
                {
                    "Name": "InstanceId",
                    "Value": "i-0c986c72"
                }
            ],
            "Statistic": "Average",
            "StateReason": "Threshold Crossed: 10 datapoints were not greater than
 or equal to the threshold (70.0). The most recent datapoints: [40.760000000000005,
 41.316].",
            "InsufficientDataActions": [],
            "OKActions": [],
            "ActionsEnabled": true,
            "MetricName": "CPUUtilization"
        },
        {
            "EvaluationPeriods": 2,
            "AlarmArn": "arn:aws:cloudwatch:us-
east-1:111122223333:alarm:myHighCpuAlarm",
            "StateUpdatedTimestamp": "2014-04-09T18:59:06.442Z",
            "AlarmConfigurationUpdatedTimestamp": "2014-04-09T22:26:05.958Z",
            "ComparisonOperator": "GreaterThanThreshold",
            "AlarmActions": [
                "arn:aws:sns:us-east-1:111122223333:HighCPUAlarm"
            ],
            "Namespace": "AWS/EC2",
            "AlarmDescription": "CPU usage exceeds 70 percent",
            "StateReasonData": "{\"version\":\"1.0\",\"queryDate\":
\"2014-04-09T18:59:06.419+0000\",\"startDate\":\"2014-04-09T18:44:00.000+0000\",
\"statistic\":\"Average\",\"period\":300,\"recentDatapoints\":[38.958,40.292],
\"threshold\":70.0}",
            "Period": 300,
            "StateValue": "OK",
            "Threshold": 70.0,
            "AlarmName": "myHighCpuAlarm",
            "Dimensions": [
                {
                    "Name": "InstanceId",
```

```
                    "Value": "i-0c986c72"
                }
            ],
            "Statistic": "Average",
            "StateReason": "Threshold Crossed: 2 datapoints were not greater than
 the threshold (70.0). The most recent datapoints: [38.958, 40.292].",
            "InsufficientDataActions": [],
            "OKActions": [],
            "ActionsEnabled": false,
            "MetricName": "CPUUtilization"
        }
    ]
}
```

- Per i dettagli sull'API, consulta DescribeAlarmsForMetric AWS CLICommand Reference.

`describe-alarms`

Il seguente esempio di codice mostra come utilizzare`describe-alarms`.

AWS CLI

Come elencare le informazioni di un allarme

L'esempio seguente utilizza il comando `describe-alarms` per fornire informazioni sull'allarme denominato "myalarm":

```
aws cloudwatch describe-alarms --alarm-names "myalarm"
```

Output:

```
{
    "MetricAlarms": [
        {
            "EvaluationPeriods": 2,
            "AlarmArn": "arn:aws:cloudwatch:us-east-1:123456789012:alarm:myalarm",
            "StateUpdatedTimestamp": "2014-04-09T18:59:06.442Z",
            "AlarmConfigurationUpdatedTimestamp": "2012-12-27T00:49:54.032Z",
            "ComparisonOperator": "GreaterThanThreshold",
            "AlarmActions": [
                "arn:aws:sns:us-east-1:123456789012:myHighCpuAlarm"
            ],
```

```
            "Namespace": "AWS/EC2",
            "AlarmDescription": "CPU usage exceeds 70 percent",
            "StateReasonData": "{\"version\":\"1.0\",\"queryDate\":
\"2014-04-09T18:59:06.419+0000\",\"startDate\":\"2014-04-09T18:44:00.000+0000\",
\"statistic\":\"Average\",\"period\":300,\"recentDatapoints\":[38.958,40.292],
\"threshold\":70.0}",
            "Period": 300,
            "StateValue": "OK",
            "Threshold": 70.0,
            "AlarmName": "myalarm",
            "Dimensions": [
                {
                    "Name": "InstanceId",
                    "Value": "i-0c986c72"
                }
            ],
            "Statistic": "Average",
            "StateReason": "Threshold Crossed: 2 datapoints were not greater than
 the threshold (70.0). The most recent datapoints: [38.958, 40.292].",
            "InsufficientDataActions": [],
            "OKActions": [],
            "ActionsEnabled": true,
            "MetricName": "CPUUtilization"
        }
    ]
}
```

- Per i dettagli sull'API, consulta DescribeAlarms AWS CLICommand Reference.

`describe-anomaly-detectors`

Il seguente esempio di codice mostra come utilizzare`describe-anomaly-detectors`.

AWS CLI

Come recuperare un elenco di modelli di rilevamento delle anomalie

L'esempio `describe-anomaly-detectors` seguente visualizza le informazioni sui modelli di rilevatore di anomalie associati al namespace `AWS/Logs` nomi nell'account specificato.

```
aws cloudwatch describe-anomaly-detectors \
    --namespace AWS/Logs
```

Output:


```
{
    "AnomalyDetectors": [
        {
            "Namespace": "AWS/Logs",
            "MetricName": "IncomingBytes",
            "Dimensions": [],
            "Stat": "SampleCount",
            "Configuration": {
                "ExcludedTimeRanges": []
            },
            "StateValue": "TRAINED",
            "SingleMetricAnomalyDetector": {
                "AccountId": "123456789012",
                "Namespace": "AWS/Logs",
                "MetricName": "IncomingBytes",
                "Dimensions": [],
                "Stat": "SampleCount"
            }
        },
        {
            "Namespace": "AWS/Logs",
            "MetricName": "IncomingBytes",
            "Dimensions": [
                {
                    "Name": "LogGroupName",
                    "Value": "demo"
                }
            ],
            "Stat": "Average",
            "Configuration": {
                "ExcludedTimeRanges": []
            },
            "StateValue": "PENDING_TRAINING",
            "SingleMetricAnomalyDetector": {
                "AccountId": "123456789012",
                "Namespace": "AWS/Logs",
                "MetricName": "IncomingBytes",
                "Dimensions": [
                    {
                        "Name": "LogGroupName",
                        "Value": "demo"
                    }
```

```
                ],
                "Stat": "Average"
            }
        }
    ]
}
```

Per ulteriori informazioni, consulta Using CloudWatch anomaly detection nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta DescribeAnomalyDetectors AWS CLICommand Reference.

`describe-insight-rules`

Il seguente esempio di codice mostra come utilizzare`describe-insight-rules`.

AWS CLI

Come recuperare un elenco di regole di Contributor Insights

L'esempio `describe-insight-rules` seguente mostra tutte le regole di Contributor Insights nell'account specificato.

```
aws cloudwatch describe-insight-rules
```

Output:

```
{
    "InsightRules": [
        {
            "Name": "Rule-A",
            "State": "ENABLED",
            "Schema": "CloudWatchLogRule/1",
            "Definition": "{\n\t\"AggregateOn\": \"Count\",\n\t\"Contribution\":
 {\n\t\t\"Filters\": [],\n\t\t\"Keys\": [\n\t\t\t\"$.requestId\"\n\t\t]\n\t},\n
\t\"LogFormat\": \"JSON\",\n\t\"Schema\": {\n\t\t\"Name\": \"CloudWatchLogRule
\",\n\t\t\"Version\": 1\n\t},\n\t\"LogGroupARNs\": [\n\t\t\"arn:aws:logs:us-
east-1:123456789012:log-group:demo\"\n\t]\n}",
            "ManagedRule": false
        },
        {
            "Name": "Rule-B",
```

```
            "State": "ENABLED",
            "Schema": "CloudWatchLogRule/1",
            "Definition": "{\n\t\"AggregateOn\": \"Count\",\n\t\"Contribution\":
 {\n\t\t\"Filters\": [],\n\t\t\"Keys\": [\n\t\t\t\"$.requestId\"\n\t\t]\n\t},\n
\t\"LogFormat\": \"JSON\",\n\t\"Schema\": {\n\t\t\"Name\": \"CloudWatchLogRule
\",\n\t\t\"Version\": 1\n\t},\n\t\"LogGroupARNs\": [\n\t\t\"arn:aws:logs:us-
east-1:123456789012:log-group:demo-1\"\n\t]\n}",
            "ManagedRule": false
        }
    ]
}
```

Per ulteriori informazioni, consulta Usa Contributor Insights per analizzare i dati ad alta cardinalità nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta Command DescribeInsightRulesReference AWS CLI .

`disable-alarm-actions`

Il seguente esempio di codice mostra come utilizzare`disable-alarm-actions`.

AWS CLI

Come disattivare le operazioni su un allarme

L'esempio seguente utilizza il comando `disable-alarm-actions` per disabilitare tutte le operazioni per l'allarme denominato myalarm:

```
aws cloudwatch disable-alarm-actions --alarm-names myalarm
```

In caso di esito positivo, il comando torna al prompt.

- Per i dettagli sull'API, consulta DisableAlarmActions AWS CLICommand Reference.

`disable-insight-rules`

Il seguente esempio di codice mostra come utilizzare`disable-insight-rules`.

AWS CLI

Come disabilitare le regole di Contributor Insights specificate

L'esempio `disable-insight-rules` seguente disabilita due regole di Contributor Insights denominate `Rule-A` e `Rule-B` nell'account specificato.

```
aws cloudwatch disable-insight-rules \
    --rule-names Rule-A Rule-B
```

Output:

```
{
    "Failures": []
}
```

Per ulteriori informazioni, consulta Usa Contributor Insights per analizzare i dati ad alta cardinalità nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta Command DisableInsightRulesReference AWS CLI .

`enable-alarm-actions`

Il seguente esempio di codice mostra come utilizzare`enable-alarm-actions`.

AWS CLI

Come abilitare tutte le operazioni relative a un allarme

L'esempio seguente utilizza il comando `enable-alarm-actions` per abilitare tutte le operazioni per l'allarme denominato myalarm:

```
aws cloudwatch enable-alarm-actions --alarm-names myalarm
```

In caso di esito positivo, il comando torna al prompt.

- Per i dettagli sull'API, consulta EnableAlarmActions AWS CLICommand Reference.

`enable-insight-rules`

Il seguente esempio di codice mostra come utilizzare`enable-insight-rules`.

AWS CLI

Come abilitare le regole di Contributor Insights specificate

L'esempio `enable-insight-rules` seguente abilita due regole di Contributor Insights denominate `Rule-A` e `Rule-B` nell'account specificato.

```
aws cloudwatch enable-insight-rules \
    --rule-names Rule-A Rule-B
```

Output:

```
{
    "Failures": []
}
```

Per ulteriori informazioni, consulta Usa Contributor Insights per analizzare i dati ad alta cardinalità nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta Command EnableInsightRulesReference AWS CLI .

`get-dashboard`

Il seguente esempio di codice mostra come utilizzare`get-dashboard`.

AWS CLI

Come recuperare le informazioni relative a una dashboard

L'esempio `get-dashboard` seguente visualizza le informazioni sulla dashboard denominata `Dashboard-A` nell'account specificato.

```
aws cloudwatch get-dashboard \
    --dashboard-name Dashboard-A
```

Output:

```
{
    "DashboardArn": "arn:aws:cloudwatch::123456789012:dashboard/Dashboard-A",
    "DashboardBody": "{\"widgets\":[{\"type\":\"metric\",\"x\":0,\"y\":0,\"width
\":6,\"height\":6,\"properties\":{\"view\":\"timeSeries\",\"stacked\":false,
\"metrics\":[[\"AWS/EC2\",\"NetworkIn\",\"InstanceId\",\"i-0131f062232ade043\"],[\".
\",\"NetworkOut\",\".\",\".\"]],\"region\":\"us-east-1\"}}]}",
    "DashboardName": "Dashboard-A"
```

```
}
```

Per ulteriori informazioni, consulta i CloudWatch dashboard di Amazon nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta GetDashboard AWS CLICommand Reference.

`get-insight-rule-report`

Il seguente esempio di codice mostra come utilizzare`get-insight-rule-report`.

AWS CLI

Come recuperare i dati di serie temporali raccolti da una regola di Contributor Insights

L'esempio `get-insight-rule-report` seguente restituisce i dati di serie temporali raccolti da una regola di Contributor Insights.

```
aws cloudwatch get-insight-rule-report \
    --rule-name Rule-A \
    --start-time 2024-10-13T20:15:00Z \
    --end-time 2024-10-13T20:30:00Z \
    --period 300
```

Output:

```
{
    "KeyLabels": [
        "PartitionKey"
    ],
    "AggregationStatistic": "Sum",
    "AggregateValue": 0.5,
    "ApproximateUniqueCount": 1,
    "Contributors": [
        {
            "Keys": [
                "RequestID"
            ],
            "ApproximateAggregateValue": 0.5,
            "Datapoints": [
                {
                    "Timestamp": "2024-10-13T21:00:00+00:00",
```

```
                    "ApproximateValue": 0.5
                }
            ]
        }
    ],
    "RuleAttributes": []
}
```

Per ulteriori informazioni, consulta Usa Contributor Insights per analizzare i dati ad alta cardinalità nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta Command GetInsightRuleReportReference AWS CLI .

`get-metric-data`

Il seguente esempio di codice mostra come utilizzare`get-metric-data`.

AWS CLI

Esempio 1: Per ottenere gli IOPS totali medi per il valore specificato EC2 utilizzando un'espressione matematica

L'`get-metric-data`esempio seguente recupera i valori delle CloudWatch metriche per l' EC2 istanza con InstanceID `i-abcdef` utilizzando un'espressione matematica metrica che combina e metriche. `EBSReadOps` `EBSWriteOps`

```
aws cloudwatch get-metric-data \
    --metric-data-queries file://file.json \
    --start-time 2024-09-29T22:10:00Z \
    --end-time 2024-09-29T22:15:00Z
```

Contenuto di `file.json`:

```
[
    {
        "Id": "m3",
        "Expression": "(m1+m2)/300",
        "Label": "Avg Total IOPS"
    },
    {
        "Id": "m1",
        "MetricStat": {
```

```
            "Metric": {
                "Namespace": "AWS/EC2",
                "MetricName": "EBSReadOps",
                "Dimensions": [
                    {
                        "Name": "InstanceId",
                        "Value": "i-abcdef"
                    }
                ]
            },
            "Period": 300,
            "Stat": "Sum",
            "Unit": "Count"
        },
        "ReturnData": false
    },
    {
        "Id": "m2",
        "MetricStat": {
            "Metric": {
                "Namespace": "AWS/EC2",
                "MetricName": "EBSWriteOps",
                "Dimensions": [
                    {
                        "Name": "InstanceId",
                        "Value": "i-abcdef"
                    }
                ]
            },
            "Period": 300,
            "Stat": "Sum",
            "Unit": "Count"
        },
        "ReturnData": false
    }
]
```

Output:

```
{
    "MetricDataResults": [
        {
            "Id": "m3",
```

```
            "Label": "Avg Total IOPS",
            "Timestamps": [
                "2024-09-29T22:10:00+00:00"
            ],
            "Values": [
                96.85
            ],
            "StatusCode": "Complete"
        }
    ],
    "Messages": []
}
```

Esempio 2: per monitorare gli addebiti stimati utilizzando i parametri di fatturazione AWS CloudWatch

L'`get-metric-data`esempio seguente recupera la `EstimatedCharges` CloudWatch metrica dallo spazio dei nomi /Billing. AWS

```
aws cloudwatch get-metric-data \
    --metric-data-queries '[{"Id":"m1","MetricStat":{"Metric":
{"Namespace":"AWS/Billing","MetricName":"EstimatedCharges","Dimensions":
[{"Name":"Currency","Value":"USD"}]},"Period":21600,"Stat":"Maximum"}}]' \
    --start-time 2024-09-26T12:00:00Z \
    --end-time 2024-09-26T18:00:00Z \
    --region us-east-1
```

Output:

```
{
    "MetricDataResults": [
        {
            "Id": "m1",
            "Label": "EstimatedCharges",
            "Timestamps": [
                "2024-09-26T12:00:00+00:00"
            ],
            "Values": [
                542.38
            ],
            "StatusCode": "Complete"
        }
    ],
```

```
    "Messages": []
}
```

Per ulteriori informazioni, consulta Using math expression with CloudWatch metrics nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta AWS CLI Command GetMetricDataReference.

`get-metric-statistics`

Il seguente esempio di codice mostra come utilizzare`get-metric-statistics`.

AWS CLI

Per ottenere l'utilizzo della CPU per istanza EC2

L'esempio seguente utilizza il `get-metric-statistics` comando per ottenere l'utilizzo della CPU per un' EC2istanza con l'ID i-abcdef.

```
aws cloudwatch get-metric-statistics --metric-name CPUUtilization --start-
time 2014-04-08T23:18:00Z --end-time 2014-04-09T23:18:00Z --period 3600 --
namespace AWS/EC2 --statistics Maximum --dimensions Name=InstanceId,Value=i-abcdef
```

Output:

```
{
    "Datapoints": [
        {
            "Timestamp": "2014-04-09T11:18:00Z",
            "Maximum": 44.79,
            "Unit": "Percent"
        },
        {
            "Timestamp": "2014-04-09T20:18:00Z",
            "Maximum": 47.92,
            "Unit": "Percent"
        },
        {
            "Timestamp": "2014-04-09T19:18:00Z",
            "Maximum": 50.85,
            "Unit": "Percent"
        },
        {
```

```
            "Timestamp": "2014-04-09T09:18:00Z",
            "Maximum": 47.92,
            "Unit": "Percent"
        },
        {
            "Timestamp": "2014-04-09T03:18:00Z",
            "Maximum": 76.84,
            "Unit": "Percent"
        },
        {
            "Timestamp": "2014-04-09T21:18:00Z",
            "Maximum": 48.96,
            "Unit": "Percent"
        },
        {
            "Timestamp": "2014-04-09T14:18:00Z",
            "Maximum": 47.92,
            "Unit": "Percent"
        },
        {
            "Timestamp": "2014-04-09T08:18:00Z",
            "Maximum": 47.92,
            "Unit": "Percent"
        },
        {
            "Timestamp": "2014-04-09T16:18:00Z",
            "Maximum": 45.55,
            "Unit": "Percent"
        },
        {
            "Timestamp": "2014-04-09T06:18:00Z",
            "Maximum": 47.92,
            "Unit": "Percent"
        },
        {
            "Timestamp": "2014-04-09T13:18:00Z",
            "Maximum": 45.08,
            "Unit": "Percent"
        },
        {
            "Timestamp": "2014-04-09T05:18:00Z",
            "Maximum": 47.92,
            "Unit": "Percent"
        },
```

```
        {
            "Timestamp": "2014-04-09T18:18:00Z",
            "Maximum": 46.88,
            "Unit": "Percent"
        },
        {
            "Timestamp": "2014-04-09T17:18:00Z",
            "Maximum": 52.08,
            "Unit": "Percent"
        },
        {
            "Timestamp": "2014-04-09T07:18:00Z",
            "Maximum": 47.92,
            "Unit": "Percent"
        },
        {
            "Timestamp": "2014-04-09T02:18:00Z",
            "Maximum": 51.23,
            "Unit": "Percent"
        },
        {
            "Timestamp": "2014-04-09T12:18:00Z",
            "Maximum": 47.67,
            "Unit": "Percent"
        },
        {
            "Timestamp": "2014-04-08T23:18:00Z",
            "Maximum": 46.88,
            "Unit": "Percent"
        },
        {
            "Timestamp": "2014-04-09T10:18:00Z",
            "Maximum": 51.91,
            "Unit": "Percent"
        },
        {
            "Timestamp": "2014-04-09T04:18:00Z",
            "Maximum": 47.13,
            "Unit": "Percent"
        },
        {
            "Timestamp": "2014-04-09T15:18:00Z",
            "Maximum": 48.96,
            "Unit": "Percent"
```

```
        },
        {
            "Timestamp": "2014-04-09T00:18:00Z",
            "Maximum": 48.16,
            "Unit": "Percent"
        },
        {
            "Timestamp": "2014-04-09T01:18:00Z",
            "Maximum": 49.18,
            "Unit": "Percent"
        }
    ],
    "Label": "CPUUtilization"
}
```

Specifica di più dimensioni

Nell'esempio seguente viene illustrato come specificare più dimensioni. Ogni dimensione è specificata come Name/Value coppia, con una virgola tra il nome e il valore. Più dimensioni sono separate da uno spazio. Se un unico parametro include più dimensioni, è necessario specificare un valore per ogni dimensione definita.

Per altri esempi di utilizzo del `get-metric-statistics` comando, consulta Get Statistics for a Metric nella Amazon CloudWatch Developer Guide.

```
aws cloudwatch get-metric-statistics --metric-name Buffers --namespace MyNameSpace
 --dimensions Name=InstanceID,Value=i-abcdef Name=InstanceType,Value=m1.small --
start-time 2016-10-15T04:00:00Z --end-time 2016-10-19T07:00:00Z --statistics Average
 --period 60
```

- Per i dettagli sull'API, consulta GetMetricStatistics AWS CLICommand Reference.

`get-metric-stream`

Il seguente esempio di codice mostra come utilizzare`get-metric-stream`.

AWS CLI

Come recuperare informazioni su un flusso di metriche

L'esempio `get-metric-stream` seguente visualizza informazioni sul flusso di metriche denominato `QuickFull-GuaFbs` nell'account specificato.

```
aws cloudwatch get-metric-stream \
    --name QuickFull-GuaFbs
```

Output:

```
{
    "Arn": "arn:aws:cloudwatch:us-east-1:123456789012:metric-stream/QuickFull-
GuaFbs",
    "Name": "QuickFull-GuaFbs",
    "FirehoseArn": "arn:aws:firehose:us-east-1:123456789012:deliverystream/
MetricStreams-QuickFull-GuaFbs-WnySbECG",
    "RoleArn": "arn:aws:iam::123456789012:role/service-role/MetricStreams-
FirehosePutRecords-JN10W9B3",
    "State": "running",
    "CreationDate": "2024-10-11T18:48:59.187000+00:00",
    "LastUpdateDate": "2024-10-11T18:48:59.187000+00:00",
    "OutputFormat": "json",
    "IncludeLinkedAccountsMetrics": false
}
```

Per ulteriori informazioni, consulta Use metric Streams nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta AWS CLI Command GetMetricStreamReference.

`get-metric-widget-image`

Il seguente esempio di codice mostra come utilizzare`get-metric-widget-image`.

AWS CLI

Per recuperare un grafico istantaneo di CPUUtilization

L'`get-metric-widget-image`esempio seguente recupera il grafico dell'istantanea per la metrica `CPUUtilization` dell' EC2 istanza con l'ID `i-abcde` e salva l'immagine recuperata come file denominato "image.png" sul computer locale.

```
aws cloudwatch get-metric-widget-image \
    --metric-widget '{"metrics":[["AWS/EC2","CPUUtilization","InstanceId","i-
abcde"]]}' \
    --output-format png \
    --output text | base64 --decode > image.png
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta GetMetricWidgetImageCommand Reference.AWS CLI

`list-dashboards`

Il seguente esempio di codice mostra come utilizzare`list-dashboards`.

AWS CLI

Come recuperare un elenco di dashboard

L'esempio `list-dashboards` seguente elenca tutte le dashboard dell'account specificato.

```
aws cloudwatch list-dashboards
```

Output:

```
{
    "DashboardEntries": [
        {
            "DashboardName": "Dashboard-A",
            "DashboardArn": "arn:aws:cloudwatch::123456789012:dashboard/Dashboard-
A",
            "LastModified": "2024-10-11T18:40:11+00:00",
            "Size": 271
        },
        {
            "DashboardName": "Dashboard-B",
            "DashboardArn": "arn:aws:cloudwatch::123456789012:dashboard/Dashboard-
B",
            "LastModified": "2024-10-11T18:44:41+00:00",
            "Size": 522
        }
    ]
}
```

Per ulteriori informazioni, consulta i CloudWatch dashboard di Amazon nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta ListDashboards AWS CLICommand Reference.

`list-metric-streams`

Il seguente esempio di codice mostra come utilizzare`list-metric-streams`.

AWS CLI

Come recuperare un elenco di flussi di metriche

L'esempio `list-metric-streams` seguente elenca tutti i flussi di metriche nell'account specificato.

```
aws cloudwatch list-metric-streams
```

Output:

```
{
    "Entries": [
        {
            "Arn": "arn:aws:cloudwatch:us-east-1:123456789012:metric-stream/
QuickFull-GuaFbs",
            "CreationDate": "2024-10-11T18:48:59.187000+00:00",
            "LastUpdateDate": "2024-10-11T18:48:59.187000+00:00",
            "Name": "QuickFull-GuaFbs",
            "FirehoseArn": "arn:aws:firehose:us-east-1:123456789012:deliverystream/
MetricStreams-QuickFull-GuaFbs-WnySbECG",
            "State": "running",
            "OutputFormat": "json"
        }
    ]
}
```

Per ulteriori informazioni, consulta Use metric Streams nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta AWS CLI Command ListMetricStreamsReference.

`list-metrics`

Il seguente esempio di codice mostra come utilizzare`list-metrics`.

AWS CLI

Come elencare i parametri per Amazon SNS

L'esempio `list-metrics` seguente mostra i parametri per Amazon SNS.

```
aws cloudwatch list-metrics \
    --namespace "AWS/SNS"
```

Output:

```
{
    "Metrics": [
        {
            "Namespace": "AWS/SNS",
            "Dimensions": [
                {
                    "Name": "TopicName",
                    "Value": "NotifyMe"
                }
            ],
            "MetricName": "PublishSize"
        },
        {
            "Namespace": "AWS/SNS",
            "Dimensions": [
                {
                    "Name": "TopicName",
                    "Value": "CFO"
                }
            ],
            "MetricName": "PublishSize"
        },
        {
            "Namespace": "AWS/SNS",
            "Dimensions": [
                {
                    "Name": "TopicName",
                    "Value": "NotifyMe"
                }
            ],
            "MetricName": "NumberOfNotificationsFailed"
        },
        {
            "Namespace": "AWS/SNS",
            "Dimensions": [
                {
```

```
                    "Name": "TopicName",
                    "Value": "NotifyMe"
                }
            ],
            "MetricName": "NumberOfNotificationsDelivered"
        },
        {
            "Namespace": "AWS/SNS",
            "Dimensions": [
                {
                    "Name": "TopicName",
                    "Value": "NotifyMe"
                }
            ],
            "MetricName": "NumberOfMessagesPublished"
        },
        {
            "Namespace": "AWS/SNS",
            "Dimensions": [
                {
                    "Name": "TopicName",
                    "Value": "CFO"
                }
            ],
            "MetricName": "NumberOfMessagesPublished"
        },
        {
            "Namespace": "AWS/SNS",
            "Dimensions": [
                {
                    "Name": "TopicName",
                    "Value": "CFO"
                }
            ],
            "MetricName": "NumberOfNotificationsDelivered"
        },
        {
            "Namespace": "AWS/SNS",
            "Dimensions": [
                {
                    "Name": "TopicName",
                    "Value": "CFO"
                }
            ],
```

```
            "MetricName": "NumberOfNotificationsFailed"
        }
    ]
}
```

- Per i dettagli sull'API, consulta ListMetrics AWS CLICommand Reference.

`list-tags-for-resource`

Il seguente esempio di codice mostra come utilizzare`list-tags-for-resource`.

AWS CLI

Come elencare i tag associati a un allarme esistente*

L'esempio `list-tags-for-resource` seguente elenca tutti i tag associati a un allarme denominato demo nell'account specificato.

```
aws cloudwatch list-tags-for-resource \
    --resource-arn arn:aws:cloudwatch:us-east-1:123456789012:alarm:demo
```

Output:

```
{
    "Tags": [
        {
            "Key": "stack",
            "Value": "Production"
        },
        {
            "Key": "team",
            "Value": "Devops"
        }
    ]
}
```

Per ulteriori informazioni, consulta Allarmi e tag nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta AWS CLI Command ListTagsForResourceReference.

`put-anomaly-detector`

Il seguente esempio di codice mostra come utilizzare`put-anomaly-detector`.

AWS CLI

Come creare un modello di rilevamento delle anomalie

L'`put-anomaly-detector`esempio seguente crea un modello di rilevamento delle anomalie per una CloudWatch metrica.

```
aws cloudwatch put-anomaly-detector \
    --namespace AWS/Logs \
    --metric-name IncomingBytes \
    --stat SampleCount
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Using CloudWatch anomaly detection nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta PutAnomalyDetector AWS CLICommand Reference.

`put-composite-alarm`

Il seguente esempio di codice mostra come utilizzare`put-composite-alarm`.

AWS CLI

Come creare un allarme CloudWatch composito

L'esempio `put-composite-alarm` seguente crea un allarme composito denominato `ProdAlarm` nell'account specificato.

```
aws cloudwatch put-composite-alarm \
    --alarm-name ProdAlarm \
    --alarm-rule "ALARM(CPUUtilizationTooHigh) AND ALARM(MemUsageTooHigh)" \
    --alarm-actions arn:aws:sns:us-east-1:123456789012:demo \
    --actions-enabled
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Creare un allarme composito nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta PutCompositeAlarm AWS CLICommand Reference.

`put-dashboard`

Il seguente esempio di codice mostra come utilizzare`put-dashboard`.

AWS CLI

Come creare una dashboard

L'esempio `put-dashboard` seguente elenca una dashboard denominata `Dashboard-A` nell'account specificato.

```
aws cloudwatch put-dashboard \
    --dashboard-name Dashboard-A \
    --dashboard-body '{"widgets":
[{"height":6,"width":6,"y":0,"x":0,"type":"metric","properties":
{"view":"timeSeries","stacked":false,"metrics":
[["Namespace","CPUUtilization","Environment","Prod","Type","App"]],"region":"us-
east-1"}}]}'
```

Output:

```
{
    "DashboardValidationMessages": []
}
```

Per ulteriori informazioni, consulta la sezione Creazione di un CloudWatch pannello di controllo nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta PutDashboard AWS CLICommand Reference.

`put-insight-rule`

Il seguente esempio di codice mostra come utilizzare`put-insight-rule`.

AWS CLI

Come creare una regola di Contributor Insights

L'esempio `put-insight-rule` seguente crea una regola di Contributor Insights denominata `VPCFlowLogsContributorInsights` nell'account specificato.

```
aws cloudwatch put-insight-rule \
    --rule-name VPCFlowLogsContributorInsights \
    --rule-definition file://insight-rule.json \
    --rule-state ENABLED
```

Contenuto di `insight-rule.json`:

```
{
    "Schema": {
        "Name": "CloudWatchLogRule",
        "Version": 1
    },
    "AggregateOn": "Count",
    "Contribution": {
        "Filters": [],
        "Keys": [
            "tcp-flag"
        ]
    },
    "LogFormat": "CLF",
    "LogGroupNames": [
        "/vpc/flowlogs/*"
    ],
    "Fields": {
        "23": "tcp-flag"
    }
}
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta la sezione Create a Contributor Insights CloudWatch nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta PutInsightRule AWS CLICommand Reference.

`put-metric-alarm`

Il seguente esempio di codice mostra come utilizzare`put-metric-alarm`.

AWS CLI

Come inviare un messaggio e-mail Amazon Simple Notification Service quando l'utilizzo della CPU supera il 70%

Nell'esempio seguente viene utilizzato il comando `put-metric-alarm` per inviare un messaggio e-mail Amazon Simple Notification Service quando l'utilizzo della CPU supera il 70%:

```
aws cloudwatch put-metric-alarm --alarm-name cpu-mon --alarm-description "Alarm
 when CPU exceeds 70 percent" --metric-name CPUUtilization --namespace AWS/
EC2 --statistic Average --period 300 --threshold 70 --comparison-
operator GreaterThanThreshold  --dimensions "Name=InstanceId,Value=i-12345678" --
evaluation-periods 2 --alarm-actions arn:aws:sns:us-east-1:111122223333:MyTopic --
unit Percent
```

In caso di esito positivo, il comando torna al prompt. Se esiste già un allarme con lo stesso nome, verrà sovrascritto dal nuovo allarme.

Come specificare più dimensioni

Nell'esempio seguente viene illustrato come specificare più dimensioni. Ogni dimensione è specificata come Name/Value coppia, con una virgola tra il nome e il valore. Più dimensioni sono separate da uno spazio:

```
aws cloudwatch put-metric-alarm --alarm-name "Default_Test_Alarm3" --alarm-
description "The default example alarm" --namespace "CW EXAMPLE METRICS"
 --metric-name Default_Test --statistic Average --period 60 --evaluation-
periods 3 --threshold 50 --comparison-operator GreaterThanOrEqualToThreshold --
dimensions Name=key1,Value=value1 Name=key2,Value=value2
```

- Per i dettagli sull'API, consulta PutMetricAlarm AWS CLICommand Reference.

`put-metric-data`

Il seguente esempio di codice mostra come utilizzare`put-metric-data`.

AWS CLI

Per pubblicare una metrica personalizzata su Amazon CloudWatch

L'esempio seguente utilizza il `put-metric-data` comando per pubblicare una metrica personalizzata su Amazon CloudWatch:

```
aws cloudwatch put-metric-data --namespace "Usage Metrics" --metric-data file://
metric.json
```

I valori del parametro stesso sono memorizzati nel file JSON, `metric.json`.

Ecco i contenuti del file:

```
[
  {
    "MetricName": "New Posts",
    "Timestamp": "Wednesday, June 12, 2013 8:28:20 PM",
    "Value": 0.50,
    "Unit": "Count"
  }
]
```

Per ulteriori informazioni, consulta Publishing Custom Metrics nella Amazon CloudWatch Developer Guide.

Come specificare più dimensioni

Nell'esempio seguente viene illustrato come specificare più dimensioni. Ogni dimensione è specificata come coppia Name=Valore. Più dimensioni sono separate da una virgola:

```
aws cloudwatch put-metric-data --metric-name Buffers --
namespace MyNameSpace --unit Bytes --value 231434333 --
dimensions InstanceID=1-23456789,InstanceType=m1.small
```

- Per i dettagli sull'API, consulta PutMetricData AWS CLICommand Reference.

`put-metric-stream`

Il seguente esempio di codice mostra come utilizzare`put-metric-stream`.

AWS CLI

Come creare un flusso di metriche

L'esempio `put-metric-stream` seguente crea un flusso di metriche denominato `QuickFull-GuaFb` nell'account specificato.

```
aws cloudwatch put-metric-stream \
```

```
    --name QuickFull-GuaFbs \
    --firehose-arn arn:aws:firehose:us-east-1:123456789012:deliverystream/
MetricStreams-QuickFull-GuaFbs-WnySbECG \
    --role-arn arn:aws:iam::123456789012:role/service-role/MetricStreams-
FirehosePutRecords-JN10W9B3 \
    --output-format json \
    --no-include-linked-accounts-metrics
```

Output:

```
{
    "Arn": "arn:aws:cloudwatch:us-east-1:123456789012:metric-stream/QuickFull-
GuaFbs"
}
```

Per ulteriori informazioni, consulta Configurare un flusso di parametri nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta PutMetricStream AWS CLICommand Reference.

`set-alarm-state`

Il seguente esempio di codice mostra come utilizzare`set-alarm-state`.

AWS CLI

Come modificare temporaneamente lo stato di un allarme

L'esempio seguente utilizza il `set-alarm-state` comando per modificare temporaneamente lo stato di un CloudWatch allarme Amazon denominato «myalarm» e impostarlo sullo stato ALARM a scopo di test:

```
aws cloudwatch set-alarm-state --alarm-name "myalarm" --state-value ALARM --state-
reason "testing purposes"
```

In caso di esito positivo, il comando torna al prompt.

- Per i dettagli sull'API, consulta SetAlarmState AWS CLICommand Reference.

`start-metric-streams`

Il seguente esempio di codice mostra come utilizzare`start-metric-streams`.

AWS CLI

Come avviare un flusso di metriche specificato

L'esempio `start-metric-streams` seguente avvia il flusso di metriche denominato `QuickFull-GuaFbs` nell'account specificato.

```
aws cloudwatch start-metric-streams \
    --names QuickFull-GuaFbs
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Use metric Streams nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta AWS CLI Command StartMetricStreamsReference.

`stop-metric-streams`

Il seguente esempio di codice mostra come utilizzare`stop-metric-streams`.

AWS CLI

Come arrestare un flusso di metriche specificato

L'esempio `stop-metric-streams` seguente arresta il flusso di metriche denominato `QuickFull-GuaFbs` nell'account specificato.

```
aws cloudwatch stop-metric-streams \
    --names QuickFull-GuaFbs
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Use metric Streams nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta AWS CLI Command StopMetricStreamsReference.

`tag-resource`

Il seguente esempio di codice mostra come utilizzare`tag-resource`.

AWS CLI

Come aggiungere uno o più tag alla risorsa specificata

L'esempio `tag-resource` seguente aggiunge 2 tag all'allarme CloudWatch denominato demo nell'account specificato.

```
aws cloudwatch tag-resource \
    --resource-arn arn:aws:cloudwatch:us-east-1:123456789012:alarm:demo \
    --tags Key=stack,Value=Production Key=team,Value=Devops
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Taggare le CloudWatch risorse Amazon nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta TagResource AWS CLICommand Reference.

`untag-resource`

Il seguente esempio di codice mostra come utilizzare`untag-resource`.

AWS CLI

Come rimuovere uno o più tag dalla risorsa specificata

L'esempio `untag-resource` seguente rimuove 2 tag dall'allarme CloudWatch denominato demo nell'account specificato.

```
aws cloudwatch untag-resource \
    --resource-arn arn:aws:cloudwatch:us-east-1:123456789012:alarm:demo \
    --tag-keys stack team
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Taggare le CloudWatch risorse Amazon nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta UntagResource AWS CLICommand Reference.

CloudWatch Esempi di log utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with CloudWatch Logs.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`create-log-group`

Il seguente esempio di codice mostra come utilizzare`create-log-group`.

AWS CLI

Il comando seguente crea un gruppo di log denominato `my-logs`.

```
aws logs create-log-group --log-group-name my-logs
```

- Per i dettagli sull'API, consulta CreateLogGroup AWS CLICommand Reference.

`create-log-stream`

Il seguente esempio di codice mostra come utilizzare`create-log-stream`.

AWS CLI

Il comando seguente crea un flusso di log denominato `20150601` nel gruppo di log `my-logs`.

```
aws logs create-log-stream --log-group-name my-logs --log-stream-name 20150601
```

- Per i dettagli sull'API, consulta CreateLogStream AWS CLICommand Reference.

`delete-log-group`

Il seguente esempio di codice mostra come utilizzare`delete-log-group`.

AWS CLI

Il comando seguente elimina un gruppo di log denominato `my-logs`.

```
aws logs delete-log-group --log-group-name my-logs
```

- Per i dettagli sull'API, consulta DeleteLogGroup AWS CLICommand Reference.

`delete-log-stream`

Il seguente esempio di codice mostra come utilizzare`delete-log-stream`.

AWS CLI

Il comando seguente elimina un flusso di log denominato `20150531` da un gruppo di log denominato `my-logs`:

```
aws logs delete-log-stream --log-group-name my-logs --log-stream-name 20150531
```

- Per i dettagli sull'API, consulta DeleteLogStream AWS CLICommand Reference.

`delete-retention-policy`

Il seguente esempio di codice mostra come utilizzare`delete-retention-policy`.

AWS CLI

Il comando seguente rimuove la policy di conservazione precedentemente applicata a un gruppo di log denominato `my-logs`:

```
aws logs delete-retention-policy --log-group-name my-logs
```

- Per i dettagli sull'API, consulta DeleteRetentionPolicy AWS CLICommand Reference.

`describe-log-groups`

Il seguente esempio di codice mostra come utilizzare`describe-log-groups`.

AWS CLI

Il comando seguente descrive un gruppo di log denominato `my-logs`.

```
aws logs describe-log-groups --log-group-name-prefix my-logs
```

Output:

```
{
    "logGroups": [
        {
            "storedBytes": 0,
            "metricFilterCount": 0,
            "creationTime": 1433189500783,
            "logGroupName": "my-logs",
            "retentionInDays": 5,
            "arn": "arn:aws:logs:us-west-2:0123456789012:log-group:my-logs:*"
        }
    ]
}
```

- Per i dettagli sull'API, consulta DescribeLogGroups AWS CLICommand Reference.

`describe-log-streams`

Il seguente esempio di codice mostra come utilizzare`describe-log-streams`.

AWS CLI

Il comando seguente mostra tutti i flussi di log che iniziano con il prefisso `2015` nel gruppo di log `my-logs`.

```
aws logs describe-log-streams --log-group-name my-logs --log-stream-name-prefix 2015
```

Output:

```
{
    "logStreams": [
        {
            "creationTime": 1433189871774,
            "arn": "arn:aws:logs:us-west-2:0123456789012:log-group:my-logs:log-
stream:20150531",
            "logStreamName": "20150531",
            "storedBytes": 0
```

```
        },
        {
            "creationTime": 1433189873898,
            "arn": "arn:aws:logs:us-west-2:0123456789012:log-group:my-logs:log-
stream:20150601",
            "logStreamName": "20150601",
            "storedBytes": 0
        }
    ]
}
```

- Per i dettagli sull'API, consulta DescribeLogStreams AWS CLICommand Reference.

`get-log-events`

Il seguente esempio di codice mostra come utilizzare`get-log-events`.

AWS CLI

Il comando seguente recupera gli eventi del log da un flusso di log denominato `20150601` nel gruppo di log `my-logs`.

```
aws logs get-log-events --log-group-name my-logs --log-stream-name 20150601
```

Output:

```
{
    "nextForwardToken":
 "f/31961209122447488583055879464742346735121166569214640130",
    "events": [
        {
            "ingestionTime": 1433190494190,
            "timestamp": 1433190184356,
            "message": "Example Event 1"
        },
        {
            "ingestionTime": 1433190516679,
            "timestamp": 1433190184356,
            "message": "Example Event 1"
        },
        {
            "ingestionTime": 1433190494190,
```

```
            "timestamp": 1433190184358,
            "message": "Example Event 2"
        }
    ],
    "nextBackwardToken":
 "b/31961209122358285602261756944988674324553373268216709120"
}
```

- Per i dettagli sull'API, consulta GetLogEvents AWS CLICommand Reference.

`put-log-events`

Il seguente esempio di codice mostra come utilizzare`put-log-events`.

AWS CLI

Il comando seguente inserisce gli eventi di log in un flusso di log denominato `20150601` nel gruppo di log `my-logs`:

```
aws logs put-log-events --log-group-name my-logs --log-stream-name 20150601 --log-events file://events
```

Output:

```
{
    "nextSequenceToken": "49542672486831074009579604567656788214806863282469607346"
}
```

L'esempio precedente legge un array di eventi JSON da un file denominato `events` nella directory corrente:

```
[
  {
    "timestamp": 1433190184356,
    "message": "Example Event 1"
  },
  {
    "timestamp": 1433190184358,
    "message": "Example Event 2"
  },
```

```
  {
    "timestamp": 1433190184360,
    "message": "Example Event 3"
  }
]
```

Ogni chiamata successiva richiede che il token di sequenza successivo fornito dalla chiamata precedente venga specificato con l'opzione del token della sequenza:

```
aws logs put-log-events --log-group-name my-logs --log-
stream-name 20150601 --log-events file://events2 --sequence-
token "49542672486831074009579604567656788214806863282469607346"
```

Output:

```
{
    "nextSequenceToken": "49542672486831074009579604567900991230369019956308219826"
}
```

- Per i dettagli sull'API, consulta PutLogEvents AWS CLICommand Reference.

`put-retention-policy`

Il seguente esempio di codice mostra come utilizzare`put-retention-policy`.

AWS CLI

Il comando seguente aggiunge una policy di conservazione di 5 giorni a un gruppo di log denominato `my-logs`:

```
aws logs put-retention-policy --log-group-name my-logs --retention-in-days 5
```

- Per i dettagli sull'API, consulta PutRetentionPolicy AWS CLICommand Reference.

CloudWatch Esempi di monitoraggio della rete utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with CloudWatch Network Monitoring.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`create-monitor`

Il seguente esempio di codice mostra come utilizzare`create-monitor`.

AWS CLI

Esempio 1: come creare un monitoraggio di rete con un periodo di aggregazione

L'esempio `create-monitor` seguente crea un monitor denominato `Example_NetworkMonitor` con `aggregationPeriod` impostato su `30` secondi. Il valore `state` iniziale del monitoraggio sarà `INACTIVE` perché non vi sono sonde associate. Lo stato diventa `ACTIVE` solo quando vengono aggiunte le sonde. È possibile utilizzare i comandi update-monitor o create-probe per aggiungere sonde a questo monitoraggio.

```
aws networkmonitor create-monitor \
     --monitor-name Example_NetworkMonitor \
     --aggregation-period 30
```

Output:

```
{
    "monitorArn": "arn:aws:networkmonitor:region:111122223333:monitor/
Example_NetworkMonitor",
    "monitorName": "Example_NetworkMonitor",
    "state": "INACTIVE",
    "aggregationPeriod": 30,
    "tags": {}
```

```
}
```

Per ulteriori informazioni, consulta How CloudWatch Network Monitor Works nella Amazon CloudWatch User Guide.

Esempio 2: come creare un monitoraggio di rete con una sonda utilizzando TCP e includendo anche i tag

L'esempio `create-monitor` seguente crea un monitor denominato `Example_NetworkMonitor`. Il comando crea inoltre una sonda che utilizza il protocollo `ICMP` e include dei tag. Poiché nella richiesta non viene passato nessun valore `aggregationPeriod`, come valore predefinito sono impostati `60` secondi. Il valore `state` del monitoraggio con la sonda rimarrà `PENDING` finché il monitoraggio rimane `ACTIVE`. Questa operazione potrebbe richiedere alcuni minuti, dopodiché `state` cambierà e potrai iniziare a `ACTIVE` visualizzare le CloudWatch metriche.

```
aws networkmonitor create-monitor \
    --monitor-name Example_NetworkMonitor \
    --probes sourceArn=arn:aws:ec2:region:111122223333:subnet/subnet-
id,destination=10.0.0.100,destinationPort=80,protocol=TCP,packetSize=56,probeTags={Name=Prob
 \
    --tags Monitor=Monitor1
```

Output:

```
{
    "monitorArn": "arn:aws:networkmonitor:region111122223333:monitor/
Example_NetworkMonitor",
    "monitorName": "Example_NetworkMonitor",
    "state": "PENDING",
    "aggregationPeriod": 60,
    "tags": {
        "Monitor": "Monitor1"
    }
}
```

Per ulteriori informazioni, consulta How CloudWatch Network Monitor Works nella Amazon CloudWatch User Guide.

Esempio 3: come creare un monitoraggio di rete con una sonda utilizzando ICMP e includendo anche i tag

L'esempio `create-monitor` seguente crea un monitor denominato `Example_NetworkMonitor` con `aggregationPeriod` impostato su `30` secondi. Il comando crea inoltre una sonda che utilizza il protocollo `ICMP` e include dei tag. Poiché nella richiesta non viene passato nessun valore `aggregationPeriod`, come valore predefinito sono impostati `60` secondi. Il valore `state` del monitoraggio con la sonda rimarrà `PENDING` finché il monitoraggio rimane `ACTIVE`. Questa operazione potrebbe richiedere alcuni minuti, dopodiché `state` cambierà e potrai iniziare a `ACTIVE` visualizzare le CloudWatch metriche.

```
aws networkmonitor create-monitor \
     --monitor-name Example_NetworkMonitor \
     --aggregation-period 30 \
     --probes sourceArn=arn:aws:ec2:region111122223333:subnet/subnet-
id,destination=10.0.0.100,protocol=ICMP,packetSize=56,probeTags={Name=Probe1} \
     --tags Monitor=Monitor1
```

Output:

```
{
    "monitorArn": "arn:aws:networkmonitor:region:111122223333:monitor/
Example_NetworkMonitor",
    "monitorName": "Example_NetworkMonitor",
    "state": "PENDING",
    "aggregationPeriod": 30,
    "tags": {
        "Monitor": "Monitor1"
    }
}
```

Per ulteriori informazioni, consulta How CloudWatch Network Monitor Works nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta CreateMonitor AWS CLICommand Reference.

`create-probe`

Il seguente esempio di codice mostra come utilizzare`create-probe`.

AWS CLI

Esempio 1: come creare una sonda che utilizza TCP e aggiungerla a un monitoraggio di rete

L'esempio `create-probe` seguente crea una sonda con `protocol` impostato su `TCP` e aggiunge la sonda a un monitor denominato `Example_NetworkMonitor`. Una volta creato, il valore `state` del monitoraggio con la sonda rimarrà `PENDING` finché il monitoraggio rimane `ACTIVE`. Questa operazione potrebbe richiedere alcuni minuti, dopodiché lo stato cambierà e potrai iniziare a `ACTIVE` visualizzare le CloudWatch metriche.

```
aws networkmonitor create-probe \
    --monitor-name Example_NetworkMonitor \
    --probe sourceArn=arn:aws:ec2:region:111122223333:subnet/subnet-
id,destination=10.0.0.100,destinationPort=80,protocol=TCP,packetSize=56,tags={Name=Probe1}
```

Output:

```
{
    "probeId": "probe-12345",
    "probeArn": "arn:aws:networkmonitor:region:111122223333:probe/probe-12345",
    "destination": "10.0.0.100",
    "destinationPort": 80,
    "packetSize": 56,
    "addressFamily": "IPV4",
    "vpcId": "vpc-12345",
    "state": "PENDING",
    "createdAt": "2024-03-29T12:41:57.314000-04:00",
    "modifiedAt": "2024-03-29T12:41:57.314000-04:00",
    "tags": {
        "Name": "Probe1"
    }
}
```

Esempio 2: come creare una sonda che utilizza ICMP e aggiungerla a un monitoraggio di rete

L'esempio `create-probe` seguente crea una sonda con `protocol` impostato su `ICMP` e aggiunge la sonda a un monitor denominato `Example_NetworkMonitor`. Una volta creato, il valore `state` del monitoraggio con la sonda rimarrà `PENDING` finché il monitoraggio rimane `ACTIVE`. Questa operazione potrebbe richiedere alcuni minuti, dopodiché lo stato cambierà e potrai iniziare a `ACTIVE` visualizzare le CloudWatch metriche.

```
aws networkmonitor create-probe \
    --monitor-name Example_NetworkMonitor \
    --probe sourceArn=arn:aws:ec2:region:012345678910:subnet/subnet-
id,destination=10.0.0.100,protocol=ICMP,packetSize=56,tags={Name=Probe1}
```

Output:

```
{
    "probeId": "probe-12345",
    "probeArn": "arn:aws:networkmonitor:region:111122223333:probe/probe-12345",
    "destination": "10.0.0.100",
    "packetSize": 56,
    "addressFamily": "IPV4",
    "vpcId": "vpc-12345",
    "state": "PENDING",
    "createdAt": "2024-03-29T12:44:02.452000-04:00",
    "modifiedAt": "2024-03-29T12:44:02.452000-04:00",
    "tags": {
        "Name": "Probe1"
    }
}
```

Per ulteriori informazioni, consulta How CloudWatch Network Monitor Works nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta CreateProbe AWS CLICommand Reference.

`delete-monitor`

Il seguente esempio di codice mostra come utilizzare`delete-monitor`.

AWS CLI

Come eliminare un monitoraggio

L'esempio `delete-monitor` seguente elimina un monitor denominato `Example_NetworkMonitor`.

```
aws networkmonitor delete-monitor \
    --monitor-name Example_NetworkMonitor
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta How CloudWatch Network Monitor Works nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta DeleteMonitor AWS CLICommand Reference.

`delete-probe`

Il seguente esempio di codice mostra come utilizzare`delete-probe`.

AWS CLI

Come eliminare una sonda

L'esempio `delete-probe` seguente elimina una sonda con l'ID `probe-12345` da un monitor di rete denominato `Example_NetworkMonitor`.

```
aws networkmonitor delete-probe \
    --monitor-name Example_NetworkMonitor \
    --probe-id probe-12345
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta How CloudWatch Network Monitor Works nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta DeleteProbe AWS CLICommand Reference.

`get-monitor`

Il seguente esempio di codice mostra come utilizzare`get-monitor`.

AWS CLI

Come ottenere informazioni sul monitoraggio

L'esempio `get-monitor` seguente ottiene informazioni su un monitor denominato `Example_NetworkMonitor`.

```
aws networkmonitor get-monitor \
    --monitor-name Example_NetworkMonitor
```

Output:

```
{
    "monitorArn": "arn:aws:networkmonitor:region:012345678910:monitor/
Example_NetworkMonitor",
```

```
    "monitorName": "Example_NetworkMonitor",
    "state": "ACTIVE",
    "aggregationPeriod": 60,
    "tags": {},
    "probes": [],
    "createdAt": "2024-04-01T17:58:07.211000-04:00",
    "modifiedAt": "2024-04-01T17:58:07.211000-04:00"
}
```

Per ulteriori informazioni, consulta How CloudWatch Network Monitor Works nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta GetMonitor AWS CLICommand Reference.

`get-probe`

Il seguente esempio di codice mostra come utilizzare`get-probe`.

AWS CLI

Come visualizzare i dettagli della sonda

L'esempio `get-probe` seguente restituisce i dettagli su una sonda con `probeID` `probe-12345` associata a un monitor `Example_NetworkMonitor`.

```
aws networkmonitor get-probe \
    --monitor-name Example_NetworkMonitor \
    --probe-id probe-12345
```

Output:

```
{
    "probeId": "probe-12345",
    "probeArn": "arn:aws:networkmonitor:region:012345678910:probe/probe-12345",
    "sourceArn": "arn:aws:ec2:region:012345678910:subnet/subnet-12345",
    "destination": "10.0.0.100",
    "destinationPort": 80,
    "protocol": "TCP",
    "packetSize": 56,
    "addressFamily": "IPV4",
    "vpcId": "vpc-12345",
```

```
    "state": "ACTIVE",
    "createdAt": "2024-03-29T12:41:57.314000-04:00",
    "modifiedAt": "2024-03-29T12:42:28.610000-04:00",
    "tags": {
        "Name": "Probe1"
    }
}
```

Per ulteriori informazioni, consulta How CloudWatch Network Monitor Works nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta GetProbe AWS CLICommand Reference.

`list-monitors`

Il seguente esempio di codice mostra come utilizzare`list-monitors`.

AWS CLI

Esempio 1: come elencare tutti i monitoraggi (monitoraggio singolo)

L'esempio `list-monitors` seguente restituisce un elenco contenente un solo monitor. Il valore `state` del monitoraggio è `ACTIVE`, con `aggregationPeriod` impostato su 60 secondi.

```
aws networkmonitor list-monitors
```

Output:

```
{
    "monitors": [{
            "monitorArn": "arn:aws:networkmonitor:region:012345678910:monitor/
Example_NetworkMonitor",
            "monitorName": "Example_NetworkMonitor",
            "state": "ACTIVE",
            "aggregationPeriod": 60,
            "tags": {
                "Monitor": "Monitor1"
            }
        }
    ]
}
```

Per ulteriori informazioni, consulta How CloudWatch Network Monitor Works nella Amazon CloudWatch User Guide.

Esempio 2: come elencare tutti i monitoraggi (monitoraggi multipli)

L'esempio `list-monitors` seguente restituisce un elenco di tre monitor. Il risultato `state` di un unico monitor è `ACTIVE` la generazione di CloudWatch metriche. Gli stati degli altri due monitor sono `INACTIVE` e non CloudWatch generano metriche. Tutti e tre i monitoraggi utilizzano un valore `aggregationPeriod` di 60 secondi.

```
aws networkmonitor list-monitors
```

Output:

```
{
    "monitors": [
        {
            "monitorArn": "arn:aws:networkmonitor:us-east-1:111122223333:monitor/
Example_NetworkMonitor",
            "monitorName": "Example_NetworkMonitor",
            "state": "INACTIVE",
            "aggregationPeriod": 60,
            "tags": {}
        },
        {
            "monitorArn": "arn:aws:networkmonitor:us-east-1:111122223333:monitor/
Example_NetworkMonitor2",
            "monitorName": "Example_NetworkMonitor2",
            "state": "ACTIVE",
            "aggregationPeriod": 60,
            "tags": {
                "Monitor": "Monitor1"
            }
        },
        {
            "monitorArn": "arn:aws:networkmonitor:us-east-1:111122223333:monitor/
TestNetworkMonitor_CLI",
            "monitorName": "TestNetworkMonitor_CLI",
            "state": "INACTIVE",
            "aggregationPeriod": 60,
            "tags": {}
        }
```

```
    ]
}
```

Per ulteriori informazioni, consulta How CloudWatch Network Monitor Works nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta ListMonitors AWS CLICommand Reference.

`list-tags-for-resource`

Il seguente esempio di codice mostra come utilizzare`list-tags-for-resource`.

AWS CLI

Come elencare i tag per una risorsa

L'esempio `list-tags-for-resource` seguente restituisce un elenco di tag per un monitor denominato `Example_NetworkMonitor`.

```
aws networkmonitor list-tags-for-resource \
    --resource-arn arn:aws:networkmonitor:region:012345678910:monitor/
Example_NetworkMonitor
```

Output:

```
{
    "tags": {
        "Environment": "Dev",
        "Application": "PetStore"
    }
}
```

Per ulteriori informazioni, consulta How CloudWatch Network Monitor Works nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta ListTagsForResource AWS CLICommand Reference.

`tag-resource`

Il seguente esempio di codice mostra come utilizzare`tag-resource`.

AWS CLI

Come taggare a una risorsa

L'esempio `tag-resource` seguente tagga un monitor denominato `Example_NetworkMonitor` con i tag `Environment=Dev` e `Application=PetStore`.

```
aws networkmonitor tag-resource \
    --resource-arn arn:aws:networkmonitor:region:012345678910:monitor/
Example_NetworkMonitor \
    --tags Environment=Dev,Application=PetStore
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta How CloudWatch Network Monitor Works nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta TagResource AWS CLICommand Reference.

`untag-resource`

Il seguente esempio di codice mostra come utilizzare`untag-resource`.

AWS CLI

Come rimuovere un tag da una risorsa

L'esempio `untag-resource` seguente rimuove un parametro `tag-keys` con la coppia chiave-valore `Environment Application` dalla relativa associazione a un monitor denominato `Example_NetworkMonitor`.

```
aws networkmonitor untag-resource \
    --resource-arn arn:aws:networkmonitor:region:012345678910:monitor/
Example_NetworkMonitor \
    --tag-keys Environment Application
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta How CloudWatch Network Monitor Works nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta UntagResource AWS CLICommand Reference.

`update-monitor`

Il seguente esempio di codice mostra come utilizzare`update-monitor`.

AWS CLI

Come creare un monitoraggio

L'esempio `update-monitor` seguente modifica il valore `aggregationPeriod` da `60` secondi a `30` secondi per un monitor.

```
aws networkmonitor update-monitor \
    --monitor-name Example_NetworkMonitor \
    --aggregation-period 30
```

Output:

```
{
    "monitorArn": "arn:aws:networkmonitor:region:012345678910:monitor/
Example_NetworkMonitor",
    "monitorName": "Example_NetworkMonitor",
    "state": "PENDING",
    "aggregationPeriod": 30,
    "tags": {
        "Monitor": "Monitor1"
    }
}
```

Per ulteriori informazioni, consulta How CloudWatch Network Monitor Works nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta UpdateMonitor AWS CLICommand Reference.

`update-probe`

Il seguente esempio di codice mostra come utilizzare`update-probe`.

AWS CLI

Come aggiornare una sonda

L'esempio `update-probe` seguente aggiorna l'indirizzo IP `destination` originale di una sonda e aggiorna anche il valore `packetSize` impostandolo su `60`.

```
aws networkmonitor update-probe \
    --monitor-name Example_NetworkMonitor \
    --probe-id probe-12345 \
    --destination 10.0.0.150 \
    --packet-size 60
```

Output:

```
{
    "probeId": "probe-12345",
    "probeArn": "arn:aws:networkmonitor:region:012345678910:probe/probe-12345",
    "sourceArn": "arn:aws:ec2:region:012345678910:subnet/subnet-12345",
    "destination": "10.0.0.150",
    "destinationPort": 80,
    "protocol": "TCP",
    "packetSize": 60,
    "addressFamily": "IPV4",
    "vpcId": "vpc-12345",
    "state": "PENDING",
    "createdAt": "2024-03-29T12:41:57.314000-04:00",
    "modifiedAt": "2024-03-29T13:52:23.115000-04:00",
    "tags": {
        "Name": "Probe1"
    }
}
```

Per ulteriori informazioni, consulta How CloudWatch Network Monitor Works nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta UpdateProbe AWS CLICommand Reference.

CloudWatch Esempi di Observability Access Monitor utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with CloudWatch Observability Access Monitor.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`create-link`

Il seguente esempio di codice mostra come utilizzare`create-link`.

AWS CLI

Come creare un link

L'esempio `create-link` seguente crea un link tra un account di origine e un sink creato in un account di monitoraggio.

```
aws oam create-link \
    --label-template sourceAccount \
    --resource-types AWS::CloudWatch::Metric \
    --sink-identifier arn:aws:oam:us-east-2:123456789012:sink/a1b2c3d4-5678-90ab-cdef-example12345
```

Output:

```
{
    "Arn": "arn:aws:oam:us-east-2:123456789111:link/a1b2c3d4-5678-90ab-cdef-example11111",
    "Id": "a1b2c3d4-5678-90ab-cdef-example11111",
    "Label": "sourceAccount",
    "LabelTemplate": "sourceAccount",
    "ResourceTypes": [
        "AWS::CloudWatch::Metric"
    ],
    "SinkArn": "arn:aws:oam:us-east-2:123456789012:sink/a1b2c3d4-5678-90ab-cdef-example12345",
    "Tags": {}
}
```

Per ulteriori informazioni, consulta l'osservabilità CloudWatch tra account nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta AWS CLI Command CreateLinkReference.

`create-sink`

Il seguente esempio di codice mostra come utilizzare`create-sink`.

AWS CLI

Come creare un sink

L'`create-sink`esempio seguente crea un sink nell'account corrente, in modo che possa essere utilizzato come account di monitoraggio nell'osservabilità CloudWatch tra account.

```
aws oam create-sink \
    --name DemoSink
```

Output:

```
{
    "Arn": "arn:aws:oam:us-east-2:123456789012:sink/a1b2c3d4-5678-90ab-cdef-
example12345",
    "Id": "a1b2c3d4-5678-90ab-cdef-example12345",
    "Name": "DemoSink",
    "Tags": {}
}
```

Per ulteriori informazioni, consulta l'osservabilità CloudWatch tra account nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta AWS CLI Command CreateSinkReference.

`delete-link`

Il seguente esempio di codice mostra come utilizzare`delete-link`.

AWS CLI

Come eliminare un link

L'esempio `delete-link` seguente elimina un link tra un sink dell'account di monitoraggio e un account di origine.

```
aws oam delete-link \
    --identifier arn:aws:oam:us-east-2:123456789111:link/a1b2c3d4-5678-90ab-cdef-example11111
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta l'osservabilità CloudWatch tra account nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta AWS CLI Command DeleteLinkReference.

`delete-sink`

Il seguente esempio di codice mostra come utilizzare`delete-sink`.

AWS CLI

Come eliminare un sink

L'esempio `delete-sink` seguente elimina un sink. È necessario eliminare tutti i link a un sink prima di poterlo eliminare.

```
aws oam delete-sink \
    --identifier arn:aws:oam:us-east-2:123456789012:sink/a1b2c3d4-5678-90ab-cdef-example12345
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta l'osservabilità CloudWatch tra account nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta AWS CLI Command DeleteSinkReference.

`get-link`

Il seguente esempio di codice mostra come utilizzare`get-link`.

AWS CLI

Come restituire informazioni complete su un link

L'esempio `get-link` seguente recupera le informazioni relative a un link.

```
aws oam get-link \
    --identifier arn:aws:oam:us-east-2:123456789111:link/a1b2c3d4-5678-90ab-cdef-example11111
```

Output:

```
{
    "Arn": "arn:aws:oam:us-east-2:123456789111:link/a1b2c3d4-5678-90ab-cdef-example11111",
    "Id": "a1b2c3d4-5678-90ab-cdef-example11111",
    "Label": "sourceAccount",
    "LabelTemplate": "sourceAccount",
    "ResourceTypes": [
        "AWS::CloudWatch::Metric"
    ],
    "SinkArn": "arn:aws:oam:us-east-2:123456789012:sink/a1b2c3d4-5678-90ab-cdef-example12345",
    "Tags": {}
}
```

Per ulteriori informazioni, consulta l'osservabilità CloudWatch tra account nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta AWS CLI Command GetLinkReference.

`get-sink-policy`

Il seguente esempio di codice mostra come utilizzare`get-sink-policy`.

AWS CLI

Come restituire l'attuale policy di sink allegata al sink

L'esempio `get-sink-policy` seguente restituisce la policy di sink corrente associata al sink.

```
aws oam get-sink-policy \
```

```
    --sink-identifier arn:aws:oam:us-east-2:123456789012:sink/a1b2c3d4-5678-90ab-
cdef-example12345
```

Output:

```
{
    "SinkArn": "arn:aws:oam:us-east-2:123456789012:sink/a1b2c3d4-5678-90ab-cdef-
example12345",
    "SinkId": "a1b2c3d4-5678-90ab-cdef-example12345",
    "Policy": "{\"Version\":\"2012-10-17\",\"Statement\":[{\"Effect\":
\"Allow\",\"Principal\":{\"AWS\":\"arn:aws:iam::123456789111:root\"},
\"Action\":[\"oam:CreateLink\",\"oam:UpdateLink\"],\"Resource\":\"*\",
\"Condition\":{\"ForAllValues:StringEquals\":{\"oam:ResourceTypes\":
[\"AWS::Logs::LogGroup\",\"AWS::CloudWatch::Metric\",\"AWS::XRay::Trace\",
\"AWS::ApplicationInsights::Application\"]}}}]}"
}
```

Per ulteriori informazioni, consulta l'osservabilità CloudWatch tra account nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta AWS CLI Command GetSinkPolicyReference.

`get-sink`

Il seguente esempio di codice mostra come utilizzare`get-sink`.

AWS CLI

Come recuperare informazioni complete su un sink di monitoraggio tra account

L'esempio `get-sink` seguente restituisce informazioni complete su un sink dell'account di monitoraggio.

```
aws oam get-sink \
    --identifier arn:aws:oam:us-east-2:123456789012:sink/a1b2c3d4-5678-90ab-cdef-
example12345
```

Output:

```
{
```

```
    "Arn": "arn:aws:oam:us-east-2:123456789012:sink/a1b2c3d4-5678-90ab-cdef-
example12345",
    "Id": "a1b2c3d4-5678-90ab-cdef-example12345",
    "Name": "DemoSink",
    "Tags": {}
}
```

Per ulteriori informazioni, consulta l'osservabilità CloudWatch tra account nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta AWS CLI Command GetSinkReference.

`list-attached-links`

Il seguente esempio di codice mostra come utilizzare`list-attached-links`.

AWS CLI

Come restituire un elenco di link agli account di origine collegati a questo sink di account di monitoraggio

L'esempio `list-attached-links` seguente restituisce un elenco di link dell'account di origine collegati al sink corrente dell'account di monitoraggio.

```
aws oam list-attached-links \
    --sink-identifier arn:aws:oam:us-east-2:123456789012:sink/a1b2c3d4-5678-90ab-
cdef-example12345
```

Output:

```
{
    "Items": [{
        "Label": "Monitoring account",
        "LinkArn": "arn:aws:oam:us-east-2:123456789111:link/a1b2c3d4-5678-90ab-cdef-
example11111",
        "ResourceTypes": [
            "AWS::ApplicationInsights::Application",
            "AWS::Logs::LogGroup",
            "AWS::CloudWatch::Metric",
            "AWS::XRay::Trace"
        ]
    }]
```

```
}
```

Per ulteriori informazioni, consulta l'osservabilità CloudWatch tra account nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta AWS CLI Command ListAttachedLinksReference.

`list-links`

Il seguente esempio di codice mostra come utilizzare`list-links`.

AWS CLI

Come restituire un elenco di link per un sink dell'account di monitoraggio

L'esempio `list-links` seguente restituisce un elenco di link per un sink dell'account di monitoraggio. Esegui questa operazione in un account di origine per restituire un elenco di link ai sink degli account di monitoraggio di questo account di origine.

```
aws oam list-links
```

Output:

```
{
    "Items": [{
        "Arn": "arn:aws:oam:us-east-2:123456789111:link/a1b2c3d4-5678-90ab-cdef-
example11111",
        "Id": "a1b2c3d4-5678-90ab-cdef-example11111",
        "Label": "sourceAccount",
        "ResourceTypes": [
            "AWS::CloudWatch::Metric"
        ],
        "SinkArn": "arn:aws:oam:us-east-2:123456789012:sink/a1b2c3d4-5678-90ab-cdef-
example12345"
    }]
}
```

Per ulteriori informazioni, consulta l'osservabilità CloudWatch tra account nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta AWS CLI Command ListLinksReference.

`list-sinks`

Il seguente esempio di codice mostra come utilizzare`list-sinks`.

AWS CLI

Come restituire l'elenco dei sink creati nell'account di monitoraggio

L'esempio `list-sinks` seguente restituisce un elenco di sink creati nell'account di monitoraggio. Esegui questa operazione in un account di monitoraggio.

```
aws oam list-sinks
```

Output:

```
{
    "Items": [
        {
            "Arn": "arn:aws:oam:us-east-2:123456789012:sink/a1b2c3d4-5678-90ab-cdef-
example12345",
            "Id": "a1b2c3d4-5678-90ab-cdef-example12345",
            "Name": "DemoSink"
        }
    ]
}
```

Per ulteriori informazioni, consulta l'osservabilità CloudWatch tra account nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta AWS CLI Command ListSinksReference.

`list-tags-for-resource`

Il seguente esempio di codice mostra come utilizzare`list-tags-for-resource`.

AWS CLI

Come visualizzare i tag associati a una risorsa

L'esempio `list-tags-for-resource` seguente visualizza i tag associati a un sink.

```
aws oam list-tags-for-resource \
```

```
    --resource-arn arn:aws:oam:us-east-2:123456789012:sink/a1b2c3d4-5678-90ab-cdef-
example12345
```

Output:

```
{
    "Tags": {
        "Team": "Devops"
    }
}
```

Per ulteriori informazioni, consulta l'osservabilità CloudWatch tra account nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta AWS CLI Command ListTagsForResourceReference.

`put-sink-policy`

Il seguente esempio di codice mostra come utilizzare`put-sink-policy`.

AWS CLI

Come creare o aggiornare una policy delle risorse

L'esempio `put-sink-policy` seguente crea la policy di risorse che concede le autorizzazioni agli account di origine per il link al sink dell'account di monitoraggio.

```
aws oam put-sink-policy \
    --policy '{"Version":"2012-10-17",       "Statement":
[{"Effect":"Allow","Principal":{"AWS":"arn:aws:iam::123456789111:root"},"Action":
["oam:CreateLink","oam:UpdateLink"],"Resource":"*","Condition":
{"ForAllValues:StringEquals":{"oam:ResourceTypes":
["AWS::Logs::LogGroup","AWS::CloudWatch::Metric","AWS::XRay::Trace","AWS::ApplicationInsight
 \
    --sink-identifier arn:aws:oam:us-east-2:123456789012:sink/a1b2c3d4-5678-90ab-
cdef-example12345
```

Output:

```
{
```

```
    "SinkArn": "arn:aws:oam:us-east-2:123456789012:sink/a1b2c3d4-5678-90ab-cdef-
example12345",
    "SinkId": "a1b2c3d4-5678-90ab-cdef-example12345",
    "Policy": "{\"Version\":\"2012-10-17\",\"Statement\":[{\"Effect\":
\"Allow\",\"Principal\":{\"AWS\":\"arn:aws:iam::123456789111:root\"},
\"Action\":[\"oam:CreateLink\",\"oam:UpdateLink\"],\"Resource\":\"*\",
\"Condition\":{\"ForAllValues:StringEquals\":{\"oam:ResourceTypes\":
[\"AWS::Logs::LogGroup\",\"AWS::CloudWatch::Metric\",\"AWS::XRay::Trace\",
\"AWS::ApplicationInsights::Application\"]}}}]}"
}
```

Per ulteriori informazioni, consulta l'osservabilità CloudWatch tra account nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta AWS CLI Command PutSinkPolicyReference.

`tag-resource`

Il seguente esempio di codice mostra come utilizzare`tag-resource`.

AWS CLI

Come assegnare uno o più tag a una risorsa specificata

L'esempio `tag-resource` seguente tagga un sink denominato `arn:aws:oam:us-east-2:123456789012:sink/a1b2c3d4-5678-90ab-cdef-example12345`.

```
aws oam tag-resource \
    --resource-arn arn:aws:oam:us-east-2:123456789012:sink/a1b2c3d4-5678-90ab-cdef-
example12345 \
    --tags team=Devops
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta l'osservabilità CloudWatch tra account nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta AWS CLI Command TagResourceReference.

`untag-resource`

Il seguente esempio di codice mostra come utilizzare`untag-resource`.

AWS CLI

Come rimuovere uno o più tag dalla risorsa specificata.

L'esempio `untag-resource` seguente rimuove un tag con la chiave `team` dal sink `arn:aws:oam:us-east-2:123456789012:sink/a1b2c3d4-5678-90ab-cdef-example12345`.

```
aws oam untag-resource \
    --resource-arn arn:aws:oam:us-east-2:123456789012:sink/f3f42f60-f0f2-425c-1234-12347bdd821f \
    --tag-keys team
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta l'osservabilità CloudWatch tra account nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta AWS CLI Command UntagResourceReference.

`update-link`

Il seguente esempio di codice mostra come utilizzare`update-link`.

AWS CLI

Come modificare i tipi di dati condivisi da un account di origine al relativo sink dell'account di monitoraggio collegato

L'esempio `update-link` seguente aggiorna il link `arn:aws:oam:us-east-2:123456789111:link/0123e691-e7ef-43fa-1234-c57c837fced0` con i tipi di risorsa `AWS::CloudWatch::Metric` e `AWS::Logs::LogGroup`.

```
aws oam update-link \
    --identifier arn:aws:oam:us-east-2:123456789111:link/a1b2c3d4-5678-90ab-cdef-example11111 \
    --resource-types "AWS::CloudWatch::Metric" "AWS::Logs::LogGroup"
```

Output:

```
{
```

```
    "Arn": "arn:aws:oam:us-east-2:123456789111:link/a1b2c3d4-5678-90ab-cdef-
example11111",
    "Id": "a1b2c3d4-5678-90ab-cdef-example11111",
    "Label": "sourceAccount",
    "LabelTemplate": "sourceAccount",
    "ResourceTypes": [
        "AWS::CloudWatch::Metric",
        "AWS::Logs::LogGroup"
    ],
    "SinkArn": "arn:aws:oam:us-east-2:123456789012:sink/a1b2c3d4-5678-90ab-cdef-
example12345",
    "Tags": {}
}
```

Per ulteriori informazioni, consulta l'osservabilità CloudWatch tra account nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta AWS CLI Command UpdateLinkReference.

CloudWatch Esempi di Observability Admin utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with CloudWatch Observability Admin.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`get-telemetry-evaluation-status-for-organization`

Il seguente esempio di codice mostra come utilizzare`get-telemetry-evaluation-status-for-organization`.

AWS CLI

Come ottenere lo stato di onboarding della telemetria per l'organizzazione

L'esempio `get-telemetry-evaluation-status-for-organization` seguente restituisce lo stato corrente di onboarding della funzionalità di configurazione della telemetria per l'organizzazione.

```
aws observabilityadmin get-telemetry-evaluation-status-for-organization
```

Output:

```
{
    "Status": "RUNNING"
}
```

Per ulteriori informazioni, consulta la sezione Controllo delle configurazioni di CloudWatch telemetria nella Amazon User Guide. CloudWatch

- Per i dettagli sull'API, consulta Command Reference. GetTelemetryEvaluationStatusForOrganizationAWS CLI

`get-telemetry-evaluation-status`

Il seguente esempio di codice mostra come utilizzare`get-telemetry-evaluation-status`.

AWS CLI

Come visualizzare lo stato di attivazione della telemetria per l'account

L'esempio `get-telemetry-evaluation-status` seguente restituisce lo stato corrente di onboarding della funzionalità di configurazione della telemetria nell'account specificato.

```
aws observabilityadmin get-telemetry-evaluation-status
```

Output:

```
{
    "Status": "RUNNING"
```

```
}
```

Per ulteriori informazioni, consulta la sezione Controllo delle configurazioni di CloudWatch telemetria nella Amazon User Guide. CloudWatch

- Per i dettagli sull'API, consulta Command Reference. GetTelemetryEvaluationStatusAWS CLI

`list-resource-telemetry-for-organization`

Il seguente esempio di codice mostra come utilizzare`list-resource-telemetry-for-organization`.

AWS CLI

Come recuperare le configurazioni di telemetria per l'organizzazione

L'`list-resource-telemetry-for-organization`esempio seguente restituisce un elenco di configurazioni di telemetria nell'organizzazione per le AWS risorse supportate dalla configurazione di telemetria.

```
aws observabilityadmin list-resource-telemetry-for-organization \
    --resource-types  AWS::EC2::Instance
```

Output:

```
{
    "TelemetryConfigurations": [
        {
            "AccountIdentifier": "111111111111",
            "TelemetryConfigurationState": {
                "Logs": "NotApplicable",
                "Metrics": "Disabled",
                "Traces": "NotApplicable"
            },
            "ResourceType": "AWS::EC2::Instance",
            "ResourceIdentifier": "i-a166400b",
            "ResourceTags": {
                "Name": "dev"
            },
            "LastUpdateTimeStamp": 1733168548521
        },
        {
```

```
            "AccountIdentifier": "222222222222",
            "TelemetryConfigurationState": {
                "Logs": "NotApplicable",
                "Metrics": "Disabled",
                "Traces": "NotApplicable"
            },
            "ResourceType": "AWS::EC2::Instance",
            "ResourceIdentifier": "i-b188560f",
            "ResourceTags": {
                "Name": "apache"
            },
            "LastUpdateTimeStamp": 1732744260182
        }
    ]
}
```

Per ulteriori informazioni, consulta la sezione Controllo delle configurazioni di CloudWatch telemetria nella Amazon User Guide. CloudWatch

- Per i dettagli sull'API, consulta Command Reference. ListResourceTelemetryForOrganizationAWS CLI

`list-resource-telemetry`

Il seguente esempio di codice mostra come utilizzare`list-resource-telemetry`.

AWS CLI

Come recuperare le configurazioni di telemetria per l'account

L'`list-resource-telemetry`esempio seguente restituisce un elenco di configurazioni di telemetria per le AWS risorse supportate dalla configurazione di telemetria nell'account specificato.

```
aws observabilityadmin list-resource-telemetry \
    --resource-types  AWS::EC2::Instance
```

Output:

```
{
    "TelemetryConfigurations": [
        {
```

```
            "AccountIdentifier": "111111111111",
            "TelemetryConfigurationState": {
                "Logs": "NotApplicable",
                "Metrics": "Disabled",
                "Traces": "NotApplicable"
            },
            "ResourceType": "AWS::EC2::Instance",
            "ResourceIdentifier": "i-0e979d278b040f856",
            "ResourceTags": {
                "Name": "apache"
            },
            "LastUpdateTimeStamp": 1732744260182
        }
    ]
}
```

Per ulteriori informazioni, consulta la sezione Controllo delle configurazioni di CloudWatch telemetria nella Amazon User Guide. CloudWatch

- Per i dettagli sull'API, consulta Command Reference. ListResourceTelemetryAWS CLI

`start-telemetry-evaluation-for-organization`

Il seguente esempio di codice mostra come utilizzare`start-telemetry-evaluation-for-organization`.

AWS CLI

Come abilitare la funzionalità di configurazione della telemetria

L'esempio `start-telemetry-evaluation-for-organization` seguente abilita la funzionalità di configurazione della telemetria per l'organizzazione.

```
aws observabilityadmin start-telemetry-evaluation-for-organization
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Attivazione del controllo CloudWatch telemetrico nella Amazon User Guide. CloudWatch

- Per i dettagli sull'API, consulta StartTelemetryEvaluationForOrganizationCommand Reference.AWS CLI

`start-telemetry-evaluation`

Il seguente esempio di codice mostra come utilizzare`start-telemetry-evaluation`.

AWS CLI

Come abilitare la funzionalità di configurazione della telemetria

L'esempio `start-telemetry-evaluation` seguente abilita la funzionalità di configurazione della telemetria nell'account specificato.

```
aws observabilityadmin start-telemetry-evaluation
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Attivazione del controllo CloudWatch telemetrico nella Amazon User Guide. CloudWatch

- Per i dettagli sull'API, consulta StartTelemetryEvaluationCommand Reference.AWS CLI

`stop-telemetry-evaluation-for-organization`

Il seguente esempio di codice mostra come utilizzare`stop-telemetry-evaluation-for-organization`.

AWS CLI

Come disabilitare la funzionalità di configurazione della telemetria

L'esempio `stop-telemetry-evaluation-for-organization` seguente disabilita la funzionalità di configurazione della telemetria per l'organizzazione.

```
aws observabilityadmin stop-telemetry-evaluation-for-organization
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta la sezione Disattivazione del controllo CloudWatch telemetrico nella Amazon User Guide. CloudWatch

- Per i dettagli sull'API, consulta StopTelemetryEvaluationForOrganizationCommand Reference.AWS CLI

`stop-telemetry-evaluation`

Il seguente esempio di codice mostra come utilizzare`stop-telemetry-evaluation`.

AWS CLI

Come disabilitare la funzionalità di configurazione della telemetria

L'esempio `stop-telemetry-evaluation` seguente disabilita la funzionalità di configurazione della telemetria nell'account specificato.

```
aws observabilityadmin stop-telemetry-evaluation
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta la sezione Disattivazione del controllo CloudWatch telemetrico nella Amazon User Guide. CloudWatch

- Per i dettagli sull'API, consulta StopTelemetryEvaluationCommand Reference.AWS CLI

CloudWatch Esempi di Synthetics utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with CloudWatch Synthetics.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`associate-resource`

Il seguente esempio di codice mostra come utilizzare. `associate-resource`

AWS CLI

Come associare un canary a un gruppo

L'esempio `associate-resource` seguente associa un canary a un gruppo denominato `demo_group`.

```
aws synthetics associate-resource \
    --group-identifier demo_group \
    --resource-arn arn:aws:synthetics:us-east-1:123456789012:canary:demo_canary
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Synthetic monitoring (canaries) nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta AssociateResource AWS CLICommand Reference.

`create-canary`

Il seguente esempio di codice mostra come utilizzare`create-canary`.

AWS CLI

Come creare un canary

L'esempio `create-canary` seguente crea un canary denominato `demo_canary`.

```
aws synthetics create-canary \
    --name demo_canary \
    --code '{"S3Bucket": "artifacts3bucket", "S3Key":"demo_canary.zip", "Handler":
 "index.lambda_handler"}' \
    --artifact-s3-location s3://amzn-s3-demo-bucket/demo_canary.zip \
    --execution-role-arn arn:aws:iam::123456789012:role/demo_canary_role \
    --schedule Expression="rate(10 minutes)" \
    --runtime-version syn-nodejs-puppeteer-9.1
```

Output:

```
{
    "Canary": {
```

```
        "Id": "a1b2c3d4-5678-90ab-cdef-example11111",
        "Name": "demo_canary",
        "Code": {
            "Handler": "index.lambda_handler"
        },
        "ExecutionRoleArn": "arn:aws:iam::123456789012:role/demo_canary_role",
        "Schedule": {
            "Expression": "rate(10 minutes)",
            "DurationInSeconds": 0
        },
        "RunConfig": {
            "TimeoutInSeconds": 600,
            "MemoryInMB": 1000,
            "ActiveTracing": false
        },
        "SuccessRetentionPeriodInDays": 31,
        "FailureRetentionPeriodInDays": 31,
        "Status": {
            "State": "CREATING",
            "StateReasonCode": "CREATE_PENDING"
        },
        "Timeline": {
            "Created": "2024-10-15T19:03:08.826000+05:30",
            "LastModified": "2024-10-15T19:03:08.826000+05:30"
        },
        "ArtifactS3Location": "amzn-s3-demo-bucket/demo_canary.zip",
        "RuntimeVersion": "syn-nodejs-puppeteer-9.1",
        "Tags": {}
    }
}
```

Per ulteriori informazioni, consulta Synthetic monitoring (canaries) nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta CreateCanary AWS CLICommand Reference.

`create-group`

Il seguente esempio di codice mostra come utilizzare`create-group`.

AWS CLI

Come creare un gruppo

L'esempio `create-group` seguente elimina un gruppo denominato `demo_group`.

```
aws synthetics create-group \
    --name demo_group
```

Output:

```
{
    "Group": {
        "Id": "example123",
        "Name": "demo_group",
        "Arn": "arn:aws:synthetics:us-east-1:123456789012:group:example123",
        "Tags": {},
        "CreatedTime": "2024-10-15T14:47:23.811000+05:30",
        "LastModifiedTime": "2024-10-15T14:47:23.811000+05:30"
    }
}
```

Per ulteriori informazioni, consulta Synthetic monitoring (canaries) nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta CreateGroup AWS CLICommand Reference.

`delete-canary`

Il seguente esempio di codice mostra come utilizzare`delete-canary`.

AWS CLI

Come eliminare definitivamente un canary

L'esempio `delete-canary` seguente elimina un canary denominato `demo_canary`.

```
aws synthetics delete-canary \
    --name demo_canary
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Synthetic monitoring (canaries) nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta DeleteCanary AWS CLICommand Reference.

`delete-group`

Il seguente esempio di codice mostra come utilizzare`delete-group`.

AWS CLI

Come eliminare un gruppo

L'esempio `delete-group` seguente elimina un gruppo denominato `demo_group`.

```
aws synthetics delete-group \
    --group-identifier demo_group
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Synthetic monitoring (canaries) nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta DeleteGroup AWS CLICommand Reference.

`describe-canaries-last-run`

Il seguente esempio di codice mostra come utilizzare`describe-canaries-last-run`.

AWS CLI

Come visualizzare le informazioni relative all'esecuzione più recente di ogni canary

L'esempio `describe-canaries-last-run` seguente restituisce l'esecuzione più recente di ogni canary creato.

```
aws synthetics describe-canaries-last-run
```

Output:

```
{
    "CanariesLastRun": [
        {
            "CanaryName": "demo_canary",
            "LastRun": {
                "Id": "a1b2c3d4-5678-90ab-cdef-example11111",
                "Name": "demo_canary",
                "Status": {
```

```
                    "State": "PASSED",
                    "StateReason": "",
                    "StateReasonCode": ""
                },
                "Timeline": {
                    "Started": "2024-10-15T19:20:39.691000+05:30",
                    "Completed": "2024-10-15T19:20:58.211000+05:30"
                },
                "ArtifactS3Location": "cw-syn-results-123456789012-us-east-1/canary/
us-east-1/demo_canary-abc-example1234/2024/10/15/13/50-39-690"
            }
        }
    ]
}
```

Per ulteriori informazioni, consulta Synthetic monitoring (canaries) nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta DescribeCanariesLastRun AWS CLICommand Reference.

`describe-canaries`

Il seguente esempio di codice mostra come utilizzare`describe-canaries`.

AWS CLI

Come elencare i canary nel tuo account

L'esempio `describe-canaries` seguente elenca i dettagli dei canary nell'account in uso.

```
aws synthetics describe-canaries
```

Output:

```
{
    "Canaries": [
        {
            "Id": "a1b2c3d4-5678-90ab-cdef-example11111",
            "Name": "demo_canary",
            "Code": {
                "SourceLocationArn": "arn:aws:lambda:us-
east-1:123456789012:layer:cwsyn-demo_canary-a1b2c3d4-5678-90ab-cdef-
example11111b8:1",
```

```
                "Handler": "pageLoadBlueprint.handler"
            },
            "ExecutionRoleArn": "arn:aws:iam::123456789012:role/service-role/
CloudWatchSyntheticsRole-demo_canary-a12-a123bc456789",
            "Schedule": {
                "Expression": "rate(5 minutes)",
                "DurationInSeconds": 0
            },
            "RunConfig": {
                "TimeoutInSeconds": 300,
                "MemoryInMB": 1000,
                "ActiveTracing": false
            },
            "SuccessRetentionPeriodInDays": 31,
            "FailureRetentionPeriodInDays": 31,
            "Status": {
            "State": "RUNNING"
            },
            "Timeline": {
                "Created": "2024-10-15T18:55:15.168000+05:30",
                "LastModified": "2024-10-15T18:55:40.540000+05:30",
                "LastStarted": "2024-10-15T18:55:40.540000+05:30"
            },
            "ArtifactS3Location": "cw-syn-results-123456789012-us-east-1/canary/us-
east-1/demo_canary-a12-a123bc456789",
            "EngineArn": "arn:aws:lambda:us-east-1:123456789012:function:cwsyn-
demo_canary-a1b2c3d4-5678-90ab-cdef-example111118:1",
            "RuntimeVersion": "syn-nodejs-puppeteer-9.1",
            "Tags": {
                "blueprint": "heartbeat"
            }
        }
    ]
}
```

Per ulteriori informazioni, consulta Synthetic monitoring (canaries) nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta DescribeCanaries AWS CLICommand Reference.

`describe-runtime-versions`

Il seguente esempio di codice mostra come utilizzare`describe-runtime-versions`.

AWS CLI

Come restituire un elenco di versioni di runtime dei canary di Synthetics

L'esempio `describe-runtime-versions` seguente restituisce l'elenco delle versioni di runtime dei canary di Synthetics.

```
aws synthetics describe-runtime-versions
```

Output:

```
{
    "RuntimeVersions": [
        {
            "VersionName": "syn-nodejs-puppeteer-9.1",
            "Description": "Security fixes and bug fix for date range error in har.
 Dependencies: Node JS 20.x, Puppeteer-core 22.12.1, Chromium 126.0.6478.126",
            "ReleaseDate": "2024-10-02T05:30:00+05:30"
        },
        {
            "VersionName": "syn-nodejs-puppeteer-9.0",
            "Description": "Upgraded Chromium and Puppeteer. Dependencies: Node JS
 20.x, Puppeteer-core 22.12.1, Chromium 126.0.6478.126",
            "ReleaseDate": "2024-07-22T05:30:00+05:30"
        },
        {
            "VersionName": "syn-nodejs-puppeteer-8.0",
            "Description": "Upgraded Chromium and Puppeteer. Dependencies: Node JS
 20.x, Puppeteer-core 22.10.0, Chromium 125.0.6422.112",
            "ReleaseDate": "2024-06-21T05:30:00+05:30"
        },
        {
            "VersionName": "syn-nodejs-puppeteer-7.0",
            "Description": "Upgraded Chromium and Puppeteer. Dependencies: Node JS
 18.x, Puppeteer-core 21.9.0, Chromium 121.0.6167.139",
            "ReleaseDate": "2024-03-08T05:30:00+05:30"
            },
        {
            "VersionName": "syn-nodejs-puppeteer-6.2",
            "Description": "Updated shared libraries for Chromium and added
 ephemeral storage monitoring. Dependencies: Node JS 18.x, Puppeteer-core 19.7.0,
 Chromium 111.0.5563.146",
            "ReleaseDate": "2024-02-02T05:30:00+05:30"
```

```
        },
        {
            "VersionName": "syn-nodejs-puppeteer-6.1",
            "Description": "Added puppeteer launch retry. Dependencies: Node JS
18.x, Puppeteer-core 19.7.0, Chromium 111.0.5563.146",
            "ReleaseDate": "2023-11-13T05:30:00+05:30",
            "DeprecationDate": "2024-03-08T13:30:00+05:30"
        },
        {
            "VersionName": "syn-nodejs-puppeteer-6.0",
            "Description": "Reduced X-Ray traces of a canary run, improved duration
metric and upgraded to NodeJS 18.x. Dependencies: Node JS 18.x, Puppeteer-core
19.7.0, Chromium 111.0.5563.146",
            "ReleaseDate": "2023-09-15T05:30:00+05:30",
            "DeprecationDate": "2024-03-08T13:30:00+05:30"
        },
        {
            "VersionName": "syn-nodejs-puppeteer-5.2",
            "Description": "Updated shared libraries for Chromium. Dependencies:
Node JS 16.x, Puppeteer-core 19.7.0, Chromium 111.0.5563.146",
            "ReleaseDate": "2024-02-01T05:30:00+05:30"
        },
        {
            "VersionName": "syn-nodejs-puppeteer-5.1",
            "Description": "Fixes a bug about missing request headers in har.
Dependencies: Node JS 16.x, Puppeteer-core 19.7.0, Chromium 111.0.5563.146",
            "ReleaseDate": "2023-08-09T05:30:00+05:30",
            "DeprecationDate": "2024-03-08T13:30:00+05:30"
        },
        {
            "VersionName": "syn-nodejs-puppeteer-5.0",
            "Description": "Upgraded Puppeteer and Chromium. Dependencies: Node JS
16.x, Puppeteer-core 19.7.0, Chromium 111.0.5563.146",
            "ReleaseDate": "2023-07-21T05:30:00+05:30",
            "DeprecationDate": "2024-03-08T13:30:00+05:30"
        },
        {
            "VersionName": "syn-nodejs-puppeteer-4.0",
            "Description": "Upgraded to NodeJS 16.x. Dependencies: Node JS 16.x,
Puppeteer-core 5.5.0, Chromium 92.0.4512.0",
            "ReleaseDate": "2023-05-01T05:30:00+05:30",
            "DeprecationDate": "2024-03-08T13:30:00+05:30"
        }
    ]
```

```
}
```

Per ulteriori informazioni, consulta Synthetic monitoring (canaries) nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta DescribeRuntimeVersions AWS CLICommand Reference.

`disassociate-resource`

Il seguente esempio di codice mostra come utilizzare`disassociate-resource`.

AWS CLI

Come rimuovere un canary da un gruppo

L'esempio `disassociate-resource` seguente rimuove un canary da un gruppo denominato `demo_group`.

```
aws synthetics disassociate-resource \
    --group-identifier demo_group \
    --resource-arn arn:aws:synthetics:us-east-1:123456789012:canary:demo_canary
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Synthetic monitoring (canaries) nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta DisassociateResource AWS CLICommand Reference.

`get-canary-runs`

Il seguente esempio di codice mostra come utilizzare`get-canary-runs`.

AWS CLI

Come recuperare un elenco di esecuzioni per un canary specificato

L'esempio `get-canary-runs` seguente recupera un elenco di esecuzioni per il canary denominato `demo_canary`.

```
aws synthetics get-canary-runs \
```

```
    --name demo_canary
```

Output:

```
{
    "CanaryRuns": [
        {
            "Id": "a1b2c3d4-5678-90ab-cdef-example11111",
            "Name": "demo_canary",
            "Status": {
                "State": "PASSED",
                "StateReason": "",
                "StateReasonCode": ""
            },
            "Timeline": {
                "Started": "2024-10-16T10:38:57.013000+05:30",
                "Completed": "2024-10-16T10:39:25.793000+05:30"
            },
            "ArtifactS3Location": "cw-syn-results-123456789012-us-east-1/canary/us-
east-1/demo_canary-abc-example1234/2024/10/15/13/50-39-690"
        }
    ]
}
```

Per ulteriori informazioni, consulta Synthetic monitoring (canaries) nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta GetCanaryRuns AWS CLICommand Reference.

`get-canary`

Il seguente esempio di codice mostra come utilizzare`get-canary`.

AWS CLI

Come restituire informazioni complete su un canary

L'esempio `get-canary` seguente recupera le informazioni complete sul canary denominato `demo_canary`.

```
aws synthetics get-canary \
```

```
    --name demo_canary
```

Output:

```
{
    "Canary": {
        "Id": "a1b2c3d4-5678-90ab-cdef-example11111",
        "Name": "demo_canary",
        "Code": {
            "SourceLocationArn": "arn:aws:lambda:us-east-1:123456789012:layer:cwsyn-
demo_canary-a1b2c3d4-5678-90ab-cdef-example111118:1",
            "Handler": "pageLoadBlueprint.handler"
        },
        "ExecutionRoleArn": "arn:aws:iam::123456789012:role/demo_canary_role",
        "Schedule": {
            "Expression": "rate(10 minutes)",
            "DurationInSeconds": 0
        },
        "RunConfig": {
            "TimeoutInSeconds": 300,
            "MemoryInMB": 1000,
            "ActiveTracing": false
        },
        "SuccessRetentionPeriodInDays": 31,
        "FailureRetentionPeriodInDays": 31,
        "Status": {
            "State": "RUNNING"
        },
        "Timeline": {
            "Created": "2024-10-15T18:55:15.168000+05:30",
            "LastModified": "2024-10-15T18:55:40.540000+05:30",
            "LastStarted": "2024-10-15T18:55:40.540000+05:30"
        },
        "ArtifactS3Location": "cw-syn-results-123456789012-us-east-1/canary/us-
east-1/demo_canary-a12-a123bc456789",
        "EngineArn": "arn:aws:lambda:us-east-1:123456789012:function:cwsyn-
demo_canary-a1b2c3d4-5678-90ab-cdef-example111118:1",
        "RuntimeVersion": "syn-nodejs-puppeteer-9.1",
        "Tags": {
            "blueprint": "heartbeat"
        }
    }
}
```

Per ulteriori informazioni, consulta Synthetic monitoring (canaries) nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta GetCanary AWS CLICommand Reference.

`get-group`

Il seguente esempio di codice mostra come utilizzare`get-group`.

AWS CLI

Come restituire informazioni su un gruppo

L'esempio `get-group` seguente restituisce informazioni sul gruppo denominato `demo_group`.

```
aws synthetics get-group \
    --group-identifier demo_group
```

Output:

```
{
    "Group": {
        "Id": "example123",
        "Name": "demo_group",
        "Arn": "arn:aws:synthetics:us-east-1:123456789012:group:example123",
        "Tags": {},
        "CreatedTime": "2024-10-15T14:47:23.811000+05:30",
        "LastModifiedTime": "2024-10-15T14:47:23.811000+05:30"
    }
}
```

Per ulteriori informazioni, consulta Synthetic monitoring (canaries) nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta GetGroup AWS CLICommand Reference.

`list-associated-groups`

Il seguente esempio di codice mostra come utilizzare`list-associated-groups`.

AWS CLI

Come recuperare un elenco di gruppi

L'esempio `list-associated-groups` seguente restituisce un elenco dei gruppi associati al canary denominato `demo_canary`.

```
aws synthetics list-associated-groups \
    --resource-arn arn:aws:synthetics:us-east-1:123456789012:canary:demo_canary
```

Output:

```
{
    "Groups": [
        {
            "Id": "example123",
            "Name": "demo_group",
            "Arn": "arn:aws:synthetics:us-east-1:123456789012:group:example123"
        }
    ]
}
```

Per ulteriori informazioni, consulta Synthetic monitoring (canaries) nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta ListAssociatedGroups AWS CLICommand Reference.

`list-group-resources`

Il seguente esempio di codice mostra come utilizzare`list-group-resources`.

AWS CLI

Per restituire un elenco ARNs dei canarini associati al gruppo specificato

L'`list-group-resources`esempio seguente restituisce un elenco ARNs dei canarini associati al gruppo denominato. `demo_group`

```
aws synthetics list-group-resources \
    --group-identifier demo_group
```

Output:

```
{
    "Resources": [
```

```
        "arn:aws:synthetics:us-east-1:123456789012:canary:demo_canary"
    ]
}
```

Per ulteriori informazioni, consulta Synthetic monitoring (canaries) nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta ListGroupResources AWS CLICommand Reference.

`list-groups`

Il seguente esempio di codice mostra come utilizzare`list-groups`.

AWS CLI

Come restituire un elenco di tutti i gruppi dell'account

L'esempio `list-groups` seguente recupera un elenco di tutti i gruppi nell'account.

```
aws synthetics list-groups
```

Output:

```
{
    "Groups": [
        {
            "Id": "example123",
            "Name": "demo_group",
            "Arn": "arn:aws:synthetics:us-east-1:123456789012:group:example123"
        }
    ]
}
```

Per ulteriori informazioni, consulta Synthetic monitoring (canaries) nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta ListGroups AWS CLICommand Reference.

`list-tags-for-resource`

Il seguente esempio di codice mostra come utilizzare`list-tags-for-resource`.

AWS CLI

Esempio 1: come visualizzare i tag associati a un canary

L'esempio `list-tags-for-resource` seguente restituisce i tag associati a un canary denominato `demo_canary`.

```
aws synthetics list-tags-for-resource \
    --resource-arn arn:aws:synthetics:us-east-1:123456789012:canary:demo_canary
```

Output:

```
{
    "Tags": {
        "blueprint": "heartbeat"
    }
}
```

Esempio 2: come visualizzare i tag associati a un gruppo

L'esempio `list-tags-for-resource` seguente restituisce i tag associati a un gruppo denominato `demo_group`.

```
aws  synthetics list-tags-for-resource \
    --resource-arn arn:aws:synthetics:us-east-1:123456789012:group:example123
```

Output:

```
{
    "Tags": {
        "team": "Devops"
    }
}
```

Per ulteriori informazioni, consulta Synthetic monitoring (canaries) nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta ListTagsForResource AWS CLICommand Reference.

`start-canary`

Il seguente esempio di codice mostra come utilizzare`start-canary`.

AWS CLI

Come eseguire un canary

L'esempio `start-canary` seguente esegue un canary denominato `demo_canary`.

```
aws synthetics start-canary \
    --name demo_canary
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Synthetic monitoring (canaries) nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta StartCanary AWS CLICommand Reference.

`stop-canary`

Il seguente esempio di codice mostra come utilizzare`stop-canary`.

AWS CLI

Come arrestare un canary

L'esempio `stop-canary` seguente arresta il canary denominato `demo_canary`.

```
aws synthetics stop-canary \
    --name demo_canary
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Synthetic monitoring (canaries) nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta StopCanary AWS CLICommand Reference.

`tag-resource`

Il seguente esempio di codice mostra come utilizzare`tag-resource`.

AWS CLI

Esempio 1: come assegnare un tag al canary

L'esempio `tag-resource` seguente assegna un tag al canary denominato `demo_canary`.

```
aws synthetics tag-resource \
    --resource-arn arn:aws:synthetics:us-east-1:123456789012:canary:demo_canary \
    --tags blueprint=heartbeat
```

Questo comando non produce alcun output.

Esempio 2: come assegnare un tag al gruppo

L'esempio `tag-resource` seguente assegna un tag al gruppo denominato `demo_group`.

```
aws synthetics tag-resource \
    --resource-arn arn:aws:synthetics:us-east-1:123456789012:group:example123 \
    --tags team=Devops
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Synthetic monitoring (canaries) nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta TagResource AWS CLICommand Reference.

`untag-resource`

Il seguente esempio di codice mostra come utilizzare`untag-resource`.

AWS CLI

Esempio 1: come rimuovere un tag dal canary

L'esempio `untag-resource` seguente rimuove un tag dal canary denominato `demo_canary`.

```
aws synthetics untag-resource \
    --resource-arn arn:aws:synthetics:us-east-1:123456789012:canary:demo_canary \
    --tag-keys blueprint
```

Questo comando non produce alcun output.

Esempio 2: come rimuovere un tag dal gruppo

L'esempio `untag-resource` seguente rimuove un tag dal gruppo denominato `demo_group`.

```
aws synthetics untag-resource \
    --resource-arn arn:aws:synthetics:us-east-1:123456789012:group:example123 \
    --tag-keys team
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Synthetic monitoring (canaries) nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta UntagResource AWS CLICommand Reference.

`update-canary`

Il seguente esempio di codice mostra come utilizzare`update-canary`.

AWS CLI

Come aggiornare un canary

L'esempio `update-canary` seguente aggiorna la configurazione di un canary denominato `demo_canary`.

```
aws synthetics update-canary \
    --name demo_canary \
    --schedule Expression="rate(15 minutes)"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Synthetic monitoring (canaries) nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta UpdateCanary AWS CLICommand Reference.

CodeArtifact esempi utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with CodeArtifact.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`associate-external-connection`

Il seguente esempio di codice mostra come utilizzare`associate-external-connection`.

AWS CLI

Come aggiungere una connessione esterna a un repository

L'esempio `associate-external-connection` seguente aggiunge una connessione esterna a npmjs.com a un repository denominato test-repo.

```
aws codeartifact associate-external-connection \
    --repository test-repo \
    --domain test-domain \
    --external-connection public:npmjs
```

Output:

```
{
    "repository": {
        "name": "test-repo",
        "administratorAccount": "111122223333",
        "domainName": "test-domain",
        "domainOwner": "111122223333",
        "arn": "arn:aws:codeartifact:us-west-2:111122223333:repository/test-domain/
test-repo",
        "upstreams": [],
        "externalConnections": [
            {
                "externalConnectionName": "public:npmjs",
                "packageFormat": "npm",
                "status": "AVAILABLE"
            }
```

```
        ]
    }
}
```

Per ulteriori informazioni, consulta Aggiungere una connessione esterna nella Guida AWS CodeArtifact per l'utente.

- Per i dettagli sull'API, consulta AssociateExternalConnection AWS CLICommand Reference.

`copy-package-versions`

Il seguente esempio di codice mostra come utilizzare`copy-package-versions`.

AWS CLI

Come copiare versioni di pacchetti da un repository all'altro

L'esempio `copy-package-versions` seguente sposta le versioni 4.0.0 e 5.0.0 di un pacchetto denominato test-package da my-repo a test-repo.

```
aws codeartifact copy-package-versions \
    --domain test-domain \
    --source-repository my-repo \
    --destination-repository test-repo \
    --format npm \
    --package test-package \
    --versions '["4.0.0", "5.0.0"]'
```

Output:

```
{
    "format": "npm",
    "package": "test-package",
    "versions": [
        {
        "version": "5.0.0",
        "revision": "REVISION-1-SAMPLE-6C81EFF7DA55CC",
        "status": "Published"
        },
        {
        "version": "4.0.0",
        "revision": "REVISION-2-SAMPLE-55C752BEE772FC",
```

```
        "status": "Published"
        }
    ]
}
```

Per ulteriori informazioni, consulta Copiare i pacchetti tra i repository nella Guida per l'AWS CodeArtifact utente.

- Per i dettagli sull'API, consulta CopyPackageVersions AWS CLICommand Reference.

`create-domain`

Il seguente esempio di codice mostra come utilizzare`create-domain`.

AWS CLI

Come creare un dominio

L'esempio `create-domain` seguente crea un dominio denominato test-domain.

```
aws codeartifact create-domain \
    --domain test-domain
```

Output:

```
{
    "domain": {
        "name": "test-domain",
        "owner": "111122223333",
        "arn": "arn:aws:codeartifact:us-west-2:111122223333:domain/test-domain",
        "status": "Active",
        "createdTime": "2020-10-20T13:16:48.559000-04:00",
        "encryptionKey": "arn:aws:kms:us-west-2:111122223333:key/a1b2c3d4-5678-90ab-
cdef-EXAMPLE11111",
        "repositoryCount": 0,
        "assetSizeBytes": 0
    }
}
```

Per ulteriori informazioni, consulta Creare un dominio nella Guida AWS CodeArtifact per l'utente.

- Per i dettagli sull'API, consulta CreateDomain AWS CLICommand Reference.

`create-repository`

Il seguente esempio di codice mostra come utilizzare`create-repository`.

AWS CLI

Come creare un repository

L'esempio `create-repository` seguente crea un repository denominato test-repo all'interno di un dominio denominato test-domain.

```
aws codeartifact create-repository \
    --domain test-domain \
    --domain-owner 111122223333 \
    --repository test-repo \
    --description "This is a test repository."
```

Output:

```
{
    "repository": {
        "name": "test-repo",
        "administratorAccount": "111122223333",
        "domainName": "test-domain",
        "domainOwner": "111122223333",
        "arn": "arn:aws:codeartifact:us-west-2:111122223333:repository/test-domain/
test-repo",
        "description": "This is a test repository.",
        "upstreams": [],
        "externalConnections": []
    }
}
```

Per ulteriori informazioni, consulta Creare un dominio nella Guida AWS CodeArtifact per l'utente.

- Per i dettagli sull'API, consulta CreateRepository AWS CLICommand Reference.

`delete-domain-permissions-policy`

Il seguente esempio di codice mostra come utilizzare`delete-domain-permissions-policy`.

AWS CLI

Come eliminare il documento della policy di autorizzazioni da un dominio

L'esempio `delete-domain-permissions-policy` seguente elimina la policy di autorizzazione da un dominio denominato test-domain.

```
aws codeartifact delete-domain-permissions-policy \
    --domain test-domain
```

Output:

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "BasicDomainPolicy",
            "Action": [
                "codeartifact:GetDomainPermissionsPolicy",
                "codeartifact:ListRepositoriesInDomain",
                "codeartifact:GetAuthorizationToken",
                "codeartifact:CreateRepository"
            ],
            "Effect": "Allow",
            "Resource": "*",
            "Principal": {
                "AWS": "arn:aws:iam::111122223333:root"
            }
        }
    ]
}
```

Per ulteriori informazioni, consulta Eliminare una politica di dominio nella Guida AWS CodeArtifact per l'utente.

- Per i dettagli sull'API, consulta DeleteDomainPermissionsPolicy AWS CLICommand Reference.

`delete-domain`

Il seguente esempio di codice mostra come utilizzare`delete-domain`.

AWS CLI

Come eliminare un dominio

L'esempio `delete-domain` seguente elimina un dominio denominato `test-domain`.

```
aws codeartifact delete-domain \
    --domain test-domain
```

Output:

```
{
    "domain": {
        "name": "test-domain",
        "owner": "417498243647",
        "arn": "arn:aws:codeartifact:us-west-2:417498243647:domain/test-domain",
        "status": "Deleted",
        "createdTime": "2020-10-20T13:16:48.559000-04:00",
        "encryptionKey": "arn:aws:kms:us-west-2:417498243647:key/c9fe2447-0795-4fda-
afbe-8464574ae162",
        "repositoryCount": 0,
        "assetSizeBytes": 0
    }
}
```

Per ulteriori informazioni, consulta Eliminare un dominio nella Guida AWS CodeArtifact per l'utente.

- Per i dettagli sull'API, consulta DeleteDomain AWS CLICommand Reference.

`delete-package-versions`

Il seguente esempio di codice mostra come utilizzare`delete-package-versions`.

AWS CLI

Come eliminare versioni di pacchetti

L'esempio `delete-package-versions` seguente elimina la versione 4.0.0 di un pacchetto denominato test-package.

```
aws codeartifact delete-package-versions \
```

```
    --domain test-domain \
    --repo test-repo \
    --format npm \
    --package test-package \
    --versions 4.0.0
```

Output:

```
{
    "successfulVersions": {
        "4.0.0": {
            "revision": "Ciqe5/9yicvkJT13b5/LdLpCyE6fqA7poa9qp+FilPs=",
            "status": "Deleted"
        }
    },
    "failedVersions": {}
}
```

Per ulteriori informazioni, consulta Eliminare una versione del pacchetto nella Guida AWS CodeArtifact per l'utente.

- Per i dettagli sull'API, consulta DeletePackageVersions AWS CLICommand Reference.

`delete-repository-permissions-policy`

Il seguente esempio di codice mostra come utilizzare`delete-repository-permissions-policy`.

AWS CLI

Come eliminare una policy di autorizzazioni da un repository

L'esempio `delete-repository-permissions-policy` seguente elimina la policy di autorizzazione da un repository denominato test-repo.

```
aws codeartifact delete-repository-permissions-policy \
    --domain test-domain \
    --repository test-repo
```

Output:

```
{
```

```
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::111122223333:root"
            },
            "Action": [
                "codeartifact:DescribePackageVersion",
                "codeartifact:DescribeRepository",
                "codeartifact:GetPackageVersionReadme",
                "codeartifact:GetRepositoryEndpoint",
                "codeartifact:ListPackages",
                "codeartifact:ListPackageVersions",
                "codeartifact:ListPackageVersionAssets",
                "codeartifact:ListPackageVersionDependencies",
                "codeartifact:ReadFromRepository"
            ],
            "Resource": "*"
        }
    ]
}
```

Per ulteriori informazioni, vedere Eliminare una politica nella Guida AWS CodeArtifact per l'utente.

- Per i dettagli sull'API, consulta DeleteRepositoryPermissionsPolicy AWS CLICommand Reference.

`delete-repository`

Il seguente esempio di codice mostra come utilizzare`delete-repository`.

AWS CLI

Come eliminare un repository

L'esempio `delete-repository` seguente elimina un repository denominato `test-repo` in un dominio denominato `test-domain`.

```
aws codeartifact delete-repository \
    --domain test-domain \
    --repository test-repo
```

Output:

```
{
    "repository": {
        "name": "test-repo",
        "administratorAccount": "111122223333",
        "domainName": "test-domain",
        "domainOwner": "111122223333",
        "arn": "arn:aws:codeartifact:us-west-2:111122223333:repository/test-domain/
test-repo",
        "description": "This is a test repository",
        "upstreams": [],
        "externalConnections": []
    }
}
```

Per ulteriori informazioni, consulta Eliminare un repository nella Guida per l'AWS CodeArtifact utente.

- Per i dettagli sull'API, consulta DeleteRepository AWS CLICommand Reference.

`describe-domain`

Il seguente esempio di codice mostra come utilizzare`describe-domain`.

AWS CLI

Come ottenere le informazioni su un dominio

L'`describe-domain`esempio seguente restituisce un DomainDescription oggetto per un dominio denominato test-domain.

```
aws codeartifact describe-domain \
    --domain test-domain
```

Output:

```
{
    "domain": {
        "name": "test-domain",
        "owner": "111122223333",
        "arn": "arn:aws:codeartifact:us-west-2:111122223333:domain/test-domain",
```

```
        "status": "Active",
        "createdTime": "2020-10-20T13:16:48.559000-04:00",
        "encryptionKey": "arn:aws:kms:us-west-2:111122223333:key/a1b2c3d4-5678-90ab-
cdef-EXAMPLE11111",
        "repositoryCount": 2,
        "assetSizeBytes": 0,
        "s3BucketArn": "arn:aws:s3:::assets-111122223333-us-west-2"
    }
}
```

Per ulteriori informazioni, vedere Panoramica del dominio nella Guida per l'AWS CodeArtifact utente.

- Per i dettagli sull'API, consulta DescribeDomain AWS CLICommand Reference.

`describe-repository`

Il seguente esempio di codice mostra come utilizzare`describe-repository`.

AWS CLI

Come ottenere informazioni su un repository

L'`describe-repository`esempio seguente restituisce un RepositoryDescription oggetto per un repository denominato test-repo.

```
aws codeartifact describe-repository \
    --domain test-domain \
    --repository test-repo
```

Output:

```
{
    "repository": {
        "name": "test-repo",
        "administratorAccount": "111122223333",
        "domainName": "test-domain",
        "domainOwner": "111122223333",
        "arn": "arn:aws:codeartifact:us-west-2:111122223333:repository/test-domain/
test-repo",
        "description": "This is a test repository.",
        "upstreams": [],
```

```
        "externalConnections": []
    }
}
```

Per ulteriori informazioni, consulta Creare un dominio nella Guida per l'utente.AWS CodeArtifact

- Per i dettagli sull'API, consulta DescribeRepository AWS CLICommand Reference.

`disassociate-external-connection`

Il seguente esempio di codice mostra come utilizzare`disassociate-external-connection`.

AWS CLI

Come rimuovere una connessione esterna da un repository

L'esempio `disassociate-external-connection` seguente rimuove una connessione esterna a npmjs.com da un repository denominato test-repo.

```
aws codeartifact disassociate-external-connection \
    --repository test-repo \
    --domain test-domain \
    --external-connection public:npmjs
```

Output:

```
{
    "repository": {
        "name": "test-repo",
        "administratorAccount": "111122223333",
        "domainName": "test-domain",
        "domainOwner": "111122223333",
        "arn": "arn:aws:codeartifact:us-west-2:111122223333:repository/test-domain/
test-repo",
        "upstreams": [],
        "externalConnections": []
    }
}
```

Per ulteriori informazioni, consulta Rimuovere una connessione esterna nella Guida AWS CodeArtifact per l'utente.

- Per i dettagli sull'API, consulta DisassociateExternalConnection AWS CLICommand Reference.

`dispose-package-versions`

Il seguente esempio di codice mostra come utilizzare`dispose-package-versions`.

AWS CLI

Come eliminare gli asset di una versione del pacchetto e impostarne lo stato su Rimosso

L'esempio `dispose-package-versions` seguente elimina gli asset della versione 4.0.0 di test-package e ne imposta lo stato su "Disposed" (Disposto).

```
aws codeartifact dispose-package-versions \
    --domain test-domain \
    --repo test-repo \
    --format npm \
    --package test-package \
    --versions 4.0.0
```

Output:

```
{
    "successfulVersions": {
        "4.0.0": {
            "revision": "Ciqe5/9yicvkJT13b5/LdLpCyE6fqA7poa9qp+FilPs=",
            "status": "Disposed"
        }
    },
    "failedVersions": {}
}
```

Per ulteriori informazioni, vedere Lavorare con i pacchetti CodeArtifact nella Guida AWS CodeArtifact per l'utente.

- Per i dettagli sull'API, consulta DisposePackageVersions AWS CLICommand Reference.

`get-authorization-token`

Il seguente esempio di codice mostra come utilizzare`get-authorization-token`.

AWS CLI

Come ottenere un token di autorizzazione

L'`get-authorization-token`esempio seguente recupera un token di CodeArtifact autorizzazione.

```
aws codeartifact get-authorization-token \
    --domain test-domain \
    --query authorizationToken \
    --output text
```

Output:

```
This command will return the authorization token. You can store the output in an
 environment variable when calling the command.
```

Per ulteriori informazioni, consulta Configura pip senza il comando login nella Guida per l'AWS CodeArtifact utente.

- Per i dettagli sull'API, consulta GetAuthorizationToken AWS CLICommand Reference.

`get-domain-permissions-policy`

Il seguente esempio di codice mostra come utilizzare`get-domain-permissions-policy`.

AWS CLI

Come ottenere il documento della policy di autorizzazioni per un dominio

L'esempio `get-domain-permissions-policy` seguente ottiene la policy di autorizzazione associata a un dominio denominato test-domain.

```
aws codeartifact get-domain-permissions-policy \
    --domain test-domain
```

Output:

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "BasicDomainPolicy",
            "Action": [
                "codeartifact:GetDomainPermissionsPolicy",
                "codeartifact:ListRepositoriesInDomain",
```

```
                "codeartifact:GetAuthorizationToken",
                "codeartifact:CreateRepository"
            ],
            "Effect": "Allow",
            "Resource": "*",
            "Principal": {
                "AWS": "arn:aws:iam::111122223333:root"
            }
        }
    ]
}
```

Per ulteriori informazioni, consulta Leggere una politica di dominio nella Guida AWS CodeArtifact per l'utente.

- Per i dettagli sull'API, consulta GetDomainPermissionsPolicy AWS CLICommand Reference.

`get-package-version-asset`

Il seguente esempio di codice mostra come utilizzare`get-package-version-asset`.

AWS CLI

Come ottenere un asset da una versione del pacchetto

L'esempio `get-package-version-asset` seguente recupera l'asset `package.tgz` per la versione 4.0.0 di un pacchetto npm denominato test-package.

```
aws codeartifact get-package-version-asset \
    --domain test-domain \
    --repository test-repo \
    --format npm \
    --package test-package \
    --package-version 4.0.0 \
    --asset 'package.tgz' \
    outfileName
```

Output:

```
The output for this command will also store the raw asset in the file provided in
 place of outfileName.

{
```

```
    "assetName": "package.tgz",
    "packageVersion": "4.0.0",
    "packageVersionRevision": "Ciqe5/9yicvkJT13b5/LdLpCyE6fqA7poa9qp+FilPs="
}
```

Per ulteriori informazioni, vedete Elencare le risorse della versione del pacchetto nella Guida AWS CodeArtifact per l'utente.

- Per i dettagli sull'API, consulta GetPackageVersionAsset AWS CLICommand Reference.

`get-package-version-readme`

Il seguente esempio di codice mostra come utilizzare`get-package-version-readme`.

AWS CLI

Come ottenere il file readme di una versione del pacchetto

L'esempio `get-package-version-readme` seguente recupera l'asset del file Readme per la versione 4.0.0 di un pacchetto npm denominato test-package.

```
aws codeartifact get-package-version-readme \
    --domain test-domain \
    --repo test-repo \
    --format npm \
    --package test-package \
    --package-version 4.0.0
```

Output:

```
{
    "format": "npm",
    "package": "test-package",
    "version": "4.0.0",
    "readme": "<div align=\"center\">\n   <a href=\https://github.com/test-package/
testpack\"> ... more content ... \n",
    "versionRevision": "Ciqe5/9yicvkJT13b5/LdLpCyE6fqA7poa9qp+FilPs="
}
```

Per ulteriori informazioni, consulta Visualizza il file readme della versione del pacchetto nella Guida per l'AWS CodeArtifact utente.

- Per i dettagli sull'API, consulta GetPackageVersionReadme AWS CLICommand Reference.

`get-repository-endpoint`

Il seguente esempio di codice mostra come utilizzare`get-repository-endpoint`.

AWS CLI

Come ottenere l'endpoint URL di un repository

L'esempio `get-repository-endpoint` seguente restituisce l'endpoint npm per il repository test-repo.

```
aws codeartifact get-repository-endpoint \
    --domain test-domain \
    --repository test-repo \
    --format npm
```

Output:

```
{
    "repositoryEndpoint": "https://test-domain-111122223333.d.codeartifact.us-
west-2.amazonaws.com/npm/test-repo/"
}
```

Per ulteriori informazioni, consulta Connect to a repository nella Guida per l'AWS CodeArtifact utente.

- Per i dettagli sull'API, consulta GetRepositoryEndpoint AWS CLICommand Reference.

`get-repository-permissions-policy`

Il seguente esempio di codice mostra come utilizzare`get-repository-permissions-policy`.

AWS CLI

Come ottenere il documento della policy di autorizzazioni per un repository

L'esempio `get-repository-permissions-policy` seguente ottiene la policy di autorizzazione associata a un repository denominato test-repo.

```
aws codeartifact get-repository-permissions-policy \
    --domain test-domain \
    --repository test-repo
```

Output:

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::111122223333:root"
            },
            "Action": [
                "codeartifact:DescribePackageVersion",
                "codeartifact:DescribeRepository",
                "codeartifact:GetPackageVersionReadme",
                "codeartifact:GetRepositoryEndpoint",
                "codeartifact:ListPackages",
                "codeartifact:ListPackageVersions",
                "codeartifact:ListPackageVersionAssets",
                "codeartifact:ListPackageVersionDependencies",
                "codeartifact:ReadFromRepository"
            ],
            "Resource": "*"
        }
    ]
}
```

Per ulteriori informazioni, consulta Leggere una politica nella Guida AWS CodeArtifact per l'utente.

- Per i dettagli sull'API, consulta GetRepositoryPermissionsPolicy AWS CLICommand Reference.

`list-domains`

Il seguente esempio di codice mostra come utilizzare`list-domains`.

AWS CLI

Come elencare i domini

L'`list-domains`esempio seguente restituisce un riepilogo di tutti i domini di proprietà dell' AWS account che effettua la chiamata.

```
aws codeartifact list-domains
```

Output:

```
{
    "domains": [
        {
            "name": "my-domain",
            "owner": "111122223333",
            "status": "Active",
            "encryptionKey": "arn:aws:kms:us-west-2:111122223333:key/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
        },
        {
            "name": "test-domain",
            "owner": "111122223333",
            "status": "Active",
            "encryptionKey": "arn:aws:kms:us-west-2:111122223333:key/
a1b2c3d4-5678-90ab-cdef-EXAMPLE22222"
        }
    ]
}
```

Per ulteriori informazioni, vedere Lavorare con i domini CodeArtifact nella Guida per l'AWS CodeArtifact utente.

- Per i dettagli sull'API, consulta ListDomains AWS CLICommand Reference.

`list-package-version-assets`

Il seguente esempio di codice mostra come utilizzare`list-package-version-assets`.

AWS CLI

Come visualizzare gli asset di una versione del pacchetto

L'esempio `list-package-version-assets` seguente recupera gli asset per la versione 4.0.0 di un pacchetto npm denominato test-package.

```
aws codeartifact list-package-version-assets \
    --domain test-domain \
    --repo test-repo \
    --format npm \
    --package test-package \
```

```
    --package-version 4.0.0
```

Output:

```
{
    "format": "npm",
    "package": "test-package",
    "version": "4.0.0",
    "versionRevision": "Ciqe5/9yicvkJT13b5/LdLpCyE6fqA7poa9qp+FilPs=",
    "assets": [
        {
            "name": "package.tgz",
            "size": 316680,
            "hashes": {
                "MD5": "60078ec6d9e76b89fb55c860832742b2",
                "SHA-1": "b44a9b6297bcb698f1c51a3545a2b3b368d59c52",
                "SHA-256":
 "d2aa8c6afc3c8591765785a37d1c5acae482a8eb3ab9729ed28922692454f2e2",
                "SHA-512":
 "3e585d15c8a594e20d7de57b362ea81754c011acb2641a19f1b72c8531ea39825896bab344ae616a0a5a824cb9
            }
        }
    ]
}
```

Per ulteriori informazioni, vedete Elencare le risorse della versione del pacchetto nella Guida AWS CodeArtifact per l'utente.

- Per i dettagli sull'API, consulta ListPackageVersionAssets AWS CLICommand Reference.

`list-package-version-dependencies`

Il seguente esempio di codice mostra come utilizzare`list-package-version-dependencies`.

AWS CLI

Come visualizzare le dipendenze di una versione del pacchetto

L'esempio `list-package-version-dependencies` seguente recupera le dipendenze per la versione 4.0.0 di un pacchetto npm denominato test-package.

```
aws codeartifact list-package-version-dependencies \
```

```
    --domain test-domain \
    --repo test-repo \
    --format npm \
    --package test-package \
    --package-version 4.0.0
```

Output:

```
{
    "format": "npm",
    "package": "test-package",
    "version": "4.0.0",
    "versionRevision": "Ciqe5/9yicvkJT13b5/LdLpCyE6fqA7poa9qp+FilPs=",
    "dependencies": [
        {
            "namespace": "testns",
            "package": "testdep1",
            "dependencyType": "regular",
            "versionRequirement": "1.8.5"
        },
        {
            "namespace": "testns",
            "package": "testdep2",
            "dependencyType": "regular",
            "versionRequirement": "1.8.5"
        }
    ]
}
```

Per ulteriori informazioni, consulta Visualizzare e aggiornare i dettagli e le dipendenze della versione del pacchetto nella Guida per l'AWS CodeArtifact utente.

- Per i dettagli sull'API, consulta ListPackageVersionDependencies AWS CLICommand Reference.

`list-package-versions`

Il seguente esempio di codice mostra come utilizzare`list-package-versions`.

AWS CLI

Come elencare le versioni di un pacchetto

L'esempio `list-package-versions` seguente restituisce un elenco di versioni del pacchetto denominato `kind-of`.

```
aws codeartifact list-package-versions \
    --package kind-of \
    --domain test-domain \
    --repository test-repo \
    --format npm
```

Output:

```
{
    "defaultDisplayVersion": "1.0.1",
    "format": "npm",
    "package": "kind-of",
    "versions": [
        {
            "version": "1.0.1",
            "revision": "REVISION-SAMPLE-1-C7F4S5E9B772FC",
            "status": "Published"
        },
        {
            "version": "1.0.0",
            "revision": "REVISION-SAMPLE-2-C752BEEF6D2CFC",
            "status": "Published"
        },
        {
            "version": "0.1.2",
            "revision": "REVISION-SAMPLE-3-654S65A5C5E1FC",
            "status": "Published"
        },
        {
            "version": "0.1.1",
            "revision": "REVISION-SAMPLE-1-C7F4S5E9B772FC"",
            "status": "Published"
        },
        {
            "version": "0.1.0",
            "revision": "REVISION-SAMPLE-4-AF669139B772FC",
            "status": "Published"
        }
    ]
```

```
}
```

Per ulteriori informazioni, consulta Elenca le versioni dei pacchetti nella Guida AWS CodeArtifact per l'utente.

- Per i dettagli sull'API, consulta ListPackageVersions AWS CLICommand Reference.

`list-packages`

Il seguente esempio di codice mostra come utilizzare`list-packages`.

AWS CLI

Come elencare i pacchetti in un repository

L'esempio `list-packages` seguente elenca i pacchetti in un repository denominato `test-repo` in un dominio denominato `test-domain`.

```
aws codeartifact list-packages \
    --domain test-domain \
    --repository test-repo
```

Output:

```
{
    "packages": [
        {
            "format": "npm",
            "package": "lodash"
        }
        {
            "format": "python",
            "package": "test-package"
        }
    ]
}
```

Per ulteriori informazioni, vedete Elencare i nomi dei pacchetti nella Guida AWS CodeArtifact per l'utente.

- Per i dettagli sull'API, consulta ListPackages AWS CLICommand Reference.

`list-repositories-in-domain`

Il seguente esempio di codice mostra come utilizzare`list-repositories-in-domain`.

AWS CLI

Come elencare i repository in un dominio

L'esempio `list-repositories-in-domain` seguente restituisce un riepilogo di tutti i repository nel dominio test-domain.

```
aws codeartifact list-repositories-in-domain \
    --domain test-domain
```

Output:

```
{
    "repositories": [
        {
            "name": "test-repo",
            "administratorAccount": "111122223333",
            "domainName": "test-domain",
            "domainOwner": "111122223333",
            "arn": "arn:aws:codeartifact:us-west-2:111122223333:repository/test-
domain/test-repo",
            "description": "This is a test repository."
        },
        {
            "name": "test-repo2",
            "administratorAccount": "111122223333",
            "domainName": "test-domain",
            "domainOwner": "111122223333",
            "arn": "arn:aws:codeartifact:us-west-2:111122223333:repository/test-
domain/test-repo2",
            "description": "This is a test repository."
        }
    ]
}
```

Per ulteriori informazioni, consulta List repositories nella AWS CodeArtifact User Guide.

- Per i dettagli sull'API, consulta ListRepositoriesInDomain AWS CLICommand Reference.

`list-repositories`

Il seguente esempio di codice mostra come utilizzare`list-repositories`.

AWS CLI

Come elencare i repository

L'`list-repositories`esempio seguente restituisce un riepilogo di tutti gli archivi nel dominio di proprietà dell' AWS account che effettua la chiamata.

```
aws codeartifact list-repositories
```

Output:

```
{
    "repositories": [
        {
            "name": "npm-store",
            "administratorAccount": "111122223333",
            "domainName": "my-domain",
            "domainOwner": "111122223333",
            "arn": "arn:aws:codeartifact:us-west-2:111122223333:repository/my-
domain/npm-store",
            "description": "Provides npm artifacts from npm, Inc."
        },
        {
            "name": "target-repo",
            "administratorAccount": "111122223333",
            "domainName": "my-domain",
            "domainOwner": "111122223333",
            "arn": "arn:aws:codeartifact:us-west-2:111122223333:repository/my-
domain/target-repo",
            "description": "test target repo"
        },
        {
            "name": "test-repo2",
            "administratorAccount": "111122223333",
            "domainName": "test-domain",
            "domainOwner": "111122223333",
            "arn": "arn:aws:codeartifact:us-west-2:111122223333:repository/test-
domain/test-repo2",
            "description": "This is a test repository."
```

```
        }
    ]
}
```

Per ulteriori informazioni, consulta List repositories nella AWS CodeArtifact User Guide.

- Per i dettagli sull'API, consulta ListRepositories AWS CLICommand Reference.

`login`

Il seguente esempio di codice mostra come utilizzare`login`.

AWS CLI

Come configurare l'autenticazione al repository con il comando di accesso

L'esempio `login` seguente configura lo strumento di gestione di pacchetti npm con un repository denominato test-repo in un dominio chiamato test-domain.

```
aws codeartifact login \
    --domain test-domain \
    --repository test-repo \
    --tool npm
```

Output:

```
Successfully configured npm to use AWS CodeArtifact repository https://test-
domain-111122223333.d.codeartifact.us-west-2.amazonaws.com/npm/test-repo/
Login expires in 12 hours at 2020-11-12 01:53:16-05:00
```

Per ulteriori informazioni, consulta Guida introduttiva alla AWS CLI nella Guida per l'AWS CodeArtifact utente.

- Per informazioni dettagliate sull'API, consulta Login in AWS CLI Command Reference.

`put-domain-permissions-policy`

Il seguente esempio di codice mostra come utilizzare`put-domain-permissions-policy`.

AWS CLI

Come associare una policy di autorizzazioni a un dominio

L'esempio `put-domain-permissions-policy` seguente associa una policy di autorizzazione definita nel file policy.json a un dominio denominato test-domain.

```
aws codeartifact put-domain-permissions-policy \
    --domain test-domain \
    --policy-document file://PATH/TO/policy.json
```

Output:

```
{
    "policy": {
        "resourceArn": "arn:aws:codeartifact:region-id:111122223333:domain/test-
domain",
        "document": "{ ...policy document content...}",
        "revision": "MQlyyTQRASRU3HB58gBtSDHXG7Q3hvxxxxxxx="
    }
}
```

Per ulteriori informazioni, consulta Impostare una politica di dominio nella Guida AWS CodeArtifact per l'utente.

- Per i dettagli sull'API, consulta PutDomainPermissionsPolicy AWS CLICommand Reference.

`put-repository-permissions-policy`

Il seguente esempio di codice mostra come utilizzare`put-repository-permissions-policy`.

AWS CLI

Come associare una policy di autorizzazioni a un repository

L'esempio `put-repository-permissions-policy` seguente associa una policy di autorizzazione definita nel file policy.json a un repository denominato test-repo.

```
aws codeartifact put-repository-permissions-policy \
    --domain test-domain \
    --repository test-repo \
    --policy-document file://PATH/TO/policy.json
```

Output:

```
{
```

```
    "policy": {
        "resourceArn": "arn:aws:codeartifact:region-id:111122223333:repository/test-
domain/test-repo",
        "document": "{ ...policy document content...}",
        "revision": "MQlyyTQRASRU3HB58gBtSDHXG7Q3hvxxxxxxx="
    }
}
```

Per ulteriori informazioni, consulta Impostare una politica nella Guida AWS CodeArtifact per l'utente.

- Per i dettagli sull'API, consulta PutRepositoryPermissionsPolicy AWS CLICommand Reference.

`update-package-versions-status`

Il seguente esempio di codice mostra come utilizzare`update-package-versions-status`.

AWS CLI

Come aggiornare lo stato della versione del pacchetto

L'esempio `update-package-versions-status` seguente aggiorna lo stato della versione 4.0.0 del pacchetto test-package su "Archived" (Archiviato).

```
aws codeartifact update-package-versions-status \
    --domain test-domain \
    --repo test-repo \
    --format npm \
    --package test-package \
    --versions 4.0.0 \
    --target-status Archived
```

Output:

```
{
    "successfulVersions": {
        "4.0.0": {
            "revision": "Ciqe5/9yicvkJT13b5/LdLpCyE6fqA7poa9qp+FilPs=",
            "status": "Archived"
        }
    },
    "failedVersions": {}
```

```
}
```

Per ulteriori informazioni, consulta Aggiornare lo stato della versione del pacchetto nella Guida AWS CodeArtifact per l'utente.

- Per i dettagli sull'API, consulta UpdatePackageVersionsStatus AWS CLICommand Reference.

`update-repository`

Il seguente esempio di codice mostra come utilizzare`update-repository`.

AWS CLI

Come aggiornare un repository

L'esempio `update-repository` seguente aggiorna la descrizione di un repository denominato test-repo in un dominio denominato test-domain impostandola su "this is an updated description" (questa è una descrizione aggiornata).

```
aws codeartifact update-repository \
    --domain test-domain \
    --repository test-repo \
    --description "this is an updated description"
```

Output:

```
{
    "repository": {
        "name": "test-repo",
        "administratorAccount": "111122223333",
        "domainName": "test-domain",
        "domainOwner": "111122223333",
        "arn": "arn:aws:codeartifact:us-west-2:111122223333:repository/test-domain/
test-repo",
        "description": "this is an updated description",
        "upstreams": [],
        "externalConnections": []
    }
}
```

Per ulteriori informazioni, consulta Visualizzare o modificare la configurazione di un repository nella Guida per l'AWS CodeArtifact utente.

- Per i dettagli sull'API, consulta UpdateRepository AWS CLICommand Reference.

CodeBuild esempi utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with CodeBuild.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`batch-delete-builds`

Il seguente esempio di codice mostra come utilizzare`batch-delete-builds`.

AWS CLI

Per eliminare le build in AWS CodeBuild.

L'`batch-delete-builds`esempio seguente elimina le build in base a CodeBuild quanto specificato. IDs

```
aws codebuild batch-delete-builds --ids my-build-project-one:a1b2c3d4-5678-9012-
abcd-11111EXAMPLE my-build-project-two:a1b2c3d4-5678-9012-abcd-22222EXAMPLE
```

Output:

```
{
    "buildsNotDeleted": [
        {
            "id": "arn:aws:codebuild:us-west-2:123456789012:build/my-build-project-
one:a1b2c3d4-5678-9012-abcd-11111EXAMPLE",
            "statusCode": "BUILD_IN_PROGRESS"
```

```
        }
    ],
    "buildsDeleted": [
        "arn:aws:codebuild:us-west-2:123456789012:build/my-build-project-
two:a1b2c3d4-5678-9012-abcd-22222EXAMPLE"
    ]
}
```

Per ulteriori informazioni, consulta Delete Builds (AWS CLI) nella Guida per AWS CodeBuild l'utente.

- Per i dettagli sull'API, consulta AWS CLI Command BatchDeleteBuildsReference.

`batch-get-build-batches`

Il seguente esempio di codice mostra come utilizzare`batch-get-build-batches`.

AWS CLI

Per visualizzare i dettagli delle build in AWS CodeBuild.

L'`batch-get-build-batches`esempio seguente ottiene informazioni sui batch di compilazione CodeBuild con quanto specificato. IDs

```
aws codebuild batch-get-build-batches \
    --ids codebuild-demo-project:e9c4f4df-3f43-41d2-ab3a-60fe2EXAMPLE
```

Output:

```
{
    "buildBatches": [
        {
            "id": "codebuild-demo-project:e9c4f4df-3f43-41d2-ab3a-60fe2EXAMPLE",
            "arn": "arn:aws:codebuild:us-west-2:123456789012:build-batch/codebuild-
demo-project:e9c4f4df-3f43-41d2-ab3a-60fe2EXAMPLE",
            "startTime": "2020-11-03T21:52:20.775000+00:00",
            "endTime": "2020-11-03T21:56:59.784000+00:00",
            "currentPhase": "SUCCEEDED",
            "buildBatchStatus": "SUCCEEDED",
            "resolvedSourceVersion": "0a6546f68309560d08a310daac92314c4d378f6b",
            "projectName": "codebuild-demo-project",
            "phases": [
                {
```

```
                    "phaseType": "SUBMITTED",
                    "phaseStatus": "SUCCEEDED",
                    "startTime": "2020-11-03T21:52:20.775000+00:00",
                    "endTime": "2020-11-03T21:52:20.976000+00:00",
                    "durationInSeconds": 0
                },
                {
                    "phaseType": "DOWNLOAD_BATCHSPEC",
                    "phaseStatus": "SUCCEEDED",
                    "startTime": "2020-11-03T21:52:20.976000+00:00",
                    "endTime": "2020-11-03T21:52:57.401000+00:00",
                    "durationInSeconds": 36
                },
                {
                    "phaseType": "IN_PROGRESS",
                    "phaseStatus": "SUCCEEDED",
                    "startTime": "2020-11-03T21:52:57.401000+00:00",
                    "endTime": "2020-11-03T21:56:59.751000+00:00",
                    "durationInSeconds": 242
                },
                {
                    "phaseType": "COMBINE_ARTIFACTS",
                    "phaseStatus": "SUCCEEDED",
                    "startTime": "2020-11-03T21:56:59.751000+00:00",
                    "endTime": "2020-11-03T21:56:59.784000+00:00",
                    "durationInSeconds": 0
                },
                {
                    "phaseType": "SUCCEEDED",
                    "startTime": "2020-11-03T21:56:59.784000+00:00"
                }
            ],
            "source": {
                "type": "GITHUB",
                "location": "https://github.com/my-repo/codebuild-demo-project.git",
                "gitCloneDepth": 1,
                "gitSubmodulesConfig": {
                    "fetchSubmodules": false
                },
                "reportBuildStatus": false,
                "insecureSsl": false
            },
            "secondarySources": [],
            "secondarySourceVersions": [],
```

```
            "artifacts": {
                "location": ""
            },
            "secondaryArtifacts": [],
            "cache": {
                "type": "NO_CACHE"
            },
            "environment": {
                "type": "LINUX_CONTAINER",
                "image": "aws/codebuild/amazonlinux2-x86_64-standard:3.0",
                "computeType": "BUILD_GENERAL1_SMALL",
                "environmentVariables": [],
                "privilegedMode": false,
                "imagePullCredentialsType": "CODEBUILD"
            },
            "logConfig": {
                "cloudWatchLogs": {
                    "status": "ENABLED"
                },
                "s3Logs": {
                    "status": "DISABLED",
                    "encryptionDisabled": false
                }
            },
            "buildTimeoutInMinutes": 60,
            "queuedTimeoutInMinutes": 480,
            "complete": true,
            "initiator": "Strohm",
            "encryptionKey": "arn:aws:kms:us-west-2:123456789012:alias/aws/s3",
            "buildBatchNumber": 6,
            "buildBatchConfig": {
                "serviceRole": "arn:aws:iam::123456789012:role/service-role/
codebuild-demo-project",
                "restrictions": {
                    "maximumBuildsAllowed": 100
                },
                "timeoutInMins": 480
            },
            "buildGroups": [
                {
                    "identifier": "DOWNLOAD_SOURCE",
                    "ignoreFailure": false,
                    "currentBuildSummary": {
```

```
                        "arn": "arn:aws:codebuild:us-west-2:123456789012:build/
codebuild-demo-project:379737d8-bc35-48ec-97fd-776d27545315",
                        "requestedOn": "2020-11-03T21:52:21.394000+00:00",
                        "buildStatus": "SUCCEEDED",
                        "primaryArtifact": {
                            "type": "no_artifacts",
                            "identifier": "DOWNLOAD_SOURCE"
                        },
                        "secondaryArtifacts": []
                    }
                },
                {
                    "identifier": "linux_small",
                    "dependsOn": [],
                    "ignoreFailure": false,
                    "currentBuildSummary": {
                        "arn": "arn:aws:codebuild:us-west-2:123456789012:build/
codebuild-demo-project:dd785171-ed84-4bb6-8ede-ceeb86e54bdb",
                        "requestedOn": "2020-11-03T21:52:57.604000+00:00",
                        "buildStatus": "SUCCEEDED",
                        "primaryArtifact": {
                            "type": "no_artifacts",
                            "identifier": "linux_small"
                        },
                        "secondaryArtifacts": []
                    }
                },
                {
                    "identifier": "linux_medium",
                    "dependsOn": [
                        "linux_small"
                    ],
                    "ignoreFailure": false,
                    "currentBuildSummary": {
                        "arn": "arn:aws:codebuild:us-west-2:123456789012:build/
codebuild-demo-project:97cf7bd4-5313-4786-8243-4aef350a1267",
                        "requestedOn": "2020-11-03T21:54:18.474000+00:00",
                        "buildStatus": "SUCCEEDED",
                        "primaryArtifact": {
                            "type": "no_artifacts",
                            "identifier": "linux_medium"
                        },
                        "secondaryArtifacts": []
                    }
```

```
                },
                {
                    "identifier": "linux_large",
                    "dependsOn": [
                        "linux_medium"
                    ],
                    "ignoreFailure": false,
                    "currentBuildSummary": {
                        "arn": "arn:aws:codebuild:us-west-2:123456789012:build/
codebuild-demo-project:60a194cd-0d03-4337-9db1-d41476a17d27",
                        "requestedOn": "2020-11-03T21:55:39.203000+00:00",
                        "buildStatus": "SUCCEEDED",
                        "primaryArtifact": {
                            "type": "no_artifacts",
                            "identifier": "linux_large"
                        },
                        "secondaryArtifacts": []
                    }
                }
            ]
        }
    ],
    "buildBatchesNotFound": []
}
```

Per ulteriori informazioni, consulta Batch builds in AWS CodeBuild < https://docs.aws.amazon.com/codebuild/ latest/userguide/batch -build.html>) __ nella Guida per l'utente.AWS CodeBuild

- Per i dettagli sull'API, consulta Command Reference. BatchGetBuildBatchesAWS CLI

`batch-get-builds`

Il seguente esempio di codice mostra come utilizzare`batch-get-builds`.

AWS CLI

Per visualizzare i dettagli delle build in AWS CodeBuild.

L'`batch-get-builds`esempio seguente ottiene informazioni sulle compilazioni CodeBuild con quanto specificato. IDs

```
aws codebuild batch-get-builds --ids codebuild-demo-project:e9c4f4df-3f43-41d2-
ab3a-60fe2EXAMPLE codebuild-demo-project:815e755f-bade-4a7e-80f0-efe51EXAMPLE
```

Output:

```
{
    "buildsNotFound": [],
    "builds": [
        {
            "artifacts": {
                "md5sum": "0e95edf915048a0c22efe6d139fff837",
                "location": "arn:aws:s3:::codepipeline-us-west-2-820783811474/
CodeBuild-Python-Pip/BuildArtif/6DJsqQa",
                "encryptionDisabled": false,
                "sha256sum":
 "cfa0df33a090966a737f64ae4fe498969fdc842a0c9aec540bf93c37ac0d05a2"
            },
            "logs": {
                "cloudWatchLogs": {
                    "status": "ENABLED"
                },
                "s3Logs": {
                    "status": "DISABLED"
                },
                "streamName": "46472baf-8f6b-43c2-9255-b3b963af2732",
                "groupName": "/aws/codebuild/codebuild-demo-project",
                "deepLink": "https://console.aws.amazon.com/cloudwatch/
home?region=us-west-2#logEvent:group=/aws/codebuild/codebuild-demo-
project;stream=46472baf-8f6b-43c2-9255-b3b963af2732"
            },
            "timeoutInMinutes": 60,
            "environment": {
                "privilegedMode": false,
                "computeType": "BUILD_GENERAL1_MEDIUM",
                "image": "aws/codebuild/windows-base:1.0",
                "environmentVariables": [],
                "type": "WINDOWS_CONTAINER"
            },
            "projectName": "codebuild-demo-project",
            "buildComplete": true,
            "source": {
                "gitCloneDepth": 1,
                "insecureSsl": false,
```

```
            "type": "CODEPIPELINE"
        },
        "buildStatus": "SUCCEEDED",
        "secondaryArtifacts": [],
        "phases": [
            {
                "durationInSeconds": 0,
                "startTime": 1548717462.122,
                "phaseType": "SUBMITTED",
                "endTime": 1548717462.484,
                "phaseStatus": "SUCCEEDED"
            },
            {
                "durationInSeconds": 0,
                "startTime": 1548717462.484,
                "phaseType": "QUEUED",
                "endTime": 1548717462.775,
                "phaseStatus": "SUCCEEDED"
            },
            {
                "durationInSeconds": 34,
                "endTime": 1548717496.909,
                "contexts": [
                    {
                        "statusCode": "",
                        "message": ""
                    }
                ],
                "startTime": 1548717462.775,
                "phaseType": "PROVISIONING",
                "phaseStatus": "SUCCEEDED"
            },
            {
                "durationInSeconds": 15,
                "endTime": 1548717512.555,
                "contexts": [
                    {
                        "statusCode": "",
                        "message": ""
                    }
                ],
                "startTime": 1548717496.909,
                "phaseType": "DOWNLOAD_SOURCE",
                "phaseStatus": "SUCCEEDED"
```

```
            },
            {
                "durationInSeconds": 0,
                "endTime": 1548717512.734,
                "contexts": [
                    {
                        "statusCode": "",
                        "message": ""
                    }
                ],
                "startTime": 1548717512.555,
                "phaseType": "INSTALL",
                "phaseStatus": "SUCCEEDED"
            },
            {
                "durationInSeconds": 0,
                "endTime": 1548717512.924,
                "contexts": [
                    {
                        "statusCode": "",
                        "message": ""
                    }
                ],
                "startTime": 1548717512.734,
                "phaseType": "PRE_BUILD",
                "phaseStatus": "SUCCEEDED"
            },
            {
                "durationInSeconds": 9,
                "endTime": 1548717522.254,
                "contexts": [
                    {
                        "statusCode": "",
                        "message": ""
                    }
                ],
                "startTime": 1548717512.924,
                "phaseType": "BUILD",
                "phaseStatus": "SUCCEEDED"
            },
            {
                "durationInSeconds": 3,
                "endTime": 1548717525.498,
                "contexts": [
```

```
                    {
                        "statusCode": "",
                        "message": ""
                    }
                ],
                "startTime": 1548717522.254,
                "phaseType": "POST_BUILD",
                "phaseStatus": "SUCCEEDED"
            },
            {
                "durationInSeconds": 9,
                "endTime": 1548717534.646,
                "contexts": [
                    {
                        "statusCode": "",
                        "message": ""
                    }
                ],
                "startTime": 1548717525.498,
                "phaseType": "UPLOAD_ARTIFACTS",
                "phaseStatus": "SUCCEEDED"
            },
            {
                "durationInSeconds": 2,
                "endTime": 1548717536.846,
                "contexts": [
                    {
                        "statusCode": "",
                        "message": ""
                    }
                ],
                "startTime": 1548717534.646,
                "phaseType": "FINALIZING",
                "phaseStatus": "SUCCEEDED"
            },
            {
                "startTime": 1548717536.846,
                "phaseType": "COMPLETED"
            }
        ],
        "startTime": 1548717462.122,
        "encryptionKey": "arn:aws:kms:us-west-2:123456789012:alias/aws/s3",
        "initiator": "codepipeline/CodeBuild-Pipeline",
        "secondarySources": [],
```

```
            "serviceRole": "arn:aws:iam::123456789012:role/service-role/my-
codebuild-service-role",
            "currentPhase": "COMPLETED",
            "id": "codebuild-demo-project:e9c4f4df-3f43-41d2-ab3a-60fe2EXAMPLE",
            "cache": {
                "type": "NO_CACHE"
            },
            "sourceVersion": "arn:aws:s3:::codepipeline-us-west-2-820783811474/
CodeBuild-Python-Pip/SourceArti/1TspnN3.zip",
            "endTime": 1548717536.846,
            "arn": "arn:aws:codebuild:us-west-2:123456789012:build/codebuild-demo-
project:e9c4f4df-3f43-41d2-ab3a-60fe2EXAMPLE",
            "queuedTimeoutInMinutes": 480,
            "resolvedSourceVersion": "f2194c1757bbdcb0f8f229254a4b3c8b27d43e0b"
        },
        {
            "artifacts": {
                "md5sum": "",
                "overrideArtifactName": false,
                "location": "arn:aws:s3:::my-artifacts/codebuild-demo-project",
                "encryptionDisabled": false,
                "sha256sum": ""
            },
            "logs": {
                "cloudWatchLogs": {
                    "status": "ENABLED"
                },
                "s3Logs": {
                    "status": "DISABLED"
                },
                "streamName": "4dea3ca4-20ec-4898-b22a-a9eb9292775d",
                "groupName": "/aws/codebuild/codebuild-demo-project",
                "deepLink": "https://console.aws.amazon.com/cloudwatch/
home?region=us-west-2#logEvent:group=/aws/codebuild/codebuild-demo-
project;stream=4dea3ca4-20ec-4898-b22a-a9eb9292775d"
            },
            "timeoutInMinutes": 60,
            "environment": {
                "privilegedMode": false,
                "computeType": "BUILD_GENERAL1_MEDIUM",
                "image": "aws/codebuild/windows-base:1.0",
                "environmentVariables": [],
                "type": "WINDOWS_CONTAINER"
            },
```

```
            "projectName": "codebuild-demo-project",
            "buildComplete": true,
            "source": {
                "gitCloneDepth": 1,
                "location": "https://github.com/my-repo/codebuild-demo-project.git",
                "insecureSsl": false,
                "reportBuildStatus": false,
                "type": "GITHUB"
            },
            "buildStatus": "SUCCEEDED",
            "secondaryArtifacts": [],
            "phases": [
                {
                    "durationInSeconds": 0,
                    "startTime": 1548716241.89,
                    "phaseType": "SUBMITTED",
                    "endTime": 1548716242.241,
                    "phaseStatus": "SUCCEEDED"
                },
                {
                    "durationInSeconds": 0,
                    "startTime": 1548716242.241,
                    "phaseType": "QUEUED",
                    "endTime": 1548716242.536,
                    "phaseStatus": "SUCCEEDED"
                },
                {
                    "durationInSeconds": 33,
                    "endTime": 1548716276.171,
                    "contexts": [
                        {
                            "statusCode": "",
                            "message": ""
                        }
                    ],
                    "startTime": 1548716242.536,
                    "phaseType": "PROVISIONING",
                    "phaseStatus": "SUCCEEDED"
                },
                {
                    "durationInSeconds": 15,
                    "endTime": 1548716291.809,
                    "contexts": [
                        {
```

```
                        "statusCode": "",
                        "message": ""
                    }
                ],
                "startTime": 1548716276.171,
                "phaseType": "DOWNLOAD_SOURCE",
                "phaseStatus": "SUCCEEDED"
            },
            {
                "durationInSeconds": 0,
                "endTime": 1548716291.993,
                "contexts": [
                    {
                        "statusCode": "",
                        "message": ""
                    }
                ],
                "startTime": 1548716291.809,
                "phaseType": "INSTALL",
                "phaseStatus": "SUCCEEDED"
            },
            {
                "durationInSeconds": 0,
                "endTime": 1548716292.191,
                "contexts": [
                    {
                        "statusCode": "",
                        "message": ""
                    }
                ],
                "startTime": 1548716291.993,
                "phaseType": "PRE_BUILD",
                "phaseStatus": "SUCCEEDED"
            },
            {
                "durationInSeconds": 9,
                "endTime": 1548716301.622,
                "contexts": [
                    {
                        "statusCode": "",
                        "message": ""
                    }
                ],
                "startTime": 1548716292.191,
```

```
                    "phaseType": "BUILD",
                    "phaseStatus": "SUCCEEDED"
                },
                {
                    "durationInSeconds": 3,
                    "endTime": 1548716304.783,
                    "contexts": [
                        {
                            "statusCode": "",
                            "message": ""
                        }
                    ],
                    "startTime": 1548716301.622,
                    "phaseType": "POST_BUILD",
                    "phaseStatus": "SUCCEEDED"
                },
                {
                    "durationInSeconds": 8,
                    "endTime": 1548716313.775,
                    "contexts": [
                        {
                            "statusCode": "",
                            "message": ""
                        }
                    ],
                    "startTime": 1548716304.783,
                    "phaseType": "UPLOAD_ARTIFACTS",
                    "phaseStatus": "SUCCEEDED"
                },
                {
                    "durationInSeconds": 2,
                    "endTime": 1548716315.935,
                    "contexts": [
                        {
                            "statusCode": "",
                            "message": ""
                        }
                    ],
                    "startTime": 1548716313.775,
                    "phaseType": "FINALIZING",
                    "phaseStatus": "SUCCEEDED"
                },
                {
                    "startTime": 1548716315.935,
```

```
                    "phaseType": "COMPLETED"
                }
            ],
            "startTime": 1548716241.89,
            "secondarySourceVersions": [],
            "initiator": "my-codebuild-project",
            "arn": "arn:aws:codebuild:us-west-2:123456789012:build/codebuild-demo-
project:815e755f-bade-4a7e-80f0-efe51EXAMPLE",
            "encryptionKey": "arn:aws:kms:us-west-2:123456789012:alias/aws/s3",
            "serviceRole": "arn:aws:iam::123456789012:role/service-role/my-
codebuild-service-role",
            "currentPhase": "COMPLETED",
            "id": "codebuild-demo-project:815e755f-bade-4a7e-80f0-efe51EXAMPLE",
            "cache": {
                "type": "NO_CACHE"
            },
            "endTime": 1548716315.935,
            "secondarySources": [],
            "queuedTimeoutInMinutes": 480,
            "resolvedSourceVersion": "f2194c1757bbdcb0f8f229254a4b3c8b27d43e0b"
        }
    ]
}
```


Per ulteriori informazioni, consulta View Build Details (AWS CLI) nella Guida per l'AWS CodeBuild utente.

- Per i dettagli sull'API, consulta BatchGetBuilds AWS CLICommand Reference.

`batch-get-projects`

Il seguente esempio di codice mostra come utilizzare`batch-get-projects`.

AWS CLI

Per ottenere un elenco dei nomi dei progetti di AWS CodeBuild compilazione.

L'`batch-get-projects`esempio seguente ottiene un elenco di progetti di CodeBuild compilazione specificati per nome.

```
aws codebuild batch-get-projects --names codebuild-demo-project codebuild-demo-
project2 my-other-demo-project
```

Nell'output seguente, l'array `projectsNotFound` elenca tutti i nomi di progetto di compilazione specificati, ma non trovati. L'array `projects` elenca i dettagli per ogni progetto di compilazione di cui sono state trovate informazioni.


```
{
    "projectsNotFound": [],
    "projects": [
        {
            "encryptionKey": "arn:aws:kms:us-west-2:123456789012:alias/aws/s3",
            "name": "codebuild-demo-project2",
            "queuedTimeoutInMinutes": 480,
            "timeoutInMinutes": 60,
            "source": {
                "buildspec": "version: 0.2\n\n#env:\n  #variables:\n     # key:
 \"value\"\n     # key: \"value\"\n  #parameter-store:\n     # key: \"value\"\n
  # key:\"value\"\n\nphases:\n  #install:\n    #commands:\n      # - command\n
  # - command\n  #pre_build:\n    #commands:\n      # - command\n      # - command
\n  build:\n    commands:\n      # - command\n      # - command\n  #post_build:\n
   #commands:\n      # - command\n      # - command\n#artifacts:\n  #files:\n    #
 - location\n    # - location\n  #name: $(date +%Y-%m-%d)\n  #discard-paths: yes\n
 #base-directory: location\n#cache:\n  #paths:\n    # - paths",
                "type": "NO_SOURCE",
                "insecureSsl": false,
                "gitCloneDepth": 1
            },
            "artifacts": {
                "type": "NO_ARTIFACTS"
            },
            "badge": {
                "badgeEnabled": false
            },
            "lastModified": 1540588091.108,
            "created": 1540588091.108,
            "arn": "arn:aws:codebuild:us-west-2:123456789012:project/test-for-
sample",
            "secondarySources": [],
            "secondaryArtifacts": [],
            "cache": {
                "type": "NO_CACHE"
            },
            "serviceRole": "arn:aws:iam::123456789012:role/service-role/my-test-
role",
            "environment": {
```

```
                "image": "aws/codebuild/java:openjdk-8",
                "privilegedMode": true,
                "type": "LINUX_CONTAINER",
                "computeType": "BUILD_GENERAL1_SMALL",
                "environmentVariables": []
            },
            "tags": []
        },
        {
            "encryptionKey": "arn:aws:kms:us-west-2:123456789012:alias/aws/s3",
            "name": "my-other-demo-project",
            "queuedTimeoutInMinutes": 480,
            "timeoutInMinutes": 60,
            "source": {
                "location": "https://github.com/iversonic/codedeploy-sample.git",
                "reportBuildStatus": false,
                "buildspec": "buildspec.yml",
                "insecureSsl": false,
                "gitCloneDepth": 1,
                "type": "GITHUB",
                "auth": {
                    "type": "OAUTH"
                }
            },
            "artifacts": {
                "type": "NO_ARTIFACTS"
            },
            "badge": {
                "badgeEnabled": false
            },
            "lastModified": 1523401711.73,
            "created": 1523401711.73,
            "arn": "arn:aws:codebuild:us-west-2:123456789012:project/Project2",
            "cache": {
                "type": "NO_CACHE"
            },
            "serviceRole": "arn:aws:iam::123456789012:role/service-role/codebuild-
Project2-service-role",
            "environment": {
                "image": "aws/codebuild/nodejs:4.4.7",
                "privilegedMode": false,
                "type": "LINUX_CONTAINER",
                "computeType": "BUILD_GENERAL1_SMALL",
                "environmentVariables": []
```

```
            },
            "tags": []
        }
    ]
}
```

Per ulteriori informazioni, consulta Visualizza i dettagli di un progetto di compilazione (AWS CLI) nella Guida per l'AWS CodeBuild utente.

- Per i dettagli sull'API, consulta BatchGetProjects AWS CLICommand Reference.

`batch-get-report-groups`

Il seguente esempio di codice mostra come utilizzare`batch-get-report-groups`.

AWS CLI

Per ottenere informazioni su uno o più gruppi di report in AWS CodeBuild.

L'esempio `batch-get-report-groups` seguente recupera informazioni sul gruppo di report con l'ARN specificato.

```
aws codebuild batch-get-report-groups \
    --report-group-arns arn:aws:codebuild:<region-ID>:<user-ID>:report-group/
<report-group-name>
```

Output:

```
{
    "reportGroups": [
        {
            "arn": "arn:aws:codebuild:<region-ID>:<user-ID>:report-group/<report-
group-name>",
            "name": "report-group-name",
            "type": "TEST",
            "exportConfig": {
                "exportConfigType": "NO_EXPORT"
            },
            "created": "2020-10-01T18:04:08.466000+00:00",
            "lastModified": "2020-10-01T18:04:08.466000+00:00",
            "tags": []
        }
    ],
```

```
    "reportGroupsNotFound": []
}
```

Per ulteriori informazioni, consulta Lavorare con i gruppi di report nella Guida AWS CodeBuild per l'utente.

- Per i dettagli sull'API, consulta BatchGetReportGroups AWS CLICommand Reference.

`batch-get-reports`

Il seguente esempio di codice mostra come utilizzare`batch-get-reports`.

AWS CLI

Per ottenere informazioni su uno o più report in AWS CodeBuild.

L'`batch-get-reports`esempio seguente recupera le informazioni sui report con il valore specificato ARNs.

```
aws codebuild batch-get-reports \
    --report-arns arn:aws:codebuild:<region-ID>:<user-ID>:report/<report-group-name>:<report 1 ID> arn:aws:codebuild:<region-ID>:<user-ID>:report/<report-group-name>:<report 2 ID>
```

Output:

```
{
    "reports": [
        {
            "arn": "arn:aws:codebuild:<region-ID>:<user-ID>:report/<report-group-name>:<report 1 ID>",
            "type": "TEST",
            "name": "<report-group-name>",
            "reportGroupArn": "arn:aws:codebuild:<region-ID>:<user-ID>:report-group/<report-group-name>",
            "executionId": "arn:aws:codebuild:<region-ID>:<user-ID>:build/test-reports:<ID>",
            "status": "FAILED",
            "created": "2020-10-01T11:25:22.531000-07:00",
            "expired": "2020-10-31T11:25:22-07:00",
            "exportConfig": {
                "exportConfigType": "NO_EXPORT"
            },
```

```
            "truncated": false,
            "testSummary": {
                "total": 28,
                "statusCounts": {
                    "ERROR": 5,
                    "FAILED": 1,
                    "SKIPPED": 4,
                    "SUCCEEDED": 18,
                    "UNKNOWN": 0
                },
                "durationInNanoSeconds": 94000000
            }
        },
        {
            "arn": "arn:aws:codebuild:<region-ID>:<user-ID>:report/<report-group-
name>:<report 2 ID>",
            "type": "TEST",
            "name": "<report-group-name>",
            "reportGroupArn": "arn:aws:codebuild:<region-ID>:<user-ID>:report-group/
<report-group-name>",
            "executionId": "arn:aws:codebuild:<region-ID>:<user-ID>:build/test-
reports:<ID>",
            "status": "FAILED",
            "created": "2020-10-01T11:13:05.816000-07:00",
            "expired": "2020-10-31T11:13:05-07:00",
            "exportConfig": {
                "exportConfigType": "NO_EXPORT"
            },
            "truncated": false,
            "testSummary": {
                "total": 28,
                "statusCounts": {
                    "ERROR": 5,
                    "FAILED": 1,
                    "SKIPPED": 4,
                    "SUCCEEDED": 18,
                    "UNKNOWN": 0
                },
                "durationInNanoSeconds": 94000000
            }
        }
    ],
    "reportsNotFound": []
```

```
}
```

Per ulteriori informazioni, vedere Utilizzo dei report nella Guida per l'AWS CodeBuild utente.

- Per i dettagli sull'API, consulta BatchGetReports AWS CLICommand Reference.

`create-project`

Il seguente esempio di codice mostra come utilizzare`create-project`.

AWS CLI

Esempio 1: creare un progetto di AWS CodeBuild compilazione

L'`create-project`esempio seguente crea un progetto di CodeBuild compilazione utilizzando i file sorgente di un bucket S3

```
aws codebuild create-project \
    --name "my-demo-project" \
    --source "{\"type\": \"S3\",\"location\": \"codebuild-us-west-2-123456789012-
input-bucket/my-source.zip\"}" \
    --artifacts {"\"type\": \"S3\",\"location\": \"codebuild-us-west-2-123456789012-
output-bucket\""} \
    --environment "{\"type\": \"LINUX_CONTAINER\",\"image\": \"aws/codebuild/
standard:1.0\",\"computeType\": \"BUILD_GENERAL1_SMALL\"}" \
    --service-role "arn:aws:iam::123456789012:role/service-role/my-codebuild-
service-role"
```

Output:

```
{
    "project": {
        "arn": "arn:aws:codebuild:us-west-2:123456789012:project/my-demo-project",
        "name": "my-cli-demo-project",
        "encryptionKey": "arn:aws:kms:us-west-2:123456789012:alias/aws/s3",
        "serviceRole": "arn:aws:iam::123456789012:role/service-role/my-codebuild-
service-role",
        "lastModified": 1556839783.274,
        "badge": {
            "badgeEnabled": false
        },
        "queuedTimeoutInMinutes": 480,
        "environment": {
```

```
            "image": "aws/codebuild/standard:1.0",
            "computeType": "BUILD_GENERAL1_SMALL",
            "type": "LINUX_CONTAINER",
            "imagePullCredentialsType": "CODEBUILD",
            "privilegedMode": false,
            "environmentVariables": []
        },
        "artifacts": {
            "location": "codebuild-us-west-2-123456789012-output-bucket",
            "name": "my-cli-demo-project",
            "namespaceType": "NONE",
            "type": "S3",
            "packaging": "NONE",
            "encryptionDisabled": false
        },
        "source": {
            "type": "S3",
            "location": "codebuild-us-west-2-123456789012-input-bucket/my-
source.zip",
            "insecureSsl": false
        },
        "timeoutInMinutes": 60,
        "cache": {
            "type": "NO_CACHE"
        },
        "created": 1556839783.274
    }
}
```

Esempio 2: creare un progetto di AWS CodeBuild compilazione utilizzando un file di input JSON per i parametri

L'`create-project`esempio seguente crea un progetto di CodeBuild compilazione passando tutti i parametri richiesti in un file di input JSON. Crea il modello del file di input eseguendo il comando solo con `--generate-cli-skeleton parameter`.

```
aws codebuild create-project --cli-input-json file://create-project.json
```

Il file di input JSON `create-project.json` include il seguente contenuto.

```
{
    "name": "codebuild-demo-project",
```

```
    "source": {
        "type": "S3",
        "location": "codebuild-region-ID-account-ID-input-bucket/MessageUtil.zip"
    },
    "artifacts": {
        "type": "S3",
        "location": "codebuild-region-ID-account-ID-output-bucket"
    },
    "environment": {
        "type": "LINUX_CONTAINER",
        "image": "aws/codebuild/standard:1.0",
        "computeType": "BUILD_GENERAL1_SMALL"
    },
    "serviceRole": "serviceIAMRole"
}
```

Output:

```
{
    "project": {
        "name": "codebuild-demo-project",
        "serviceRole": "serviceIAMRole",
        "tags": [],
        "artifacts": {
            "packaging": "NONE",
            "type": "S3",
            "location": "codebuild-region-ID-account-ID-output-bucket",
            "name": "message-util.zip"
        },
        "lastModified": 1472661575.244,
        "timeoutInMinutes": 60,
        "created": 1472661575.244,
        "environment": {
            "computeType": "BUILD_GENERAL1_SMALL",
            "image": "aws/codebuild/standard:1.0",
            "type": "LINUX_CONTAINER",
            "environmentVariables": []
        },
        "source": {
            "type": "S3",
            "location": "codebuild-region-ID-account-ID-input-bucket/
MessageUtil.zip"
        },
```

```
        "encryptionKey": "arn:aws:kms:region-ID:account-ID:alias/aws/s3",
        "arn": "arn:aws:codebuild:region-ID:account-ID:project/codebuild-demo-
project"
    }
}
```

Per ulteriori informazioni, consulta Create a Build Project (AWS CLI) nella Guida per l'AWS CodeBuild utente.

- Per i dettagli sull'API, consulta CreateProject AWS CLICommand Reference.

`create-report-group`

Il seguente esempio di codice mostra come utilizzare`create-report-group`.

AWS CLI

Per creare un gruppo di report in AWS CodeBuild.

L'esempio `create-report-group` seguente crea un nuovo gruppo di report.

```
aws codebuild create-report-group \
    --cli-input-json file://create-report-group-source.json
```

Contenuto create-report-group-source di.json:

```
{
    "name": "cli-created-report-group",
    "type": "TEST",
    "exportConfig": {
        "exportConfigType": "S3",
        "s3Destination": {
            "bucket": "amzn-s3-demo-bucket",
            "path": "",
            "packaging": "ZIP",
            "encryptionDisabled": true
        }
    }
}
```

Output:

```
{
    "reportGroup": {
        "arn": "arn:aws:codebuild:<region-ID>:<user-ID>:report-group/cli-created-
report-group",
        "name": "cli-created-report-group",
        "type": "TEST",
        "exportConfig": {
            "exportConfigType": "S3",
            "s3Destination": {
                "bucket": "amzn-s3-demo-bucket",
                "path": "",
                "packaging": "ZIP",
                "encryptionDisabled": true
            }
        },
        "created": 1602020026.775,
        "lastModified": 1602020026.775
    }
}
```

Per ulteriori informazioni, consulta Lavorare con i gruppi di report nella Guida per l'AWS CodeBuild utente.

- Per i dettagli sull'API, consulta CreateReportGroup AWS CLICommand Reference.

`create-webhook`

Il seguente esempio di codice mostra come utilizzare`create-webhook`.

AWS CLI

Per creare filtri webhook per un progetto AWS CodeBuild

L'`create-webhook`esempio seguente crea un webhook per un CodeBuild progetto denominato `my-project` che ha due gruppi di filtri. Il primo gruppo di filtri specifica le richieste pull create, aggiornate o riaperte su ramificazioni con nomi di riferimento Git che corrispondono all'espressione regolare `^refs/heads/master$` e riferimenti head che corrispondono a `^refs/heads/myBranch$`. Il secondo gruppo di filtri specifica le richieste push su ramificazioni con nomi di riferimento Git che non corrispondono all'espressione regolare `^refs/heads/myBranch$`.

```
aws codebuild create-webhook \
```

```
    --project-name my-project \
    --filter-groups "[[{\"type\":\"EVENT\",\"pattern\":\"PULL_REQUEST_CREATED,
 PULL_REQUEST_UPDATED, PULL_REQUEST_REOPENED\"},{\"type\":\"HEAD_REF\",\"pattern
\":\"^refs/heads/myBranch$\",\"excludeMatchedPattern\":true},{\"type\":\"BASE_REF
\",\"pattern\":\"^refs/heads/master$\",\"excludeMatchedPattern\":true}],[{\"type\":
\"EVENT\",\"pattern\":\"PUSH\"},{\"type\":\"HEAD_REF\",\"pattern\":\"^refs/heads/
myBranch$\",\"excludeMatchedPattern\":true}]]"
```

Output:

```
{
    "webhook": {
        "payloadUrl": "https://codebuild.us-west-2.amazonaws.com/webhooks?
t=eyJlbmNyeXB0ZWREYXRhIjoiVVl5MGtoeGRwSzZFRXl2Wnh4bld1Z0tKZ291TVpQNEtFamQ3RDlDYWpRaGIreVFrdm
        "url": "https://api.github.com/repos/iversonic/codedeploy-sample/
hooks/105190656",
        "lastModifiedSecret": 1556311319.069,
        "filterGroups": [
            [
                {
                    "type": "EVENT",
                    "pattern": "PULL_REQUEST_CREATED, PULL_REQUEST_UPDATED,
 PULL_REQUEST_REOPENED",
                    "excludeMatchedPattern": false
                },
                {
                    "type": "HEAD_REF",
                    "pattern": "refs/heads/myBranch$",
                    "excludeMatchedPattern": true
                },
                {
                    "type": "BASE_REF",
                    "pattern": "refs/heads/master$",
                    "excludeMatchedPattern": true
                }
            ],
            [
                {
                    "type": "EVENT",
                    "pattern": "PUSH",
                    "excludeMatchedPattern": false
                },
                {
```

```
                    "type": "HEAD_REF",
                    "pattern": "refs/heads/myBranch$",
                    "excludeMatchedPattern": true
                }
            ]
        ]
    }
}
```

Per ulteriori informazioni, consulta Filter GitHub Webhook Events (SDK) nella Guida per l'AWS CodeBuild utente.

- Per i dettagli sull'API, consulta AWS CLI Command CreateWebhookReference.

`delete-build-batch`

Il seguente esempio di codice mostra come utilizzare`delete-build-batch`.

AWS CLI

Per eliminare un batch, compila AWS CodeBuild.

L'esempio `delete-build-batch` seguente elimina la compilazione in batch specificata.

```
aws codebuild delete-build-batch \
    --id <project-name>:<batch-ID>
```

Output:

```
{
    "statusCode": "BATCH_DELETED",
    "buildsDeleted": [
        "arn:aws:codebuild:<region-ID>:<account-ID>:build/<project-name>:<build-
ID>",
        "arn:aws:codebuild:<region-ID>:<account-ID>:build/<project-name>:<build-
ID>",
        "arn:aws:codebuild:<region-ID>:<account-ID>:build/<project-name>:<build-
ID>",
        "arn:aws:codebuild:<region-ID>:<account-ID>:build/<project-name>:<build-ID>"
    ],
    "buildsNotDeleted": []
}
```

Per ulteriori informazioni, consulta Batch builds AWS CodeBuild in the AWS CodeBuild User Guide.

- Per i dettagli sull'API, consulta DeleteBuildBatch AWS CLICommand Reference.

`delete-project`

Il seguente esempio di codice mostra come utilizzare`delete-project`.

AWS CLI

Per eliminare un progetto di AWS CodeBuild compilazione

L'`delete-project`esempio seguente elimina il progetto di CodeBuild compilazione specificato.

```
aws codebuild delete-project --name my-project
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Eliminare un progetto di compilazione (AWS CLI) nella Guida per l'AWS CodeBuild utente.

- Per i dettagli sull'API, consulta DeleteProject AWS CLICommand Reference.

`delete-report-group`

Il seguente esempio di codice mostra come utilizzare`delete-report-group`.

AWS CLI

Per eliminare un report si raggruppa in AWS CodeBuild.

L'esempio `delete-report-group` seguente elimina il gruppo di report con l'ARN specificato.

```
aws codebuild delete-report-group \
    --arn arn:aws:codebuild:<region-ID>:<user-ID>:report-group/<report-group-name>
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Lavorare con i gruppi di report nella Guida AWS CodeBuild per l'utente.

- Per i dettagli sull'API, consulta DeleteReportGroup AWS CLICommand Reference.

`delete-report`

Il seguente esempio di codice mostra come utilizzare`delete-report`.

AWS CLI

Per eliminare un rapporto in AWS CodeBuild.

L'esempio `delete-report` seguente elimina il report specificato.

```
aws codebuild delete-report \
    --arn arn:aws:codebuild:<region-ID>:<account-ID>:report/<report-group-name>:<report-ID>
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Lavorare con i report nella Guida AWS CodeBuild per l'utente.

- Per i dettagli sull'API, consulta DeleteReport AWS CLICommand Reference.

`delete-source-credentials`

Il seguente esempio di codice mostra come utilizzare`delete-source-credentials`.

AWS CLI

Come disconnettersi da un provider di origine e rimuovere i relativi token di accesso.

L'esempio `delete-source-credentials` seguente effettua la disconnessione da un provider di origine e rimuove i relativi token. Le credenziali di origine sono determinate dall'ARN delle credenziali di origine utilizzate per connettersi al provider di origine.

```
aws codebuild delete-source-credentials --arn arn-of-your-credentials
```

Output:

```
{
    "arn": "arn:aws:codebuild:your-region:your-account-id:token/your-server-type"
}
```

Per ulteriori informazioni, consulta Connect Source Provider with Access Tokens (CLI) nella Guida per AWS CodeBuild l'utente.

- Per i dettagli sull'API, consulta AWS CLI Command DeleteSourceCredentialsReference.

`delete-webhook`

Il seguente esempio di codice mostra come utilizzare`delete-webhook`.

AWS CLI

Per eliminare un filtro webhook da un progetto AWS CodeBuild

L'`delete-webhook`esempio seguente elimina un webhook dal progetto specificato. CodeBuild

```
aws codebuild delete-webhook --project-name my-project
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Stop Running Builds Automaticamente (AWS CLI) nella Guida per AWS CodeBuild l'utente.

- Per i dettagli sull'API, consulta AWS CLI Command DeleteWebhookReference.

`describe-code-coverages`

Il seguente esempio di codice mostra come utilizzare`describe-code-coverages`.

AWS CLI

Per ottenere informazioni dettagliate sulla copertura del codice, i risultati dei test in AWS CodeBuild.

L'esempio `describe-code-coverages` seguente ottiene informazioni sui risultati del test di copertura del codice nel report specificato.

```
aws codebuild describe-code-coverages \
    --report-arn arn:aws:codebuild:<region-ID>:<account-ID>:report/<report-group-name>:<report-ID>
```

Output:

```
{
    "codeCoverages": [
        {
```

```
            "id": "20a0adcc-db13-4b66-804b-ecaf9f852855",
            "reportARN": "arn:aws:codebuild:<region-ID>:972506530580:report/<report-
group-name>:<report-ID>",
            "filePath": "<source-file-1-path>",
            "lineCoveragePercentage": 83.33,
            "linesCovered": 5,
            "linesMissed": 1,
            "branchCoveragePercentage": 50.0,
            "branchesCovered": 1,
            "branchesMissed": 1,
            "expired": "2020-11-20T21:22:45+00:00"
        },
        {
            "id": "0887162d-bf57-4cf1-a164-e432373d1a83",
            "reportARN": "arn:aws:codebuild:<region-ID>:972506530580:report/<report-
group-name>:<report-ID>",
            "filePath": "<source-file-2-path>",
            "lineCoveragePercentage": 90.9,
            "linesCovered": 10,
            "linesMissed": 1,
            "branchCoveragePercentage": 50.0,
            "branchesCovered": 1,
            "branchesMissed": 1,
            "expired": "2020-11-20T21:22:45+00:00"
        }
    ]
}
```

Per ulteriori informazioni, consulta i report sulla copertura del codice nella Guida AWS CodeBuild per l'utente.

- Per i dettagli sull'API, consulta DescribeCodeCoverages AWS CLICommand Reference.

`describe-test-cases`

Il seguente esempio di codice mostra come utilizzare`describe-test-cases`.

AWS CLI

Per ottenere informazioni dettagliate sui casi di test in AWS CodeBuild.

L'esempio `describe-test-cases` seguente ottiene informazioni sui casi di test nel report specificato.

```
aws codebuild describe-test-cases \
    --report-arn arn:aws:codebuild:<region-ID>:<account-ID>:report/<report-group-name>:<report-ID>
```

Output:

```
{
    "testCases": [
        {
            "reportArn": "arn:aws:codebuild:<region-ID>:<account-ID>:report/<report-group-name>:<report-ID>",
            "testRawDataPath": "<test-report-path>",
            "prefix": "NUnit.Tests.Assemblies.MockTestFixture",
            "name": "NUnit.Tests.Assemblies.MockTestFixture.NotRunnableTest",
            "status": "ERROR",
            "durationInNanoSeconds": 0,
            "message": "No arguments were provided\n",
            "expired": "2020-11-20T17:52:10+00:00"
        },
        {
            "reportArn": "arn:aws:codebuild:<region-ID>:<account-ID>:report/<report-group-name>:<report-ID>",
            "testRawDataPath": "<test-report-path>",
            "prefix": "NUnit.Tests.Assemblies.MockTestFixture",
            "name": "NUnit.Tests.Assemblies.MockTestFixture.TestWithException",
            "status": "ERROR",
            "durationInNanoSeconds": 0,
            "message": "System.ApplicationException : Intentional Exception
\nat NUnit.Tests.Assemblies.MockTestFixture.MethodThrowsException()\nat
 NUnit.Tests.Assemblies.MockTestFixture.TestWithException()\n\n",
            "expired": "2020-11-20T17:52:10+00:00"
        }
    ]
}
```

Per ulteriori informazioni, consulta Working with test reporting AWS CodeBuild nella Guida per l'AWS CodeBuild utente.

- Per i dettagli sull'API, consulta DescribeTestCases AWS CLICommand Reference.

`import-source-credentials`

Il seguente esempio di codice mostra come utilizzare`import-source-credentials`.

AWS CLI

Connect un AWS CodeBuild utente a un provider di origine importando le credenziali per il provider di origine.

L'esempio `import-source-credentials` seguente importa un token per un repository Bitbucket che utilizza BASIC_AUTH per il tipo di autenticazione.

```
aws codebuild import-source-credentials --server-type BITBUCKET --auth-
type BASIC_AUTH --token my-Bitbucket-password --username my-Bitbucket-username
```

Output:

```
{
    "arn": "arn:aws:codebuild:us-west-2:123456789012:token/bitbucket"
}
```

Per ulteriori informazioni, consulta Connect Source Provider with Access Tokens (CLI) nella Guida per AWS CodeBuild l'utente.

- Per i dettagli sull'API, consulta AWS CLI Command ImportSourceCredentialsReference.

`invalidate-project-cache`

Il seguente esempio di codice mostra come utilizzare`invalidate-project-cache`.

AWS CLI

Per reimpostare la cache per un progetto di AWS CodeBuild compilazione.

L'`invalidate-project-cache`esempio seguente reimposta la cache per il CodeBuild progetto specificato.

```
aws codebuild invalidate-project-cache --project-name my-project
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Build Caching in CodeBuild nella Guida per l'AWS CodeBuild utente.

- Per i dettagli sull'API, consulta InvalidateProjectCache AWS CLICommand Reference.

`list-build-batches-for-project`

Il seguente esempio di codice mostra come utilizzare`list-build-batches-for-project`.

AWS CLI

Per elencare le build in batch per uno specifico progetto di compilazione in AWS CodeBuild.

L'`list-build-batches-for-project`esempio seguente elenca le build CodeBuild in batch per il progetto specificato.

```
aws codebuild list-build-batches-for-project \
    --project-name "<project-name>"
```

Output:

```
{
    "ids": [
        "<project-name>:<batch-ID>",
        "<project-name>:<batch-ID>"
    ]
}
```

Per ulteriori informazioni, consulta Batch builds AWS CodeBuild in the AWS CodeBuild User Guide.

- Per i dettagli sull'API, consulta ListBuildBatchesForProject AWS CLICommand Reference.

`list-build-batches`

Il seguente esempio di codice mostra come utilizzare`list-build-batches`.

AWS CLI

Per elencare le build in AWS CodeBuild batch.

L'`list-build-batches`esempio seguente elenca le build CodeBuild in batch per l'account corrente.

```
aws codebuild list-build-batches
```

Output:

```
{
    "ids": [
        "<project-name>:<batch-ID>",
        "<project-name>:<batch-ID>"
    ]
}
```

Per ulteriori informazioni, consulta Batch builds in AWS CodeBuild < https://docs.aws.amazon.com/codebuild/ latest/userguide/batch -build.html>) __ nella Guida per l'utente.AWS CodeBuild

- Per i dettagli sull'API, consulta Command Reference. ListBuildBatchesAWS CLI

`list-builds-for-project`

Il seguente esempio di codice mostra come utilizzare`list-builds-for-project`.

AWS CLI

Per visualizzare un elenco di build per un progetto di AWS CodeBuild compilazione.

L'`list-builds-for-project`esempio seguente elenca la build IDs in ordine decrescente per il progetto di CodeBuild compilazione specificato.

```
aws codebuild list-builds-for-project --project-name codebuild-demo-project --sort-order DESCENDING
```

Output:

```
{
    "ids": [
        "codebuild-demo-project:1a2b3c4d-5678-90ab-cdef-11111example",
        "codebuild-demo-project:1a2b3c4d-5678-90ab-cdef-22222example",
        "codebuild-demo-project:1a2b3c4d-5678-90ab-cdef-33333example",
        "codebuild-demo-project:1a2b3c4d-5678-90ab-cdef-44444example",
```

```
        "codebuild-demo-project:1a2b3c4d-5678-90ab-cdef-55555example"
    ]
}
```

Per ulteriori informazioni, consulta Visualizzare un elenco di build IDs for a Build Project (AWS CLI) nella Guida per l'AWS CodeBuild utente

- Per i dettagli sull'API, consulta ListBuildsForProject AWS CLICommand Reference.

`list-builds`

Il seguente esempio di codice mostra come utilizzare`list-builds`.

AWS CLI

Per ottenere un elenco di AWS CodeBuild build. IDs

L'`list-builds`esempio seguente ottiene un elenco CodeBuild IDs ordinato in ordine crescente.

```
aws codebuild list-builds --sort-order ASCENDING
```

L'output include un valore `nextToken` che indica che esiste altro output disponibile.

```
{
    "nextToken": "4AEA6u7J...The full token has been omitted for
 brevity...MzY2OA==",
    "ids": [
        "codebuild-demo-project:815e755f-bade-4a7e-80f0-efe51EXAMPLE"
        "codebuild-demo-project:84a7f3d1-d40e-4956-b4cf-7a9d4EXAMPLE"
            ... The full list of build IDs has been omitted for brevity ...
        "codebuild-demo-project:931d0b72-bf6f-4040-a472-5c707EXAMPLE"
    ]
}
```

Esegui di nuovo questo comando e specifica il valore `nextToken` nella risposta precedente come parametro per ottenere la parte successiva dell'output. Ripeti l'operazione finché non viene più visualizzato un valore `nextToken` nella risposta.

```
aws codebuild list-builds --sort-order ASCENDING --next-
token 4AEA6u7J...The full token has been omitted for brevity...MzY2OA==
```

Parte successiva dell'output:

```
{
    "ids": [
        "codebuild-demo-project:49015049-21cf-4b50-9708-df115EXAMPLE",
        "codebuild-demo-project:543e7206-68a3-46d6-a4da-759abEXAMPLE",
            ... The full list of build IDs has been omitted for brevity ...
        "codebuild-demo-project:c282f198-4582-4b38-bdc0-26f96EXAMPLE"
    ]
}
```

Per ulteriori informazioni, consulta Visualizza un elenco di build IDs (AWS CLI) nella Guida per l'AWS CodeBuild utente

- Per i dettagli sull'API, consulta ListBuilds AWS CLICommand Reference.

`list-curated-environment-images`

Il seguente esempio di codice mostra come utilizzare`list-curated-environment-images`.

AWS CLI

Per ottenere un elenco di immagini Docker gestite da AWS CodeBuild che puoi utilizzare per le tue build.

L'`list-curated-environment-images`esempio seguente elenca le immagini Docker gestite da CodeBuild che possono essere utilizzate per le build. :

```
aws codebuild list-curated-environment-images
```

Output:

```
{
    "platforms": [
        {
            "platform": "AMAZON_LINUX",
            "languages": [
                {
                    "language": "JAVA",
                    "images": [
                        {
                            "description": "AWS ElasticBeanstalk - Java 7 Running on
 Amazon Linux 64bit v2.1.3",
                            "name": "aws/codebuild/eb-java-7-amazonlinux-64:2.1.3",
```

```
                            "versions": [
                                "aws/codebuild/eb-java-7-amazonlinux-64:2.1.3-1.0.0"
                            ]
                        },
                        {
                            "description": "AWS ElasticBeanstalk - Java 8 Running on
 Amazon Linux 64bit v2.1.3",
                            "name": "aws/codebuild/eb-java-8-amazonlinux-64:2.1.3",
                            "versions": [
                                "aws/codebuild/eb-java-8-amazonlinux-64:2.1.3-1.0.0"
                            ]
                        },
                        ... LIST TRUNCATED FOR BREVITY ...
                    ]
                }
            ]
        }
    ]
}
```

Per ulteriori informazioni, consulta le immagini Docker fornite da CodeBuild nella Guida per l'AWS CodeBuild utente

- Per i dettagli sull'API, consulta ListCuratedEnvironmentImages AWS CLICommand Reference.

`list-projects`

Il seguente esempio di codice mostra come utilizzare`list-projects`.

AWS CLI

Per ottenere un elenco dei nomi dei progetti di AWS CodeBuild compilazione.

L'`list-projects`esempio seguente ottiene un elenco di progetti di CodeBuild compilazione ordinati per nome in ordine crescente.

```
aws codebuild list-projects --sort-by NAME --sort-order ASCENDING
```

L'output include un valore `nextToken` che indica che esiste altro output disponibile.

```
{
    "nextToken": "Ci33ACF6...The full token has been omitted for brevity...U
+AkMx8=",
```

```
    "projects": [
        "codebuild-demo-project",
        "codebuild-demo-project2",
            ... The full list of build project names has been omitted for
 brevity ...
        "codebuild-demo-project99"
    ]
}
```

Esegui di nuovo questo comando e specifica il valore `nextToken` della risposta precedente come parametro per ottenere la parte successiva dell'output. Ripeti l'operazione finché non viene più visualizzato un valore `nextToken` nella risposta.

```
aws codebuild list-projects  --sort-by NAME --sort-order ASCENDING --next-
token Ci33ACF6...The full token has been omitted for brevity...U+AkMx8=

{
    "projects": [
        "codebuild-demo-project100",
        "codebuild-demo-project101",

  ... The full list of build project names has been omitted for brevity ...
        "codebuild-demo-project122"

  ]
}
```

Per ulteriori informazioni, consulta Visualizza un elenco di nomi di progetti di compilazione (AWS CLI) nella Guida per l'AWS CodeBuild utente.

- Per i dettagli sull'API, consulta ListProjects AWS CLICommand Reference.

`list-report-groups`

Il seguente esempio di codice mostra come utilizzare`list-report-groups`.

AWS CLI

Per ottenere un elenco del gruppo di report ARNs in AWS CodeBuild.

L'`list-report-groups`esempio seguente recupera il gruppo di rapporti ARNs per l'account nella regione.

```
aws codebuild list-report-groups
```

Output:

```
{
    "reportGroups": [
        "arn:aws:codebuild:<region-ID>:<user-ID>:report-group/report-group-1",
        "arn:aws:codebuild:<region-ID>:<user-ID>:report-group/report-group-2",
        "arn:aws:codebuild:<region-ID>:<user-ID>:report-group/report-group-3"
    ]
}
```

Per ulteriori informazioni, consulta Lavorare con i gruppi di report nella Guida per l'AWS CodeBuild utente.

- Per i dettagli sull'API, consulta ListReportGroups AWS CLICommand Reference.

`list-reports-for-report-group`

Il seguente esempio di codice mostra come utilizzare`list-reports-for-report-group`.

AWS CLI

Per ottenere un elenco dei report in un report, raggruppa in AWS CodeBuild.

L'esempio `list-report-for-report-groups` seguente recupera i report nel gruppo di report specificato per l'account nella Regione.

```
aws codebuild list-reports-for-report-group \
    --report-group-arn arn:aws:codebuild:<region-ID>:<user-ID>:report-group/<report-group-name>
```

Output:

```
{
    "reports": [
        "arn:aws:codebuild:<region-ID>:<user-ID>:report/report-1",
        "arn:aws:codebuild:<region-ID>:<user-ID>:report/report-2",
        "arn:aws:codebuild:<region-ID>:<user-ID>:report/report-3"
    ]
}
```

Per ulteriori informazioni, consulta Lavorare con i gruppi di report nella Guida AWS CodeBuild per l'utente.

- Per i dettagli sull'API, consulta ListReportsForReportGroup AWS CLICommand Reference.

`list-reports`

Il seguente esempio di codice mostra come utilizzare`list-reports`.

AWS CLI

Per ottenere un elenco dei report relativi all'account corrente in AWS CodeBuild.

L'`list-reports`esempio seguente recupera ARNs i report per l'account corrente.

```
aws codebuild list-reports
```

Output:

```
{
    "reports": [
        "arn:aws:codebuild:<region-ID>:<user-ID>:report/<report-group-name>:<report
 ID>",
        "arn:aws:codebuild:<region-ID>:<user-ID>:report/<report-group-name>:<report
 ID>",
        "arn:aws:codebuild:<region-ID>:<user-ID>:report/<report-group-name>:<report
 ID>"
    ]
}
```

Per ulteriori informazioni, vedere Utilizzo dei report nella Guida per l'AWS CodeBuild utente.

- Per i dettagli sull'API, consulta ListReports AWS CLICommand Reference.

`list-shared-projects`

Il seguente esempio di codice mostra come utilizzare`list-shared-projects`.

AWS CLI

In cui elencare il progetto condiviso AWS CodeBuild.

L'`list-shared-projects`esempio seguente elenca i progetti CodeBuild condivisi disponibili per l'account corrente.

```
aws codebuild list-shared-projects
```

Output:

```
{
    "projects": [
        "arn:aws:codebuild:<region-ID>:<account-ID>:project/<shared-project-
name-1>",
        "arn:aws:codebuild:<region-ID>:<account-ID>:project/<shared-project-name-2>"
    ]
}
```

Per ulteriori informazioni, vedere Lavorare con progetti condivisi nella Guida AWS CodeBuild per l'utente.

- Per i dettagli sull'API, consulta ListSharedProjects AWS CLICommand Reference.

`list-shared-report-groups`

Il seguente esempio di codice mostra come utilizzare`list-shared-report-groups`.

AWS CLI

Per ottenere un elenco del gruppo di report condiviso ARNs in AWS CodeBuild.

L'`list-shared-report-groups`esempio seguente recupera il gruppo di report ARNs per l'account nella regione.

```
aws codebuild list-shared-report-groups
```

Output:

```
{
    "reportGroups": [
        "arn:aws:codebuild:<region-ID>:<user-ID>:report-group/report-group-1",
        "arn:aws:codebuild:<region-ID>:<user-ID>:report-group/report-group-2",
        "arn:aws:codebuild:<region-ID>:<user-ID>:report-group/report-group-3"
    ]
```

```
}
```

Per ulteriori informazioni, consulta Lavorare con i gruppi di report nella Guida per l'AWS CodeBuild utente.

- Per i dettagli sull'API, consulta ListSharedReportGroups AWS CLICommand Reference.

`list-source-credentials`

Il seguente esempio di codice mostra come utilizzare`list-source-credentials`.

AWS CLI

Per visualizzare un elenco di sourceCredentialsObjects

L'`list-source-credentials`esempio seguente elenca i token per un AWS account collegato a un account Bitbucket e a un account. GitHub Ogni oggetto `sourceCredentialsInfos` nella risposta contiene informazioni sulle credenziali di origine connesse.

```
aws codebuild list-source-credentials
```

Output:

```
{
    "sourceCredentialsInfos": [
        {
            "serverType": "BITBUCKET",
            "arn": "arn:aws:codebuild:us-west-2:123456789012:token/bitbucket",
            "authType": "BASIC_AUTH"
        },
        {
            "serverType": "GITHUB",
            "arn": "arn:aws:codebuild:us-west-2:123456789012:token/github",
            "authType": "OAUTH"
        }
    ]
}
```

Per ulteriori informazioni, consulta Connect Source Provider with Access Tokens (CLI) nella Guida per AWS CodeBuild l'utente.

- Per i dettagli sull'API, consulta AWS CLI Command ListSourceCredentialsReference.

`retry-build-batch`

Il seguente esempio di codice mostra come utilizzare`retry-build-batch`.

AWS CLI

Per riprovare la compilazione di un batch non riuscito. AWS CodeBuild

L'esempio `retry-build-batch` seguente riavvia la compilazione in batch specificata.

```
aws codebuild retry-build-batch \
    --id <project-name>:<batch-ID>
```

Output:

```
{
    "buildBatch": {
        "id": "<project-name>:<batch-ID>",
        "arn": "arn:aws:codebuild:<region-ID>:<account-ID>:build-batch/<project-
name>:<batch-ID>",
        "startTime": "2020-10-21T17:26:23.099000+00:00",
        "currentPhase": "SUBMITTED",
        "buildBatchStatus": "IN_PROGRESS",
        "resolvedSourceVersion": "3a9e11cb419e8fff14b03883dc4e64f6155aaa7e",
        "projectName": "<project-name>",
        "phases": [
            {
                "phaseType": "SUBMITTED",
                "phaseStatus": "SUCCEEDED",
                "startTime": "2020-10-21T17:26:23.099000+00:00",
                "endTime": "2020-10-21T17:26:23.457000+00:00",
                "durationInSeconds": 0
            },
            {
                "phaseType": "DOWNLOAD_BATCHSPEC",
                "phaseStatus": "SUCCEEDED",
                "startTime": "2020-10-21T17:26:23.457000+00:00",
                "endTime": "2020-10-21T17:26:54.902000+00:00",
                "durationInSeconds": 31
            },
            {
                "phaseType": "IN_PROGRESS",
                "phaseStatus": "CLIENT_ERROR",
                "startTime": "2020-10-21T17:26:54.902000+00:00",
```

```
                "endTime": "2020-10-21T17:28:16.060000+00:00",
                "durationInSeconds": 81
            },
            {
                "phaseType": "FAILED",
                "phaseStatus": "RETRY",
                "startTime": "2020-10-21T17:28:16.060000+00:00",
                "endTime": "2020-10-21T17:29:39.709000+00:00",
                "durationInSeconds": 83
            },
            {
                "phaseType": "SUBMITTED",
                "startTime": "2020-10-21T17:29:39.709000+00:00"
            }
        ],
        "source": {
            "type": "GITHUB",
            "location": "https://github.com/strohm-a/<project-name>-graph.git",
            "gitCloneDepth": 1,
            "gitSubmodulesConfig": {
                "fetchSubmodules": false
            },
            "reportBuildStatus": false,
            "insecureSsl": false
        },
        "secondarySources": [],
        "secondarySourceVersions": [],
        "artifacts": {
            "location": ""
        },
        "secondaryArtifacts": [],
        "cache": {
            "type": "NO_CACHE"
        },
        "environment": {
            "type": "LINUX_CONTAINER",
            "image": "aws/codebuild/amazonlinux2-x86_64-standard:3.0",
            "computeType": "BUILD_GENERAL1_SMALL",
            "environmentVariables": [],
            "privilegedMode": false,
            "imagePullCredentialsType": "CODEBUILD"
        },
        "logConfig": {
            "cloudWatchLogs": {
```

```
                "status": "ENABLED"
            },
            "s3Logs": {
                "status": "DISABLED",
                "encryptionDisabled": false
            }
        },
        "buildTimeoutInMinutes": 60,
        "queuedTimeoutInMinutes": 480,
        "complete": false,
        "initiator": "<username>",
        "encryptionKey": "arn:aws:kms:<region-ID>:<account-ID>:alias/aws/s3",
        "buildBatchNumber": 4,
        "buildBatchConfig": {
            "serviceRole": "arn:aws:iam::<account-ID>:role/service-role/<project-
name>",
            "restrictions": {
                "maximumBuildsAllowed": 100
            },
            "timeoutInMins": 480
        },
        "buildGroups": [
            {
                "identifier": "DOWNLOAD_SOURCE",
                "ignoreFailure": false,
                "currentBuildSummary": {
                    "arn": "arn:aws:codebuild:<region-ID>:<account-ID>:build/
<project-name>:<build-ID>",
                    "requestedOn": "2020-10-21T17:26:23.889000+00:00",
                    "buildStatus": "SUCCEEDED",
                    "primaryArtifact": {
                        "type": "no_artifacts",
                        "identifier": "DOWNLOAD_SOURCE"
                    },
                    "secondaryArtifacts": []
                }
            },
            {
                "identifier": "linux_small",
                "dependsOn": [],
                "ignoreFailure": false,
                "currentBuildSummary": {
                    "arn": "arn:aws:codebuild:<region-ID>:<account-ID>:build/
<project-name>:<build-ID>",
```

```
                    "requestedOn": "2020-10-21T17:26:55.115000+00:00",
                    "buildStatus": "FAILED",
                    "primaryArtifact": {
                        "type": "no_artifacts",
                        "identifier": "linux_small"
                    },
                    "secondaryArtifacts": []
                }
            },
            {
                "identifier": "linux_medium",
                "dependsOn": [
                    "linux_small"
                ],
                "ignoreFailure": false,
                "currentBuildSummary": {
                    "arn": "arn:aws:codebuild:<region-ID>:<account-ID>:build/
<project-name>:<build-ID>",
                    "requestedOn": "2020-10-21T17:26:54.594000+00:00",
                    "buildStatus": "STOPPED"
                }
            },
            {
                "identifier": "linux_large",
                "dependsOn": [
                    "linux_medium"
                ],
                "ignoreFailure": false,
                "currentBuildSummary": {
                    "arn": "arn:aws:codebuild:<region-ID>:<account-ID>:build/
<project-name>:<build-ID>",
                    "requestedOn": "2020-10-21T17:26:54.701000+00:00",
                    "buildStatus": "STOPPED"
                }
            }
        ]
    }
}
```

Per ulteriori informazioni, consulta Batch builds AWS CodeBuild in the AWS CodeBuild User Guide.

- Per i dettagli sull'API, consulta RetryBuildBatch AWS CLICommand Reference.

`retry-build`

Il seguente esempio di codice mostra come utilizzare`retry-build`.

AWS CLI

Per riprovare una compilazione non riuscita. AWS CodeBuild

L'esempio `retry-build` seguente riavvia la compilazione specificata.

```
aws codebuild retry-build \
    --id <project-name>:<build-ID>
```

Output:

```
{
    "build": {
        "id": "<project-name>:<build-ID>",
        "arn": "arn:aws:codebuild:<region-ID>:<account-ID>:build/<project-
name>:<build-ID>",
        "buildNumber": 9,
        "startTime": "2020-10-21T17:51:38.161000+00:00",
        "currentPhase": "QUEUED",
        "buildStatus": "IN_PROGRESS",
        "projectName": "<project-name>",
        "phases": [
            {
                "phaseType": "SUBMITTED",
                "phaseStatus": "SUCCEEDED",
                "startTime": "2020-10-21T17:51:38.161000+00:00",
                "endTime": "2020-10-21T17:51:38.210000+00:00",
                "durationInSeconds": 0
            },
            {
                "phaseType": "QUEUED",
                "startTime": "2020-10-21T17:51:38.210000+00:00"
            }
        ],
        "source": {
            "type": "GITHUB",
            "location": "<GitHub-repo-URL>",
            "gitCloneDepth": 1,
            "gitSubmodulesConfig": {
                "fetchSubmodules": false
```

```
            },
            "reportBuildStatus": false,
            "insecureSsl": false
        },
        "secondarySources": [],
        "secondarySourceVersions": [],
        "artifacts": {
            "location": ""
        },
        "secondaryArtifacts": [],
        "cache": {
            "type": "NO_CACHE"
        },
        "environment": {
            "type": "LINUX_CONTAINER",
            "image": "aws/codebuild/amazonlinux2-x86_64-standard:3.0",
            "computeType": "BUILD_GENERAL1_SMALL",
            "environmentVariables": [],
            "privilegedMode": false,
            "imagePullCredentialsType": "CODEBUILD"
        },
        "serviceRole": "arn:aws:iam::<account-ID>:role/service-role/<service-role-
name>",
        "logs": {
            "deepLink": "https://console.aws.amazon.com/cloudwatch/home?
region=<region-ID>#logEvent:group=null;stream=null",
            "cloudWatchLogsArn": "arn:aws:logs:<region-ID>:<account-ID>:log-
group:null:log-stream:null",
            "cloudWatchLogs": {
                "status": "ENABLED"
            },
            "s3Logs": {
                "status": "DISABLED",
                "encryptionDisabled": false
            }
        },
        "timeoutInMinutes": 60,
        "queuedTimeoutInMinutes": 480,
        "buildComplete": false,
        "initiator": "<username>",
        "encryptionKey": "arn:aws:kms:<region-ID>:<account-ID>:alias/aws/s3"
    }
}
```

Per ulteriori informazioni, consulta Batch builds AWS CodeBuild in the AWS CodeBuild User Guide.

- Per i dettagli sull'API, consulta RetryBuild AWS CLICommand Reference.

`start-build-batch`

Il seguente esempio di codice mostra come utilizzare`start-build-batch`.

AWS CLI

Per avviare una compilazione in batch AWS CodeBuild.

L'esempio `start-build-batch` seguente avvia la compilazione in batch del progetto specificato.

```
aws codebuild start-build-batch \
    --project-name <project-name>
```

Output:

```
{
    "buildBatch": {
        "id": "<project-name>:<batch-ID>",
        "arn": "arn:aws:codebuild:<region-ID>:<account-ID>:build-batch/<project-
name>:<batch-ID>",
        "startTime": "2020-10-21T16:54:24.740000+00:00",
        "currentPhase": "SUBMITTED",
        "buildBatchStatus": "IN_PROGRESS",
        "projectName": "<project-name>",
        "source": {
            "type": "GITHUB",
            "location": "<GitHub-repo-URL>",
            "gitCloneDepth": 1,
            "gitSubmodulesConfig": {
                "fetchSubmodules": false
            },
            "reportBuildStatus": false,
            "insecureSsl": false
        },
        "secondarySources": [],
        "secondarySourceVersions": [],
        "artifacts": {
```

```
            "location": ""
        },
        "secondaryArtifacts": [],
        "cache": {
            "type": "NO_CACHE"
        },
        "environment": {
            "type": "LINUX_CONTAINER",
            "image": "aws/codebuild/amazonlinux2-x86_64-standard:3.0",
            "computeType": "BUILD_GENERAL1_SMALL",
            "environmentVariables": [],
            "privilegedMode": false,
            "imagePullCredentialsType": "CODEBUILD"
        },
        "logConfig": {
            "cloudWatchLogs": {
                "status": "ENABLED"
            },
            "s3Logs": {
                "status": "DISABLED",
                "encryptionDisabled": false
            }
        },
        "buildTimeoutInMinutes": 60,
        "queuedTimeoutInMinutes": 480,
        "complete": false,
        "initiator": "<username>",
        "encryptionKey": "arn:aws:kms:<region-ID>:<account-ID>:alias/aws/s3",
        "buildBatchNumber": 3,
        "buildBatchConfig": {
            "serviceRole": "arn:aws:iam::<account-ID>:role/service-role/<service-
role-name>",
            "restrictions": {
                "maximumBuildsAllowed": 100
            },
            "timeoutInMins": 480
        }
    }
}
```

Per ulteriori informazioni, consulta Batch builds AWS CodeBuild in the AWS CodeBuild User Guide.

- Per i dettagli sull'API, consulta StartBuildBatch AWS CLICommand Reference.

`start-build`

Il seguente esempio di codice mostra come utilizzare`start-build`.

AWS CLI

Per iniziare a eseguire la build di un progetto di AWS CodeBuild compilazione.

L'`start-build`esempio seguente avvia una compilazione per il CodeBuild progetto specificato. La compilazione sostituisce sia l'impostazione del progetto relativamente al numero di minuti di permanenza della compilazione nella coda prima del timeout sia le impostazioni relative agli artefatti del progetto.

```
aws codebuild start-build \
    --project-name "my-demo-project" \
    --queued-timeout-in-minutes-override 5 \
    --artifacts-override {"\"type\": \"S3\",\"location\": \"arn:aws:s3:::artifacts-
override\",\"overrideArtifactName\":true"}
```

Output:

```
{
    "build": {
        "serviceRole": "arn:aws:iam::123456789012:role/service-role/my-codebuild-
service-role",
        "buildStatus": "IN_PROGRESS",
        "buildComplete": false,
        "projectName": "my-demo-project",
        "timeoutInMinutes": 60,
        "source": {
            "insecureSsl": false,
            "type": "S3",
            "location": "codebuild-us-west-2-123456789012-input-bucket/my-
source.zip"
        },
        "queuedTimeoutInMinutes": 5,
        "encryptionKey": "arn:aws:kms:us-west-2:123456789012:alias/aws/s3",
        "currentPhase": "QUEUED",
        "startTime": 1556905683.568,
        "environment": {
            "computeType": "BUILD_GENERAL1_MEDIUM",
            "environmentVariables": [],
```

```
            "type": "LINUX_CONTAINER",
            "privilegedMode": false,
            "image": "aws/codebuild/standard:1.0",
            "imagePullCredentialsType": "CODEBUILD"
        },
        "phases": [
            {
                "phaseStatus": "SUCCEEDED",
                "startTime": 1556905683.568,
                "phaseType": "SUBMITTED",
                "durationInSeconds": 0,
                "endTime": 1556905684.524
            },
            {
                "startTime": 1556905684.524,
                "phaseType": "QUEUED"
            }
        ],
        "logs": {
            "deepLink": "https://console.aws.amazon.com/cloudwatch/home?region=us-
west-2#logEvent:group=null;stream=null"
        },
        "artifacts": {
            "encryptionDisabled": false,
            "location": "arn:aws:s3:::artifacts-override/my-demo-project",
            "overrideArtifactName": true
        },
        "cache": {
            "type": "NO_CACHE"
        },
        "id": "my-demo-project::12345678-a1b2-c3d4-e5f6-11111EXAMPLE",
        "initiator": "my-aws-account-name",
        "arn": "arn:aws:codebuild:us-west-2:123456789012:build/my-demo-
project::12345678-a1b2-c3d4-e5f6-11111EXAMPLE"
    }
}
```

Per ulteriori informazioni, consulta Run a Build (AWS CLI) nella Guida per l'AWS CodeBuild utente.

- Per i dettagli sull'API, consulta StartBuild AWS CLICommand Reference.

stop-build-batch

Il seguente esempio di codice mostra come utilizzare`stop-build-batch`.

AWS CLI

Per interrompere un batch in corso, compila. AWS CodeBuild

L'esempio `stop-build-batch` seguente arresta la compilazione in batch specificata.

```
aws codebuild stop-build-batch \
    --id <project-name>:<batch-ID>
```

Output:

```
{
    "buildBatch": {
        "id": "<project-name>:<batch-ID>",
        "arn": "arn:aws:codebuild:<region-ID>:<account-ID>:build-batch/<project-
name>:<batch-ID>",
        "startTime": "2020-10-21T16:54:24.740000+00:00",
        "endTime": "2020-10-21T16:56:05.152000+00:00",
        "currentPhase": "STOPPED",
        "buildBatchStatus": "STOPPED",
        "resolvedSourceVersion": "aef7744ed069c51098e15c360f4102cd2cd1ad64",
        "projectName": "<project-name>",
        "phases": [
            {
                "phaseType": "SUBMITTED",
                "phaseStatus": "SUCCEEDED",
                "startTime": "2020-10-21T16:54:24.740000+00:00",
                "endTime": "2020-10-21T16:54:25.039000+00:00",
                "durationInSeconds": 0
            },
            {
                "phaseType": "DOWNLOAD_BATCHSPEC",
                "phaseStatus": "SUCCEEDED",
                "startTime": "2020-10-21T16:54:25.039000+00:00",
                "endTime": "2020-10-21T16:54:56.583000+00:00",
                "durationInSeconds": 31
            },
            {
                "phaseType": "IN_PROGRESS",
                "phaseStatus": "STOPPED",
```

```
                "startTime": "2020-10-21T16:54:56.583000+00:00",
                "endTime": "2020-10-21T16:56:05.152000+00:00",
                "durationInSeconds": 68
            },
            {
                "phaseType": "STOPPED",
                "startTime": "2020-10-21T16:56:05.152000+00:00"
            }
        ],
        "source": {
            "type": "GITHUB",
            "location": "<GitHub-repo-URL>",
            "gitCloneDepth": 1,
            "gitSubmodulesConfig": {
                "fetchSubmodules": false
            },
            "reportBuildStatus": false,
            "insecureSsl": false
        },
        "secondarySources": [],
        "secondarySourceVersions": [],
        "artifacts": {
            "location": ""
        },
        "secondaryArtifacts": [],
        "cache": {
            "type": "NO_CACHE"
        },
        "environment": {
            "type": "LINUX_CONTAINER",
            "image": "aws/codebuild/amazonlinux2-x86_64-standard:3.0",
            "computeType": "BUILD_GENERAL1_SMALL",
            "environmentVariables": [],
            "privilegedMode": false,
            "imagePullCredentialsType": "CODEBUILD"
        },
        "logConfig": {
            "cloudWatchLogs": {
                "status": "ENABLED"
            },
            "s3Logs": {
                "status": "DISABLED",
                "encryptionDisabled": false
            }
```

```
        },
        "buildTimeoutInMinutes": 60,
        "queuedTimeoutInMinutes": 480,
        "complete": true,
        "initiator": "Strohm",
        "encryptionKey": "arn:aws:kms:<region-ID>:<account-ID>:alias/aws/s3",
        "buildBatchNumber": 3,
        "buildBatchConfig": {
            "serviceRole": "arn:aws:iam::<account-ID>:role/service-role/<project-
name>",
            "restrictions": {
                "maximumBuildsAllowed": 100
            },
            "timeoutInMins": 480
        },
        "buildGroups": [
            {
                "identifier": "DOWNLOAD_SOURCE",
                "ignoreFailure": false,
                "currentBuildSummary": {
                    "arn": "arn:aws:codebuild:<region-ID>:<account-ID>:build/
<project-name>:<build-ID>",
                    "requestedOn": "2020-10-21T16:54:25.468000+00:00",
                    "buildStatus": "SUCCEEDED",
                    "primaryArtifact": {
                        "type": "no_artifacts",
                        "identifier": "DOWNLOAD_SOURCE"
                    },
                    "secondaryArtifacts": []
                }
            },
            {
                "identifier": "linux_small",
                "dependsOn": [],
                "ignoreFailure": false,
                "currentBuildSummary": {
                    "arn": "arn:aws:codebuild:<region-ID>:<account-ID>:build/
<project-name>:<build-ID>",
                    "requestedOn": "2020-10-21T16:54:56.833000+00:00",
                    "buildStatus": "IN_PROGRESS"
                }
            },
            {
                "identifier": "linux_medium",
```

```
                "dependsOn": [
                    "linux_small"
                ],
                "ignoreFailure": false,
                "currentBuildSummary": {
                    "arn": "arn:aws:codebuild:<region-ID>:<account-ID>:build/
<project-name>:<build-ID>",
                    "requestedOn": "2020-10-21T16:54:56.211000+00:00",
                    "buildStatus": "PENDING"
                }
            },
            {
                "identifier": "linux_large",
                "dependsOn": [
                    "linux_medium"
                ],
                "ignoreFailure": false,
                "currentBuildSummary": {
                    "arn": "arn:aws:codebuild:<region-ID>:<account-ID>:build/
<project-name>:<build-ID>",
                    "requestedOn": "2020-10-21T16:54:56.330000+00:00",
                    "buildStatus": "PENDING"
                }
            }
        ]
    }
}
```

Per ulteriori informazioni, consulta Batch builds AWS CodeBuild in the AWS CodeBuild User Guide.

- Per i dettagli sull'API, consulta StopBuildBatch AWS CLICommand Reference.

`stop-build`

Il seguente esempio di codice mostra come utilizzare`stop-build`.

AWS CLI

Per interrompere la compilazione di un progetto di AWS CodeBuild compilazione.

L'`stop-build`esempio seguente interrompe la CodeBuild build specificata.

```
aws codebuild stop-build --id my-demo-project:12345678-a1b2-c3d4-e5f6-11111EXAMPLE
```

Output:

```
{
    "build": {
        "startTime": 1556906956.318,
        "initiator": "my-aws-account-name",
        "projectName": "my-demo-project",
        "currentPhase": "COMPLETED",
        "cache": {
            "type": "NO_CACHE"
        },
        "source": {
            "insecureSsl": false,
            "location": "codebuild-us-west-2-123456789012-input-bucket/my-
source.zip",
            "type": "S3"
        },
        "id": "my-demo-project:1a2b3c4d-5678-90ab-cdef-11111EXAMPLE",
        "endTime": 1556906974.781,
        "phases": [
            {
                "durationInSeconds": 0,
                "phaseType": "SUBMITTED",
                "endTime": 1556906956.935,
                "phaseStatus": "SUCCEEDED",
                "startTime": 1556906956.318
            },
            {
                "durationInSeconds": 1,
                "phaseType": "QUEUED",
                "endTime": 1556906958.272,
                "phaseStatus": "SUCCEEDED",
                "startTime": 1556906956.935
            },
            {
                "phaseType": "PROVISIONING",
                "phaseStatus": "SUCCEEDED",
                "durationInSeconds": 14,
                "contexts": [
                    {
                        "message": "",
```

```
                    "statusCode": ""
                }
            ],
            "endTime": 1556906972.847,
            "startTime": 1556906958.272
        },
        {
            "phaseType": "DOWNLOAD_SOURCE",
            "phaseStatus": "SUCCEEDED",
            "durationInSeconds": 0,
            "contexts": [
                    {
                    "message": "",
                    "statusCode": ""
                }
            ],
            "endTime": 1556906973.552,
            "startTime": 1556906972.847
        },
        {
            "phaseType": "INSTALL",
            "phaseStatus": "SUCCEEDED",
            "durationInSeconds": 0,
            "contexts": [
                {
                    "message": "",
                    "statusCode": ""
                }
            ],
            "endTime": 1556906973.75,
            "startTime": 1556906973.552
        },
        {
            "phaseType": "PRE_BUILD",
            "phaseStatus": "SUCCEEDED",
            "durationInSeconds": 0,
            "contexts": [
                {
                    "message": "",
                    "statusCode": ""
                }
            ],
            "endTime": 1556906973.937,
            "startTime": 1556906973.75
```

```
            },
            {
                "durationInSeconds": 0,
                "phaseType": "BUILD",
                "endTime": 1556906974.781,
                "phaseStatus": "STOPPED",
                "startTime": 1556906973.937
            },
            {
                "phaseType": "COMPLETED",
                "startTime": 1556906974.781
            }
        ],
        "artifacts": {
            "location": "arn:aws:s3:::artifacts-override/my-demo-project",
            "encryptionDisabled": false,
            "overrideArtifactName": true
        },
        "buildComplete": true,
        "buildStatus": "STOPPED",
        "encryptionKey": "arn:aws:kms:us-west-2:123456789012:alias/aws/s3",
        "serviceRole": "arn:aws:iam::123456789012:role/service-role/my-codebuild-
service-role",
        "queuedTimeoutInMinutes": 5,
        "timeoutInMinutes": 60,
        "environment": {
            "type": "LINUX_CONTAINER",
            "environmentVariables": [],
            "computeType": "BUILD_GENERAL1_MEDIUM",
            "privilegedMode": false,
            "image": "aws/codebuild/standard:1.0",
            "imagePullCredentialsType": "CODEBUILD"
        },
        "logs": {
            "streamName": "1a2b3c4d-5678-90ab-cdef-11111EXAMPLE",
            "deepLink": "https://console.aws.amazon.com/cloudwatch/home?region=us-
west-2#logEvent:group=/aws/codebuild/my-demo-project;stream=1a2b3c4d-5678-90ab-
cdef-11111EXAMPLE",
            "groupName": "/aws/codebuild/my-demo-project"
        },
        "arn": "arn:aws:codebuild:us-west-2:123456789012:build/my-demo-
project:1a2b3c4d-5678-90ab-cdef-11111EXAMPLE"
    }
```

```
}
```

Per ulteriori informazioni, consulta Stop a Build (AWS CLI) nella Guida per l'AWS CodeBuild utente.

- Per i dettagli sull'API, consulta StopBuild AWS CLICommand Reference.

`update-project`

Il seguente esempio di codice mostra come utilizzare`update-project`.

AWS CLI

Per modificare le impostazioni di un progetto di AWS CodeBuild compilazione.

L'`update-project`esempio seguente modifica le impostazioni del progetto di CodeBuild compilazione specificato denominato my-demo-project.

```
aws codebuild update-project --name "my-demo-project" \
    --description "This project is updated" \
    --source "{\"type\": \"S3\",\"location\": \"codebuild-us-west-2-123456789012-
input-bucket/my-source-2.zip\"}" \
    --artifacts {"\"type\": \"S3\",\"location\": \"codebuild-us-west-2-123456789012-
output-bucket-2\""} \
    --environment "{\"type\": \"LINUX_CONTAINER\",\"image\": \"aws/codebuild/
standard:1.0\",\"computeType\": \"BUILD_GENERAL1_MEDIUM\"}" \
    --service-role "arn:aws:iam::123456789012:role/service-role/my-codebuild-
service-role"
```

L'output mostra le impostazioni aggiornate.

```
{
    "project": {
        "arn": "arn:aws:codebuild:us-west-2:123456789012:project/my-demo-project",
        "environment": {
            "privilegedMode": false,
            "environmentVariables": [],
            "type": "LINUX_CONTAINER",
            "image": "aws/codebuild/standard:1.0",
            "computeType": "BUILD_GENERAL1_MEDIUM",
            "imagePullCredentialsType": "CODEBUILD"
```

```
        },
        "queuedTimeoutInMinutes": 480,
        "description": "This project is updated",
        "artifacts": {
            "packaging": "NONE",
            "name": "my-demo-project",
            "type": "S3",
            "namespaceType": "NONE",
            "encryptionDisabled": false,
            "location": "codebuild-us-west-2-123456789012-output-bucket-2"
        },
        "encryptionKey": "arn:aws:kms:us-west-2:123456789012:alias/aws/s3",
        "badge": {
            "badgeEnabled": false
        },
        "serviceRole": "arn:aws:iam::123456789012:role/service-role/my-codebuild-
service-role",
        "lastModified": 1556840545.967,
        "tags": [],
        "timeoutInMinutes": 60,
        "created": 1556839783.274,
        "name": "my-demo-project",
        "cache": {
            "type": "NO_CACHE"
        },
        "source": {
            "type": "S3",
            "insecureSsl": false,
            "location": "codebuild-us-west-2-123456789012-input-bucket/my-
source-2.zip"
        }
    }
}
```

Per ulteriori informazioni, consulta Modifica delle impostazioni di un progetto di compilazione (AWS CLI) nella Guida per l'AWS CodeBuild utente

- Per i dettagli sull'API, consulta UpdateProject AWS CLICommand Reference.

`update-report-group`

Il seguente esempio di codice mostra come utilizzare`update-report-group`.

AWS CLI

Per aggiornare un gruppo di report in AWS CodeBuild.

L'esempio `update-report-group` seguente cambia il tipo di esportazione del gruppo di report impostandolo su "NO_EXPORT".

```
aws codebuild update-report-group \
    --arn arn:aws:codebuild:<region-ID>:<user-ID>:report-group/cli-created-report-group \
    --export-config="exportConfigType=NO_EXPORT"
```

Output:

```
{
    "reportGroup": {
        "arn": "arn:aws:codebuild:<region-ID>:<user-ID>:report-group/cli-created-report-group",
        "name": "cli-created-report-group",
        "type": "TEST",
        "exportConfig": {
            "exportConfigType": "NO_EXPORT"
        },
        "created": 1602020686.009,
        "lastModified": 1602021033.454,
        "tags": []
    }
}
```

Per ulteriori informazioni, consulta Lavorare con i gruppi di report nella Guida AWS CodeBuild per l'utente.

- Per i dettagli sull'API, consulta UpdateReportGroup AWS CLICommand Reference.

`update-webhook`

Il seguente esempio di codice mostra come utilizzare`update-webhook`.

AWS CLI

Per aggiornare il webhook di un progetto AWS CodeBuild

L'`update-webhook`esempio seguente aggiorna un webhook per il CodeBuild progetto specificato con due gruppi di filtri. Il `--rotate-secret` parametro specifica che la chiave segreta del progetto deve GitHub ruotare ogni volta che una modifica al codice attiva una compilazione. Il primo gruppo di filtri specifica le richieste pull create, aggiornate o riaperte su ramificazioni con nomi di riferimento Git che corrispondono all'espressione regolare `^refs/heads/master$` e riferimenti head che corrispondono a `^refs/heads/myBranch$`. Il secondo gruppo di filtri specifica le richieste push su ramificazioni con nomi di riferimento Git che non corrispondono all'espressione regolare `^refs/heads/myBranch$`.

```
aws codebuild update-webhook \
    --project-name Project2 \
    --rotate-secret \
    --filter-groups "[[{\"type\":\"EVENT\",\"pattern\":\"PULL_REQUEST_CREATED,
 PULL_REQUEST_UPDATED, PULL_REQUEST_REOPENED\"},{\"type\":\"HEAD_REF\",\"pattern
\":\"^refs/heads/myBranch$\",\"excludeMatchedPattern\":true},{\"type\":\"BASE_REF
\",\"pattern\":\"^refs/heads/master$\",\"excludeMatchedPattern\":true}],[{\"type\":
\"EVENT\",\"pattern\":\"PUSH\"},{\"type\":\"HEAD_REF\",\"pattern\":\"^refs/heads/
myBranch$\",\"excludeMatchedPattern\":true}]]"
```

Output:

```
{
    "webhook": {
        "filterGroups": [
            [
                {
                    "pattern": "PULL_REQUEST_CREATED, PULL_REQUEST_UPDATED,
 PULL_REQUEST_REOPENED",
                    "type": "EVENT"
                },
                {
                    "excludeMatchedPattern": true,
                    "pattern": "refs/heads/myBranch$",
                    "type": "HEAD_REF"
                },
                {
                    "excludeMatchedPattern": true,
                    "pattern": "refs/heads/master$",
                    "type": "BASE_REF"
                }
            ],
            [
```

```
                {
                    "pattern": "PUSH",
                    "type": "EVENT"
                },
                {
                    "excludeMatchedPattern": true,
                    "pattern": "refs/heads/myBranch$",
                    "type": "HEAD_REF"
                }
            ]
        ],
        "lastModifiedSecret": 1556312220.133
    }
}
```

Per ulteriori informazioni, consulta Modifica delle impostazioni di un progetto di compilazione (AWS CLI) nella Guida per l'AWS CodeBuild utente

- Per i dettagli sull'API, consulta UpdateWebhook AWS CLICommand Reference.

CodeCommit esempi utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with CodeCommit.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`associate-approval-rule-template-with-repository`

Il seguente esempio di codice mostra come utilizzare`associate-approval-rule-template-with-repository`.

AWS CLI

Come associare un modello di regola di approvazione a un repository

L'esempio `associate-approval-rule-template-with-repository` seguente associa il modello di regola di approvazione specificato a un repository denominato `MyDemoRepo`.

```
aws codecommit associate-approval-rule-template-with-repository \
    --repository-name MyDemoRepo  \
    --approval-rule-template-name 2-approver-rule-for-main
```

Questo comando non produce alcun output.

Per ulteriori informazioni, vedere Associare un modello di regola di approvazione a un repository nella Guida per l'AWS CodeCommit utente.

- Per i dettagli sull'API, consulta AssociateApprovalRuleTemplateWithRepository AWS CLICommand Reference.

`batch-associate-approval-rule-template-with-repositories`

Il seguente esempio di codice mostra come utilizzare`batch-associate-approval-rule-template-with-repositories`.

AWS CLI

Come associare un modello di regola di approvazione a più repository in un'unica operazione

L'esempio `batch-associate-approval-rule-template-with-repositories` seguente associa il modello di regola di approvazione specificato ai repository denominati `MyDemoRepo` e `MyOtherDemoRepo`.

Nota: i modelli di regole di approvazione sono specifici della AWS regione in cui vengono creati. Possono essere associati solo ai repository di quella AWS regione.

```
aws codecommit batch-associate-approval-rule-template-with-repositories \
    --repository-names MyDemoRepo, MyOtherDemoRepo  \
    --approval-rule-template-name 2-approver-rule-for-main
```

Output:

```
{
```

```
    "associatedRepositoryNames": [
        "MyDemoRepo",
        "MyOtherDemoRepo"
    ],
    "errors": []
}
```

Per ulteriori informazioni, consulta Associare un modello di regola di approvazione a un repository nella Guida per l'AWS CodeCommit utente.

- Per i dettagli sull'API, consulta BatchAssociateApprovalRuleTemplateWithRepositories AWS CLICommand Reference.

`batch-describe-merge-conflicts`

Il seguente esempio di codice mostra come utilizzare`batch-describe-merge-conflicts`.

AWS CLI

Come ottenere informazioni sui conflitti di unione in tutti i file oppure in un sottoinsieme di file in un'unione tra due specificatori di commit

L'esempio `batch-describe-merge-conflicts` seguente determina i conflitti di unione per l'unione di una ramificazione di origine denominata `feature-randomizationfeature` con una ramificazione di destinazione denominata `main` utilizzando la strategia `THREE_WAY_MERGE` in un repository denominato `MyDemoRepo`.

```
aws codecommit batch-describe-merge-conflicts \
    --source-commit-specifier feature-randomizationfeature \
    --destination-commit-specifier main \
    --merge-option THREE_WAY_MERGE \
    --repository-name MyDemoRepo
```

Output:

```
{
    "conflicts": [
        {
            "conflictMetadata": {
                "filePath": "readme.md",
                "fileSizes": {
                    "source": 139,
```

```
                    "destination": 230,
                    "base": 85
                },
                "fileModes": {
                    "source": "NORMAL",
                    "destination": "NORMAL",
                    "base": "NORMAL"
                },
                "objectTypes": {
                    "source": "FILE",
                    "destination": "FILE",
                    "base": "FILE"
                },
                "numberOfConflicts": 1,
                "isBinaryFile": {
                    "source": false,
                    "destination": false,
                    "base": false
                },
                "contentConflict": true,
                "fileModeConflict": false,
                "objectTypeConflict": false,
                "mergeOperations": {
                    "source": "M",
                    "destination": "M"
                }
            },
            "mergeHunks": [
                {
                    "isConflict": true,
                    "source": {
                        "startLine": 0,
                        "endLine": 3,
                        "hunkContent": "VGhpcyBpEXAMPLE=="
                    },
                    "destination": {
                        "startLine": 0,
                        "endLine": 1,
                        "hunkContent": "VXNlIHRoEXAMPLE="
                    }
                }
            ]
        }
    ],
```

```
    "errors": [],
    "destinationCommitId": "86958e0aEXAMPLE",
    "sourceCommitId": "6ccd57fdEXAMPLE",
    "baseCommitId": "767b6958EXAMPLE"
}
```

Per ulteriori informazioni, consulta Resolve Conflicts in a Pull Request nella Guida per AWS CodeCommit l'utente.

- Per i dettagli sull'API, consulta BatchDescribeMergeConflicts AWS CLICommand Reference.

`batch-disassociate-approval-rule-template-from-repositories`

Il seguente esempio di codice mostra come utilizzare`batch-disassociate-approval-rule-template-from-repositories`.

AWS CLI

Come annullare l'associazione tra un modello di regola di approvazione e più repository in un'unica operazione

L'esempio `batch-disassociate-approval-rule-template-from-repositories` seguente annulla l'associazione tra il modello di regola di approvazione specificato e i repository denominati `MyDemoRepo` e `MyOtherDemoRepo`.

```
aws codecommit batch-disassociate-approval-rule-template-from-repositories \
    --repository-names MyDemoRepo, MyOtherDemoRepo  \
    --approval-rule-template-name 1-approval-rule-for-all pull requests
```

Output:

```
{
    "disassociatedRepositoryNames": [
        "MyDemoRepo",
        "MyOtherDemoRepo"
    ],
    "errors": []
}
```

Per ulteriori informazioni, consulta Dissociare un modello di regola di approvazione nella Guida per l'AWS CodeCommit utente.

- Per i dettagli sull'API, consulta BatchDisassociateApprovalRuleTemplateFromRepositories AWS CLICommand Reference.

`batch-get-commits`

Il seguente esempio di codice mostra come utilizzare`batch-get-commits`.

AWS CLI

Come visualizzare informazioni su più commit

L'esempio `batch-get-commits` seguente visualizza i dettagli dei commit specificati.

```
aws codecommit batch-get-commits  \
    --repository-name MyDemoRepo  \
    --commit-ids 317f8570EXAMPLE 4c925148EXAMPLE
```

Output:

```
{
    "commits": [
      {
        "additionalData": "",
        "committer": {
            "date": "1508280564 -0800",
            "name": "Mary Major",
            "email": "mary_major@example.com"
        },
        "author": {
            "date": "1508280564 -0800",
            "name": "Mary Major",
            "email": "mary_major@example.com"
        },
        "commitId": "317f8570EXAMPLE",
        "treeId": "1f330709EXAMPLE",
        "parents": [
            "6e147360EXAMPLE"
        ],
        "message": "Change variable name and add new response element"
    },
    {
        "additionalData": "",
        "committer": {
```

```
            "date": "1508280542 -0800",
            "name": "Li Juan",
            "email": "li_juan@example.com"
        },
        "author": {
            "date": "1508280542 -0800",
            "name": "Li Juan",
            "email": "li_juan@example.com"
        },
        "commitId": "4c925148EXAMPLE",
        "treeId": "1f330709EXAMPLE",
        "parents": [
            "317f8570EXAMPLE"
        ],
        "message": "Added new class"
    }
}
```

Per ulteriori informazioni, consulta View Commit Details nella Guida AWS CodeCommit per l'utente.

- Per i dettagli sull'API, consulta BatchGetCommits AWS CLICommand Reference.

`batch-get-repositories`

Il seguente esempio di codice mostra come utilizzare`batch-get-repositories`.

AWS CLI

Come visualizzare i dettagli su più repository

Questo esempio mostra i dettagli su più AWS CodeCommit repository.

```
aws codecommit batch-get-repositories \
    --repository-names MyDemoRepo MyOtherDemoRepo
```

Output:

```
{
    "repositoriesNotFound": [],
    "repositories": [
         {
            "creationDate": 1429203623.625,
```

```
            "defaultBranch": "main",
            "repositoryName": "MyDemoRepo",
            "cloneUrlSsh": "ssh://git-codecommit.us-east-2.amazonaws.com/v1/repos/
MyDemoRepo",
            "lastModifiedDate": 1430783812.0869999,
            "repositoryDescription": "My demonstration repository",
            "cloneUrlHttp": "https://codecommit.us-east-2.amazonaws.com/v1/repos/
MyDemoRepo",
            "repositoryId": "f7579e13-b83e-4027-aaef-650c0EXAMPLE",
            "Arn": "arn:aws:codecommit:us-east-2:111111111111:MyDemoRepo"
            "accountId": "111111111111"
        },
        {
            "creationDate": 1429203623.627,
            "defaultBranch": "main",
            "repositoryName": "MyOtherDemoRepo",
            "cloneUrlSsh": "ssh://git-codecommit.us-east-2.amazonaws.com/v1/repos/
MyOtherDemoRepo",
            "lastModifiedDate": 1430783812.0889999,
            "repositoryDescription": "My other demonstration repository",
            "cloneUrlHttp": "https://codecommit.us-east-2.amazonaws.com/v1/repos/
MyOtherDemoRepo",
            "repositoryId": "cfc29ac4-b0cb-44dc-9990-f6f51EXAMPLE",
            "Arn": "arn:aws:codecommit:us-east-2:111111111111:MyOtherDemoRepo"
            "accountId": "111111111111"
        }
    ],
    "repositoriesNotFound": []
}
```

- Per i dettagli sull'API, consulta BatchGetRepositories AWS CLICommand Reference.

`create-approval-rule-template`

Il seguente esempio di codice mostra come utilizzare`create-approval-rule-template`.

AWS CLI

Come creare un modello di regola di approvazione

L'esempio `create-approval-rule-template` seguente crea un modello di regola di approvazione denominato `2-approver-rule-for-main ``. The template requires`

two users who assume the role of ``CodeCommitReview per approvare qualsiasi richiesta pull prima che possa essere unita alla ramificazione main.

```
aws codecommit create-approval-rule-template \
    --approval-rule-template-name 2-approver-rule-for-main \
    --approval-rule-template-description  "Requires two developers from the team to
 approve the pull request if the destination branch is main" \
    --approval-rule-template-content "{\"Version\": \"2018-11-08\",
\"DestinationReferences\": [\"refs/heads/main\"],\"Statements\": [{\"Type
\": \"Approvers\",\"NumberOfApprovalsNeeded\": 2,\"ApprovalPoolMembers\":
 [\"arn:aws:sts::123456789012:assumed-role/CodeCommitReview/*\"]}]}"
```

Output:

```
{
    "approvalRuleTemplate": {
        "approvalRuleTemplateName": "2-approver-rule-for-main",
        "creationDate": 1571356106.936,
        "approvalRuleTemplateId": "dd8b17fe-EXAMPLE",
        "approvalRuleTemplateContent": "{\"Version\": \"2018-11-08\",
\"DestinationReferences\": [\"refs/heads/main\"],\"Statements\": [{\"Type
\": \"Approvers\",\"NumberOfApprovalsNeeded\": 2,\"ApprovalPoolMembers\":
 [\"arn:aws:sts::123456789012:assumed-role/CodeCommitReview/*\"]}]}",
        "lastModifiedUser": "arn:aws:iam::123456789012:user/Mary_Major",
        "approvalRuleTemplateDescription": "Requires two developers from the team to
 approve the pull request if the destination branch is main",
        "lastModifiedDate": 1571356106.936,
        "ruleContentSha256": "4711b576EXAMPLE"
    }
}
```

Per ulteriori informazioni, vedere Creare un modello di regola di approvazione nella Guida AWS CodeCommit per l'utente.

- Per i dettagli sull'API, consulta CreateApprovalRuleTemplate AWS CLICommand Reference.

`create-branch`

Il seguente esempio di codice mostra come utilizzare`create-branch`.

AWS CLI

Come creare una ramificazione

Questo esempio crea un ramo in un AWS CodeCommit repository. Questo comando restituisce un output solo se ci sono errori.

Comando:

```
aws codecommit create-branch --repository-name MyDemoRepo --branch-name MyNewBranch
 --commit-id 317f8570EXAMPLE
```

Output:

```
None.
```

- Per i dettagli sull'API, consulta CreateBranch AWS CLICommand Reference.

`create-commit`

Il seguente esempio di codice mostra come utilizzare`create-commit`.

AWS CLI

Come creare un commit

L'esempio `create-commit` seguente crea un commit iniziale per un repository che aggiunge un file `readme.md` a un repository denominato `MyDemoRepo` nella ramificazione `main`.

```
aws codecommit create-commit \
    --repository-name MyDemoRepo \
    --branch-name main \
    --put-files "filePath=readme.md,fileContent='Welcome to our team repository.'"
```

Output:

```
{
    "filesAdded": [
        {
            "blobId": "5e1c309d-EXAMPLE",
            "absolutePath": "readme.md",
            "fileMode": "NORMAL"
```

```
        }
    ],
    "commitId": "4df8b524-EXAMPLE",
    "treeId": "55b57003-EXAMPLE",
    "filesDeleted": [],
    "filesUpdated": []
}
```

Per ulteriori informazioni, consulta Create a Commit AWS CodeCommit nella Guida AWS CodeCommit per l'utente.

- Per i dettagli sull'API, consulta CreateCommit AWS CLICommand Reference.

`create-pull-request-approval-rule`

Il seguente esempio di codice mostra come utilizzare`create-pull-request-approval-rule`.

AWS CLI

Come creare una regola di approvazione per una richiesta pull

L'esempio `create-pull-request-approval-rule` seguente crea una regola di approvazione denominata `Require two approved approvers` per la richiesta pull specificata. La regola specifica che sono necessarie due approvazioni da un pool di approvazioni. Il pool include tutti gli utenti che accedono CodeCommit assumendo il ruolo di `CodeCommitReview` nell'`123456789012` AWS account. Include anche un utente IAM o un utente federato denominato `Nikhil_Jayashankar` dallo stesso AWS account.

```
aws codecommit create-pull-request-approval-rule  \
    --approval-rule-name "Require two approved approvers"  \
    --approval-rule-content "{\"Version\": \"2018-11-08\",\"Statements\":
 [{\"Type\": \"Approvers\",\"NumberOfApprovalsNeeded\": 2,\"ApprovalPoolMembers
\": [\"CodeCommitApprovers:123456789012:Nikhil_Jayashankar\",
 \"arn:aws:sts::123456789012:assumed-role/CodeCommitReview/*\"]}]}"
```

Output:

```
{
    "approvalRule": {
        "approvalRuleName": "Require two approved approvers",
        "lastModifiedDate": 1570752871.932,
        "ruleContentSha256": "7c44e6ebEXAMPLE",
```

```
        "creationDate": 1570752871.932,
        "approvalRuleId": "aac33506-EXAMPLE",
        "approvalRuleContent": "{\"Version\": \"2018-11-08\",\"Statements\":
 [{\"Type\": \"Approvers\",\"NumberOfApprovalsNeeded\": 2,\"ApprovalPoolMembers
\": [\"CodeCommitApprovers:123456789012:Nikhil_Jayashankar\",
 \"arn:aws:sts::123456789012:assumed-role/CodeCommitReview/*\"]}]}",
        "lastModifiedUser": "arn:aws:iam::123456789012:user/Mary_Major"
    }
}
```

Per ulteriori informazioni, consulta Creare una regola di approvazione nella Guida per l'AWS CodeCommit utente.

- Per i dettagli sull'API, consulta CreatePullRequestApprovalRule AWS CLICommand Reference.

`create-pull-request`

Il seguente esempio di codice mostra come utilizzare`create-pull-request`.

AWS CLI

Come creare una richiesta pull

L'`create-pull-request`esempio seguente crea una richiesta pull denominata «Pronunciation difficulty analyzer» con una descrizione di «Please review these changes by Tuesday» destinata al ramo di origine «jane-branch» e deve essere unita al ramo predefinito «main» in un repository denominato ". AWS CodeCommit MyDemoRepo

```
aws codecommit create-pull-request \
    --title "My Pull Request" \
    --description "Please review these changes by Tuesday" \
    --client-request-token 123Example \
    --targets repositoryName=MyDemoRepo,sourceReference=MyNewBranch
```

Output:

```
{
    "pullRequest": {
        "approvalRules": [
            {
                "approvalRuleContent": "{\"Version\": \"2018-11-08\",
\"DestinationReferences\": [\"refs/heads/main\"],\"Statements\": [{\"Type
```

```
\": \"Approvers\",\"NumberOfApprovalsNeeded\": 2,\"ApprovalPoolMembers\":
 [\"arn:aws:sts::123456789012:assumed-role/CodeCommitReview/*\"]}]}",
                "approvalRuleId": "dd8b17fe-EXAMPLE",
                "approvalRuleName": "2-approver-rule-for-main",
                "creationDate": 1571356106.936,
                "lastModifiedDate": 571356106.936,
                "lastModifiedUser": "arn:aws:iam::123456789012:user/Mary_Major",
                "originApprovalRuleTemplate": {
                    "approvalRuleTemplateId": "dd3d22fe-EXAMPLE",
                    "approvalRuleTemplateName": "2-approver-rule-for-main"
                },
                "ruleContentSha256": "4711b576EXAMPLE"
            }
        ],
        "authorArn": "arn:aws:iam::111111111111:user/Jane_Doe",
        "description": "Please review these changes by Tuesday",
        "title": "Pronunciation difficulty analyzer",
        "pullRequestTargets": [
            {
                "destinationCommit": "5d036259EXAMPLE",
                "destinationReference": "refs/heads/main",
                "repositoryName": "MyDemoRepo",
                "sourceCommit": "317f8570EXAMPLE",
                "sourceReference": "refs/heads/jane-branch",
                "mergeMetadata": {
                    "isMerged": false
                }
            }
        ],
        "lastActivityDate": 1508962823.285,
        "pullRequestId": "42",
        "clientRequestToken": "123Example",
        "pullRequestStatus": "OPEN",
        "creationDate": 1508962823.285
    }
}
```

- Per CreatePullRequesti AWS CLI dettagli sull'API, consulta Command Reference.

`create-repository`

Il seguente esempio di codice mostra come utilizzare`create-repository`.

AWS CLI

Come creare un repository

Questo esempio crea un repository e lo associa all'account dell' AWS utente.

Comando:

```
aws codecommit create-repository --repository-name MyDemoRepo --repository-
description "My demonstration repository"
```

Output:

```
{
    "repositoryMetadata": {
        "repositoryName": "MyDemoRepo",
                "cloneUrlSsh": "ssh://git-codecommit.us-east-1.amazonaws.com/v1/
repos/MyDemoRepo",
                "lastModifiedDate": 1444766838.027,
        "repositoryDescription": "My demonstration repository",
                "cloneUrlHttp": "https://git-codecommit.us-east-1.amazonaws.com/v1/
repos/MyDemoRepo",
        "repositoryId": "f7579e13-b83e-4027-aaef-650c0EXAMPLE",
                "Arn": "arn:aws:codecommit:us-
east-1:111111111111EXAMPLE:MyDemoRepo",
        "accountId": "111111111111"
    }
}
```

- Per i dettagli sull'API, consulta AWS CLI Command CreateRepositoryReference.

`create-unreferenced-merge-commit`

Il seguente esempio di codice mostra come utilizzare`create-unreferenced-merge-commit`.

AWS CLI

Come creare un commit non referenziato che rappresenti il risultato dell'unione di due specificatori di commit

L'esempio `create-unreferenced-merge-commit` seguente crea un commit che rappresenta i risultati di un'unione tra una ramificazione di origine denominata `bugfix-1234` con una di

destinazione denominata `main` utilizzando la strategia THREE_WAY_MERGE in un repository denominato `MyDemoRepo`.

```
aws codecommit create-unreferenced-merge-commit \
    --source-commit-specifier bugfix-1234 \
    --destination-commit-specifier main \
    --merge-option THREE_WAY_MERGE \
    --repository-name MyDemoRepo \
    --name "Maria Garcia" \
    --email "maria_garcia@example.com" \
    --commit-message "Testing the results of this merge."
```

Output:

```
{
    "commitId": "4f178133EXAMPLE",
    "treeId": "389765daEXAMPLE"
}
```

Per ulteriori informazioni, consulta Resolve Conflicts in a Pull Request nella Guida per AWS CodeCommit l'utente.

- Per i dettagli sull'API, consulta CreateUnreferencedMergeCommit AWS CLICommand Reference.

`credential-helper`

Il seguente esempio di codice mostra come utilizzare`credential-helper`.

AWS CLI

Per configurare l'helper per le credenziali incluso nella CLI con AWS AWS CodeCommit

L'`credential-helper`utilità non è progettata per essere chiamata direttamente dalla AWS CLI. ma per essere utilizzata come parametro con il comando `git config` per configurare il computer locale. Consente a Git di utilizzare HTTPS e una versione con firma crittografica delle credenziali utente IAM o del ruolo dell' EC2 istanza Amazon ogni volta che Git deve autenticarsi per interagire con AWS i repository. CodeCommit

```
git config --global credential.helper '!aws codecommit credential-helper $@'
git config --global credential.UseHttpPath true
```

Output:

```
[credential]
    helper = !aws codecommit credential-helper $@
    UseHttpPath = true
```

Per ulteriori informazioni, consulta Configurazione per l' AWS CodeCommit utilizzo di altri metodi nella Guida per l'utente.AWS CodeCommit Esamina attentamente il contenuto, quindi segui le procedure in uno dei seguenti argomenti: Per le connessioni HTTPS su Linux, macOS o Unix o Per le connessioni HTTPS su Windows nella Guida per l'AWS CodeCommit utente.

- Per i dettagli sull'API, consulta AWS CLI Command CredentialHelperReference.

`delete-approval-rule-template`

Il seguente esempio di codice mostra come utilizzare`delete-approval-rule-template`.

AWS CLI

Come eliminare un modello di regola di approvazione

L'esempio `delete-approval-rule-template` seguente elimina il modello di regola di approvazione specificato.

```
aws codecommit delete-approval-rule-template  \
    --approval-rule-template-name 1-approver-for-all-pull-requests
```

Output:

```
{
    "approvalRuleTemplateId": "41de97b7-EXAMPLE"
}
```

Per ulteriori informazioni, vedere Eliminare un modello di regola di approvazione nella Guida AWS CodeCommit per l'utente.

- Per i dettagli sull'API, consulta DeleteApprovalRuleTemplate AWS CLICommand Reference.

`delete-branch`

Il seguente esempio di codice mostra come utilizzare`delete-branch`.

AWS CLI

Come eliminare una ramificazione

Questo esempio mostra come eliminare un ramo in un AWS CodeCommit repository.

Comando:

```
aws codecommit delete-branch --repository-name MyDemoRepo --branch-name MyNewBranch
```

Output:

```
{
  "branch": {
      "commitId": "317f8570EXAMPLE",
      "branchName": "MyNewBranch"
  }
}
```

- Per i dettagli sull'API, consulta DeleteBranch AWS CLICommand Reference.

`delete-comment-content`

Il seguente esempio di codice mostra come utilizzare`delete-comment-content`.

AWS CLI

Come eliminare i contenuti di un commento

È possibile eliminare i contenuti di un commento solo se è stato creato personalmente. Questo esempio mostra come eliminare il contenuto di un commento con l'ID `ff30b348EXAMPLEb9aa670f` generato dal sistema.

```
aws codecommit delete-comment-content \
    --comment-id ff30b348EXAMPLEb9aa670f
```

Output:

```
{
    "comment": {
        "creationDate": 1508369768.142,
```

```
        "deleted": true,
        "lastModifiedDate": 1508369842.278,
        "clientRequestToken": "123Example",
        "commentId": "ff30b348EXAMPLEb9aa670f",
        "authorArn": "arn:aws:iam::111111111111:user/Li_Juan",
        "callerReactions": [],
        "reactionCounts":
        {
            "CLAP" : 1
        }
    }
}
```

- Per i dettagli sull'API, consulta DeleteCommentContent AWS CLICommand Reference.

`delete-file`

Il seguente esempio di codice mostra come utilizzare`delete-file`.

AWS CLI

Come eliminare un file

L'esempio `delete-file` seguente elimina un file denominato `README.md` da una ramificazione denominata `main` con ID di commit più recente `c5709475EXAMPLE` in un repository denominato `MyDemoRepo`.

```
aws codecommit delete-file \
    --repository-name MyDemoRepo \
    --branch-name main \
    --file-path README.md \
    --parent-commit-id c5709475EXAMPLE
```

Output:

```
{
    "blobId":"559b44fEXAMPLE",
    "commitId":"353cf655EXAMPLE",
    "filePath":"README.md",
    "treeId":"6bc824cEXAMPLE"
}
```

Per ulteriori informazioni, consulta Modifica o eliminazione di un file AWS CodeCommit nella guida di riferimento dell'AWS CodeCommit API.

- Per i dettagli sull'API, consulta DeleteFile AWS CLICommand Reference.

`delete-pull-request-approval-rule`

Il seguente esempio di codice mostra come utilizzare`delete-pull-request-approval-rule`.

AWS CLI

Come eliminare una regola di approvazione per una richiesta pull

L'esempio `delete-pull-request-approval-rule` seguente elimina la regola di approvazione denominata `My Approval Rule` per la richiesta pull specificata.

```
aws codecommit delete-pull-request-approval-rule  \
    --approval-rule-name "My Approval Rule"  \
    --pull-request-id 15
```

Output:

```
{
    "approvalRuleId": "077d8e8a8-EXAMPLE"
}
```

Per ulteriori informazioni, consulta Modificare o eliminare una regola di approvazione nella Guida AWS CodeCommit per l'utente.

- Per i dettagli sull'API, consulta DeletePullRequestApprovalRule AWS CLICommand Reference.

`delete-repository`

Il seguente esempio di codice mostra come utilizzare`delete-repository`.

AWS CLI

Come eliminare un repository

Questo esempio mostra come eliminare un AWS CodeCommit repository.

Comando:

```
aws codecommit delete-repository --repository-name MyDemoRepo
```

Output:

```
{
  "repositoryId": "f7579e13-b83e-4027-aaef-650c0EXAMPLE"
}
```

- Per i dettagli sull'API, consulta DeleteRepository AWS CLICommand Reference.

`describe-merge-conflicts`

Il seguente esempio di codice mostra come utilizzare`describe-merge-conflicts`.

AWS CLI

Come ottenere informazioni dettagliate sui conflitti di unione

L'esempio `describe-merge-conflicts` seguente determina i conflitti di unione per un file denominato `readme.md` nella ramificazione di origine e in quella di destinazione specificate utilizzando la strategia THREE_WAY_MERGE.

```
aws codecommit describe-merge-conflicts \
    --source-commit-specifier feature-randomizationfeature \
    --destination-commit-specifier main \
    --merge-option THREE_WAY_MERGE \
    --file-path readme.md \
    --repository-name MyDemoRepo
```

Output:

```
{
    "conflictMetadata": {
        "filePath": "readme.md",
        "fileSizes": {
            "source": 139,
            "destination": 230,
            "base": 85
        },
        "fileModes": {
            "source": "NORMAL",
```

```
            "destination": "NORMAL",
            "base": "NORMAL"
        },
        "objectTypes": {
            "source": "FILE",
            "destination": "FILE",
            "base": "FILE"
        },
        "numberOfConflicts": 1,
        "isBinaryFile": {
            "source": false,
            "destination": false,
            "base": false
        },
        "contentConflict": true,
        "fileModeConflict": false,
        "objectTypeConflict": false,
        "mergeOperations": {
            "source": "M",
            "destination": "M"
        }
    },
    "mergeHunks": [
        {
            "isConflict": true,
            "source": {
                "startLine": 0,
                "endLine": 3,
                "hunkContent": "VGhpcyBpEXAMPLE="
            },
            "destination": {
                "startLine": 0,
                "endLine": 1,
                "hunkContent": "VXNlIHRoEXAMPLE="
            }
        }
    ],
    "destinationCommitId": "86958e0aEXAMPLE",
    "sourceCommitId": "6ccd57fdEXAMPLE",
    "baseCommitId": "767b69580EXAMPLE"
}
```

Per ulteriori informazioni, consulta Resolve Conflicts in a Pull Request nella Guida per AWS CodeCommit l'utente.

- Per i dettagli sull'API, consulta DescribeMergeConflicts AWS CLICommand Reference.

`describe-pull-request-events`

Il seguente esempio di codice mostra come utilizzare`describe-pull-request-events`.

AWS CLI

Come visualizzare gli eventi in una richiesta pull

L'esempio `describe-pull-request-events` seguente recupera gli eventi per una richiesta pull con ID 8.

```
aws codecommit describe-pull-request-events --pull-request-id 8
```

Output:

```
{
    "pullRequestEvents": [
        {
            "pullRequestId": "8",
            "pullRequestEventType": "PULL_REQUEST_CREATED",
            "eventDate": 1510341779.53,
            "actor": "arn:aws:iam::111111111111:user/Zhang_Wei"
        },
        {
            "pullRequestStatusChangedEventMetadata": {
                "pullRequestStatus": "CLOSED"
            },
            "pullRequestId": "8",
            "pullRequestEventType": "PULL_REQUEST_STATUS_CHANGED",
            "eventDate": 1510341930.72,
            "actor": "arn:aws:iam::111111111111:user/Jane_Doe"
        }
    ]
}
```

- Per i dettagli sull'API, consulta DescribePullRequestEvents AWS CLICommand Reference.

`disassociate-approval-rule-template-from-repository`

Il seguente esempio di codice mostra come utilizzare`disassociate-approval-rule-template-from-repository`.

AWS CLI

Come annullare l'associazione tra un modello di regola di approvazione e un repository

L'esempio `disassociate-approval-rule-template-from-repository` seguente annulla l'associazione tra il modello di regola di approvazione specificato e un repository denominato `MyDemoRepo`.

```
aws codecommit disassociate-approval-rule-template-from-repository \
    --repository-name MyDemoRepo  \
    --approval-rule-template-name 1-approver-rule-for-all-pull-requests
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Dissociare un modello di regola di approvazione nella Guida per l'AWS CodeCommit utente.

- Per i dettagli sull'API, consulta DisassociateApprovalRuleTemplateFromRepository AWS CLICommand Reference.

`evaluate-pull-request-approval-rules`

Il seguente esempio di codice mostra come utilizzare`evaluate-pull-request-approval-rules`.

AWS CLI

Come valutare se una richiesta pull soddisfa tutte le relative regole di approvazione

L'esempio `evaluate-pull-request-approval-rules` seguente valuta lo stato delle regole di approvazione nella richiesta pull specificata. In questo esempio, una regola di approvazione non è stata soddisfatta per richiesta pull, l'output del comando restituisce un valore `false` per `approved`.

```
aws codecommit evaluate-pull-request-approval-rules \
    --pull-request-id 27  \
```

```
    --revision-id 9f29d167EXAMPLE
```

Output:

```
{
    "evaluation": {
        "approved": false,
        "approvalRulesNotSatisfied": [
            "Require two approved approvers"
        ],
        "overridden": false,
        "approvalRulesSatisfied": []
    }
}
```

Per ulteriori informazioni, consulta Merge a Pull Request nella Guida per l'AWS CodeCommit utente.

- Per i dettagli sull'API, consulta EvaluatePullRequestApprovalRules AWS CLICommand Reference.

`get-approval-rule-template`

Il seguente esempio di codice mostra come utilizzare`get-approval-rule-template`.

AWS CLI

Come ottenere il contenuto di un modello di regola di approvazione

L'esempio `get-approval-rule-template` seguente ottiene il contenuto di un modello di regola di approvazione denominato `1-approver-rule-for-all-pull-requests`.

```
aws codecommit get-approval-rule-template \
    --approval-rule-template-name 1-approver-rule-for-all-pull-requests
```

Output:

```
{
    "approvalRuleTemplate": {
        "approvalRuleTemplateContent": "{\"Version\": \"2018-11-08\",\"Statements\":
 [{\"Type\": \"Approvers\",\"NumberOfApprovalsNeeded\": 1,\"ApprovalPoolMembers\":
 [\"arn:aws:sts::123456789012:assumed-role/CodeCommitReview/*\"]}]}",
```

```
        "ruleContentSha256": "621181bbEXAMPLE",
        "lastModifiedDate": 1571356106.936,
        "creationDate": 1571356106.936,
        "approvalRuleTemplateName": "1-approver-rule-for-all-pull-requests",
        "lastModifiedUser": "arn:aws:iam::123456789012:user/Li_Juan",
        "approvalRuleTemplateId": "a29abb15-EXAMPLE",
        "approvalRuleTemplateDescription": "All pull requests must be approved by
 one developer on the team."
    }
}
```

Per ulteriori informazioni, vedere Gestire i modelli di regole di approvazione nella Guida AWS CodeCommit per l'utente.

- Per i dettagli sull'API, consulta GetApprovalRuleTemplate AWS CLICommand Reference.

`get-blob`

Il seguente esempio di codice mostra come utilizzare`get-blob`.

AWS CLI

Come visualizzare le informazioni su un oggetto blob Git

L'`get-blob`esempio seguente recupera informazioni su un blob Git con l'ID di '2EB4AF3BExample' in un repository denominato ". AWS CodeCommit MyDemoRepo

```
aws codecommit get-blob  --repository-name MyDemoRepo  --blob-id 2eb4af3bEXAMPLE
```

Output:

```
{
    "content": "QSBCaW5hcnkgTGFyToEXAMPLE="
}
```

- Per i dettagli sull'API, consulta Command Reference. GetBlobAWS CLI

`get-branch`

Il seguente esempio di codice mostra come utilizzare`get-branch`.

AWS CLI

Come ottenere informazioni su una ramificazione

Questo esempio ottiene informazioni su un ramo in un AWS CodeCommit repository.

Comando:

```
aws codecommit get-branch --repository-name MyDemoRepo --branch-name MyNewBranch
```

Output:

```
{
  "BranchInfo": {
        "commitID": "317f8570EXAMPLE",
                "branchName": "MyNewBranch"
  }
}
```

- Per i dettagli sull'API, consulta GetBranch AWS CLICommand Reference.

`get-comment-reactions`

Il seguente esempio di codice mostra come utilizzare`get-comment-reactions`.

AWS CLI

Come visualizzare le reazioni emoji a un commento

L'esempio `get-comment-reactions` seguente elenca tutte le reazioni emoji a un commento con ID `abcd1234EXAMPLEb5678efgh`. Se il carattere della shell supporta la visualizzazione di Emoji versione 1.0, l'emoji viene visualizzata nell'output per `emoji`.

```
aws codecommit get-comment-reactions \
    --comment-id abcd1234EXAMPLEb5678efgh
```

Output:

```
{
    "reactionsForComment": {
        [
           {
```

```
                "reaction": {
                    "emoji:"??",
                    "shortCode": "thumbsup",
                    "unicode": "U+1F44D"
                },
                "users": [
                    "arn:aws:iam::123456789012:user/Li_Juan",
                    "arn:aws:iam::123456789012:user/Mary_Major",
                    "arn:aws:iam::123456789012:user/Jorge_Souza"
                ]
            },
            {
                "reaction": {
                    "emoji": "??",
                    "shortCode": "thumbsdown",
                    "unicode": "U+1F44E"
                },
                "users": [
                    "arn:aws:iam::123456789012:user/Nikhil_Jayashankar"
                ]
            },
            {
                "reaction": {
                    "emoji": "??",
                    "shortCode": "confused",
                    "unicode": "U+1F615"
                },
                "users": [
                    "arn:aws:iam::123456789012:user/Saanvi_Sarkar"
                ]
            }
        ]
    }
}
```

Per ulteriori informazioni, consulta Commentare un commit AWS CodeCommit nella Guida per l'AWS CodeCommit utente.

- Per i dettagli sull'API, consulta GetCommentReactions AWS CLICommand Reference.

`get-comment`

Il seguente esempio di codice mostra come utilizzare`get-comment`.

AWS CLI

Come visualizzare i dettagli di un commento

Questo esempio mostra come visualizzare i dettagli di un commento con ID `ff30b348EXAMPLEb9aa670f` generato dal sistema.

```
aws codecommit get-comment \
    --comment-id ff30b348EXAMPLEb9aa670f
```

Output:

```
{
    "comment": {
        "authorArn": "arn:aws:iam::111111111111:user/Li_Juan",
        "clientRequestToken": "123Example",
        "commentId": "ff30b348EXAMPLEb9aa670f",
        "content": "Whoops - I meant to add this comment to the line, but I don't
 see how to delete it.",
        "creationDate": 1508369768.142,
        "deleted": false,
        "commentId": "",
        "lastModifiedDate": 1508369842.278,
        "callerReactions": [],
        "reactionCounts":
        {
            "SMILE" : 6,
            "THUMBSUP" : 1
        }
    }
}
```

- Per i dettagli sull'API, consulta GetComment AWS CLICommand Reference.

`get-comments-for-compared-commit`

Il seguente esempio di codice mostra come utilizzare`get-comments-for-compared-commit`.

AWS CLI

Come visualizzare i commenti su un commit

Questo esempio mostra come visualizzare i commenti fatti sul confronto tra due commit in un repository denominato MyDemoRepo.

```
aws codecommit get-comments-for-compared-commit \
    --repository-name MyDemoRepo \
    --before-commit-ID 6e147360EXAMPLE \
    --after-commit-id 317f8570EXAMPLE
```

Output:

```
{
    "commentsForComparedCommitData": [
        {
            "afterBlobId": "1f330709EXAMPLE",
            "afterCommitId": "317f8570EXAMPLE",
            "beforeBlobId": "80906a4cEXAMPLE",
            "beforeCommitId": "6e147360EXAMPLE",
            "comments": [
                {
                    "authorArn": "arn:aws:iam::111111111111:user/Li_Juan",
                    "clientRequestToken": "123Example",
                    "commentId": "ff30b348EXAMPLEb9aa670f",
                    "content": "Whoops - I meant to add this comment to the line,
 not the file, but I don't see how to delete it.",
                    "creationDate": 1508369768.142,
                    "deleted": false,
                    "CommentId": "123abc-EXAMPLE",
                    "lastModifiedDate": 1508369842.278,
                    "callerReactions": [],
                    "reactionCounts":
                    {
                        "SMILE" : 6,
                        "THUMBSUP" : 1
                    }
                },
                {
                    "authorArn": "arn:aws:iam::111111111111:user/Li_Juan",
                    "clientRequestToken": "123Example",
                    "commentId": "553b509bEXAMPLE56198325",
                    "content": "Can you add a test case for this?",
                    "creationDate": 1508369612.240,
                    "deleted": false,
                    "commentId": "456def-EXAMPLE",
```

```
                    "lastModifiedDate": 1508369612.240,
                    "callerReactions": [],
                    "reactionCounts":
                    {
                        "THUMBSUP" : 2
                    }
                }
            ],
            "location": {
                "filePath": "cl_sample.js",
                "filePosition": 1232,
                "relativeFileVersion": "after"
            },
            "repositoryName": "MyDemoRepo"
        }
    ],
    "nextToken": "exampleToken"
}
```

- Per i dettagli sull'API, consulta GetCommentsForComparedCommit AWS CLICommand Reference.

`get-comments-for-pull-request`

Il seguente esempio di codice mostra come utilizzare`get-comments-for-pull-request`.

AWS CLI

Come visualizzare i commenti per una richiesta pull

Questo esempio come visualizzare i commenti per una richiesta pull in un repository denominato `MyDemoRepo`.

```
aws codecommit get-comments-for-pull-request \
    --repository-name MyDemoRepo \
    --before-commit-ID 317f8570EXAMPLE \
    --after-commit-id 5d036259EXAMPLE
```

Output:

```
{
    "commentsForPullRequestData": [
```

```
        {
            "afterBlobId": "1f330709EXAMPLE",
            "afterCommitId": "5d036259EXAMPLE",
            "beforeBlobId": "80906a4cEXAMPLE",
            "beforeCommitId": "317f8570EXAMPLE",
            "comments": [
                {
                    "authorArn": "arn:aws:iam::111111111111:user/Saanvi_Sarkar",
                    "clientRequestToken": "",
                    "commentId": "abcd1234EXAMPLEb5678efgh",
                    "content": "These don't appear to be used anywhere. Can we
remove them?",
                    "creationDate": 1508369622.123,
                    "deleted": false,
                    "lastModifiedDate": 1508369622.123,
                    "callerReactions": [],
                    "reactionCounts":
                    {
                        "THUMBSUP" : 6,
                        "CONFUSED" : 1
                    }
                },
                {
                    "authorArn": "arn:aws:iam::111111111111:user/Li_Juan",
                    "clientRequestToken": "",
                    "commentId": "442b498bEXAMPLE5756813",
                    "content": "Good catch. I'll remove them.",
                    "creationDate": 1508369829.104,
                    "deleted": false,
                    "lastModifiedDate": 150836912.273,
                    "callerReactions": ["THUMBSUP"]
                    "reactionCounts":
                    {
                        "THUMBSUP" : 14
                    }
                }
            ],
            "location": {
                "filePath": "ahs_count.py",
                "filePosition": 367,
                "relativeFileVersion": "AFTER"
            },
            "repositoryName": "MyDemoRepo",
            "pullRequestId": "42"
```

```
        }
    ],
    "nextToken": "exampleToken"
}
```

- Per i dettagli sull'API, consulta GetCommentsForPullRequest AWS CLICommand Reference.

`get-commit`

Il seguente esempio di codice mostra come utilizzare`get-commit`.

AWS CLI

Come visualizzare le informazioni su un commit in un repository

Questo esempio mostra i dettagli su un commit con l'ID generato dal sistema '7e9fd3091thisisanexamplethisanexample1' in un repository denominato ". AWS CodeCommit MyDemoRepo

Comando:

```
aws codecommit get-commit --repository-name MyDemoRepo --commit-
id 7e9fd3091thisisanexamplethisisanexample1
```

Output:

```
{
  "commit": {
      "additionalData": "",
      "committer": {
          "date": "1484167798 -0800",
          "name": "Mary Major",
          "email": "mary_major@example.com"
      },
      "author": {
          "date": "1484167798 -0800",
          "name": "Mary Major",
          "email": "mary_major@example.com"
      },
      "treeId": "347a3408thisisanexampletreeidexample",
      "parents": [
```

```
            "7aa87a031thisisanexamplethisisanexample1"
        ],
        "message": "Fix incorrect variable name"
    }
}
```

- Per i dettagli sull'GetCommitAPI, consulta Command Reference.AWS CLI

`get-differences`

Il seguente esempio di codice mostra come utilizzare`get-differences`.

AWS CLI

Come ottenere informazioni sulle differenze relative a uno specificatore di commit in un repository

Questo esempio mostra le informazioni sui metadati relative alle modifiche tra due specificatori di commit (branch, tag, HEAD o altri riferimenti completi, come commit IDs) in una cartella rinominata nel repository denominata. AWS CodeCommit MyDemoRepo L'esempio include diverse opzioni non obbligatorie, tra cui --before-commit-specifier, --before-path e --after-path, per illustrare in modo più completo come utilizzare queste opzioni per limitare i risultati. La risposta include le autorizzazioni in modalità file.

Comando:

```
aws codecommit get-differences --repository-name MyDemoRepo --before-
commit-specifier 955bba12thisisanexamplethisisanexample --after-commit-
specifier 14a95463thisisanexamplethisisanexample --before-path tmp/example-folder --
after-path tmp/renamed-folder
```

Output:

```
{
  "differences": [
      {
          "afterBlob": {
              "path": "blob.txt",
              "blobId": "2eb4af3b1thisisanexamplethisisanexample1",
              "mode": "100644"
          },
          "changeType": "M",
```

```
            "beforeBlob": {
                "path": "blob.txt",
                "blobId": "bf7fcf281thisisanexamplethisisanexample1",
                "mode": "100644"
            }
        }
    ]
}
```

- Per i dettagli sull'API, consulta Command Reference. GetDifferencesAWS CLI

`get-file`

Il seguente esempio di codice mostra come utilizzare`get-file`.

AWS CLI

Per ottenere il contenuto codificato in base 64 di un file in un repository AWS CodeCommit

L'esempio `get-file` seguente ottiene il contenuto con codifica Base64 di un file denominato `README.md` da una ramificazione denominata `main` in un repository denominato `MyDemoRepo`.

```
aws codecommit get-file \
    --repository-name MyDemoRepo \
    --commit-specifier main \
    --file-path README.md
```

Output:

```
{
    "blobId":"559b44fEXAMPLE",
    "commitId":"c5709475EXAMPLE",
    "fileContent":"IyBQaHVzEXAMPLE",
    "filePath":"README.md",
    "fileMode":"NORMAL",
    "fileSize":1563
}
```

Per ulteriori informazioni, consulta la guida di riferimento GetFiledelle AWS CodeCommit API.

- Per i dettagli sull'API, consulta GetFile AWS CLICommand Reference.

`get-folder`

Il seguente esempio di codice mostra come utilizzare`get-folder`.

AWS CLI

Per ottenere il contenuto di una cartella in un AWS CodeCommit repository

L'esempio `get-folder` seguente ottiene il contenuto di una cartella di primo livello da un repository denominato `MyDemoRepo`.

```
aws codecommit get-folder --repository-name MyDemoRepo --folder-path ""
```

Output:

```
{
    "commitId":"c5709475EXAMPLE",
    "files":[
        {
            "absolutePath":".gitignore",
            "blobId":"74094e8bEXAMPLE",
            "fileMode":"NORMAL",
            "relativePath":".gitignore"
        },
        {
            "absolutePath":"Gemfile",
            "blobId":"9ceb72f6EXAMPLE",
            "fileMode":"NORMAL",
            "relativePath":"Gemfile"
        },
        {
            "absolutePath":"Gemfile.lock",
            "blobId":"795c4a2aEXAMPLE",
            "fileMode":"NORMAL",
            "relativePath":"Gemfile.lock"
        },
        {
            "absolutePath":"LICENSE.txt",
            "blobId":"0c7932c8EXAMPLE",
            "fileMode":"NORMAL",
            "relativePath":"LICENSE.txt"
        },
        {
```

```
            "absolutePath":"README.md",
            "blobId":"559b44feEXAMPLE",
            "fileMode":"NORMAL",
            "relativePath":"README.md"
        }
    ],
    "folderPath":"",
    "subFolders":[
        {
            "absolutePath":"public",
            "relativePath":"public",
            "treeId":"d5e92ae3aEXAMPLE"
        },
        {
            "absolutePath":"tmp",
            "relativePath":"tmp",
            "treeId":"d564d0bcEXAMPLE"
        }
    ],
    "subModules":[],
    "symbolicLinks":[],
    "treeId":"7b3c4dadEXAMPLE"
}
```

Per ulteriori informazioni, consulta GetFolder la guida di riferimento delle AWS CodeCommit API.

- Per i dettagli sull'API, consulta GetFolder AWS CLICommand Reference.

`get-merge-commit`

Il seguente esempio di codice mostra come utilizzare`get-merge-commit`.

AWS CLI

Come ottenere informazioni dettagliate su un commit di unione

L'esempio `get-merge-commit` seguente mostra i dettagli sul commit di un'unione per la ramificazione di origine denominata `bugfix-bug1234` con una di destinazione denominata `main` in un repository denominato `MyDemoRepo`.

```
aws codecommit get-merge-commit \
    --source-commit-specifier bugfix-bug1234 \
```

```
    --destination-commit-specifier main \
    --repository-name MyDemoRepo
```

Output:

```
{
    "sourceCommitId": "c5709475EXAMPLE",
    "destinationCommitId": "317f8570EXAMPLE",
    "baseCommitId": "fb12a539EXAMPLE",
    "mergeCommitId": "ffc4d608eEXAMPLE"
}
```

Per ulteriori informazioni, consulta View Commit Details nella Guida AWS CodeCommit per l'utente.

- Per i dettagli sull'API, consulta GetMergeCommit AWS CLICommand Reference.

`get-merge-conflicts`

Il seguente esempio di codice mostra come utilizzare`get-merge-conflicts`.

AWS CLI

Come visualizzare eventuali conflitti di unione per una richiesta pull

L'esempio `get-merge-conflicts` seguente mostra se sono presenti conflitti di unione tra una ramificazione di origine denominata `feature-randomizationfeature` e una di destinazione denominata main in un repository denominato `MyDemoRepo`.

```
aws codecommit get-merge-conflicts \
    --repository-name MyDemoRepo \
    --source-commit-specifier feature-randomizationfeature \
    --destination-commit-specifier main \
    --merge-option THREE_WAY_MERGE
```

Output:

```
{
    "mergeable": false,
    "destinationCommitId": "86958e0aEXAMPLE",
```

```
    "sourceCommitId": "6ccd57fdEXAMPLE",
    "baseCommitId": "767b6958EXAMPLE",
    "conflictMetadataList": [
        {
            "filePath": "readme.md",
            "fileSizes": {
                "source": 139,
                "destination": 230,
                "base": 85
            },
            "fileModes": {
                "source": "NORMAL",
                "destination": "NORMAL",
                "base": "NORMAL"
            },
            "objectTypes": {
                "source": "FILE",
                "destination": "FILE",
                "base": "FILE"
            },
            "numberOfConflicts": 1,
            "isBinaryFile": {
                "source": false,
                "destination": false,
                "base": false
            },
            "contentConflict": true,
            "fileModeConflict": false,
            "objectTypeConflict": false,
            "mergeOperations": {
                "source": "M",
                "destination": "M"
            }
        }
    ]
}
```

- Per i dettagli sull'API, consulta GetMergeConflicts AWS CLICommand Reference.

`get-merge-options`

Il seguente esempio di codice mostra come utilizzare`get-merge-options`.

AWS CLI

Come ottenere informazioni sulle opzioni di unione disponibili per unire due ramificazioni specifiche

L'esempio `get-merge-options` seguente determina le opzioni di unione disponibili per unire una ramificazione di origine denominata `bugfix-bug1234` con una ramificazione di destinazione denominata `main` in un repository denominato `MyDemoRepo`.

```
aws codecommit get-merge-options \
    --source-commit-specifier bugfix-bug1234 \
    --destination-commit-specifier main \
    --repository-name MyDemoRepo
```

Output:

```
{
    "mergeOptions": [
        "FAST_FORWARD_MERGE",
        "SQUASH_MERGE",
        "THREE_WAY_MERGE"
    ],
    "sourceCommitId": "18059494EXAMPLE",
    "destinationCommitId": "ffd3311dEXAMPLE",
    "baseCommitId": "ffd3311dEXAMPLE"
}
```

Per ulteriori informazioni, consulta Resolve Conflicts in a Pull Request nella Guida per AWS CodeCommit l'utente.

- Per i dettagli sull'API, consulta GetMergeOptions AWS CLICommand Reference.

`get-pull-request-approval-states`

Il seguente esempio di codice mostra come utilizzare`get-pull-request-approval-states`.

AWS CLI

Come visualizzare le approvazioni in una richiesta pull

L'esempio `get-pull-request-approval-states` seguente restituisce le approvazioni per la richiesta pull specificata.

```
aws codecommit get-pull-request-approval-states \
    --pull-request-id 8 \
    --revision-id 9f29d167EXAMPLE
```

Output:

```
{
    "approvals": [
        {
            "userArn": "arn:aws:iam::123456789012:user/Mary_Major",
            "approvalState": "APPROVE"
        }
    ]
}
```

Per ulteriori informazioni, consulta View Pull Requests nella Guida AWS CodeCommit per l'utente.

- Per i dettagli sull'API, consulta GetPullRequestApprovalStates AWS CLICommand Reference.

`get-pull-request-override-state`

Il seguente esempio di codice mostra come utilizzare`get-pull-request-override-state`.

AWS CLI

Come ottenere informazioni sullo stato di sostituzione di una richiesta pull

L'esempio `get-pull-request-override-state` seguente restituisce lo stato della sostituzione per la richiesta pull specificata. In questo esempio, le regole di approvazione per la richiesta pull sono state sostituite da un utente di nome Mary Major, pertanto l'output restituisce un valore `true`:

```
aws codecommit get-pull-request-override-state \
    --pull-request-id 34  \
    --revision-id 9f29d167EXAMPLE
```

Output:

```
{
    "overridden": true,
    "overrider": "arn:aws:iam::123456789012:user/Mary_Major"
```

```
}
```

Per ulteriori informazioni, consulta Ignorare le regole di approvazione su una pull request nella Guida per l'AWS CodeCommit utente.

- Per i dettagli sull'API, consulta GetPullRequestOverrideState AWS CLICommand Reference.

`get-pull-request`

Il seguente esempio di codice mostra come utilizzare`get-pull-request`.

AWS CLI

Come visualizzare i dettagli di una richiesta pull

Questo esempio mostra come visualizzare le informazioni su una richiesta pull con ID 27.

```
aws codecommit get-pull-request \
    --pull-request-id 27
```

Output:

```
{
    "pullRequest": {
        "approvalRules": [
            {
                "approvalRuleContent": "{\"Version\": \"2018-11-08\",\"Statements\":
 [{\"Type\": \"Approvers\",\"NumberOfApprovalsNeeded\": 2,\"ApprovalPoolMembers\":
 [\"arn:aws:sts::123456789012:assumed-role/CodeCommitReview/*\"]}]}",
                "approvalRuleId": "dd8b17fe-EXAMPLE",
                "approvalRuleName": "2-approver-rule-for-main",
                "creationDate": 1571356106.936,
                "lastModifiedDate": 571356106.936,
                "lastModifiedUser": "arn:aws:iam::123456789012:user/Mary_Major",
                "ruleContentSha256": "4711b576EXAMPLE"
            }
        ],
        "lastActivityDate": 1562619583.565,
        "pullRequestTargets": [
            {
                "sourceCommit": "ca45e279EXAMPLE",
                "sourceReference": "refs/heads/bugfix-1234",
                "mergeBase": "a99f5ddbEXAMPLE",
```

```
                "destinationReference": "refs/heads/main",
                "mergeMetadata": {
                    "isMerged": false
                },
                "destinationCommit": "2abfc6beEXAMPLE",
                "repositoryName": "MyDemoRepo"
            }
        ],
        "revisionId": "e47def21EXAMPLE",
        "title": "Quick fix for bug 1234",
        "authorArn": "arn:aws:iam::123456789012:user/Nikhil_Jayashankar",
        "clientRequestToken": "d8d7612e-EXAMPLE",
        "creationDate": 1562619583.565,
        "pullRequestId": "27",
        "pullRequestStatus": "OPEN"
    }
}
```

- Per i dettagli sull'API, consulta GetPullRequest AWS CLICommand Reference.

`get-repository-triggers`

Il seguente esempio di codice mostra come utilizzare`get-repository-triggers`.

AWS CLI

Come ottenere informazioni sui trigger in un repository

Questo esempio mostra i dettagli sui trigger configurati per un AWS CodeCommit repository denominato. MyDemoRepo

```
aws codecommit get-repository-triggers \
    --repository-name MyDemoRepo
```

Output:

```
{
    "configurationId": "f7579e13-b83e-4027-aaef-650c0EXAMPLE",
    "triggers": [
        {
            "destinationArn": "arn:aws:sns:us-
east-1:111111111111:MyCodeCommitTopic",
```

```
            "branches": [
                "main",
                "preprod"
            ],
            "name": "MyFirstTrigger",
            "customData": "",
            "events": [
                "all"
            ]
        },
        {
            "destinationArn": "arn:aws:lambda:us-
east-1:111111111111:function:MyCodeCommitPythonFunction",
            "branches": [],
            "name": "MySecondTrigger",
            "customData": "EXAMPLE",
            "events": [
                "all"
            ]
        }
    ]
}
```

- Per i dettagli sull'API, vedere GetRepositoryTriggersin AWS CLI Command Reference.

`get-repository`

Il seguente esempio di codice mostra come utilizzare`get-repository`.

AWS CLI

Come ottenere informazioni su un repository

Questo esempio mostra i dettagli su un AWS CodeCommit repository.

```
aws codecommit get-repository \
    --repository-name MyDemoRepo
```

Output:

```
{
    "repositoryMetadata": {
```

```
        "creationDate": 1429203623.625,
        "defaultBranch": "main",
        "repositoryName": "MyDemoRepo",
        "cloneUrlSsh": "ssh://git-codecommit.us-east-1.amazonaws.com/v1/repos/v1/
repos/MyDemoRepo",
        "lastModifiedDate": 1430783812.0869999,
        "repositoryDescription": "My demonstration repository",
        "cloneUrlHttp": "https://codecommit.us-east-1.amazonaws.com/v1/repos/
MyDemoRepo",
        "repositoryId": "f7579e13-b83e-4027-aaef-650c0EXAMPLE",
        "Arn": "arn:aws:codecommit:us-east-1:80398EXAMPLE:MyDemoRepo
        "accountId": "111111111111"
    }
}
```

- Per i dettagli sull'API, consulta GetRepository AWS CLICommand Reference.

`list-approval-rule-templates`

Il seguente esempio di codice mostra come utilizzare`list-approval-rule-templates`.

AWS CLI

Per elencare tutti i modelli di regole di approvazione in una AWS regione

L'esempio `list-approval-rule-templates` seguente elenca tutti i modelli di regola di approvazione nella Regione specificata. Se non viene specificata alcuna AWS regione come parametro, il comando restituisce i modelli di regole di approvazione per l'area specificata nel profilo AWS CLI utilizzato per eseguire il comando.

```
aws codecommit list-approval-rule-templates \
    --region us-east-2
```

Output:

```
{
    "approvalRuleTemplateNames": [
        "2-approver-rule-for-main",
        "1-approver-rule-for-all-pull-requests"
    ]
}
```

Per ulteriori informazioni, vedere Gestire i modelli di regole di approvazione nella Guida per l'AWS CodeCommit utente.

- Per i dettagli sull'API, consulta ListApprovalRuleTemplates AWS CLICommand Reference.

`list-associated-approval-rule-templates-for-repository`

Il seguente esempio di codice mostra come utilizzare`list-associated-approval-rule-templates-for-repository`.

AWS CLI

Come elencare tutti i modelli associati a un repository

L'esempio `list-associated-approval-rule-templates-for-repository` seguente elenca tutti i modelli di regola di approvazione associati a un repository denominato `MyDemoRepo`.

```
aws codecommit list-associated-approval-rule-templates-for-repository \
    --repository-name MyDemoRepo
```

Output:

```
{
  "approvalRuleTemplateNames": [
    "2-approver-rule-for-main",
    "1-approver-rule-for-all-pull-requests"
  ]
}
```

Per ulteriori informazioni, vedere Gestire i modelli di regole di approvazione nella Guida AWS CodeCommit per l'utente.

- Per i dettagli sull'API, consulta ListAssociatedApprovalRuleTemplatesForRepository AWS CLICommand Reference.

`list-branches`

Il seguente esempio di codice mostra come utilizzare`list-branches`.

AWS CLI

Come visualizzare un elenco di nomi delle ramificazioni

Questo esempio elenca tutti i nomi delle filiali in un AWS CodeCommit repository.

```
aws codecommit list-branches \
    --repository-name MyDemoRepo
```

Output:

```
{
    "branches": [
        "MyNewBranch",
        "main"
    ]
}
```

- Per i dettagli sull'API, consulta ListBranches AWS CLICommand Reference.

`list-pull-requests`

Il seguente esempio di codice mostra come utilizzare`list-pull-requests`.

AWS CLI

Come visualizzare un elenco di richieste pull in un repository

Questo esempio dimostra come elencare le richieste pull create da un utente IAM con l'ARN 'arn:aws:iam: :1111:user/LI_JUAN' e lo stato di 'CLOSED' in un repository denominato ": AWS CodeCommit MyDemoRepo

```
aws codecommit list-pull-requests --author-arn arn:aws:iam::111111111111:user/
Li_Juan --pull-request-status CLOSED --repository-name MyDemoRepo
```

Output:

```
{
 "nextToken": "",
 "pullRequestIds": ["2","12","16","22","23","35","30","39","47"]
}
```

- Per ListPullRequestsi AWS CLI dettagli sull'API, consulta Command Reference.

`list-repositories-for-approval-rule-template`

Il seguente esempio di codice mostra come utilizzare`list-repositories-for-approval-rule-template`.

AWS CLI

Come elencare tutti i repository associati a un modello

L'esempio `list-repositories-for-approval-rule-template` seguente elenca tutti i repository associati al modello di regola di approvazione specificato.

```
aws codecommit list-repositories-for-approval-rule-template \
    --approval-rule-template-name 2-approver-rule-for-main
```

Output:

```
{
    "repositoryNames": [
        "MyDemoRepo",
        "MyClonedRepo"
    ]
}
```

Per ulteriori informazioni, vedere Gestire i modelli di regole di approvazione nella Guida AWS CodeCommit per l'utente.

- Per i dettagli sull'API, consulta ListRepositoriesForApprovalRuleTemplate AWS CLICommand Reference.

`list-repositories`

Il seguente esempio di codice mostra come utilizzare`list-repositories`.

AWS CLI

Come visualizzare un elenco di repository

Questo esempio elenca tutti i AWS CodeCommit repository associati all' AWS account dell'utente.

Comando:

```
aws codecommit list-repositories
```

Output:

```
{
  "repositories": [
      {
         "repositoryName": "MyDemoRepo"
         "repositoryId": "f7579e13-b83e-4027-aaef-650c0EXAMPLE",
      },
      {
         "repositoryName": "MyOtherDemoRepo"
         "repositoryId": "cfc29ac4-b0cb-44dc-9990-f6f51EXAMPLE"
      }
  ]
}
```

- Per i dettagli sull'API, consulta ListRepositories AWS CLICommand Reference.

`list-tags-for-resource`

Il seguente esempio di codice mostra come utilizzare`list-tags-for-resource`.

AWS CLI

Per visualizzare i AWS tag di un repository

L'esempio `list-tags-for-resource` seguente elenca le chiavi e i valori di tag per il repository specificato.

```
aws codecommit list-tags-for-resource \
    --resource-arn arn:aws:codecommit:us-west-2:111111111111:MyDemoRepo
```

Output:

```
{
    "tags": {
        "Status": "Secret",
        "Team": "Saanvi"
    }
```

```
}
```

Per ulteriori informazioni, consulta View Tags for a Repository nella Guida per l'AWS CodeCommit utente.

- Per i dettagli sull'API, consulta ListTagsForResource AWS CLICommand Reference.

`merge-branches-by-fast-forward`

Il seguente esempio di codice mostra come utilizzare`merge-branches-by-fast-forward`.

AWS CLI

Come unire due ramificazioni utilizzando la strategia di unione ad avanzamento rapido

L'esempio `merge-branches-by-fast-forward` seguente unisce la ramificazione di origine specificata con quella di destinazione specificata in un repository denominato `MyDemoRepo`.

```
aws codecommit merge-branches-by-fast-forward \
    --source-commit-specifier bugfix-bug1234 \
    --destination-commit-specifier bugfix-bug1233 \
    --repository-name MyDemoRepo
```

Output:

```
{
    "commitId": "4f178133EXAMPLE",
    "treeId": "389765daEXAMPLE"
}
```

Per ulteriori informazioni, consulta Compare and Merge Branches nella Guida per l'AWS CodeCommit utente.

- Per i dettagli sull'API, consulta MergeBranchesByFastForward AWS CLICommand Reference.

`merge-branches-by-squash`

Il seguente esempio di codice mostra come utilizzare`merge-branches-by-squash`.

AWS CLI

Come unire due ramificazioni utilizzando la strategia di unione squash

L'esempio `merge-branches-by-squash` seguente unisce la ramificazione di origine specificata con quella di destinazione specificata in un repository denominato `MyDemoRepo`.

```
aws codecommit merge-branches-by-squash \
    --source-commit-specifier bugfix-bug1234 \
    --destination-commit-specifier bugfix-bug1233 \
    --author-name "Maria Garcia" \
    --email "maria_garcia@example.com" \
    --commit-message "Merging two fix branches to prepare for a general patch." \
    --repository-name MyDemoRepo
```

Output:

```
{
    "commitId": "4f178133EXAMPLE",
    "treeId": "389765daEXAMPLE"
}
```

Per ulteriori informazioni, consulta Compare and Merge Branches nella Guida per l'AWS CodeCommit utente.

- Per i dettagli sull'API, consulta MergeBranchesBySquash AWS CLICommand Reference.

`merge-branches-by-three-way`

Il seguente esempio di codice mostra come utilizzare`merge-branches-by-three-way`.

AWS CLI

Come unire due ramificazioni utilizzando la strategia di unione a tre vie

L'esempio `merge-branches-by-three-way` seguente unisce la ramificazione di origine specificata con quella di destinazione specificata in un repository denominato `MyDemoRepo`.

```
aws codecommit merge-branches-by-three-way \
    --source-commit-specifier main \
    --destination-commit-specifier bugfix-bug1234 \
    --author-name "Jorge Souza" --email "jorge_souza@example.com" \
    --commit-message "Merging changes from main to bugfix branch before additional
 testing." \
    --repository-name MyDemoRepo
```

Output:

```
{
    "commitId": "4f178133EXAMPLE",
    "treeId": "389765daEXAMPLE"
}
```

Per ulteriori informazioni, consulta Compare and Merge Branches nella Guida per l'AWS CodeCommit utente.

- Per i dettagli sull'API, consulta MergeBranchesByThreeWay AWS CLICommand Reference.

`merge-pull-request-by-fast-forward`

Il seguente esempio di codice mostra come utilizzare`merge-pull-request-by-fast-forward`.

AWS CLI

Come unire e chiudere una richiesta pull

Questo esempio mostra come unire e chiudere una richiesta di pull con ID 47 e ID commit di origine 99132ab0EXAMPLE in un repository denominato `MyDemoRepo`.

```
aws codecommit merge-pull-request-by-fast-forward \
    --pull-request-id 47 \
    --source-commit-id 99132ab0EXAMPLE \
    --repository-name MyDemoRepo
```

Output:

```
{
    "pullRequest": {
        "approvalRules": [
            {
                "approvalRuleContent": "{\"Version\": \"2018-11-08\",\"Statements\":
 [{\"Type\": \"Approvers\",\"NumberOfApprovalsNeeded\": 1,\"ApprovalPoolMembers\":
 [\"arn:aws:sts::123456789012:assumed-role/CodeCommitReview/*\"]}]}",
                "approvalRuleId": "dd8b17fe-EXAMPLE",
                "approvalRuleName": "I want one approver for this pull request",
                "creationDate": 1571356106.936,
                "lastModifiedDate": 571356106.936,
```

```
                "lastModifiedUser": "arn:aws:iam::123456789012:user/Mary_Major",
                "ruleContentSha256": "4711b576EXAMPLE"
            }
        ],
        "authorArn": "arn:aws:iam::123456789012:user/Li_Juan",
        "clientRequestToken": "",
        "creationDate": 1508530823.142,
        "description": "Review the latest changes and updates to the global
 variables",
        "lastActivityDate": 1508887223.155,
        "pullRequestId": "47",
        "pullRequestStatus": "CLOSED",
        "pullRequestTargets": [
            {
                "destinationCommit": "9f31c968EXAMPLE",
                "destinationReference": "refs/heads/main",
                "mergeMetadata": {
                    "isMerged": true,
                    "mergedBy": "arn:aws:iam::123456789012:user/Mary_Major"
                },
                "repositoryName": "MyDemoRepo",
                "sourceCommit": "99132ab0EXAMPLE",
                "sourceReference": "refs/heads/variables-branch"
            }
        ],
        "title": "Consolidation of global variables"
    }
}
```

Per ulteriori informazioni, consulta Merge a Pull Request nella Guida per l'AWS CodeCommit utente.

- Per i dettagli sull'API, consulta MergePullRequestByFastForward AWS CLICommand Reference.

`merge-pull-request-by-squash`

Il seguente esempio di codice mostra come utilizzare`merge-pull-request-by-squash`.

AWS CLI

Come unire una richiesta pull utilizzando la strategia di unione squash

L'esempio `merge-pull-request-by-squash` seguente unisce e chiude la richiesta pull specificata utilizzando la strategia di risoluzione dei conflitti ACCEPT_SOURCE in un repository denominato `MyDemoRepo`.

```
aws codecommit merge-pull-request-by-squash \
    --pull-request-id 47 \
    --source-commit-id 99132ab0EXAMPLE \
    --repository-name MyDemoRepo \
    --conflict-detail-level LINE_LEVEL \
    --conflict-resolution-strategy ACCEPT_SOURCE \
    --name "Jorge Souza" --email "jorge_souza@example.com" \
    --commit-message "Merging pull request 47 by squash and accepting source in
 merge conflicts"
```

Output:

```
{
    "pullRequest": {
        "approvalRules": [
            {
                "approvalRuleContent": "{\"Version\": \"2018-11-08\",
\"DestinationReferences\": [\"refs/heads/main\"],\"Statements\": [{\"Type
\": \"Approvers\",\"NumberOfApprovalsNeeded\": 2,\"ApprovalPoolMembers\":
 [\"arn:aws:sts::123456789012:assumed-role/CodeCommitReview/*\"]}]}",
                "approvalRuleId": "dd8b17fe-EXAMPLE",
                "approvalRuleName": "2-approver-rule-for-main",
                "creationDate": 1571356106.936,
                "lastModifiedDate": 571356106.936,
                "lastModifiedUser": "arn:aws:iam::123456789012:user/Mary_Major",
                "originApprovalRuleTemplate": {
                    "approvalRuleTemplateId": "dd8b17fe-EXAMPLE",
                    "approvalRuleTemplateName": "2-approver-rule-for-main"
                },
                "ruleContentSha256": "4711b576EXAMPLE"
            }
        ],
        "authorArn": "arn:aws:iam::123456789012:user/Li_Juan",
        "clientRequestToken": "",
        "creationDate": 1508530823.142,
        "description": "Review the latest changes and updates to the global
 variables",
        "lastActivityDate": 1508887223.155,
        "pullRequestId": "47",
```

```
        "pullRequestStatus": "CLOSED",
        "pullRequestTargets": [
            {
                "destinationCommit": "9f31c968EXAMPLE",
                "destinationReference": "refs/heads/main",
                "mergeMetadata": {
                    "isMerged": true,
                    "mergedBy": "arn:aws:iam::123456789012:user/Mary_Major"
                },
                "repositoryName": "MyDemoRepo",
                "sourceCommit": "99132ab0EXAMPLE",
                "sourceReference": "refs/heads/variables-branch"
            }
        ],
        "title": "Consolidation of global variables"
    }
}
```

Per ulteriori informazioni, consulta Merge a Pull Request nella Guida per l'AWS CodeCommit utente.

- Per i dettagli sull'API, consulta MergePullRequestBySquash AWS CLICommand Reference.

`merge-pull-request-by-three-way`

Il seguente esempio di codice mostra come utilizzare`merge-pull-request-by-three-way`.

AWS CLI

Come unire una richiesta pull utilizzando la strategia di unione a tre vie

L'esempio `merge-pull-request-by-three-way` seguente unisce e chiude la richiesta pull specificata utilizzando le opzioni predefinite per i dettagli dei conflitti e la strategia di risoluzione dei conflitti in un repository denominato `MyDemoRepo`.

```
aws codecommit merge-pull-request-by-three-way \
    --pull-request-id 47 \
    --source-commit-id 99132ab0EXAMPLE \
    --repository-name MyDemoRepo \
    --name "Maria Garcia" \
    --email "maria_garcia@example.com" \
    --commit-message "Merging pull request 47 by three-way with default options"
```

Output:


```
{
    "pullRequest": {
        "approvalRules": [
            {
                "approvalRuleContent": "{\"Version\": \"2018-11-08\",
\"DestinationReferences\": [\"refs/heads/main\"],\"Statements\": [{\"Type
\": \"Approvers\",\"NumberOfApprovalsNeeded\": 2,\"ApprovalPoolMembers\":
 [\"arn:aws:sts::123456789012:assumed-role/CodeCommitReview/*\"]}]}",
                "approvalRuleId": "dd8b17fe-EXAMPLE",
                "approvalRuleName": "2-approver-rule-for-main",
                "creationDate": 1571356106.936,
                "lastModifiedDate": 571356106.936,
                "lastModifiedUser": "arn:aws:iam::123456789012:user/Mary_Major",
                "originApprovalRuleTemplate": {
                    "approvalRuleTemplateId": "dd8b17fe-EXAMPLE",
                    "approvalRuleTemplateName": "2-approver-rule-for-main"
                },
                "ruleContentSha256": "4711b576EXAMPLE"
            }
        ],
        "authorArn": "arn:aws:iam::123456789012:user/Li_Juan",
        "clientRequestToken": "",
        "creationDate": 1508530823.142,
        "description": "Review the latest changes and updates to the global
 variables",
        "lastActivityDate": 1508887223.155,
        "pullRequestId": "47",
        "pullRequestStatus": "CLOSED",
        "pullRequestTargets": [
            {
                "destinationCommit": "9f31c968EXAMPLE",
                "destinationReference": "refs/heads/main",
                "mergeMetadata": {
                    "isMerged": true,
                    "mergedBy": "arn:aws:iam::123456789012:user/Mary_Major"
                },
                "repositoryName": "MyDemoRepo",
                "sourceCommit": "99132ab0EXAMPLE",
                "sourceReference": "refs/heads/variables-branch"
            }
        ],
        "title": "Consolidation of global variables"
```

```
    }
}
```

Per ulteriori informazioni, consulta Merge a Pull Request nella Guida per l'AWS CodeCommit utente.

- Per i dettagli sull'API, consulta MergePullRequestByThreeWay AWS CLICommand Reference.

`override-pull-request-approval-rules`

Il seguente esempio di codice mostra come utilizzare`override-pull-request-approval-rules`.

AWS CLI

Come sostituire i requisiti della regola di approvazione in una richiesta pull

L'esempio `override-pull-request-approval-rules` seguente sostituisce le regole di approvazione nella richiesta pull specificata. Per revocare invece una sostituzione, imposta il valore del parametro `--override-status` su `REVOKE`.

```
aws codecommit override-pull-request-approval-rules \
    --pull-request-id 34  \
    --revision-id 927df8d8EXAMPLE \
    --override-status OVERRIDE
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Ignorare le regole di approvazione su una pull request nella Guida per l'AWS CodeCommit utente.

- Per i dettagli sull'API, consulta OverridePullRequestApprovalRules AWS CLICommand Reference.

`post-comment-for-compared-commit`

Il seguente esempio di codice mostra come utilizzare`post-comment-for-compared-commit`.

AWS CLI

Come creare un commento su un commit

Questo esempio mostra come aggiungere il commento "Can you add a test case for this?" quando si modifica il file cl_sample.js nel confronto tra due commit in un repository denominato MyDemoRepo.

```
aws codecommit post-comment-for-compared-commit \
    --repository-name MyDemoRepo \
    --before-commit-id 317f8570EXAMPLE \
    --after-commit-id 5d036259EXAMPLE \
    --client-request-token 123Example \
    --content "Can you add a test case for this?" \
    --location filePath=cl_sample.js,filePosition=1232,relativeFileVersion=AFTER
```

Output:

```
{
    "afterBlobId": "1f330709EXAMPLE",
    "afterCommitId": "317f8570EXAMPLE",
    "beforeBlobId": "80906a4cEXAMPLE",
    "beforeCommitId": "6e147360EXAMPLE",
    "comment": {
        "authorArn": "arn:aws:iam::111111111111:user/Li_Juan",
        "clientRequestToken": "",
        "commentId": "553b509bEXAMPLE56198325",
        "content": "Can you add a test case for this?",
        "creationDate": 1508369612.203,
        "deleted": false,
        "commentId": "abc123-EXAMPLE",
        "lastModifiedDate": 1508369612.203,
        "callerReactions": [],
        "reactionCounts": []
    },
    "location": {
        "filePath": "cl_sample.js",
        "filePosition": 1232,
        "relativeFileVersion": "AFTER"
    ,
    "repositoryName": "MyDemoRepo"
    }
}
```

- Per i dettagli sull'API, consulta PostCommentForComparedCommit AWS CLICommand Reference.

`post-comment-for-pull-request`

Il seguente esempio di codice mostra come utilizzare`post-comment-for-pull-request`.

AWS CLI

Come aggiungere un commento a una richiesta pull

L'esempio `post-comment-for-pull-request` seguente aggiunge il commento "These don't appear to be used anywhere. Can we remove them?" quando si modifica il file `ahs_count.py` in una richiesta pull con ID 47 in un repository denominato `MyDemoRepo`.

```
aws codecommit post-comment-for-pull-request \
    --pull-request-id "47" \
    --repository-name MyDemoRepo \
    --before-commit-id 317f8570EXAMPLE \
    --after-commit-id 5d036259EXAMPLE \
    --client-request-token 123Example \
    --content "These don't appear to be used anywhere. Can we remove them?" \
    --location filePath=ahs_count.py,filePosition=367,relativeFileVersion=AFTER
```

Output:

```
{
     "afterBlobId": "1f330709EXAMPLE",
     "afterCommitId": "5d036259EXAMPLE",
     "beforeBlobId": "80906a4cEXAMPLE",
     "beforeCommitId": "317f8570EXAMPLE",
     "comment": {
            "authorArn": "arn:aws:iam::111111111111:user/Saanvi_Sarkar",
            "clientRequestToken": "123Example",
            "commentId": "abcd1234EXAMPLEb5678efgh",
            "content": "These don't appear to be used anywhere. Can we remove
 them?",
            "creationDate": 1508369622.123,
            "deleted": false,
            "CommentId": "",
            "lastModifiedDate": 1508369622.123,
            "callerReactions": [],
            "reactionCounts": []
        },
        "location": {
            "filePath": "ahs_count.py",
```

```
            "filePosition": 367,
            "relativeFileVersion": "AFTER"
        },
    "repositoryName": "MyDemoRepo",
    "pullRequestId": "47"
}
```

- Per i dettagli sull'API, consulta PostCommentForPullRequest AWS CLICommand Reference.

`post-comment-reply`

Il seguente esempio di codice mostra come utilizzare`post-comment-reply`.

AWS CLI

Come rispondere a un commento su un commit o su una richiesta pull

Questo esempio mostra come aggiungere la risposta `"Good catch. I'll remove them."` al commento con l'ID `abcd1234EXAMPLEb5678efgh` generato dal sistema.

```
aws codecommit post-comment-reply \
    --in-reply-to abcd1234EXAMPLEb5678efgh \
    --content "Good catch. I'll remove them." \
    --client-request-token 123Example
```

Output:

```
{
    "comment": {
        "authorArn": "arn:aws:iam::111111111111:user/Li_Juan",
        "clientRequestToken": "123Example",
        "commentId": "442b498bEXAMPLE5756813",
        "content": "Good catch. I'll remove them.",
        "creationDate": 1508369829.136,
        "deleted": false,
        "CommentId": "abcd1234EXAMPLEb5678efgh",
        "lastModifiedDate": 150836912.221,
        "callerReactions": [],
        "reactionCounts": []
    }
}
```

- Per i dettagli sull'API, consulta PostCommentReply AWS CLICommand Reference.

`put-comment-reaction`

Il seguente esempio di codice mostra come utilizzare`put-comment-reaction`.

AWS CLI

Come rispondere a un commento su un commit con un'emoji

L'esempio `put-comment-reaction` seguente risponde a un commento con ID `abcd1234EXAMPLEb5678efgh` con una reazione emoji il cui valore è `:thumbsup:`.

```
aws codecommit put-comment-reaction \
    --comment-id abcd1234EXAMPLEb5678efgh \
    --reaction-value :thumbsup:
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Commentare un commit AWS CodeCommit nella Guida per l'AWS CodeCommit utente.

- Per i dettagli sull'API, consulta PutCommentReaction AWS CLICommand Reference.

`put-file`

Il seguente esempio di codice mostra come utilizzare`put-file`.

AWS CLI

Come aggiungere un file a un repository

L'`put-file`esempio seguente aggiunge un file denominato 'ExampleSolution.py' a un repository denominato '' a un ramo denominato 'feature-randomizationfeatureMyDemoRepo' il cui commit più recente ha un ID di '4C925148Example'.

```
aws codecommit put-file \
    --repository-name MyDemoRepo \
    --branch-name feature-randomizationfeature \
    --file-content file://MyDirectory/ExampleSolution.py \
    --file-path /solutions/ExampleSolution.py \
```

```
    --parent-commit-id 4c925148EXAMPLE \
    --name "Maria Garcia" \
    --email "maria_garcia@example.com" \
    --commit-message "I added a third randomization routine."
```

Output:

```
{
    "blobId": "2eb4af3bEXAMPLE",
    "commitId": "317f8570EXAMPLE",
    "treeId": "347a3408EXAMPLE"
}
```

- Per i dettagli sull'API, consulta Command Reference. PutFileAWS CLI

`put-repository-triggers`

Il seguente esempio di codice mostra come utilizzare`put-repository-triggers`.

AWS CLI

Come aggiungere o aggiornare un trigger in un repository

Questo esempio dimostra come aggiornare i trigger denominati " e 'MyFirstTriggerMySecondTrigger' utilizzando un file JSON già creato (qui denominato MyTriggers .json) che contiene la struttura di tutti i trigger per un repository denominato. MyDemoRepo Per informazioni su come ottenere il codice JSON per i trigger esistenti, consulta il comando. get-repository-triggers

```
aws codecommit put-repository-triggers \
    --repository-name MyDemoRepo file://MyTriggers.json
```

Contenuto di `MyTriggers.json`:

```
{
    "repositoryName": "MyDemoRepo",
    "triggers": [
        {
            "destinationArn": "arn:aws:sns:us-
east-1:80398EXAMPLE:MyCodeCommitTopic",
            "branches": [
```

```
                "main",
                "preprod"
            ],
            "name": "MyFirstTrigger",
            "customData": "",
            "events": [
                "all"
            ]
        },
        {
            "destinationArn": "arn:aws:lambda:us-
east-1:111111111111:function:MyCodeCommitPythonFunction",
            "branches": [],
            "name": "MySecondTrigger",
            "customData": "EXAMPLE",
            "events": [
                "all"
            ]
        }
    ]
}
```

Output:

```
{
    "configurationId": "6fa51cd8-35c1-EXAMPLE"
}
```

- Per i dettagli sull'API, consulta AWS CLI Command PutRepositoryTriggersReference.

`tag-resource`

Il seguente esempio di codice mostra come utilizzare`tag-resource`.

AWS CLI

Per aggiungere AWS tag a un repository esistente

L'esempio `tag-resource` seguente applica due tag al repository specificato.

```
aws codecommit tag-resource \
    --resource-arn arn:aws:codecommit:us-west-2:111111111111:MyDemoRepo \
```

```
    --tags Status=Secret,Team=Saanvi
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Aggiungere un tag a un repository nella Guida per l'AWS CodeCommit utente.

- Per i dettagli sull'API, consulta TagResource AWS CLICommand Reference.

`test-repository-triggers`

Il seguente esempio di codice mostra come utilizzare`test-repository-triggers`.

AWS CLI

Come testare i trigger in un repository

Questo esempio dimostra come testare un trigger denominato 'MyFirstTrigger' in un AWS CodeCommit repository denominato. MyDemoRepo In questo esempio, gli eventi nel repository attivano le notifiche da un argomento Amazon Simple Notification Service (Amazon SNS).

Comando:

```
aws codecommit test-repository-triggers --repository-name MyDemoRepo
 --triggers name=MyFirstTrigger,destinationArn=arn:aws:sns:us-
east-1:111111111111:MyCodeCommitTopic,branches=mainline,preprod,events=all
```

Output:

```
{
  "successfulExecutions": [
      "MyFirstTrigger"
  ],
  "failedExecutions": []
}
```

- Per i dettagli sull'API, vedere TestRepositoryTriggersin AWS CLI Command Reference.

`untag-resource`

Il seguente esempio di codice mostra come utilizzare`untag-resource`.

AWS CLI

Per rimuovere i AWS tag da un repository

L'esempio `untag-resource` seguente rimuove il tag con la chiave specificata dal repository denominato `MyDemoRepo`.

```
aws codecommit untag-resource \
    --resource-arn arn:aws:codecommit:us-west-2:111111111111:MyDemoRepo \
    --tag-keys Status
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Rimuovere un tag da un repository nella Guida per l'AWS CodeCommit utente.

- Per i dettagli sull'API, consulta UntagResource AWS CLICommand Reference.

`update-approval-rule-template-content`

Il seguente esempio di codice mostra come utilizzare`update-approval-rule-template-content`.

AWS CLI

Come aggiornare il contenuto di un modello di regola di approvazione

L'esempio `update-approval-rule-template-content` seguente modifica il contenuto di un modello di regola di approvazione specificato per ridefinire il pool di approvazioni per gli utenti che assumono il ruolo `CodeCommitReview`.

```
aws codecommit update-approval-rule-template-content \
    --approval-rule-template-name 1-approver-rule  \
    --new-rule-content "{\"Version\": \"2018-11-08\",\"DestinationReferences
\": [\"refs/heads/main\"],\"Statements\": [{\"Type\": \"Approvers
\",\"NumberOfApprovalsNeeded\": 2,\"ApprovalPoolMembers\":
 [\"arn:aws:sts::123456789012:assumed-role/CodeCommitReview/*\"]}]}"
```

Output:

```
{
    "approvalRuleTemplate": {
```

```
        "creationDate": 1571352720.773,
        "approvalRuleTemplateDescription": "Requires 1 approval for all pull
 requests from the CodeCommitReview pool",
        "lastModifiedDate": 1571358728.41,
        "approvalRuleTemplateId": "41de97b7-EXAMPLE",
        "approvalRuleTemplateContent": "{\"Version\": \"2018-11-08\",\"Statements\":
 [{\"Type\": \"Approvers\",\"NumberOfApprovalsNeeded\": 1,\"ApprovalPoolMembers\":
 [\"arn:aws:sts::123456789012:assumed-role/CodeCommitReview/*\"]}]}",
        "approvalRuleTemplateName": "1-approver-rule-for-all-pull-requests",
        "ruleContentSha256": "2f6c21a5EXAMPLE",
        "lastModifiedUser": "arn:aws:iam::123456789012:user/Li_Juan"
    }
}
```

Per ulteriori informazioni, vedere Gestire i modelli di regole di approvazione nella Guida AWS CodeCommit per l'utente.

- Per i dettagli sull'API, consulta UpdateApprovalRuleTemplateContent AWS CLICommand Reference.

update-approval-rule-template-description

Il seguente esempio di codice mostra come utilizzareupdate-approval-rule-template-description.

AWS CLI

Come aggiornare la descrizione di un modello di regola di approvazione

L'esempio update-approval-rule-template-description seguente modifica la descrizione del modello di regola di approvazione specificato in Requires 1 approval for all pull requests from the CodeCommitReview pool:

```
aws codecommit update-approval-rule-template-description \
    --approval-rule-template-name 1-approver-rule-for-all-pull-requests  \
    --approval-rule-template-description "Requires 1 approval for all pull requests
 from the CodeCommitReview pool"
```

Output:

```
{
  "approvalRuleTemplate": {
```

```
    "creationDate": 1571352720.773,
    "approvalRuleTemplateDescription": "Requires 1 approval for all pull requests
 from the CodeCommitReview pool",
    "lastModifiedDate": 1571358728.41,
    "approvalRuleTemplateId": "41de97b7-EXAMPLE",
    "approvalRuleTemplateContent": "{\"Version\": \"2018-11-08\",\"Statements\":
 [{\"Type\": \"Approvers\",\"NumberOfApprovalsNeeded\": 1,\"ApprovalPoolMembers\":
 [\"arn:aws:sts::123456789012:assumed-role/CodeCommitReview/*\"]}]}",
    "approvalRuleTemplateName": "1-approver-rule-for-all-pull-requests",
    "ruleContentSha256": "2f6c21a5EXAMPLE",
    "lastModifiedUser": "arn:aws:iam::123456789012:user/Li_Juan"
  }
}
```

Per ulteriori informazioni, vedere Gestire i modelli di regole di approvazione nella Guida AWS CodeCommit per l'utente.

- Per i dettagli sull'API, consulta UpdateApprovalRuleTemplateDescription AWS CLICommand Reference.

`update-approval-rule-template-name`

Il seguente esempio di codice mostra come utilizzare`update-approval-rule-template-name`.

AWS CLI

Come aggiornare il nome di un modello di regola di approvazione

L'`update-approval-rule-template-name`esempio seguente modifica il nome di un modello di regola di approvazione da `1-approver-rule` 1- approver-rule-for-all -pull-requests`.

```
aws codecommit update-approval-rule-template-name \
    --old-approval-rule-template-name 1-approver-rule  \
    --new-approval-rule-template-name 1-approver-rule-for-all-pull-requests
```

Output:

```
{
  "approvalRuleTemplate": {
    "approvalRuleTemplateName": "1-approver-rule-for-all-pull-requests",
    "lastModifiedDate": 1571358241.619,
    "approvalRuleTemplateId": "41de97b7-EXAMPLE",
```

```
    "approvalRuleTemplateContent": "{\"Version\": \"2018-11-08\",\"Statements\":
 [{\"Type\": \"Approvers\",\"NumberOfApprovalsNeeded\": 1,\"ApprovalPoolMembers\":
 [\"arn:aws:sts::123456789012:assumed-role/CodeCommitReview/*\"]}]}",
    "creationDate": 1571352720.773,
    "lastModifiedUser": "arn:aws:iam::123456789012:user/Mary_Major",
    "approvalRuleTemplateDescription": "All pull requests must be approved by one
 developer on the team.",
    "ruleContentSha256": "2f6c21a5cEXAMPLE"
  }
}
```

Per ulteriori informazioni, vedere Gestire i modelli di regole di approvazione nella Guida per l'utente.AWS CodeCommit

- Per i dettagli sull'API, consulta UpdateApprovalRuleTemplateName AWS CLICommand Reference.

`update-comment`

Il seguente esempio di codice mostra come utilizzare`update-comment`.

AWS CLI

Come aggiornare un commento su un commit

Questo esempio mostra come aggiungere il contenuto `"Fixed as requested. I'll update the pull request."` a un commento con ID `442b498bEXAMPLE5756813`.

```
aws codecommit update-comment \
    --comment-id 442b498bEXAMPLE5756813 \
    --content "Fixed as requested. I'll update the pull request."
```

Output:

```
{
    "comment": {
        "authorArn": "arn:aws:iam::111111111111:user/Li_Juan",
        "clientRequestToken": "",
        "commentId": "442b498bEXAMPLE5756813",
        "content": "Fixed as requested. I'll update the pull request.",
        "creationDate": 1508369929.783,
        "deleted": false,
```

```
        "lastModifiedDate": 1508369929.287,
        "callerReactions": [],
        "reactionCounts":
            {
                "THUMBSUP" : 2
            }
    }
}
```

- Per i dettagli sull'API, consulta UpdateComment AWS CLICommand Reference.

`update-default-branch`

Il seguente esempio di codice mostra come utilizzare`update-default-branch`.

AWS CLI

Come modificare la ramificazione predefinita per un repository

Questo esempio modifica il ramo predefinito per un AWS CodeCommit repository. Questo comando restituisce un output solo se ci sono errori.

Comando:

```
aws codecommit update-default-branch --repository-name MyDemoRepo --default-branch-
name MyNewBranch
```

Output:

```
None.
```

- Per i dettagli sull'API, vedere UpdateDefaultBranchin AWS CLI Command Reference.

`update-pull-request-approval-rule-content`

Il seguente esempio di codice mostra come utilizzare`update-pull-request-approval-rule-content`.

AWS CLI

Come modificare una regola di approvazione per una richiesta pull

L'`update-pull-request-approval-rule-content`esempio seguente aggiorna la regola di approvazione specificata per richiedere l'approvazione di un utente da un pool di approvazione che include qualsiasi utente IAM nell'123456789012 AWS account.

```
aws codecommit update-pull-request-approval-rule-content \
    --pull-request-id 27  \
    --approval-rule-name "Require two approved approvers" \
    --approval-rule-content "{Version: 2018-11-08, Statements: [{Type:
 \"Approvers\", NumberOfApprovalsNeeded: 1, ApprovalPoolMembers:
[\"CodeCommitApprovers:123456789012:user/*\"]}]}"
```

Output:

```
{
    "approvalRule": {
        "approvalRuleContent": "{Version: 2018-11-08, Statements:
 [{Type: \"Approvers\", NumberOfApprovalsNeeded: 1, ApprovalPoolMembers:
[\"CodeCommitApprovers:123456789012:user/*\"]}]}",
        "approvalRuleId": "aac33506-EXAMPLE",
        "originApprovalRuleTemplate": {},
        "creationDate": 1570752871.932,
        "lastModifiedDate": 1570754058.333,
        "approvalRuleName": Require two approved approvers",
        "lastModifiedUser": "arn:aws:iam::123456789012:user/Mary_Major",
        "ruleContentSha256": "cd93921cEXAMPLE",
    }
}
```

Per ulteriori informazioni, consulta Modificare o eliminare una regola di approvazione nella Guida AWS CodeCommit per l'utente.

- Per i dettagli sull'API, consulta UpdatePullRequestApprovalRuleContent AWS CLICommand Reference.

`update-pull-request-approval-state`

Il seguente esempio di codice mostra come utilizzare`update-pull-request-approval-state`.

AWS CLI

Come approvare o revocare un'approvazione per una richiesta pull

L'esempio `update-pull-request-approval-state` seguente approva una richiesta pull con ID 27 e ID revisione `9f29d167EXAMPLE`. Se invece si desidera revocare l'approvazione, imposta il valore del parametro `--approval-state` su `REVOKE`.

```
aws codecommit update-pull-request-approval-state \
    --pull-request-id 27  \
    --revision-id 9f29d167EXAMPLE  \
    --approval-state "APPROVE"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Review a Pull Request nella Guida AWS CodeCommit per l'utente.

- Per i dettagli sull'API, consulta UpdatePullRequestApprovalState AWS CLICommand Reference.

`update-pull-request-description`

Il seguente esempio di codice mostra come utilizzare`update-pull-request-description`.

AWS CLI

Come modificare la descrizione di una richiesta pull

Questo esempio mostra come modificare la descrizione di una richiesta pull con ID 47.

```
aws codecommit update-pull-request-description \
    --pull-request-id 47 \
    --description "Updated the pull request to remove unused global variable."
```

Output:

```
{
    "pullRequest": {
        "authorArn": "arn:aws:iam::111111111111:user/Li_Juan",
        "clientRequestToken": "",
        "creationDate": 1508530823.155,
        "description": "Updated the pull request to remove unused global variable.",
        "lastActivityDate": 1508372423.204,
```

```
        "pullRequestId": "47",
        "pullRequestStatus": "OPEN",
        "pullRequestTargets": [
            {
                "destinationCommit": "9f31c968EXAMPLE",
                "destinationReference": "refs/heads/main",
                "mergeMetadata": {
                    "isMerged": false,
                },
                "repositoryName": "MyDemoRepo",
                "sourceCommit": "99132ab0EXAMPLE",
                "sourceReference": "refs/heads/variables-branch"
            }
        ],
        "title": "Consolidation of global variables"
    }
}
```

- Per i dettagli sull'API, consulta UpdatePullRequestDescription AWS CLICommand Reference.

`update-pull-request-status`

Il seguente esempio di codice mostra come utilizzare`update-pull-request-status`.

AWS CLI

Come modificare lo stato di una richiesta pull

Questo esempio dimostra come modificare lo stato di una pull request con l'ID di in uno status `42` di `CLOSED` in un AWS CodeCommit repository denominato. `MyDemoRepo`

```
aws codecommit update-pull-request-status \
    --pull-request-id 42 \
    --pull-request-status CLOSED
```

Output:

```
{
    "pullRequest": {
        "approvalRules": [
            {
```

```
                "approvalRuleContent": "{\"Version\": \"2018-11-08\",\"Statements\":
 [{\"Type\": \"Approvers\",\"NumberOfApprovalsNeeded\": 2,\"ApprovalPoolMembers\":
 [\"arn:aws:sts::123456789012:assumed-role/CodeCommitReview/*\"]}]}",
                "approvalRuleId": "dd8b17fe-EXAMPLE",
                "approvalRuleName": "2-approvers-needed-for-this-change",
                "creationDate": 1571356106.936,
                "lastModifiedDate": 571356106.936,
                "lastModifiedUser": "arn:aws:iam::123456789012:user/Mary_Major",
                "ruleContentSha256": "4711b576EXAMPLE"
            }
        ],
        "authorArn": "arn:aws:iam::123456789012:user/Li_Juan",
        "clientRequestToken": "",
        "creationDate": 1508530823.165,
        "description": "Updated the pull request to remove unused global variable.",
        "lastActivityDate": 1508372423.12,
        "pullRequestId": "47",
        "pullRequestStatus": "CLOSED",
        "pullRequestTargets": [
            {
                "destinationCommit": "9f31c968EXAMPLE",
                "destinationReference": "refs/heads/main",
                "mergeMetadata": {
                    "isMerged": false,
                },
                "repositoryName": "MyDemoRepo",
                "sourceCommit": "99132ab0EXAMPLE",
                "sourceReference": "refs/heads/variables-branch"
            }
        ],
        "title": "Consolidation of global variables"
    }
}
```

- Per i dettagli sull'API, vedere UpdatePullRequestStatusin AWS CLI Command Reference.

`update-pull-request-title`

Il seguente esempio di codice mostra come utilizzareupdate-pull-request-title.

AWS CLI

Come modificare il titolo di una richiesta pull

Questo esempio mostra come modificare il titolo di una richiesta pull con ID 47.

```
aws codecommit update-pull-request-title \
    --pull-request-id 47 \
    --title "Consolidation of global variables - updated review"
```

Output:

```
{
    "pullRequest": {
        "approvalRules": [
            {
                "approvalRuleContent": "{\"Version\": \"2018-11-08\",
\"DestinationReferences\": [\"refs/heads/main\"],\"Statements\": [{\"Type
\": \"Approvers\",\"NumberOfApprovalsNeeded\": 2,\"ApprovalPoolMembers\":
 [\"arn:aws:sts::123456789012:assumed-role/CodeCommitReview/*\"]}]}",
                "approvalRuleId": "dd8b17fe-EXAMPLE",
                "approvalRuleName": "2-approver-rule-for-main",
                "creationDate": 1571356106.936,
                "lastModifiedDate": 571356106.936,
                "lastModifiedUser": "arn:aws:iam::123456789012:user/Mary_Major",
                "originApprovalRuleTemplate": {
                    "approvalRuleTemplateId": "dd8b26gr-EXAMPLE",
                    "approvalRuleTemplateName": "2-approver-rule-for-main"
                },
                "ruleContentSha256": "4711b576EXAMPLE"
            }
        ],
        "authorArn": "arn:aws:iam::123456789012:user/Li_Juan",
        "clientRequestToken": "",
        "creationDate": 1508530823.12,
        "description": "Review the latest changes and updates to the global
 variables. I have updated this request with some changes, including removing some
 unused variables.",
        "lastActivityDate": 1508372657.188,
        "pullRequestId": "47",
        "pullRequestStatus": "OPEN",
        "pullRequestTargets": [
            {
                "destinationCommit": "9f31c968EXAMPLE",
                "destinationReference": "refs/heads/main",
                "mergeMetadata": {
                    "isMerged": false,
```

```
                },
                "repositoryName": "MyDemoRepo",
                "sourceCommit": "99132ab0EXAMPLE",
                "sourceReference": "refs/heads/variables-branch"
            }
        ],
        "title": "Consolidation of global variables - updated review"
    }
}
```

- Per i dettagli sull'API, consulta UpdatePullRequestTitle AWS CLICommand Reference.

`update-repository-description`

Il seguente esempio di codice mostra come utilizzare`update-repository-description`.

AWS CLI

Come modificare la descrizione di un repository

Questo esempio modifica la descrizione di un AWS CodeCommit repository. Questo comando restituisce un output solo se ci sono errori.

Comando:

```
aws codecommit update-repository-description --repository-name MyDemoRepo --
repository-description "This description was changed"
```

Output:

```
None.
```

- Per i dettagli sull'API, vedere UpdateRepositoryDescriptionin AWS CLI Command Reference.

`update-repository-name`

Il seguente esempio di codice mostra come utilizzare`update-repository-name`.

AWS CLI

Come modificare il nome di un repository

Questo esempio modifica il nome di un AWS CodeCommit repository. Questo comando restituisce un output solo se ci sono errori. La modifica del nome del AWS CodeCommit repository cambierà il protocollo SSH e HTTPS di URLs cui gli utenti hanno bisogno per connettersi al repository. Gli utenti non riusciranno a connettersi a questo repository finché non aggiorneranno le impostazioni di connessione. Inoltre, poiché l'ARN del repository cambierà, la modifica del nome del repository invaliderà le policy dell'utente IAM basate su tale ARN.

Comando:

```
aws codecommit update-repository-name --old-name MyDemoRepo --new-
name MyRenamedDemoRepo
```

Output:

```
None.
```

- Per i dettagli sull'API, consulta Command UpdateRepositoryNameReference AWS CLI .

CodeDeploy esempi utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with CodeDeploy.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`add-tags-to-on-premises-instances`

Il seguente esempio di codice mostra come utilizzare`add-tags-to-on-premises-instances`.

AWS CLI

Come aggiungere tag a istanze on-premises

L'`add-tags-to-on-premises-instances`esempio seguente associa lo AWS CodeDeploy stesso tag di istanza locale a due istanze locali. Non registra le istanze locali con. AWS CodeDeploy

```
aws deploy add-tags-to-on-premises-instances \
    --instance-names AssetTag12010298EX AssetTag23121309EX \
    --tags Key=Name,Value=CodeDeployDemo-OnPrem
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta AWS CLI Command AddTagsToOnPremisesInstancesReference.

`batch-get-application-revisions`

Il seguente esempio di codice mostra come utilizzare`batch-get-application-revisions`.

AWS CLI

Come recuperare le informazioni sulle revisioni di un'applicazione

L'`batch-get-application-revisions`esempio seguente recupera le informazioni sulla revisione specificata memorizzata in un GitHub repository.

```
aws deploy batch-get-application-revisions \
    --application-name my-codedeploy-application \
    --revisions "[{\"gitHubLocation\": {\"commitId\":
 \"fa85936EXAMPLEa31736c051f10d77297EXAMPLE\",\"repository\": \"my-github-token/my-
repository\"},\"revisionType\": \"GitHub\"}]"
```

Output:

```
{
    "revisions": [
        {
            "genericRevisionInfo": {
                "description": "Application revision registered by Deployment ID: d-
A1B2C3111",
```

```
                "lastUsedTime": 1556912355.884,
                "registerTime": 1556912355.884,
                "firstUsedTime": 1556912355.884,
                "deploymentGroups": []
            },
            "revisionLocation": {
                "revisionType": "GitHub",
                "gitHubLocation": {
                    "commitId": "fa85936EXAMPLEa31736c051f10d77297EXAMPLE",
                    "repository": "my-github-token/my-repository"
                }
            }
        }
    ],
    "applicationName": "my-codedeploy-application",
    "errorMessage": ""
}
```

Per ulteriori informazioni, consulta BatchGetApplicationRevisions nella documentazione di riferimento dell'API AWS CodeDeploy .

- Per i dettagli sull'API, vedere BatchGetApplicationRevisionsin AWS CLI Command Reference.

`batch-get-applications`

Il seguente esempio di codice mostra come utilizzare`batch-get-applications`.

AWS CLI

Come ottenere informazioni su più applicazioni

L'`batch-get-applications`esempio seguente visualizza informazioni su più applicazioni associate all' AWS account dell'utente.

```
aws deploy batch-get-applications --application-names WordPress_App MyOther_App
```

Output:

```
{
    "applicationsInfo": [
        {
            "applicationName": "WordPress_App",
            "applicationId": "d9dd6993-f171-44fa-a811-211e4EXAMPLE",
```

```
            "createTime": 1407878168.078,
            "linkedToGitHub": false
        },
        {
            "applicationName": "MyOther_App",
            "applicationId": "8ca57519-31da-42b2-9194-8bb16EXAMPLE",
            "createTime": 1407453571.63,
            "linkedToGitHub": false
        }
    ]
}
```

- Per i dettagli sull'API, consulta BatchGetApplications AWS CLICommand Reference.

`batch-get-deployment-groups`

Il seguente esempio di codice mostra come utilizzare`batch-get-deployment-groups`.

AWS CLI

Come recuperare le informazioni su uno o più gruppi di distribuzione

L'`batch-get-deployment-groups`esempio seguente recupera informazioni su due gruppi di distribuzione associati all' CodeDeploy applicazione specificata.

```
aws deploy batch-get-deployment-groups \
    --application-name my-codedeploy-application \
    --deployment-group-names "[\"my-deployment-group-1\",\"my-deployment-group-2\"]"
```

Output:

```
{
    "deploymentGroupsInfo": [
        {
            "deploymentStyle": {
                "deploymentOption": "WITHOUT_TRAFFIC_CONTROL",
                "deploymentType": "IN_PLACE"
            },
            "autoRollbackConfiguration": {
                "enabled": false
            },
            "onPremisesTagSet": {
```

```
                "onPremisesTagSetList": []
            },
            "serviceRoleArn": "arn:aws:iam::123456789012:role/
CodeDeployServiceRole",
            "lastAttemptedDeployment": {
                "endTime": 1556912366.415,
                "status": "Failed",
                "createTime": 1556912355.884,
                "deploymentId": "d-A1B2C3111"
            },
            "autoScalingGroups": [],
            "deploymentGroupName": "my-deployment-group-1",
            "ec2TagSet": {
                "ec2TagSetList": [
                    [
                        {
                            "Type": "KEY_AND_VALUE",
                            "Value": "my-EC2-instance",
                            "Key": "Name"
                        }
                    ]
                ]
            },
            "deploymentGroupId": "a1b2c3d4-5678-90ab-cdef-11111example",
            "triggerConfigurations": [],
            "applicationName": "my-codedeploy-application",
            "computePlatform": "Server",
            "deploymentConfigName": "CodeDeployDefault.AllAtOnce"
        },
        {
            "deploymentStyle": {
                "deploymentOption": "WITHOUT_TRAFFIC_CONTROL",
                "deploymentType": "IN_PLACE"
            },
            "autoRollbackConfiguration": {
                "enabled": false
            },
            "onPremisesTagSet": {
                "onPremisesTagSetList": []
            },
            "serviceRoleArn": "arn:aws:iam::123456789012:role/
CodeDeployServiceRole",
            "autoScalingGroups": [],
            "deploymentGroupName": "my-deployment-group-2",
```

```
            "ec2TagSet": {
                "ec2TagSetList": [
                    [
                        {
                            "Type": "KEY_AND_VALUE",
                            "Value": "my-EC2-instance",
                            "Key": "Name"
                            }
                    ]
                ]
            },
            "deploymentGroupId": "a1b2c3d4-5678-90ab-cdef-22222example",
            "triggerConfigurations": [],
            "applicationName": "my-codedeploy-application",
            "computePlatform": "Server",
            "deploymentConfigName": "CodeDeployDefault.AllAtOnce"
        }
    ],
    "errorMessage": ""
}
```

Per ulteriori informazioni, consulta BatchGetDeploymentGroups nella documentazione di riferimento dell'API AWS CodeDeploy .

- Per i dettagli sull'API, vedere BatchGetDeploymentGroupsin AWS CLI Command Reference.

`batch-get-deployment-targets`

Il seguente esempio di codice mostra come utilizzare`batch-get-deployment-targets`.

AWS CLI

Come recuperare le destinazioni associate a un'implementazione

L'esempio `batch-get-deployment-targets` seguente restituisce informazioni su una delle destinazioni associate all'implementazione specificata.

```
aws deploy batch-get-deployment-targets \
    --deployment-id "d-1A2B3C4D5" \
    --target-ids "i-01a2b3c4d5e6f1111"
```

Output:

```
{
    "deploymentTargets": [
        {
            "deploymentTargetType": "InstanceTarget",
            "instanceTarget": {
                "lifecycleEvents": [
                    {
                        "startTime": 1556918592.162,
                        "lifecycleEventName": "ApplicationStop",
                        "status": "Succeeded",
                        "endTime": 1556918592.247,
                        "diagnostics": {
                            "scriptName": "",
                            "errorCode": "Success",
                            "logTail": "",
                            "message": "Succeeded"
                        }
                    },
                    {
                        "startTime": 1556918593.193,
                        "lifecycleEventName": "DownloadBundle",
                        "status": "Succeeded",
                        "endTime": 1556918593.981,
                        "diagnostics": {
                            "scriptName": "",
                            "errorCode": "Success",
                            "logTail": "",
                            "message": "Succeeded"
                        }
                    },
                    {
                        "startTime": 1556918594.805,
                        "lifecycleEventName": "BeforeInstall",
                        "status": "Succeeded",
                        "endTime": 1556918681.807,
                        "diagnostics": {
                            "scriptName": "",
                            "errorCode": "Success",
                            "logTail": "",
                            "message": "Succeeded"
                        }
                    }
                ],
```

```
                "targetArn": "arn:aws:ec2:us-west-2:123456789012:instance/
i-01a2b3c4d5e6f1111",
                "deploymentId": "d-1A2B3C4D5",
                "lastUpdatedAt": 1556918687.504,
                "targetId": "i-01a2b3c4d5e6f1111",
                "status": "Succeeded"
            }
        }
    ]
}
```

Per ulteriori informazioni, consulta BatchGetDeploymentTargets nella documentazione di riferimento dell'API AWS CodeDeploy .

- Per i dettagli sull'API, consulta BatchGetDeploymentTargets AWS CLICommand Reference.

`batch-get-deployments`

Il seguente esempio di codice mostra come utilizzare`batch-get-deployments`.

AWS CLI

Come ottenere informazioni su più implementazioni

L'`batch-get-deployments`esempio seguente visualizza informazioni su più distribuzioni associate all'account dell' AWS utente.

```
aws deploy batch-get-deployments --deployment-ids d-A1B2C3111 d-A1B2C3222
```

Output:

```
{
    "deploymentsInfo": [
        {
            "applicationName": "WordPress_App",
            "status": "Failed",
            "deploymentOverview": {
                "Failed": 0,
                "InProgress": 0,
                "Skipped": 0,
                "Succeeded": 1,
                "Pending": 0
            },
```

```
            "deploymentConfigName": "CodeDeployDefault.OneAtATime",
            "creator": "user",
            "deploymentGroupName": "WordPress_DG",
            "revision": {
                "revisionType": "S3",
                "s3Location": {
                "bundleType": "zip",
                "version": "uTecLusEXAMPLEFXtfUcyfV8bEXAMPLE",
                "bucket": "amzn-s3-demo-bucket",
                "key": "WordPressApp.zip"
                }
            },
            "deploymentId": "d-A1B2C3111",
            "createTime": 1408480721.9,
            "completeTime": 1408480741.822
        },
        {
            "applicationName": "MyOther_App",
            "status": "Failed",
            "deploymentOverview": {
                "Failed": 1,
                "InProgress": 0,
                "Skipped": 0,
                "Succeeded": 0,
                "Pending": 0
            },
            "deploymentConfigName": "CodeDeployDefault.OneAtATime",
            "creator": "user",
            "errorInformation": {
                "message": "Deployment failed: Constraint default violated: No hosts
succeeded.",
                "code": "HEALTH_CONSTRAINTS"
            },
            "deploymentGroupName": "MyOther_DG",
            "revision": {
                "revisionType": "S3",
                "s3Location": {
                "bundleType": "zip",
                "eTag": "\"dd56cfdEXAMPLE8e768f9d77fEXAMPLE\"",
                "bucket": "amzn-s3-demo-bucket",
                "key": "MyOtherApp.zip"
                }
            },
            "deploymentId": "d-A1B2C3222",
```

```
            "createTime": 1409764576.589,
            "completeTime": 1409764596.101
        }
    ]
}
```

- Per i dettagli sull'API, consulta AWS CLI Command BatchGetDeploymentsReference.

`batch-get-on-premises-instances`

Il seguente esempio di codice mostra come utilizzare`batch-get-on-premises-instances`.

AWS CLI

Come ottenere informazioni su una o più istanze on-premises

L'esempio `batch-get-on-premises-instances` seguente ottiene informazioni su due istanze on-premises.

```
aws deploy batch-get-on-premises-instances --instance-
names AssetTag12010298EX AssetTag23121309EX
```

Output:

```
{
    "instanceInfos": [
        {
            "iamUserArn": "arn:aws:iam::123456789012:user/AWS/CodeDeploy/
AssetTag12010298EX",
            "tags": [
                {
                    "Value": "CodeDeployDemo-OnPrem",
                    "Key": "Name"
                }
            ],
            "instanceName": "AssetTag12010298EX",
            "registerTime": 1425579465.228,
            "instanceArn": "arn:aws:codedeploy:us-west-2:123456789012:instance/
AssetTag12010298EX_4IwLNI2Alh"
        },
        {
```

```
            "iamUserArn": "arn:aws:iam::123456789012:user/AWS/CodeDeploy/
AssetTag23121309EX",
            "tags": [
                {
                    "Value": "CodeDeployDemo-OnPrem",
                    "Key": "Name"
                }
            ],
            "instanceName": "AssetTag23121309EX",
            "registerTime": 1425595585.988,
            "instanceArn": "arn:aws:codedeploy:us-west-2:80398EXAMPLE:instance/
AssetTag23121309EX_PomUy64Was"
        }
    ]
}
```

- Per i dettagli sull'API, consulta BatchGetOnPremisesInstances AWS CLICommand Reference.

`continue-deployment`

Il seguente esempio di codice mostra come utilizzare`continue-deployment`.

AWS CLI

Come iniziare a reinstradare il traffico senza attendere che il tempo specificato scada.

L'esempio `continue-deployment` seguente inizia a reinstradare il traffico dalle istanze nell'ambiente originale pronte per iniziare a spostare il traffico verso le istanze nell'ambiente sostitutivo.

```
aws deploy continue-deployment \
    --deployment-id "d-A1B2C3111" \
    --deployment-wait-type "READY_WAIT"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta ContinueDeployment nella documentazione di riferimento dell'API AWS CodeDeploy .

- Per i dettagli sull'API, consulta ContinueDeployment AWS CLICommand Reference.

`create-application`

Il seguente esempio di codice mostra come utilizzare`create-application`.

AWS CLI

Come creare un'applicazione

L'`create-application`esempio seguente crea un'applicazione e la associa all' AWS account dell'utente.

```
aws deploy create-application --application-name MyOther_App
```

Output:

```
{
    "applicationId": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE"
}
```

- Per i dettagli sull'API, vedere CreateApplicationin AWS CLI Command Reference.

`create-deployment-config`

Il seguente esempio di codice mostra come utilizzare`create-deployment-config`.

AWS CLI

Come creare una configurazione di implementazione personalizzata

L'`create-deployment-config`esempio seguente crea una configurazione di distribuzione personalizzata e la associa all' AWS account dell'utente.

```
aws deploy create-deployment-config \
    --deployment-config-name ThreeQuartersHealthy \
    --minimum-healthy-hosts type=FLEET_PERCENT,value=75
```

Output:

```
{
    "deploymentConfigId": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE"
}
```

- Per i dettagli sull'API, vedere CreateDeploymentConfigin AWS CLI Command Reference.

`create-deployment-group`

Il seguente esempio di codice mostra come utilizzare`create-deployment-group`.

AWS CLI

Come creare un gruppo di distribuzione

L'`create-deployment-group`esempio seguente crea un gruppo di distribuzione e lo associa all'applicazione specificata e all' AWS account dell'utente.

```
aws deploy create-deployment-group \
    --application-name WordPress_App \
    --auto-scaling-groups CodeDeployDemo-ASG \
    --deployment-config-name CodeDeployDefault.OneAtATime \
    --deployment-group-name WordPress_DG \
    --ec2-tag-filters Key=Name,Value=CodeDeployDemo,Type=KEY_AND_VALUE \
    --service-role-arn arn:aws:iam::123456789012:role/CodeDeployDemoRole
```

Output:

```
{
    "deploymentGroupId": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE"
}
```

- Per i dettagli sull'API, vedere CreateDeploymentGroupin AWS CLI Command Reference.

`create-deployment`

Il seguente esempio di codice mostra come utilizzare`create-deployment`.

AWS CLI

Esempio 1: creare una CodeDeploy distribuzione utilizzando la piattaforma di calcolo EC2 /On-premise

L'`create-deployment`esempio seguente crea una distribuzione e la associa all'account dell'utente. AWS

```
aws deploy create-deployment \
    --application-name WordPress_App \
    --deployment-config-name CodeDeployDefault.OneAtATime \
    --deployment-group-name WordPress_DG \
    --description "My demo deployment" \
    --s3-location bucket=amzn-s3-demo-
bucket,bundleType=zip,eTag=dd56cfdEXAMPLE8e768f9d77fEXAMPLE,key=WordPressApp.zip
```

Output:

```
{
    "deploymentId": "d-A1B2C3111"
}
```

Esempio 2: creare una CodeDeploy distribuzione utilizzando la piattaforma di calcolo Amazon ECS

L'esempio `create-deployment` seguente utilizza i due file seguenti per distribuire un servizio Amazon ECS.

Contenuto del file `create-deployment.json`:

```
{
    "applicationName": "ecs-deployment",
    "deploymentGroupName": "ecs-deployment-dg",
    "revision": {
        "revisionType": "S3",
        "s3Location": {
            "bucket": "ecs-deployment-bucket",
            "key": "appspec.yaml",
            "bundleType": "YAML"
        }
    }
}
```

A sua volta, questo file recupera il file `appspec.yaml` seguente da un bucket S3 denominato `ecs-deployment-bucket`.

```
version: 0.0
Resources:
  - TargetService:
```

```
      Type: AWS::ECS::Service
      Properties:
        TaskDefinition: "arn:aws:ecs:region:123456789012:task-definition/ecs-task-
def:2"
        LoadBalancerInfo:
          ContainerName: "sample-app"
          ContainerPort: 80
        PlatformVersion: "LATEST"
```

Comando:

```
aws deploy create-deployment \
    --cli-input-json file://create-deployment.json \
    --region us-east-1
```

Output:

```
{
    "deploymentId": "d-1234ABCDE"
}
```

Per ulteriori informazioni, consulta CreateDeployment nella documentazione di riferimento dell'API AWS CodeDeploy .

- Per i dettagli sull'API, consulta AWS CLI Command CreateDeploymentReference.

`delete-application`

Il seguente esempio di codice mostra come utilizzare`delete-application`.

AWS CLI

Come eliminare un'applicazione

L'`delete-application`esempio seguente elimina l'applicazione specificata associata all' AWS account dell'utente.

```
aws deploy delete-application --application-name WordPress_App
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, vedere DeleteApplicationin AWS CLI Command Reference.

`delete-deployment-config`

Il seguente esempio di codice mostra come utilizzare`delete-deployment-config`.

AWS CLI

Come eliminare una configurazione di implementazione

L'`delete-deployment-config`esempio seguente elimina una configurazione di distribuzione personalizzata associata all' AWS account dell'utente.

```
aws deploy delete-deployment-config --deployment-config-name ThreeQuartersHealthy
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, vedere DeleteDeploymentConfigin AWS CLI Command Reference.

`delete-deployment-group`

Il seguente esempio di codice mostra come utilizzare`delete-deployment-group`.

AWS CLI

Come eliminare un gruppo di distribuzione

L'esempio `delete-deployment-group` seguente elimina un gruppo di distribuzione associato all'applicazione specificata.

```
aws deploy delete-deployment-group \
    --application-name WordPress_App \
    --deployment-group-name WordPress_DG
```

Output:

```
{
    "hooksNotCleanedUp": []
}
```

- Per i dettagli sull'API, consulta DeleteDeploymentGroup AWS CLICommand Reference.

`delete-git-hub-account-token`

Il seguente esempio di codice mostra come utilizzare`delete-git-hub-account-token`.

AWS CLI

To elimina la connessione di un GitHub account

L'`delete-git-hub-account-token`esempio seguente elimina la connessione dell'account specificato GitHub .

```
aws deploy delete-git-hub-account-token --token-name my-github-account
```

Output:

```
{
    "tokenName": "my-github-account"
}
```

Per ulteriori informazioni, consulta DeleteGitHubAccountToken nella documentazione di riferimento dell'API AWS CodeDeploy .

- Per i dettagli sull'API, vedere DeleteGitHubAccountTokenin AWS CLI Command Reference.

`deregister-on-premises-instance`

Il seguente esempio di codice mostra come utilizzare`deregister-on-premises-instance`.

AWS CLI

Come annullare la registrazione di un'istanza on-premises

L'`deregister-on-premises-instance`esempio seguente annulla la registrazione di un'istanza locale con AWS CodeDeploy, ma non elimina l'utente IAM associato all'istanza, né si dissocia nei tag dell'istanza locale AWS CodeDeploy dall'istanza. Inoltre, non disinstalla l' AWS CodeDeploy agente dall'istanza né rimuove il file di configurazione locale dall'istanza.

```
aws deploy deregister-on-premises-instance --instance-name AssetTag12010298EX
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta DeregisterOnPremisesInstance AWS CLICommand Reference.

`deregister`

Il seguente esempio di codice mostra come utilizzare`deregister`.

AWS CLI

Come annullare la registrazione di un'istanza on-premises

L'`deregister`esempio seguente annulla la registrazione di un'istanza locale con. AWS CodeDeploy Non elimina l'utente IAM associato all'istanza. Nei tag locali si dissocia AWS CodeDeploy dall'istanza. Non disinstalla l' AWS CodeDeploy agente dall'istanza né rimuove il file di configurazione locale dall'istanza.

```
aws deploy deregister \
    --instance-name AssetTag12010298EX \
    --no-delete-iam-user \
    --region us-west-2
```

Output:

```
Retrieving on-premises instance information... DONE
IamUserArn: arn:aws:iam::80398EXAMPLE:user/AWS/CodeDeploy/AssetTag12010298EX
Tags: Key=Name,Value=CodeDeployDemo-OnPrem
Removing tags from the on-premises instance... DONE
Deregistering the on-premises instance... DONE
Run the following command on the on-premises instance to uninstall the codedeploy-
agent:
aws deploy uninstall
```

- Per informazioni dettagliate sull'API, consulta Deregister in AWS CLI Command Reference.

`get-application-revision`

Il seguente esempio di codice mostra come utilizzare`get-application-revision`.

AWS CLI

Come ottenere informazioni su una revisione di un'applicazione

L'esempio `get-application-revision` seguente visualizza le informazioni su una revisione dell'applicazione associata all'account specificato.

```
aws deploy get-application-revision \
    --application-name WordPress_App \
    --s3-location bucket=amzn-s3-demo-
bucket,bundleType=zip,eTag=dd56cfdEXAMPLE8e768f9d77fEXAMPLE,key=WordPressApp.zip
```

Output:

```
{
    "applicationName": "WordPress_App",
    "revisionInfo": {
        "description": "Application revision registered by Deployment ID: d-
A1B2C3111",
        "registerTime": 1411076520.009,
        "deploymentGroups": "WordPress_DG",
        "lastUsedTime": 1411076520.009,
        "firstUsedTime": 1411076520.009
    },
    "revision": {
        "revisionType": "S3",
        "s3Location": {
            "bundleType": "zip",
            "eTag": "dd56cfdEXAMPLE8e768f9d77fEXAMPLE",
            "bucket": "amzn-s3-demo-bucket",
            "key": "WordPressApp.zip"
        }
    }
}
```

- Per i dettagli sull'API, consulta GetApplicationRevision AWS CLICommand Reference.

`get-application`

Il seguente esempio di codice mostra come utilizzare`get-application`.

AWS CLI

Come ottenere informazioni su un'applicazione

L'`get-application`esempio seguente visualizza informazioni su un'applicazione associata all' AWS account dell'utente.

```
aws deploy get-application --application-name WordPress_App
```

Output:

```
{
    "application": {
        "applicationName": "WordPress_App",
        "applicationId": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
        "createTime": 1407878168.078,
        "linkedToGitHub": false
    }
}
```

- Per i dettagli sull'API, consulta GetApplication AWS CLICommand Reference.

`get-deployment-config`

Il seguente esempio di codice mostra come utilizzare`get-deployment-config`.

AWS CLI

Come ottenere informazioni su una configurazione di implementazione

L'`get-deployment-config`esempio seguente visualizza informazioni su una configurazione di distribuzione associata all' AWS account dell'utente.

```
aws deploy get-deployment-config --deployment-config-name ThreeQuartersHealthy
```

Output:

```
{
    "deploymentConfigInfo": {
        "deploymentConfigId": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
        "minimumHealthyHosts": {
            "type": "FLEET_PERCENT",
            "value": 75
        },
        "createTime": 1411081164.379,
        "deploymentConfigName": "ThreeQuartersHealthy"
    }
```

```
}
```

- Per i dettagli sull'API, vedere GetDeploymentConfigin AWS CLI Command Reference.

`get-deployment-group`

Il seguente esempio di codice mostra come utilizzare`get-deployment-group`.

AWS CLI

Come visualizzare le informazioni sui gruppi di distribuzione

L'esempio `get-deployment-group` seguente visualizza le informazioni su un gruppo di distribuzione associato all'account specificato.

```
aws deploy get-deployment-group \
    --application-name WordPress_App \
    --deployment-group-name WordPress_DG
```

Output:

```
{
    "deploymentGroupInfo": {
        "applicationName": "WordPress_App",
        "autoScalingGroups": [
            "CodeDeployDemo-ASG"
        ],
        "deploymentConfigName": "CodeDeployDefault.OneAtATime",
        "ec2TagFilters": [
            {
                "Type": "KEY_AND_VALUE",
                "Value": "CodeDeployDemo",
                "Key": "Name"
            }
        ],
        "deploymentGroupId": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
        "serviceRoleArn": "arn:aws:iam::123456789012:role/CodeDeployDemoRole",
        "deploymentGroupName": "WordPress_DG"
    }
}
```

- Per i dettagli sull'API, consulta GetDeploymentGroup AWS CLICommand Reference.

`get-deployment-instance`

Il seguente esempio di codice mostra come utilizzare`get-deployment-instance`.

AWS CLI

Come ottenere informazioni su un'istanza dell'implementazione

L'esempio `get-deployment-instance` seguente visualizza le informazioni su un'istanza di implementazione associata all'implementazione specificata.

```
aws deploy get-deployment-instance --deployment-id d-QA4G4F9EX --instance-
id i-902e9fEX
```

Output:

```
{
    "instanceSummary": {
        "instanceId": "arn:aws:ec2:us-east-1:80398EXAMPLE:instance/i-902e9fEX",
        "lifecycleEvents": [
            {
                "status": "Succeeded",
                "endTime": 1408480726.569,
                "startTime": 1408480726.437,
                "lifecycleEventName": "ApplicationStop"
            },
            {
                "status": "Succeeded",
                "endTime": 1408480728.016,
                "startTime": 1408480727.665,
                "lifecycleEventName": "DownloadBundle"
            },
            {
                "status": "Succeeded",
                "endTime": 1408480729.744,
                "startTime": 1408480729.125,
                "lifecycleEventName": "BeforeInstall"
            },
            {
                "status": "Succeeded",
                "endTime": 1408480730.979,
                "startTime": 1408480730.844,
                "lifecycleEventName": "Install"
```

```
            },
            {
                "status": "Failed",
                "endTime": 1408480732.603,
                "startTime": 1408480732.1,
                "lifecycleEventName": "AfterInstall"
            },
            {
                "status": "Skipped",
                "endTime": 1408480732.606,
                "lifecycleEventName": "ApplicationStart"
            },
            {
                "status": "Skipped",
                "endTime": 1408480732.606,
                "lifecycleEventName": "ValidateService"
            }
        ],
        "deploymentId": "d-QA4G4F9EX",
        "lastUpdatedAt": 1408480733.152,
        "status": "Failed"
    }
}
```

- Per i dettagli sull'API, consulta GetDeploymentInstance AWS CLICommand Reference.

`get-deployment-target`

Il seguente esempio di codice mostra come utilizzare`get-deployment-target`.

AWS CLI

Come restituire informazioni su una destinazione di implementazione

L'esempio `get-deployment-target` seguente restituisce le informazioni su una destinazione di implementazione associata all'implementazione specificata.

```
aws deploy get-deployment-target \
    --deployment-id "d-A1B2C3111" \
    --target-id "i-a1b2c3d4e5f611111"
```

Output:

```
{
    "deploymentTarget": {
        "deploymentTargetType": "InstanceTarget",
        "instanceTarget": {
            "lastUpdatedAt": 1556918687.504,
            "targetId": "i-a1b2c3d4e5f611111",
            "targetArn": "arn:aws:ec2:us-west-2:123456789012:instance/i-
a1b2c3d4e5f611111",
            "status": "Succeeded",
            "lifecycleEvents": [
                {
                    "status": "Succeeded",
                    "diagnostics": {
                        "errorCode": "Success",
                        "message": "Succeeded",
                        "logTail": "",
                        "scriptName": ""
                    },
                    "lifecycleEventName": "ApplicationStop",
                    "startTime": 1556918592.162,
                    "endTime": 1556918592.247
                },
                {
                    "status": "Succeeded",
                    "diagnostics": {
                        "errorCode": "Success",
                        "message": "Succeeded",
                        "logTail": "",
                        "scriptName": ""
                    },
                    "lifecycleEventName": "DownloadBundle",
                    "startTime": 1556918593.193,
                    "endTime": 1556918593.981
                },
                {
                    "status": "Succeeded",
                    "diagnostics": {
                        "errorCode": "Success",
                        "message": "Succeeded",
                        "logTail": "",
                        "scriptName": ""
                    },
                    "lifecycleEventName": "BeforeInstall",
```

```
                    "startTime": 1556918594.805,
                    "endTime": 1556918681.807
                },
                {
                    "status": "Succeeded",
                    "diagnostics": {
                        "errorCode": "Success",
                        "message": "Succeeded",
                        "logTail": "",
                        "scriptName": ""
                    },
                    "lifecycleEventName": "Install",
                    "startTime": 1556918682.696,
                    "endTime": 1556918683.005
                },
                {
                    "status": "Succeeded",
                    "diagnostics": {
                        "errorCode": "Success",
                        "message": "Succeeded",
                        "logTail": "",
                        "scriptName": ""
                    },
                    "lifecycleEventName": "AfterInstall",
                    "startTime": 1556918684.135,
                    "endTime": 1556918684.216
                },
                {
                    "status": "Succeeded",
                    "diagnostics": {
                        "errorCode": "Success",
                        "message": "Succeeded",
                        "logTail": "",
                        "scriptName": ""
                    },
                    "lifecycleEventName": "ApplicationStart",
                    "startTime": 1556918685.211,
                    "endTime": 1556918685.295
                },
                {
                    "status": "Succeeded",
                    "diagnostics": {
                        "errorCode": "Success",
                        "message": "Succeeded",
```

```
                        "logTail": "",
                        "scriptName": ""
                    },
                    "lifecycleEventName": "ValidateService",
                    "startTime": 1556918686.65,
                    "endTime": 1556918686.747
                }
            ],
            "deploymentId": "d-A1B2C3111"
        }
    }
}
```

Per ulteriori informazioni, consulta GetDeploymentTarget nella documentazione di riferimento dell'API AWS CodeDeploy .

- Per i dettagli sull'API, consulta GetDeploymentTarget AWS CLICommand Reference.

`get-deployment`

Il seguente esempio di codice mostra come utilizzare`get-deployment`.

AWS CLI

Come ottenere informazioni su un'implementazione

L'`get-deployment`esempio seguente visualizza informazioni su una distribuzione associata all' AWS account dell'utente.

```
aws deploy get-deployment --deployment-id d-A1B2C3123
```

Output:

```
{
    "deploymentInfo": {
        "applicationName": "WordPress_App",
        "status": "Succeeded",
        "deploymentOverview": {
            "Failed": 0,
            "InProgress": 0,
            "Skipped": 0,
            "Succeeded": 1,
            "Pending": 0
```

```
        },
        "deploymentConfigName": "CodeDeployDefault.OneAtATime",
        "creator": "user",
        "description": "My WordPress app deployment",
        "revision": {
            "revisionType": "S3",
            "s3Location":  {
            "bundleType": "zip",
            "eTag": "\"dd56cfdEXAMPLE8e768f9d77fEXAMPLE\"",
            "bucket": "amzn-s3-demo-bucket",
            "key": "WordPressApp.zip"
            }
        },
        "deploymentId": "d-A1B2C3123",
        "deploymentGroupName": "WordPress_DG",
        "createTime": 1409764576.589,
        "completeTime": 1409764596.101,
        "ignoreApplicationStopFailures": false
    }
}
```

- Per i dettagli sull'API, consulta GetDeployment AWS CLICommand Reference.

`get-on-premises-instance`

Il seguente esempio di codice mostra come utilizzare`get-on-premises-instance`.

AWS CLI

Come ottenere informazioni su un'istanza on-premises

L'esempio `get-on-premises-instance` seguente recupera le informazioni sull'istanza on-premises specificata.

```
aws deploy get-on-premises-instance --instance-name AssetTag12010298EX
```

Output:

```
{
    "instanceInfo": {
    "iamUserArn": "arn:aws:iam::123456789012:user/AWS/CodeDeploy/
AssetTag12010298EX",
```

```
        "tags": [
        {
            "Value": "CodeDeployDemo-OnPrem",
            "Key": "Name"
        }
        ],
        "instanceName": "AssetTag12010298EX",
        "registerTime": 1425579465.228,
        "instanceArn": "arn:aws:codedeploy:us-east-1:123456789012:instance/
AssetTag12010298EX_4IwLNI2Alh"
    }
}
```

- Per i dettagli sull'API, consulta GetOnPremisesInstance AWS CLICommand Reference.

`install`

Il seguente esempio di codice mostra come utilizzare`install`.

AWS CLI

Come installare un'istanza on-premises

L'`install`esempio seguente copia il file di configurazione locale dalla posizione specificata sull'istanza alla posizione sull'istanza in cui l' AWS CodeDeploy agente si aspetta di trovarlo. Inoltre, installa l' AWS CodeDeploy agente sull'istanza. Non crea alcun utente IAM, né registra l'istanza locale né associa alcun tag di istanza locale all' AWS CodeDeploy istanza. AWS CodeDeploy

```
aws deploy install \
    --override-config \
    --config-file C:\temp\codedeploy.onpremises.yml \
    --region us-west-2 \
    --agent-installer s3://aws-codedeploy-us-west-2/latest/codedeploy-agent.msi
```

Output:

```
Creating the on-premises instance configuration file... DONE
Installing the AWS CodeDeploy Agent... DONE
```

- Per informazioni dettagliate sull'API, consulta Install in AWS CLI Command Reference.

`list-application-revisions`

Il seguente esempio di codice mostra come utilizzare. `list-application-revisions`

AWS CLI

Come ottenere informazioni sulle revisioni di un'applicazione

L'esempio `list-application-revisions` seguente visualizza le informazioni su tutte le revisioni associate all'applicazione specificata.

```
aws deploy list-application-revisions \
    --application-name WordPress_App \
    --s-3-bucket amzn-s3-demo-bucket \
    --deployed exclude \
    --s-3-key-prefix WordPress_ \
    --sort-by lastUsedTime \
    --sort-order descending
```

Output:

```
{
    "revisions": [
        {
            "revisionType": "S3",
            "s3Location": {
                "version": "uTecLusvCB_JqHFXtfUcyfV8bEXAMPLE",
                "bucket": "amzn-s3-demo-bucket",
                "key": "WordPress_App.zip",
                "bundleType": "zip"
            }
        },
        {
            "revisionType": "S3",
            "s3Location": {
                "version": "tMk.UxgDpMEVb7V187ZM6wVAWEXAMPLE",
                "bucket": "amzn-s3-demo-bucket",
                "key": "WordPress_App_2-0.zip",
                "bundleType": "zip"
            }
        }
    ]
}
```

- Per i dettagli sull'API, consulta ListApplicationRevisions AWS CLICommand Reference.

`list-applications`

Il seguente esempio di codice mostra come utilizzare`list-applications`.

AWS CLI

Come ottenere informazioni sulle applicazione

L'`list-applications`esempio seguente visualizza informazioni su tutte le applicazioni associate all' AWS account dell'utente.

```
aws deploy list-applications
```

Output:

```
{
    "applications": [
        "WordPress_App",
        "MyOther_App"
    ]
}
```

- Per i dettagli sull'API, consulta ListApplications AWS CLICommand Reference.

`list-deployment-configs`

Il seguente esempio di codice mostra come utilizzare`list-deployment-configs`.

AWS CLI

Come ottenere informazioni sulle configurazioni di implementazione

L'`list-deployment-configs`esempio seguente visualizza informazioni su tutte le configurazioni di distribuzione associate all' AWS account dell'utente.

```
aws deploy list-deployment-configs
```

Output:

```
{
    "deploymentConfigsList": [
        "ThreeQuartersHealthy",
        "CodeDeployDefault.AllAtOnce",
        "CodeDeployDefault.HalfAtATime",
        "CodeDeployDefault.OneAtATime"
    ]
}
```

- Per i dettagli sull'API, vedere ListDeploymentConfigsin AWS CLI Command Reference.

`list-deployment-groups`

Il seguente esempio di codice mostra come utilizzare`list-deployment-groups`.

AWS CLI

Come ottenere informazioni sui gruppi di distribuzione

L'esempio `list-deployment-groups` seguente visualizza le informazioni sui gruppi di distribuzione associati all'account specificato.

```
aws deploy list-deployment-groups --application-name WordPress_App
```

Output:

```
{
    "applicationName": "WordPress_App",
    "deploymentGroups": [
        "WordPress_DG",
        "WordPress_Beta_DG"
    ]
}
```

- Per i dettagli sull'API, consulta ListDeploymentGroups AWS CLICommand Reference.

`list-deployment-instances`

Il seguente esempio di codice mostra come utilizzare`list-deployment-instances`.

AWS CLI

Come ottenere informazioni sulle istanze dell'implementazione

L'esempio `list-deployment-instances` seguente visualizza le informazioni su tutte le istanze di implementazione associate all'implementazione specificata.

```
aws deploy list-deployment-instances \
    --deployment-id d-A1B2C3111 \
    --instance-status-filter Succeeded
```

Output:

```
{
    "instancesList": [
        "i-EXAMPLE11",
        "i-EXAMPLE22"
    ]
}
```

- Per i dettagli sull'API, consulta ListDeploymentInstances AWS CLICommand Reference.

`list-deployment-targets`

Il seguente esempio di codice mostra come utilizzare`list-deployment-targets`.

AWS CLI

Per recuperare un elenco di destinazioni IDs associate a una distribuzione

L'`list-deployment-targets`esempio seguente recupera un elenco di oggetti IDs associati a distribuzioni con lo stato «Failed» o "». InProgress

```
aws deploy list-deployment-targets \
    --deployment-id "d-A1B2C3111" \
    --target-filters "{\"TargetStatus\":[\"Failed\",\"InProgress\"]}"
```

Output:

```
{
    "targetIds": [
```

```
        "i-0f1558aaf90e5f1f9"
    ]
}
```

Per ulteriori informazioni, consulta ListDeploymentTargets nella documentazione di riferimento dell'API AWS CodeDeploy .

- Per i dettagli sull'API, consulta Command ListDeploymentTargetsReference AWS CLI .

`list-deployments`

Il seguente esempio di codice mostra come utilizzare`list-deployments`.

AWS CLI

Come ottenere informazioni sulle implementazioni

L'esempio `list-deployments` seguente visualizza le informazioni su tutte le implementazioni associate all'applicazione e al gruppo di distribuzione specificati.

```
aws deploy list-deployments \
    --application-name WordPress_App \
    --create-time-range start=2014-08-19T00:00:00,end=2014-08-20T00:00:00 \
    --deployment-group-name WordPress_DG \
    --include-only-statuses Failed
```

Output:

```
{
    "deployments": [
        "d-EXAMPLE11",
        "d-EXAMPLE22",
        "d-EXAMPLE33"
    ]
}
```

- Per i dettagli sull'API, consulta ListDeployments AWS CLICommand Reference.

`list-git-hub-account-token-names`

Il seguente esempio di codice mostra come utilizzare`list-git-hub-account-token-names`.

AWS CLI

Elenca i nomi delle connessioni memorizzate agli GitHub account

L'`list-git-hub-account-token-names`esempio seguente elenca i nomi delle connessioni memorizzate agli GitHub account AWS dell'utente corrente.

```
aws deploy list-git-hub-account-token-names
```

Output:

```
{
    "tokenNameList": [
        "my-first-token",
        "my-second-token",
        "my-third-token"
    ]
}
```

Per ulteriori informazioni, consulta ListGitHubAccountTokenNames nella documentazione di riferimento dell'API AWS CodeDeploy .

- Per i dettagli sull'API, vedere ListGitHubAccountTokenNamesin AWS CLI Command Reference.

`list-on-premises-instances`

Il seguente esempio di codice mostra come utilizzare`list-on-premises-instances`.

AWS CLI

Come ottenere informazioni su una o più istanze on-premises

L'`list-on-premises-instances`esempio seguente recupera un elenco di nomi di istanze locali disponibili per le istanze registrate in AWS CodeDeploy e a cui è associato anche il tag di istanza locale specificato. AWS CodeDeploy

```
aws deploy list-on-premises-instances \
    --registration-status Registered \
    --tag-filters Key=Name,Value=CodeDeployDemo-OnPrem,Type=KEY_AND_VALUE
```

Output:

```
{
    "instanceNames": [
        "AssetTag12010298EX"
    ]
}
```

- Per i dettagli sull'API, consulta ListOnPremisesInstancesCommand Reference.AWS CLI

`list-tags-for-resource`

Il seguente esempio di codice mostra come utilizzare`list-tags-for-resource`.

AWS CLI

Come elencare i tag per una risorsa (applicazione)

L'`list-tags-for-resource`esempio seguente elenca i tag applicati a un'applicazione denominata TestApp in CodeDeploy.

```
aws deploy list-tags-for-resource \
    --resource-arn arn:aws:codedeploy:us-west-2:111122223333:application:testApp
```

Output:

```
{
    "Tags": [
        {
            "Key": "Type",
            "Value": "testType"
        },
        {
            "Key": "Name",
            "Value": "testName"
        }
    ]
}
```

Per ulteriori informazioni, consulta la sezione Tagging delle istanze per i gruppi di distribuzione CodeDeploy nella Guida per l'AWS CodeDeploy utente.

- Per i dettagli sull'API, consulta AWS CLI Command ListTagsForResourceReference.

`push`

Il seguente esempio di codice mostra come utilizzare`push`.

AWS CLI

Per raggruppare e distribuire una revisione dell'applicazione AWS CodeDeploy compatibile su Amazon S3

L'esempio `push` seguente raggruppa e implementa una revisione dell'applicazione in Amazon S3 e associa la revisione all'applicazione specificata.

```
aws deploy push \
    --application-name WordPress_App \
    --description "This is my deployment" \
    --ignore-hidden-files \
    --s3-location s3://amzn-s3-demo-bucket/WordPressApp.zip \
    --source /tmp/MyLocalDeploymentFolder/
```

L'output descrive come usare il comando `create-deployment` per creare un'implementazione che utilizza la revisione dell'applicazione caricata.

```
To deploy with this revision, run:
aws deploy create-deployment --application-name WordPress_App --
deployment-config-name <deployment-config-name> --deployment-group-
name <deployment-group-name> --s3-location bucket=amzn-s3-demo-
bucket,key=WordPressApp.zip,bundleType=zip,eTag="cecc9b8EXAMPLE50a6e71fdb88EXAMPLE",version=
```

- Per informazioni dettagliate sull'API, consulta Push in AWS CLI Command Reference.

`register-application-revision`

Il seguente esempio di codice mostra come usare. `register-application-revision`

AWS CLI

Come registrare le informazioni su una revisione di un'applicazione già caricata

L'`register-application-revision`esempio seguente registra le informazioni su una revisione di un'applicazione già caricata archiviata in Amazon S3 con. AWS CodeDeploy

```
aws deploy register-application-revision \
    --application-name WordPress_App \
    --description "Revised WordPress application" \
    --s3-location bucket=amzn-s3-demo-
bucket,key=RevisedWordPressApp.zip,bundleType=zip,eTag=cecc9b8a08eac650a6e71fdb88EXAMPLE
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta Command Reference. RegisterApplicationRevisionAWS CLI

`register-on-premises-instance`

Il seguente esempio di codice mostra come utilizzare`register-on-premises-instance`.

AWS CLI

Come registrare un'istanza on-premises

L'`register-on-premises-instance`esempio seguente registra un'istanza locale con. AWS CodeDeploy Non crea l'utente IAM specificato, né associa in AWS CodeDeploy nessuna istanza locale i tag all'istanza registrata.

```
aws deploy register-on-premises-instance \
    --instance-name AssetTag12010298EX \
    --iam-user-arn arn:aws:iam::80398EXAMPLE:user/CodeDeployDemoUser-OnPrem
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta AWS CLI Command RegisterOnPremisesInstanceReference.

`register`

Il seguente esempio di codice mostra come utilizzare`register`.

AWS CLI

Come registrare un'istanza on-premises

L'`register`esempio seguente registra un'istanza locale con AWS CodeDeploy, la associa AWS CodeDeploy all'istanza registrata nel tag di istanza locale specificato e crea un file di

configurazione locale che può essere copiato nell'istanza. Non crea l'utente IAM né installa l'agente sull'istanza. AWS CodeDeploy

```
aws deploy register \
    --instance-name AssetTag12010298EX \
    --iam-user-arn arn:aws:iam::80398EXAMPLE:user/CodeDeployUser-OnPrem \
    --tags Key=Name,Value=CodeDeployDemo-OnPrem \
    --region us-west-2
```

Output:

```
Registering the on-premises instance... DONE
Adding tags to the on-premises instance... DONE
Copy the on-premises configuration file named codedeploy.onpremises.yml to the on-
premises instance, and run the following command on the on-premises instance to
 install and configure the AWS CodeDeploy Agent:
aws deploy install --config-file codedeploy.onpremises.yml
```

- Per informazioni dettagliate sull'API, consulta Register in AWS CLI Command Reference.

`remove-tags-from-on-premises-instances`

Il seguente esempio di codice mostra come utilizzare`remove-tags-from-on-premises-instances`.

AWS CLI

Come rimuovere i tag da una o più istanze on-premises

L'`remove-tags-from-on-premises-instances`esempio seguente dissocia i tag locali specificati AWS CodeDeploy dalle istanze locali. Non annulla la registrazione delle istanze locali né disinstalla l' AWS CodeDeploy agente dall'istanza AWS CodeDeploy, né rimuove il file di configurazione locale dalle istanze, né elimina gli utenti IAM associati alle istanze.

```
aws deploy remove-tags-from-on-premises-instances \
    --instance-names AssetTag12010298EX AssetTag23121309EX \
    --tags Key=Name,Value=CodeDeployDemo-OnPrem
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta Command Reference. RemoveTagsFromOnPremisesInstancesAWS CLI

`stop-deployment`

Il seguente esempio di codice mostra come utilizzare`stop-deployment`.

AWS CLI

Come tentare di arrestare un'implementazione

L'`stop-deployment`esempio seguente tenta di interrompere una distribuzione in corso associata all' AWS account dell'utente.

aws deploy stop-deployment --deployment-id d-A1B2C3111

Output:

```
{
    "status": "Succeeded",
    "statusMessage": "No more commands will be scheduled for execution in the
 deployment instances"
}
```

- Per i dettagli sull'API, consulta StopDeployment AWS CLICommand Reference.

`tag-resource`

Il seguente esempio di codice mostra come utilizzare`tag-resource`.

AWS CLI

Come taggare una risorsa (applicazione)

L'`tag-resource`esempio seguente aggiunge due tag con le chiavi Name e Type e i valori TestName e TestType a un'applicazione denominata TestApp in. CodeDeploy :

```
aws deploy tag-resource \
    --resource-arn  arn:aws:codedeploy:us-west-2:111122223333:application:testApp \
    --tags Key=Name,Value=testName Key=Type,Value=testType
```

Se il comando ha esito positivo, non produce alcun output.

Per ulteriori informazioni, consulta la sezione Tagging delle istanze per i gruppi di distribuzione CodeDeploy nella Guida per l'AWS CodeDeploy utente.

- Per i dettagli sull'API, consulta AWS CLI Command TagResourceReference.

`uninstall`

Il seguente esempio di codice mostra come utilizzare`uninstall`.

AWS CLI

Come disinstallare un'istanza on-premises

L'`uninstall`esempio seguente disinstalla l' AWS CodeDeploy agente dall'istanza locale e rimuove il file di configurazione locale dall'istanza. Non annulla la registrazione dell'istanza AWS CodeDeploy, né dissocia i tag dell'istanza locale AWS CodeDeploy dall'istanza, né elimina l'utente IAM associato all'istanza.

```
aws deploy uninstall
```

Questo comando non produce alcun output.

- Per informazioni dettagliate sull'API, consulta Uninstall in AWS CLI Command Reference.

`untag-resource`

Il seguente esempio di codice mostra come utilizzare. `untag-resource`

AWS CLI

Come rimuovere i tag da una risorsa (applicazione)

L'`untag-resource`esempio seguente rimuove due tag con le chiavi Name e Type da un'applicazione denominata TestApp in CodeDeploy.

```
aws deploy untag-resource \
    --resource-arn  arn:aws:codedeploy:us-west-2:111122223333:application:testApp \
    --tag-keys Name Type
```

Se il comando ha esito positivo, non produce alcun output.

Per ulteriori informazioni, consulta Tagging instances for deployment group CodeDeploy nella Guida per l'AWS CodeDeploy utente.

- Per i dettagli sull'API, consulta AWS CLI Command UntagResourceReference.

`update-application`

Il seguente esempio di codice mostra come utilizzare`update-application`.

AWS CLI

Come modificare i dettagli di un'applicazione

L'`update-application`esempio seguente modifica il nome di un'applicazione associata all' AWS account dell'utente.

```
aws deploy update-application \
    --application-name WordPress_App \
    --new-application-name My_WordPress_App
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta UpdateApplication AWS CLICommand Reference.

`update-deployment-group`

Il seguente esempio di codice mostra come utilizzare`update-deployment-group`.

AWS CLI

Come visualizzare le informazioni su un gruppo di distribuzione

L'esempio `update-deployment-group` seguente modifica le impostazione di un gruppo di distribuzione associato all'applicazione specificata.

```
aws deploy update-deployment-group \
    --application-name WordPress_App \
    --auto-scaling-groups My_CodeDeployDemo_ASG \
    --current-deployment-group-name WordPress_DG \
    --deployment-config-name CodeDeployDefault.AllAtOnce \
```

```
    --ec2-tag-filters Key=Name,Type=KEY_AND_VALUE,Value=My_CodeDeployDemo \
    --new-deployment-group-name My_WordPress_DepGroup \
    --service-role-arn arn:aws:iam::80398EXAMPLE:role/CodeDeployDemo-2
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta UpdateDeploymentGroup AWS CLICommand Reference.

CodeGuru Esempi di revisori che utilizzano AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with CodeGuru Reviewer.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`associate-repository`

Il seguente esempio di codice mostra come utilizzare`associate-repository`.

AWS CLI

Esempio 1: come creare un'associazione di repository Bitbucket

L'esempio `associate-repository` seguente crea un'associazione di repository utilizzando un repository Bitbucket esistente.

```
aws codeguru-reviewer associate-repository \
    --repository 'Bitbucket={Owner=sample-owner, Name=mySampleRepo,
 ConnectionArn=arn:aws:codestar-connections:us-west-2:123456789012:connection/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 }'
```

Output:

```
{
    "RepositoryAssociation": {
        "ProviderType": "Bitbucket",
        "Name": "mySampleRepo",
        "LastUpdatedTimeStamp": 1596216896.979,
        "AssociationId": "association:a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
        "CreatedTimeStamp": 1596216896.979,
        "ConnectionArn": "arn:aws:codestar-connections:us-
west-2:123456789012:connection/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
        "State": "Associating",
        "StateReason": "Pending Repository Association",
        "AssociationArn": "arn:aws:codeguru-reviewer:us-
west-2:123456789012:association:a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
        "Owner": "sample-owner"
    }
}
```

Per ulteriori informazioni, consulta Creare un'associazione di repository Bitbucket in Amazon Reviewer nella Amazon CodeGuru Reviewer User Guide. CodeGuru

Esempio 2: creare un'associazione di repository aziendali GitHub

L'`associate-repository`esempio seguente crea un'associazione di repository utilizzando un repository GitHub Enterprise esistente.

```
aws codeguru-reviewer associate-repository \
    --repository 'GitHubEnterpriseServer={Owner=sample-owner, Name=mySampleRepo,
 ConnectionArn=arn:aws:codestar-connections:us-west-2:123456789012:connection/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 }'
```

Output:

```
{
    "RepositoryAssociation": {
        "ProviderType": "GitHubEnterpriseServer",
        "Name": "mySampleRepo",
        "LastUpdatedTimeStamp": 1596216896.979,
        "AssociationId": "association:a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
        "CreatedTimeStamp": 1596216896.979,
```

```
        "ConnectionArn": "arn:aws:codestar-connections:us-
west-2:123456789012:connection/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
        "State": "Associating",
        "StateReason": "Pending Repository Association",
        "AssociationArn": "arn:aws:codeguru-reviewer:us-
west-2:123456789012:association:a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
        "Owner": "sample-owner"
    }
}
```

Per ulteriori informazioni, consulta Creare un'associazione di repository GitHub Enterprise Server in Amazon CodeGuru Reviewer nella Amazon Codeguru Reviewer User Guide.

Esempio 3: creare un'associazione di repository AWS CodeCommit

L'`associate-repository`esempio seguente crea un'associazione di repository utilizzando un repository esistente AWS CodeCommit .

```
aws codeguru-reviewer associate-repository \
    --repository CodeCommit={Name=mySampleRepo}
```

Output:

```
{
    "RepositoryAssociation": {
        "AssociationId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
        "Name": "My-ecs-beta-repo",
        "LastUpdatedTimeStamp": 1595634764.029,
        "ProviderType": "CodeCommit",
        "CreatedTimeStamp": 1595634764.029,
        "Owner": "544120495673",
        "State": "Associating",
        "StateReason": "Pending Repository Association",
        "AssociationArn": "arn:aws:codeguru-reviewer:us-
west-2:544120495673:association:a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
    }
}
```

Per ulteriori informazioni, consulta Creare un'associazione di AWS CodeCommit repository in Amazon CodeGuru Reviewer nella Amazon CodeGuru Reviewer User Guide.

- Per i dettagli sull'API, consulta Command AssociateRepositoryReference AWS CLI .

`create-code-review`

Il seguente esempio di codice mostra come utilizzare`create-code-review`.

AWS CLI

Come creare una revisione del codice.

Quanto segue `create-code-review` crea una revisione del codice nel `mainline` ramo di un AWS CodeCommit repository denominato`my-repository-name`.

```
aws codeguru-reviewer create-code-review \
    --name my-code-review \
    --repository-association-arn arn:aws:codeguru-reviewer:us-
west-2:123456789012:association:a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 \
    --type '{"RepositoryAnalysis": {"RepositoryHead": {"BranchName": "mainline"}}}'
```

Output:

```
{
    "CodeReview": {
        "Name": "my-code-review",
        "CodeReviewArn": "arn:aws:codeguru-reviewer:us-
west-2:123456789012:association:a1b2c3d4-5678-90ab-cdef-EXAMPLE22222:code-
review:RepositoryAnalysis-my-code-review",
        "RepositoryName": "my-repository-name",
        "Owner": "123456789012",
        "ProviderType": "CodeCommit",
        "State": "Pending",
        "StateReason": "CodeGuru Reviewer has received the request, and a code
 review is scheduled.",
        "CreatedTimeStamp": 1618873489.195,
        "LastUpdatedTimeStamp": 1618873489.195,
        "Type": "RepositoryAnalysis",
        "SourceCodeType": {
            "RepositoryHead": {
                "BranchName": "mainline"
            }
        },
        "AssociationArn": "arn:aws:codeguru-reviewer:us-
west-2:123456789012:association:a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
    }
}
```

Per ulteriori informazioni, consulta Create code review in Amazon CodeGuru Reviewer nella Amazon CodeGuru Reviewer User Guide.

- Per i dettagli sull'API, consulta AWS CLI Command CreateCodeReviewReference.

`describe-code-review`

Il seguente esempio di codice mostra come utilizzare`describe-code-review`.

AWS CLI

Elenca i dettagli relativi alla revisione del codice.

Di seguito `describe-code-review` sono elencate le informazioni su una revisione del codice nel ramo «mainline» di un AWS CodeCommit repository denominato "»my-repo-name.

```
aws codeguru-reviewer put-recommendation-feedback \
    --code-review-arn arn:aws:codeguru-reviewer:us-
west-2:123456789012:association:a1b2c3d4-5678-90ab-cdef-EXAMPLE11111:code-
review:RepositoryAnalysis-my-repository-name-branch-abcdefgh12345678 \
    --recommendation-
id 3be1b2e5d7ef6e298a06499379ee290c9c596cf688fdcadb08285ddb0dd390eb \
    --reactions ThumbsUp
```

Output

```
{
        "CodeReview": {
            "Name": "My-ecs-beta-repo-master-xs6di4kfd4j269dz",
            "CodeReviewArn": "arn:aws:codeguru-reviewer:us-
west-2:123456789012:association:a1b2c3d4-5678-90ab-cdef-EXAMPLE22222:code-
review:RepositoryAnalysis-my-repo-name",
            "RepositoryName": "My-ecs-beta-repo",
            "Owner": "123456789012",
            "ProviderType": "CodeCommit",
            "State": "Pending",
            "StateReason": "CodeGuru Reviewer is reviewing the source code.",
            "CreatedTimeStamp": 1618874226.226,
            "LastUpdatedTimeStamp": 1618874233.689,
            "Type": "RepositoryAnalysis",
            "SourceCodeType": {
                "RepositoryHead": {
```

```
                    "BranchName": "mainline"
                }
            },
            "AssociationArn": "arn:aws:codeguru-reviewer:us-
west-2:123456789012:association:a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
        }
    }
```

Per ulteriori informazioni, consulta Visualizza i dettagli della revisione del codice nella Guida per l'utente di Amazon CodeGuru Reviewer.

- Per i dettagli sull'API, consulta DescribeCodeReview AWS CLICommand Reference.

`describe-recommendation-feedback`

Il seguente esempio di codice mostra come utilizzare`describe-recommendation-feedback`.

AWS CLI

Come visualizzare informazioni sul feedback su una raccomandazione

L'esempio `describe-recommendation-feedback` seguente visualizza le informazioni sul feedback relativo a una raccomandazione. Questa raccomandazione ha una sola reazione ThumbsUp.

```
aws codeguru-reviewer describe-recommendation-feedback \
    --code-review-arn arn:aws:codeguru-reviewer:us-
west-2:123456789012:association:a1b2c3d4-5678-90ab-cdef-EXAMPLE11111:code-
review:RepositoryAnalysis-my-repository-name-branch-abcdefgh12345678 \
    --recommendation-
id 3be1b2e5d7ef6e298a06499379ee290c9c596cf688fdcadb08285ddb0dd390eb
```

Output:

```
{
    "RecommendationFeedback": {
        "CodeReviewArn": "arn:aws:codeguru-reviewer:us-
west-2:123456789012:association:a1b2c3d4-5678-90ab-cdef-EXAMPLE11111:code-
review:RepositoryAnalysis-my-repository-name-branch-abcdefgh12345678",
        "RecommendationId":
 "3be1b2e5d7ef6e298a06499379ee290c9c596cf688fdcadb08285ddb0dd390eb",
        "Reactions": [
```

```
            "ThumbsUp"
        ],
        "UserId": "aws-user-id",
        "CreatedTimeStamp": 1618877070.313,
        "LastUpdatedTimeStamp": 1618877948.881
    }
}
```

Per ulteriori informazioni, consulta Visualizzare consigli e fornire feedback e Passaggio 4: fornire feedback nella Amazon CodeGuru Reviewer User Guide.

- Per i dettagli sull'API, consulta DescribeRecommendationFeedback AWS CLICommand Reference.

`describe-repository-association`

Il seguente esempio di codice mostra come utilizzare`describe-repository-association`.

AWS CLI

Esempio 1: per restituire informazioni su un'associazione di GitHub repository

L'`describe-repository-association`esempio seguente restituisce informazioni su un'associazione di repository che utilizza un repository GitHub Enterprise e si trova nello stato. `Associated`

```
aws codeguru-reviewer describe-repository-association \
    --association-arn arn:aws:codeguru-reviewer:us-
west-2:123456789012:association:a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Output:

```
{
    "RepositoryAssociation": {
        "AssociationId": "b822717e-0711-4e8a-bada-0e738289c75e",
        "Name": "mySampleRepo",
        "LastUpdatedTimeStamp": 1588102637.649,
        "ProviderType": "GitHub",
        "CreatedTimeStamp": 1588102615.636,
        "Owner": "sample-owner",
        "State": "Associated",
        "StateReason": "Pull Request Notification configuration successful",
```

```
        "AssociationArn": "arn:aws:codeguru-reviewer:us-
west-2:123456789012:association:a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
    }
}
```

Per ulteriori informazioni, consulta Creare un'associazione di repository GitHub Enterprise Server in Amazon CodeGuru Reviewer nella Amazon CodeGuru Reviewer User Guide.

Esempio 2: come restituire informazioni su un'associazione di repository non riuscita

L'`describe-repository-association`esempio seguente restituisce informazioni su un'associazione di repository che utilizza un repository GitHub Enterprise e si trova nello stato. `Failed`

```
aws codeguru-reviewer describe-repository-association \
    --association-arn arn:aws:codeguru-reviewer:us-
west-2:123456789012:association:a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Output:

```
{
    "RepositoryAssociation": {
        "ProviderType": "GitHubEnterpriseServer",
        "Name": "mySampleRepo",
        "LastUpdatedTimeStamp": 1596217036.892,
        "AssociationId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
        "CreatedTimeStamp": 1596216896.979,
        "ConnectionArn": "arn:aws:codestar-connections:us-
west-2:123456789012:connection/a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
        "State": "Failed",
        "StateReason": "Failed, Please retry.",
        "AssociationArn": "arn:aws:codeguru-reviewer:us-
west-2:123456789012:association:a1b2c3d4-5678-90ab-cdef-EXAMPLE33333",
        "Owner": "sample-owner"
    }
}
```

Per ulteriori informazioni, consulta Creare un'associazione di repository GitHub Enterprise Server in Amazon CodeGuru Reviewer nella Amazon CodeGuru Reviewer User Guide.

Esempio 3: come restituire informazioni su un'associazione di repository dissociata

L'`describe-repository-association`esempio seguente restituisce informazioni su un'associazione di repository che utilizza un repository GitHub Enterprise e si trova nello stato. `Disassociating`

```
aws codeguru-reviewer describe-repository-association \
    --association-arn arn:aws:codeguru-reviewer:us-
west-2:123456789012:association:a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Output:

```
{
    "RepositoryAssociation": {
        "ProviderType": "GitHubEnterpriseServer",
        "Name": "mySampleRepo",
        "LastUpdatedTimeStamp": 1596217036.892,
        "AssociationId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
        "CreatedTimeStamp": 1596216896.979,
        "ConnectionArn": "arn:aws:codestar-connections:us-
west-2:123456789012:connection/a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
        "State": "Disassociating",
        "StateReason": "Source code access removal in progress",
        "AssociationArn": "arn:aws:codeguru-reviewer:us-
west-2:123456789012:association:a1b2c3d4-5678-90ab-cdef-EXAMPLE33333",
        "Owner": "sample-owner"
    }
}
```

Per ulteriori informazioni, consulta Creare un'associazione di repository GitHub Enterprise Server in Amazon CodeGuru Reviewer nella Amazon CodeGuru Reviewer User Guide.

- Per i dettagli sull'API, consulta Command DescribeRepositoryAssociationReference AWS CLI .

`disassociate-repository`

Il seguente esempio di codice mostra come utilizzare`disassociate-repository`.

AWS CLI

Come annullare un'associazione di repository

Di seguito `disassociate-repository` viene dissociata un'associazione di repository che utilizza un repository. AWS CodeCommit

```
aws codeguru-reviewer disassociate-repository \
    --association-arn arn:aws:codeguru-reviewer:us-
west-2:123456789012:association:a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Output:

```
{
    "RepositoryAssociation": {
        "AssociationId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
        "AssociationArn": "arn:aws:codeguru-reviewer:us-
west-2:123456789012:association:a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
        "Name": "my-repository",
        "Owner": "123456789012",
        "ProviderType": "CodeCommit",
        "State": "Disassociating",
        "LastUpdatedTimeStamp": 1618939174.759,
        "CreatedTimeStamp": 1595636947.096
    },
    "Tags": {
        "Status": "Secret",
        "Team": "Saanvi"
    }
}
```

Per ulteriori informazioni, consulta Dissociare un repository in CodeGuru Reviewer nella Amazon CodeGuru Reviewer User Guide.

- Per i dettagli sull'API, consulta DisassociateRepositoryCommand Reference.AWS CLI

`list-code-reviews`

Il seguente esempio di codice mostra come utilizzare`list-code-reviews`.

AWS CLI

Per elencare le recensioni di codice create nel tuo AWS account negli ultimi 90 giorni.

L'esempio `list-code-reviews` seguente elenca le revisioni del codice create negli ultimi 90 giorni utilizzando richieste pull.

```
aws codeguru-reviewer list-code-reviews \
    --type PullRequest
```

Output:


```
{
    "CodeReviewSummaries": [
        {
            "LastUpdatedTimeStamp": 1588897288.054,
            "Name": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
            "ProviderType": "GitHub",
            "PullRequestId": "5",
            "MetricsSummary": {
                "MeteredLinesOfCodeCount": 24,
                "FindingsCount": 1
            },
            "CreatedTimeStamp": 1588897068.512,
            "State": "Completed",
            "CodeReviewArn": "arn:aws:codeguru-reviewer:us-west-2:123456789012:code-
review:a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
            "Owner": "sample-owner",
            "RepositoryName": "sample-repository-name",
            "Type": "PullRequest"
        },
        {
            "LastUpdatedTimeStamp": 1588869793.263,
            "Name": "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
            "ProviderType": "GitHub",
            "PullRequestId": "4",
            "MetricsSummary": {
                "MeteredLinesOfCodeCount": 29,
                "FindingsCount": 0
            },
            "CreatedTimeStamp": 1588869575.949,
            "State": "Completed",
            "CodeReviewArn": "arn:aws:codeguru-reviewer:us-west-2:123456789012:code-
review:a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
            "Owner": "sample-owner",
            "RepositoryName": "sample-repository-name",
            "Type": "PullRequest"
        },
        {
            "LastUpdatedTimeStamp": 1588870511.211,
            "Name": "a1b2c3d4-5678-90ab-cdef-EXAMPLE33333",
            "ProviderType": "GitHub",
            "PullRequestId": "4",
            "MetricsSummary": {
```

```
                "MeteredLinesOfCodeCount": 2,
                "FindingsCount": 0
            },
            "CreatedTimeStamp": 1588870292.425,
            "State": "Completed",
            "CodeReviewArn": "arn:aws:codeguru-reviewer:us-west-2:123456789012:code-
review:a1b2c3d4-5678-90ab-cdef-EXAMPLE33333",
            "Owner": "sample-owner",
            "RepositoryName": "sample-repository-name",
            "Type": "PullRequest"
        },
        {
            "LastUpdatedTimeStamp": 1588118522.452,
            "Name": "a1b2c3d4-5678-90ab-cdef-EXAMPLE44444",
            "ProviderType": "GitHub",
            "PullRequestId": "3",
            "MetricsSummary": {
                "MeteredLinesOfCodeCount": 29,
                "FindingsCount": 0
            },
            "CreatedTimeStamp": 1588118301.131,
            "State": "Completed",
            "CodeReviewArn": "arn:aws:codeguru-reviewer:us-west-2:123456789012:code-
review:a1b2c3d4-5678-90ab-cdef-EXAMPLE44444",
            "Owner": "sample-owner",
            "RepositoryName": "sample-repository-name",
            "Type": "PullRequest"
        },
        {
            "LastUpdatedTimeStamp": 1588112205.207,
            "Name": "a1b2c3d4-5678-90ab-cdef-EXAMPLE55555",
            "ProviderType": "GitHub",
            "PullRequestId": "2",
            "MetricsSummary": {
                "MeteredLinesOfCodeCount": 25,
                "FindingsCount": 0
            },
            "CreatedTimeStamp": 1588111987.443,
            "State": "Completed",
            "CodeReviewArn": "arn:aws:codeguru-reviewer:us-west-2:123456789012:code-
review:a1b2c3d4-5678-90ab-cdef-EXAMPLE55555",
            "Owner": "sample-owner",
            "RepositoryName": "sample-repository-name",
            "Type": "PullRequest"
```

```
        },
        {
            "LastUpdatedTimeStamp": 1588104489.981,
            "Name": "a1b2c3d4-5678-90ab-cdef-EXAMPLE66666",
            "ProviderType": "GitHub",
            "PullRequestId": "1",
            "MetricsSummary": {
                "MeteredLinesOfCodeCount": 25,
                "FindingsCount": 0
            },
            "CreatedTimeStamp": 1588104270.223,
            "State": "Completed",
            "CodeReviewArn": "arn:aws:codeguru-reviewer:us-west-2:123456789012:code-
review:a1b2c3d4-5678-90ab-cdef-EXAMPLE66666",
            "Owner": "sample-owner",
            "RepositoryName": "sample-repository-name",
            "Type": "PullRequest"
        }
    ]
}
```

Per ulteriori informazioni, consulta Visualizza tutte le revisioni del codice nella Amazon CodeGuru Reviewer User Guide.

- Per i dettagli sull'API, consulta ListCodeReviews AWS CLICommand Reference.

`list-recommendation-feedback`

Il seguente esempio di codice mostra come utilizzare`list-recommendation-feedback`.

AWS CLI

Come elencare il feedback dei clienti relativo a una raccomandazione su un repository associato

L'esempio `list-recommendation-feedback` seguente elenca i feedback dei clienti per tutte le raccomandazioni relative alla revisione del codice. Questa revisione del codice contiene un feedback, un "ThumbsUp«, da parte di un cliente.

```
aws codeguru-reviewer list-recommendation-feedback \
    --code-review-arn arn:aws:codeguru-reviewer:us-
west-2:123456789012:association:a1b2c3d4-5678-90ab-cdef-EXAMPLE11111:code-
review:RepositoryAnalysis-my-repository-name-branch-abcdefgh12345678
```

Output:

```
{
    "RecommendationFeedbackSummaries": [
        {
            "RecommendationId":
 "3be1b2e5d7ef6e298a06499379ee290c9c596cf688fdcadb08285ddb0dd390eb",
            "Reactions": [
                "ThumbsUp"
            ],
            "UserId": "aws-user-id"
        }
    ]
}
```

Per ulteriori informazioni, consulta la Fase 4: Fornire feedback nella Guida per l'utente di Amazon CodeGuru Reviewer.

- Per i dettagli sull'API, consulta ListRecommendationFeedback AWS CLICommand Reference.

`list-recommendations`

Il seguente esempio di codice mostra come utilizzare`list-recommendations`.

AWS CLI

Come elencare le raccomandazioni per completare la revisione del codice

L'esempio `list-recommendations` seguente elenca le raccomandazioni per una revisione del codice completata. Questa revisione del codice contiene una raccomandazione.

```
aws codeguru-reviewer list-recommendations \
    --code-review-arn arn:aws:codeguru-reviewer:us-west-2:544120495673:code-
review:a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Output:

```
{
    "RecommendationSummaries": [
        {
            "Description": "\n\n**Problem**  \n You are using a `ConcurrentHashMap`,
 but your usage of `containsKey()` and `get()` may not be thread-safe at lines: **63
```

```
 and 64**. In between the check and the `get()` another thread can remove the key
 and the `get()` will return `null`. The remove that can remove the key is at line:
 **59**.\n\n**Fix**  \n Consider calling `get()`, checking instead of your current
 check if the returned object is `null`, and then using that object only, without
 calling `get()` again.\n\n**More info**  \n [View an example on GitHub](https://
github.com/apache/hadoop/blob/f16cf877e565084c66bc63605659b157c4394dc8/hadoop-tools/
hadoop-aws/src/main/java/org/apache/hadoop/fs/s3a/s3guard/S3Guard.java#L302-L304)
 (external link).",
            "RecommendationId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
            "StartLine": 63,
            "EndLine": 64,
            "FilePath": "src/main/java/com/company/sample/application/
CreateOrderThread.java"
        }
    ]
}
```

Per ulteriori informazioni, consulta la Fase 4: Fornire feedback nella Guida per l'utente di Amazon CodeGuru Reviewer.

- Per i dettagli sull'API, consulta ListRecommendations AWS CLICommand Reference.

`list-repository-associations`

Il seguente esempio di codice mostra come utilizzare`list-repository-associations`.

AWS CLI

Per elencare le associazioni di repository presenti nel tuo account AWS

L'esempio `list-repository-associations` seguente restituisce un elenco di oggetti di riepilogo dell'associazione di repository nell'account in uso. È possibile filtrare l'elenco restituito per `ProviderType`, `Name`, `State` e `Owner`.

```
aws codeguru-reviewer list-repository-associations
```

Output:

```
{
    "RepositoryAssociationSummaries": [
        {
            "LastUpdatedTimeStamp": 1595886609.616,
            "Name": "test",
```

```
            "AssociationId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
            "Owner": "sample-owner",
            "State": "Associated",
            "AssociationArn": "arn:aws:codeguru-reviewer:us-
west-2:123456789012:association:a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
            "ProviderType": "Bitbucket"
        },
        {
            "LastUpdatedTimeStamp": 1595636969.035,
            "Name": "CodeDeploy-CodePipeline-ECS-Tutorial",
            "AssociationId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
            "Owner": "123456789012",
            "State": "Associated",
            "AssociationArn": "arn:aws:codeguru-reviewer:us-
west-2:123456789012:association:a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
            "ProviderType": "CodeCommit"
        },
        {
            "LastUpdatedTimeStamp": 1595634785.983,
            "Name": "My-ecs-beta-repo",
            "AssociationId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE33333",
            "Owner": "123456789012",
            "State": "Associated",
            "AssociationArn": "arn:aws:codeguru-reviewer:us-
west-2:123456789012:association:a1b2c3d4-5678-90ab-cdef-EXAMPLE33333",
            "ProviderType": "CodeCommit"
        },
        {
            "LastUpdatedTimeStamp": 1590712811.77,
            "Name": "MyTestCodeCommit",
            "AssociationId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE44444",
            "Owner": "123456789012",
            "State": "Associated",
            "AssociationArn": "arn:aws:codeguru-reviewer:us-
west-2:123456789012:association:a1b2c3d4-5678-90ab-cdef-EXAMPLE44444",
            "ProviderType": "CodeCommit"
        },
        {
            "LastUpdatedTimeStamp": 1588102637.649,
            "Name": "aws-codeguru-profiler-sample-application",
            "AssociationId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE55555",
            "Owner": "sample-owner",
            "State": "Associated",
```

```
            "AssociationArn": "arn:aws:codeguru-reviewer:us-
west-2:123456789012:association:a1b2c3d4-5678-90ab-cdef-EXAMPLE55555",
            "ProviderType": "GitHub"
        },
        {
            "LastUpdatedTimeStamp": 1588028233.995,
            "Name": "codeguru-profiler-demo-app",
            "AssociationId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE66666",
            "Owner": "sample-owner",
            "State": "Associated",
            "AssociationArn": "arn:aws:codeguru-reviewer:us-
west-2:123456789012:association:a1b2c3d4-5678-90ab-cdef-EXAMPLE66666",
            "ProviderType": "GitHub"
        }
    ]
}
```

Per ulteriori informazioni, consulta Visualizza tutte le associazioni di repository in CodeGuru Reviewer nella Amazon CodeGuru Reviewer User Guide.

- Per i dettagli sull'API, consulta Command ListRepositoryAssociationsReference AWS CLI .

`list-tags-for-resource`

Il seguente esempio di codice mostra come utilizzare`list-tags-for-resource`.

AWS CLI

Come elencare i tag in un repository associato

L'esempio `list-tags-for-resource` seguente elenca i tag in un repository associato. Questo repository associato ha due tag.

```
aws codeguru-reviewer list-tags-for-resource \
    --resource-arn arn:aws:codeguru-reviewer:us-
west-2:123456789012:association:a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Output:

```
{
    "Tags": {
        "Status": "Secret",
```

```
        "Team": "Saanvi"
    }
}
```

Per ulteriori informazioni, consulta Visualizza tag per un repository associato a CodeGuru Reviewer (AWS CLI) nella Amazon CodeGuru Reviewer User Guide.

- Per i dettagli sull'API, consulta ListTagsForResourceCommand Reference.AWS CLI

`put-recommendation-feedback`

Il seguente esempio di codice mostra come utilizzare`put-recommendation-feedback`.

AWS CLI

Come aggiungere una raccomandazione a una revisione del codice

L'esempio `put-recommendation-feedback` seguente fornisce una raccomandazione ThumbsUp per una revisione del codice.

```
aws codeguru-reviewer put-recommendation-feedback \
    --code-review-arn \arn:aws:codeguru-reviewer:us-
west-2:123456789012:association:a1b2c3d4-5678-90ab-cdef-EXAMPLE11111:code-
review:RepositoryAnalysis-my-repository-name-branch-abcdefgh12345678 \
    --recommendation-
id 3be1b2e5d7ef6e298a06499379ee290c9c596cf688fdcadb08285ddb0dd390eb \
    --reactions ThumbsUp
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta la Fase 4: Fornire feedback nella Guida per l'utente di Amazon CodeGuru Reviewer.

- Per i dettagli sull'API, consulta PutRecommendationFeedback AWS CLICommand Reference.

`tag-resource`

Il seguente esempio di codice mostra come utilizzare`tag-resource`.

AWS CLI

Come aggiungere un tag a un repository associato

L'esempio `tag-resource` seguente aggiunge due tag a un repository associato.

```
aws codeguru-reviewer tag-resource \
    --resource-arn arn:aws:codeguru-reviewer:us-
west-2:123456789012:association:a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 \
    --tags Status=Secret,Team=Saanvi
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Aggiungere un tag a un repository associato a CodeGuru Reviewer (AWS CLI) e Aggiungere o aggiornare tag per un repository CodeGuru associato a Reviewer (AWS CLI) nella Amazon Reviewer User Guide. CodeGuru

- Per i dettagli sull'API, consulta Command Reference. TagResourceAWS CLI

`untag-resource`

Il seguente esempio di codice mostra come utilizzare`untag-resource`.

AWS CLI

Come rimuovere i tag da un repository associato

L'esempio `untag-resource` seguente rimuove due tag con le chiavi "Secret" e "Team" da un repository associato.

```
aws codeguru-reviewer untag-resource \
    --resource-arn arn:aws:codeguru-reviewer:us-
west-2:123456789012:association:a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 \
    --tag-keys Status Team
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Rimuovere i tag da un repository associato a CodeGuru Reviewer (AWS CLI) nella Amazon CodeGuru Reviewer User Guide.

- Per i dettagli sull'API, consulta UntagResourceCommand Reference.AWS CLI

CodePipeline esempi utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with CodePipeline.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`acknowledge-job`

Il seguente esempio di codice mostra come utilizzare`acknowledge-job`.

AWS CLI

Come recuperare le informazioni relative a un processo specificato

Questo esempio restituisce le informazioni su un processo specificato, incluso il relativo stato, se esistente. Viene utilizzato solo per i worker del processo e le azioni personalizzate. Per determinare il valore di nonce e l'ID del lavoro, usa aws poll-for-jobs codepipeline.

Comando:

```
aws codepipeline acknowledge-job --job-id f4f4ff82-2d11-EXAMPLE --nonce 3
```

Output:

```
{
  "status": "InProgress"
}
```

- Per i dettagli sull'API, consulta Command AcknowledgeJobReference AWS CLI .

`create-custom-action-type`

Il seguente esempio di codice mostra come utilizzare`create-custom-action-type`.

AWS CLI

Come creare un'azione personalizzata

Questo esempio crea un'azione personalizzata per AWS CodePipeline l'utilizzo di un file JSON già creato (qui denominato MyCustomAction .json) che contiene la struttura dell'azione personalizzata. Per ulteriori informazioni sui requisiti per la creazione di un'azione personalizzata, inclusa la struttura del file, consulta la Guida per l'utente. AWS CodePipeline

```
aws codepipeline create-custom-action-type --cli-input-json file://
MyCustomAction.json
```

Contenuto del file `MyCustomAction.json`:

```
{
    "category": "Build",
    "provider": "MyJenkinsProviderName",
    "version": "1",
    "settings": {
        "entityUrlTemplate": "https://192.0.2.4/job/{Config:ProjectName}/",
        "executionUrlTemplate": "https://192.0.2.4/job/{Config:ProjectName}/
lastSuccessfulBuild/{ExternalExecutionId}/"
    },
    "configurationProperties": [
        {
            "name": "MyJenkinsExampleBuildProject",
            "required": true,
            "key": true,
            "secret": false,
            "queryable": false,
            "description": "The name of the build project must be provided when this
 action is added to the pipeline.",
            "type": "String"
        }
    ],
    "inputArtifactDetails": {
        "maximumCount": 1,
        "minimumCount": 0
    },
    "outputArtifactDetails": {
        "maximumCount": 1,
        "minimumCount": 0
    }
```

```
}
```

Questo comando restituisce la struttura dell'azione personalizzata.

- Per i dettagli sull'API, consulta CreateCustomActionType AWS CLICommand Reference.

`create-pipeline`

Il seguente esempio di codice mostra come utilizzare`create-pipeline`.

AWS CLI

Come creare una pipeline

Questo esempio crea una pipeline AWS CodePipeline utilizzando un file JSON già creato (qui denominato MySecondPipeline .json) che contiene la struttura della pipeline. Per ulteriori informazioni sui requisiti per la creazione di una pipeline, inclusa la struttura del file, consulta la Guida per l'utente. AWS CodePipeline

Comando:

```
aws codepipeline create-pipeline --cli-input-json file://MySecondPipeline.json
```

Contenuto del file JSON di esempio:

```
{
 "pipeline": {
  "roleArn": "arn:aws:iam::111111111111:role/AWS-CodePipeline-Service",
  "stages": [
    {
      "name": "Source",
      "actions": [
        {
          "inputArtifacts": [],
          "name": "Source",
          "actionTypeId": {
            "category": "Source",
            "owner": "AWS",
            "version": "1",
            "provider": "S3"
          },
          "outputArtifacts": [
            {
```

```
            "name": "MyApp"
          }
        ],
        "configuration": {
          "S3Bucket": "awscodepipeline-demo-bucket",
          "S3ObjectKey": "aws-codepipeline-s3-aws-codedeploy_linux.zip"
        },
        "runOrder": 1
      }
    ]
  },
  {
    "name": "Beta",
    "actions": [
      {
        "inputArtifacts": [
          {
            "name": "MyApp"
          }
        ],
        "name": "CodePipelineDemoFleet",
        "actionTypeId": {
          "category": "Deploy",
          "owner": "AWS",
          "version": "1",
          "provider": "CodeDeploy"
        },
        "outputArtifacts": [],
        "configuration": {
          "ApplicationName": "CodePipelineDemoApplication",
          "DeploymentGroupName": "CodePipelineDemoFleet"
        },
        "runOrder": 1
      }
    ]
  }
 ],
 "artifactStore": {
   "type": "S3",
   "location": "codepipeline-us-east-1-11EXAMPLE11"
 },
 "name": "MySecondPipeline",
 "version": 1
}
```

```
}
```

Output:

```
This command returns the structure of the pipeline.
```

- Per i dettagli sull'API, consulta CreatePipeline AWS CLICommand Reference.

`delete-custom-action-type`

Il seguente esempio di codice mostra come utilizzare`delete-custom-action-type`.

AWS CLI

Come eliminare un'azione personalizzata

Questo esempio elimina un'azione personalizzata AWS CodePipeline utilizzando un file JSON già creato (qui denominato DeleteMyCustomAction .json) che contiene il tipo di azione, il nome del provider e il numero di versione dell'azione da eliminare. Utilizzate il list-action-types comando per visualizzare i valori corretti per categoria, versione e provider.

Comando:

```
aws codepipeline delete-custom-action-type --cli-input-json file://
DeleteMyCustomAction.json
```

Contenuto del file JSON di esempio:

```
{
  "category": "Build",
  "version": "1",
  "provider": "MyJenkinsProviderName"
}
```

Output:

```
None.
```

- Per i dettagli sull'API, consulta DeleteCustomActionType AWS CLICommand Reference.

`delete-pipeline`

Il seguente esempio di codice mostra come utilizzare`delete-pipeline`.

AWS CLI

Come eliminare una pipeline

Questo esempio elimina una pipeline denominata MySecondPipeline from. AWS CodePipeline Utilizzate il comando list-pipelines per visualizzare un elenco di pipeline associate al vostro account. AWS

Comando:

```
aws codepipeline delete-pipeline --name MySecondPipeline
```

Output:

```
None.
```

- Per i dettagli sull'API, consulta DeletePipelineCommand Reference.AWS CLI

`delete-webhook`

Il seguente esempio di codice mostra come utilizzare`delete-webhook`.

AWS CLI

Come eliminare un webhook

L'`delete-webhook`esempio seguente elimina un webhook per un'azione di origine della GitHub versione 1. Prima di eliminare il webhook, devi annullarne la registrazione utilizzando il comando `deregister-webhook-with-third-party`.

```
aws codepipeline delete-webhook \
    --name my-webhook
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Eliminare il webhook dalla GitHub fonte nella Guida per l'AWS CodePipeline utente.

- Per i dettagli sull'API, consulta DeleteWebhook AWS CLICommand Reference.

`deregister-webhook-with-third-party`

Il seguente esempio di codice mostra come utilizzare`deregister-webhook-with-third-party`.

AWS CLI

Come annullare la registrazione di un webhook

L'`deregister-webhook-with-third-party`esempio seguente elimina un webhook per un'azione di origine della GitHub versione 1. Prima di eliminare il webhook, devi annullarne la registrazione.

```
aws codepipeline deregister-webhook-with-third-party \
    --webhook-name my-webhook
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Eliminare il webhook dalla GitHub fonte nella Guida per l'AWS CodePipeline utente.

- Per i dettagli sull'API, consulta DeregisterWebhookWithThirdParty AWS CLICommand Reference.

`disable-stage-transition`

Il seguente esempio di codice mostra come utilizzare`disable-stage-transition`.

AWS CLI

Come disabilitare una transizione verso una fase in una pipeline

Questo esempio disabilita le transizioni alla fase Beta della MyFirstPipeline pipeline in. AWS CodePipeline

Comando:

```
aws codepipeline disable-stage-transition --pipeline-name MyFirstPipeline --stage-
name Beta  --transition-type Inbound
```

Output:

```
None.
```

- Per i dettagli sull'API, consulta Command DisableStageTransitionReference AWS CLI .

`enable-stage-transition`

Il seguente esempio di codice mostra come utilizzare`enable-stage-transition`.

AWS CLI

Come abilitare una transizione verso una fase in una pipeline

Questo esempio abilita le transizioni alla fase Beta della MyFirstPipeline pipeline in. AWS CodePipeline

Comando:

```
aws codepipeline enable-stage-transition --pipeline-name MyFirstPipeline --stage-name Beta  --transition-type Inbound
```

Output:

```
None.
```

- Per i dettagli sull'API, consulta AWS CLI Command EnableStageTransitionReference.

`get-job-details`

Il seguente esempio di codice mostra come utilizzare`get-job-details`.

AWS CLI

Come ottenere i dettagli di un processo

Questo esempio restituisce i dettagli su un processo il cui ID è rappresentato da f4f4ff82-2d11-EXAMPLE. Questo comando viene utilizzato solo per le azioni personalizzate. Quando viene chiamato questo comando, AWS CodePipeline restituisce le credenziali temporanee per il bucket Amazon S3 utilizzato per archiviare gli artefatti per la pipeline, se necessario per l'azione personalizzata. Questo comando restituirà anche tutti i valori del segreto definiti per l'azione, se definiti.

Comando:

```
aws codepipeline get-job-details --job-id f4f4ff82-2d11-EXAMPLE
```

Output:

```
{
 "jobDetails": {
  "accountId": "111111111111",
  "data": {
    "actionConfiguration": {
      "__type": "ActionConfiguration",
      "configuration": {
        "ProjectName": "MyJenkinsExampleTestProject"
      }
    },
    "actionTypeId": {
      "__type": "ActionTypeId",
      "category": "Test",
      "owner": "Custom",
      "provider": "MyJenkinsProviderName",
      "version": "1"
    },
    "artifactCredentials": {
      "__type": "AWSSessionCredentials",
      "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
      "secretAccessKey": "wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY",
      "sessionToken":
 "fICCQD6m7oRw0uXOjANBgkqhkiG9w0BAQUFADCBiDELMAkGA1UEBhMCVVMxCzAJBgNVBAgTAldBMRAwDgYDVQQHEwd
+a4GmWIWJ21uUSfwfEvySWtC2XADZ4nB+BLYgVIk60CpiwsZ3G93vUEIO3IyNoH/
f0wYK8m9TrDHudUZg3qX4waLG5M43q7Wgc/
MbQITxOUSQv7c7ugFFDzQGBzZswY6786m86gpEIbb3OhjZnzcvQAaRHhdlQWIMm2nrAgMBAAEwDQYJKoZIhvcNAQEFBQ
+auNKyExzyLwaxlAoo7TJHidbtS4J5iNmZgXL0FkbFFBjvSfpJIlJ00zbhNYS5f6GuoEDmFJl0ZxBHjJnyp378OD8uTs
    },
    "inputArtifacts": [
      {
        "__type": "Artifact",
        "location": {
          "s3Location": {
            "bucketName": "codepipeline-us-east-1-11EXAMPLE11",
            "objectKey": "MySecondPipeline/MyAppBuild/EXAMPLE"
          },
          "type": "S3"
```

```
        },
        "name": "MyAppBuild"
      }
    ],
    "outputArtifacts": [],
    "pipelineContext": {
      "__type": "PipelineContext",
      "action": {
        "name": "MyJenkinsTest-Action"
      },
      "pipelineName": "MySecondPipeline",
      "stage": {
        "name": "Testing"
      }
    }
  },
  "id": "f4f4ff82-2d11-EXAMPLE"
 }
}
```

- Per i dettagli sull'API, consulta Command Reference. GetJobDetailsAWS CLI

`get-pipeline-state`

Il seguente esempio di codice mostra come utilizzare`get-pipeline-state`.

AWS CLI

Come ottenere informazioni sullo stato di una pipeline

Questo esempio restituisce lo stato più recente di una pipeline denominata MyFirstPipeline.

Comando:

```
aws codepipeline get-pipeline-state --name MyFirstPipeline
```

Output:

```
{
 "created": 1446137312.204,
 "pipelineName": "MyFirstPipeline",
 "pipelineVersion": 1,
 "stageStates": [
```

```
  {
    "actionStates": [
      {
        "actionName": "Source",
        "entityUrl": "https://console.aws.amazon.com/s3/home?#",
        "latestExecution": {
          "lastStatusChange": 1446137358.328,
          "status": "Succeeded"
        }
      }
    ],
    "stageName": "Source"
  },
  {
    "actionStates": [
      {
        "actionName": "CodePipelineDemoFleet",
        "entityUrl": "https://console.aws.amazon.com/codedeploy/home?#/applications/
CodePipelineDemoApplication/deployment-groups/CodePipelineDemoFleet",
        "latestExecution": {
          "externalExecutionId": "d-EXAMPLE",
          "externalExecutionUrl": "https://console.aws.amazon.com/codedeploy/home?#/
deployments/d-EXAMPLE",
          "lastStatusChange": 1446137493.131,
          "status": "Succeeded",
          "summary": "Deployment Succeeded"
        }
      }
    ],
    "inboundTransitionState": {
      "enabled": true
    },
    "stageName": "Beta"
  }
 ],
 "updated": 1446137312.204
}
```

- Per i dettagli sull'API, vedere GetPipelineStatein AWS CLI Command Reference.

`get-pipeline`

Il seguente esempio di codice mostra come utilizzare`get-pipeline`.

AWS CLI

Come visualizzare la struttura di una pipeline

Questo esempio restituisce la struttura di una pipeline denominata MyFirstPipeline.

Comando:

```
aws codepipeline get-pipeline --name MyFirstPipeline
```

Output:

```
{
  "pipeline": {
      "roleArn": "arn:aws:iam::111111111111:role/AWS-CodePipeline-Service",
      "stages": [
          {
              "name": "Source",
              "actions": [
                  {
                      "inputArtifacts": [],
                      "name": "Source",
                      "actionTypeId": {
                          "category": "Source",
                          "owner": "AWS",
                          "version": "1",
                          "provider": "S3"
                      },
                      "outputArtifacts": [
                          {
                              "name": "MyApp"
                          }
                      ],
                      "configuration": {
                          "S3Bucket": "awscodepipeline-demo-bucket",
                          "S3ObjectKey": "aws-codepipeline-s3-aws-
codedeploy_linux.zip"
                      },
                      "runOrder": 1
                  }
              ]
          },
          {
```

```
            "name": "Beta",
            "actions": [
                {
                    "inputArtifacts": [
                        {
                            "name": "MyApp"
                        }
                    ],
                    "name": "CodePipelineDemoFleet",
                    "actionTypeId": {
                        "category": "Deploy",
                        "owner": "AWS",
                        "version": "1",
                        "provider": "CodeDeploy"
                    },
                    "outputArtifacts": [],
                    "configuration": {
                        "ApplicationName": "CodePipelineDemoApplication",
                        "DeploymentGroupName": "CodePipelineDemoFleet"
                    },
                    "runOrder": 1
                }
            ]
        }
    ],
    "artifactStore": {
        "type": "S3",
        "location": "codepipeline-us-east-1-11EXAMPLE11"
    },
    "name": "MyFirstPipeline",
    "version": 1
  }
}
```

- Per i dettagli sull'API, vedere GetPipelinein AWS CLI Command Reference.

`list-action-executions`

Il seguente esempio di codice mostra come utilizzare`list-action-executions`.

AWS CLI

Come elencare le esecuzioni di azioni

L'esempio `list-action-executions` seguente visualizza i dettagli di esecuzione di un'azione per una pipeline, ad esempio l'ID di esecuzione dell'azione, gli artefatti di input, gli artefatti di output, i risultati dell'esecuzione e lo stato.

```
aws codepipeline list-action-executions \
    --pipeline-name myPipeline
```

Output:

```
{
    "actionExecutionDetails": [
        {
            "pipelineExecutionId": "EXAMPLE0-adfc-488e-bf4c-1111111720d3",
            "actionExecutionId": "EXAMPLE4-2ee8-4853-bd6a-111111158148",
            "pipelineVersion": 12,
            "stageName": "Deploy",
            "actionName": "Deploy",
            "startTime": 1598572628.6,
            "lastUpdateTime": 1598572661.255,
            "status": "Succeeded",
            "input": {
                "actionTypeId": {
                    "category": "Deploy",
                    "owner": "AWS",
                    "provider": "CodeDeploy",
                    "version": "1"
                },
                "configuration": {
                    "ApplicationName": "my-application",
                    "DeploymentGroupName": "my-deployment-group"
                },
                "resolvedConfiguration": {
                    "ApplicationName": "my-application",
                    "DeploymentGroupName": "my-deployment-group"
                },
                "region": "us-east-1",
                "inputArtifacts": [
                    {
                        "name": "SourceArtifact",
                        "s3location": {
                            "bucket": "artifact-bucket",
                            "key": "myPipeline/SourceArti/key"
                        }
```

```
                }
            ],
            "namespace": "DeployVariables"
        },
        "output": {
            "outputArtifacts": [],
            "executionResult": {
                "externalExecutionId": "d-EXAMPLEE5",
                "externalExecutionSummary": "Deployment Succeeded",
                "externalExecutionUrl": "https://myaddress.com"
            },
            "outputVariables": {}
        }
    },
    {
        "pipelineExecutionId": "EXAMPLE0-adfc-488e-bf4c-1111111720d3",
        "actionExecutionId": "EXAMPLE5-abb4-4192-9031-11111113a7b0",
        "pipelineVersion": 12,
        "stageName": "Source",
        "actionName": "Source",
        "startTime": 1598572624.387,
        "lastUpdateTime": 1598572628.16,
        "status": "Succeeded",
        "input": {
            "actionTypeId": {
                "category": "Source",
                "owner": "AWS",
                "provider": "CodeCommit",
                "version": "1"
            },
            "configuration": {
                "BranchName": "production",
                "PollForSourceChanges": "false",
                "RepositoryName": "my-repo"
            },
            "resolvedConfiguration": {
                "BranchName": "production",
                "PollForSourceChanges": "false",
                "RepositoryName": "my-repo"
            },
            "region": "us-east-1",
            "inputArtifacts": [],
            "namespace": "SourceVariables"
        },
```

```
            "output": {
                "outputArtifacts": [
                    {
                        "name": "SourceArtifact",
                        "s3location": {
                            "bucket": "amzn-s3-demo-bucket",
                            "key": "myPipeline/SourceArti/key"
                        }
                    }
                ],
                "executionResult": {
                    "externalExecutionId":
 "1111111ad99dcd35914c00b7fbea13995EXAMPLE",
                    "externalExecutionSummary": "Edited template.yml",
                    "externalExecutionUrl": "https://myaddress.com"
                },
                "outputVariables": {
                    "AuthorDate": "2020-05-08T17:45:43Z",
                    "BranchName": "production",
                    "CommitId": "EXAMPLEad99dcd35914c00b7fbea139951111111",
                    "CommitMessage": "Edited template.yml",
                    "CommitterDate": "2020-05-08T17:45:43Z",
                    "RepositoryName": "my-repo"
                }
            }
        },
. . . .
```

Per ulteriori informazioni, consulta Visualizza esecuzioni di azioni (CLI) nella Guida per AWS CodePipeline l'utente.

- Per i dettagli sull'API, consulta AWS CLI Command ListActionExecutionsReference.

`list-action-types`

Il seguente esempio di codice mostra come utilizzare`list-action-types`.

AWS CLI

Come visualizzare i tipi di azione disponibili

Utilizzato da solo, il list-action-types comando restituisce la struttura di tutte le azioni disponibili per l' AWS account. Questo esempio utilizza l'action-owner-filter opzione -- per restituire solo azioni personalizzate.

Comando:

```
aws codepipeline list-action-types --action-owner-filter Custom
```

Output:

```
{
  "actionTypes": [
      {
          "inputArtifactDetails": {
              "maximumCount": 5,
              "minimumCount": 0
          },
          "actionConfigurationProperties": [
              {
                  "secret": false,
                  "required": true,
                  "name": "MyJenkinsExampleBuildProject",
                  "key": true,
                  "queryable": true
              }
          ],
          "outputArtifactDetails": {
              "maximumCount": 5,
              "minimumCount": 0
          },
          "id": {
              "category": "Build",
              "owner": "Custom",
              "version": "1",
              "provider": "MyJenkinsProviderName"
          },
          "settings": {
              "entityUrlTemplate": "http://192.0.2.4/job/{Config:ProjectName}",
              "executionUrlTemplate": "http://192.0.2.4/job/{Config:ProjectName}/
{ExternalExecutionId}"
          }
      },
      {
```

```
            "inputArtifactDetails": {
                "maximumCount": 5,
                "minimumCount": 0
            },
            "actionConfigurationProperties": [
                {
                    "secret": false,
                    "required": true,
                    "name": "MyJenkinsExampleTestProject",
                    "key": true,
                    "queryable": true
                }
            ],
            "outputArtifactDetails": {
                "maximumCount": 5,
                "minimumCount": 0
            },
            "id": {
                "category": "Test",
                "owner": "Custom",
                "version": "1",
                "provider": "MyJenkinsProviderName"
            },
            "settings": {
                "entityUrlTemplate": "http://192.0.2.4/job/{Config:ProjectName}",
                "executionUrlTemplate": "http://192.0.2.4/job/{Config:ProjectName}/
{ExternalExecutionId}"
            }
        }
    ]
}
```

- Per i dettagli sull'API, vedere ListActionTypesin AWS CLI Command Reference.

`list-pipeline-executions`

Il seguente esempio di codice mostra come utilizzare`list-pipeline-executions`.

AWS CLI

Come visualizzare la cronologia di esecuzione delle pipeline

L'`list-pipeline-executions`esempio seguente mostra la cronologia di esecuzione di una pipeline nel tuo AWS account.

```
aws codepipeline list-pipeline-executions \
    --pipeline-name MyPipeline
```

Output:

```
{
    "pipelineExecutionSummaries": [
        {
            "lastUpdateTime": 1496380678.648,
            "pipelineExecutionId": "7cf7f7cb-3137-539g-j458-d7eu3EXAMPLE",
            "startTime": 1496380258.243,
            "status": "Succeeded"
        },
        {
            "lastUpdateTime": 1496591045.634,
            "pipelineExecutionId": "3137f7cb-8d494hj4-039j-d84l-d7eu3EXAMPLE",
            "startTime": 1496590401.222,
            "status": "Succeeded"
        },
        {
            "lastUpdateTime": 1496946071.6456,
            "pipelineExecutionId": "4992f7jf-7cf7-913k-k334-d7eu3EXAMPLE",
            "startTime": 1496945471.5645,
            "status": "Succeeded"
        }
    ]
}
```

Per ulteriori informazioni, consulta Visualizza la cronologia di esecuzione nella Guida per l'AWS CodePipeline utente.

- Per i dettagli sull'API, consulta ListPipelineExecutions AWS CLICommand Reference.

`list-pipelines`

Il seguente esempio di codice mostra come utilizzare`list-pipelines`.

AWS CLI

Come visualizzare un elenco di pipeline

Questo esempio elenca tutte le AWS CodePipeline pipeline associate all' AWS account dell'utente.

Comando:

```
aws codepipeline list-pipelines
```

Output:

```
{
  "pipelines": [
      {
          "updated": 1439504274.641,
          "version": 1,
          "name": "MyFirstPipeline",
          "created": 1439504274.641
      },
      {
          "updated": 1436461837.992,
          "version": 2,
          "name": "MySecondPipeline",
          "created": 1436460801.381
      }
      ]
}
```

- Per i dettagli sull'API, vedere ListPipelinesin AWS CLI Command Reference.

`list-tags-for-resource`

Il seguente esempio di codice mostra come utilizzare`list-tags-for-resource`.

AWS CLI

Come elencare i tag

L'esempio `list-tags-for-resource` seguente recupera un elenco di tutti i tag collegati alla risorsa pipeline specificata.

```
aws codepipeline list-tags-for-resource \
    --resource-arn arn:aws:codepipeline:us-east-1:123456789012:MyPipeline
```

Output:

```
{
    "tags": {
        "Project": "ProjectA",
        "IscontainerBased": "true"
    }
}
```

Per ulteriori informazioni, consulta Visualizza i tag per una pipeline (CLI) nella Guida per AWS CodePipeline l'utente.

- Per i dettagli sull'API, consulta AWS CLI Command ListTagsForResourceReference.

`list-webhooks`

Il seguente esempio di codice mostra come utilizzare`list-webhooks`.

AWS CLI

Come elencare i webhook

L'esempio `list-webhooks` seguente recupera un elenco di tutti i tag collegati alla risorsa pipeline specificata.

```
aws codepipeline list-webhooks \
    --endpoint-url "https://codepipeline.eu-central-1.amazonaws.com" \
    --region "eu-central-1"
```

Output:

```
{
    "webhooks": [
        {
            "url": "https://webhooks.domain.com/
trigger111111111EXAMPLE11111111111111111": {
                "authenticationConfiguration": {
                    "SecretToken": "Secret"
```

```
                },
                "name": "my-webhook",
                "authentication": "GITHUB_HMAC",
                "targetPipeline": "my-Pipeline",
                "targetAction": "Source",
                "filters": [
                    {
                        "jsonPath": "$.ref",
                        "matchEquals": "refs/heads/{Branch}"
                    }
                ]
            },
            "arn": "arn:aws:codepipeline:eu-central-1:123456789012:webhook:my-
webhook"
        }
    ]
}
```

Per ulteriori informazioni, consulta Elenca i webhook presenti nel tuo account nella Guida per l'AWS CodePipeline utente.

- Per i dettagli sull'API, consulta AWS CLI Command ListWebhooksReference.

`poll-for-jobs`

Il seguente esempio di codice mostra come utilizzare`poll-for-jobs`.

AWS CLI

Come visualizzare tutti i processi disponibili

Questo esempio restituisce le informazioni su qualsiasi processo su cui può agire un worker. Questo esempio utilizza un file JSON predefinito (MyActionTypeInfo.json) per fornire informazioni sul tipo di azione per cui il job worker elabora i job. Questo comando viene utilizzato solo per le azioni personalizzate. Quando viene chiamato questo comando, AWS CodePipeline restituisce le credenziali temporanee per il bucket Amazon S3 utilizzato per archiviare gli artefatti per la pipeline. Questo comando restituirà anche tutti i valori del segreto definiti per l'azione, se definiti.

Comando:

```
aws codepipeline poll-for-jobs --cli-input-json file://MyActionTypeInfo.json
```

Contenuto del file JSON di esempio:

```
{
  "actionTypeId": {
    "category": "Test",
    "owner": "Custom",
    "provider": "MyJenkinsProviderName",
    "version": "1"
  },
  "maxBatchSize": 5,
  "queryParam": {
      "ProjectName": "MyJenkinsTestProject"
  }
}
```

Output:

```
{
 "jobs": [
  {
    "accountId": "111111111111",
    "data": {
      "actionConfiguration": {
        "__type": "ActionConfiguration",
        "configuration": {
          "ProjectName": "MyJenkinsExampleTestProject"
        }
      },
      "actionTypeId": {
        "__type": "ActionTypeId",
        "category": "Test",
        "owner": "Custom",
        "provider": "MyJenkinsProviderName",
        "version": "1"
      },
      "artifactCredentials": {
        "__type": "AWSSessionCredentials",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "secretAccessKey": "wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY",
        "sessionToken":
 "fICCQD6m7oRw0uXOjANBgkqhkiG9w0BAQUFADCBiDELMAkGA1UEBhMCVVMxCzAJBgNVBAgTAldBMRAwDgYDVQQHEwd
+a4GmWIWJ21uUSfwfEvySWtC2XADZ4nB+BLYgVIk60CpiwsZ3G93vUEIO3IyNoH/
f0wYK8m9TrDHudUZg3qX4waLG5M43q7Wgc/
```

```
MbQITxOUSQv7c7ugFFDzQGBzZswY6786m86gpEIbb3OhjZnzcvQAaRHhdlQWIMm2nrAgMBAAEwDQYJKoZIhvcNAQEFBQ
+auNKyExzyLwaxlAoo7TJHidbtS4J5iNmZgXL0FkbFFBjvSfpJIlJ00zbhNYS5f6GuoEDmFJl0ZxBHjJnyp378OD8uTs
      },
      "inputArtifacts": [
        {
          "__type": "Artifact",
          "location": {
            "s3Location": {
              "bucketName": "codepipeline-us-east-1-11EXAMPLE11",
              "objectKey": "MySecondPipeline/MyAppBuild/EXAMPLE"
            },
            "type": "S3"
          },
          "name": "MyAppBuild"
        }
      ],
      "outputArtifacts": [],
      "pipelineContext": {
        "__type": "PipelineContext",
        "action": {
          "name": "MyJenkinsTest-Action"
        },
        "pipelineName": "MySecondPipeline",
        "stage": {
          "name": "Testing"
        }
      }
    },
    "id": "ef66c259-64f9-EXAMPLE",
    "nonce": "3"
  }
 ]
}
```


- Per i dettagli sull'API, consulta Command Reference. PollForJobsAWS CLI

`put-webhook`

Il seguente esempio di codice mostra come utilizzare`put-webhook`.

AWS CLI

Come creare un webhook

L'`put-webhook`esempio seguente crea un webhook per un'azione di origine della GitHub versione 1. Dopo aver creato il webhook, è necessario utilizzare il comando register-webhook-with-third -party per registrarlo.

```
aws codepipeline put-webhook \
    --cli-input-json file://webhook_json.json \
    --region "eu-central-1"
```

Contenuto di `webhook_json.json`:

```
{
    "webhook": {
        "name": "my-webhook",
        "targetPipeline": "pipeline_name",
        "targetAction": "source_action_name",
        "filters": [
            {
                "jsonPath": "$.ref",
                "matchEquals": "refs/heads/{Branch}"
            }
        ],
        "authentication": "GITHUB_HMAC",
        "authenticationConfiguration": {
            "SecretToken": "secret"
        }
    }
}
```

Output:

```
{
    "webhook": {
        "url": "https://webhooks.domain.com/
trigger111111111EXAMPLE11111111111111111",
        "definition": {
            "authenticationConfiguration": {
                "SecretToken": "secret"
            },
            "name": "my-webhook",
            "authentication": "GITHUB_HMAC",
            "targetPipeline": "pipeline_name",
            "targetAction": "Source",
```

```
            "filters": [
                {
                    "jsonPath": "$.ref",
                    "matchEquals": "refs/heads/{Branch}"
                }
            ]
        },
        "arn": "arn:aws:codepipeline:eu-central-1:123456789012:webhook:my-webhook"
    },
    "tags": [
        {
            "key": "Project",
            "value": "ProjectA"
        }
    ]
}
```

Per ulteriori informazioni, consulta Creare un webhook per una GitHub fonte nella Guida per l'AWS CodePipeline utente.

- Per i dettagli sull'API, consulta PutWebhook AWS CLICommand Reference.

`retry-stage-execution`

Il seguente esempio di codice mostra come utilizzare`retry-stage-execution`.

AWS CLI

Come riprovare a eseguire un'azione non riuscita

L'esempio `retry-stage-execution` seguente ritenta una fase con un'azione non riuscita.

```
aws codepipeline retry-stage-execution \
    --pipeline-name MyPipeline \
    --stage-name Deploy \
    --pipeline-execution-id b59babff-5f34-EXAMPLE \
    --retry-mode FAILED_ACTIONS
```

Output:

```
{
    "pipelineExecutionId": "b59babff-5f34-EXAMPLE"
```

```
}
```

Per ulteriori informazioni, consulta Retry failed actions (CLI) nella Guida per AWS CodePipeline l'utente.

- Per i dettagli sull'API, consulta AWS CLI Command RetryStageExecutionReference.

`start-pipeline-execution`

Il seguente esempio di codice mostra come utilizzare`start-pipeline-execution`.

AWS CLI

Come eseguire la revisione più recente tramite una pipeline

Questo esempio esegue l'ultima revisione presente nella fase di origine di una pipeline tramite la pipeline denominata "». MyFirstPipeline

Comando:

```
aws codepipeline start-pipeline-execution --name MyFirstPipeline
```

Output:

```
{
  "pipelineExecutionId": "3137f7cb-7cf7-EXAMPLE"
}
```

- Per i dettagli sull'API, consulta Command StartPipelineExecutionReference AWS CLI .

`stop-pipeline-execution`

Il seguente esempio di codice mostra come utilizzare`stop-pipeline-execution`.

AWS CLI

Come arrestare l'esecuzione di una pipeline

L'esempio `stop-pipeline-execution` seguente attende, per impostazione predefinita, il completamento delle azioni in corso e quindi arresta la pipeline. Non è possibile scegliere di

interrompere e attendere se l'esecuzione è già in uno stato di arresto. È possibile scegliere di interrompere e abbandonare un'esecuzione già in uno stato di arresto.

```
aws codepipeline stop-pipeline-execution \
    --pipeline-name MyFirstPipeline \
    --pipeline-execution-id d-EXAMPLE \
    --reason "Stopping pipeline after the build action is done"
```

Questo comando non restituisce alcun output.

Per ulteriori informazioni, consulta Arrestare l'esecuzione di una pipeline (CLI) nella Guida per AWS CodePipeline l'utente.

- Per i dettagli sull'API, consulta AWS CLI Command StopPipelineExecutionReference.

`tag-resource`

Il seguente esempio di codice mostra come utilizzare`tag-resource`.

AWS CLI

Come taggare a una risorsa

L'esempio `tag-resource` seguente associa un set di tag forniti a una connessione. Utilizza questo comando per aggiungere o modificare tag.

```
aws codepipeline tag-resource \
    --resource-arn arn:aws:codepipeline:us-east-1:123456789012:MyPipeline \
    --tags key=Project,value=ProjectA key=IscontainerBased,value=true
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Aggiungere tag a una pipeline (CLI) nella Guida per AWS CodePipeline l'utente.

- Per i dettagli sull'API, consulta AWS CLI Command TagResourceReference.

`untag-resource`

Il seguente esempio di codice mostra come utilizzare`untag-resource`.

AWS CLI

Per rimuovere i AWS tag da una risorsa di connessione

L'esempio `untag-resource` seguente rimuove un tag dalla risorsa specificata.

```
aws codepipeline untag-resource \
    --resource-arn arn:aws:codepipeline:us-east-1:123456789012:MyPipeline \
    --tag-keys Project IscontainerBased
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Rimuovere tag da una pipeline (CLI) nella Guida per AWS CodePipeline l'utente.

- Per i dettagli sull'API, consulta AWS CLI Command UntagResourceReference.

`update-pipeline`

Il seguente esempio di codice mostra come utilizzare`update-pipeline`.

AWS CLI

Come aggiornare la struttura di una pipeline

Questo esempio utilizza il comando update-pipeline con l'argomento --. cli-input-json Questo esempio utilizza un file JSON predefinito (MyFirstPipeline.json) per aggiornare la struttura di una pipeline. AWS CodePipeline riconosce il nome della pipeline contenuto nel file JSON, quindi applica tutte le modifiche apportate ai campi modificati nella struttura della pipeline per aggiornare la pipeline.

Utilizza le linee guida seguenti per creare il file JSON predefinito:

Se utilizzi una struttura di pipeline recuperata mediante il comando get-pipeline, rimuovi la sezione dei metadati dalla struttura della pipeline nel file JSON, ovvero le righe "metadata": { } e i campi "created", "pipelineARN" e "updated" al loro interno. Il nome della pipeline non può essere modificato.

Comando:

```
aws codepipeline update-pipeline --cli-input-json file://MyFirstPipeline.json
```

Contenuto del file JSON di esempio:

```
{
 "pipeline": {
  "roleArn": "arn:aws:iam::111111111111:role/AWS-CodePipeline-Service",
  "stages": [
    {
      "name": "Source",
      "actions": [
        {
          "inputArtifacts": [],
          "name": "Source",
          "actionTypeId": {
            "category": "Source",
            "owner": "AWS",
            "version": "1",
            "provider": "S3"
          },
          "outputArtifacts": [
            {
              "name": "MyApp"
            }
          ],
          "configuration": {
            "S3Bucket": "awscodepipeline-demo-bucket2",
            "S3ObjectKey": "aws-codepipeline-s3-aws-codedeploy_linux.zip"
          },
          "runOrder": 1
        }
      ]
    },
    {
      "name": "Beta",
      "actions": [
        {
          "inputArtifacts": [
            {
              "name": "MyApp"
            }
          ],
          "name": "CodePipelineDemoFleet",
          "actionTypeId": {
            "category": "Deploy",
            "owner": "AWS",
```

```
            "version": "1",
            "provider": "CodeDeploy"
          },
          "outputArtifacts": [],
          "configuration": {
            "ApplicationName": "CodePipelineDemoApplication",
            "DeploymentGroupName": "CodePipelineDemoFleet"
          },
          "runOrder": 1
        }
      ]
    }
  ],
  "artifactStore": {
    "type": "S3",
    "location": "codepipeline-us-east-1-11EXAMPLE11"
  },
  "name": "MyFirstPipeline",
  "version": 1
 }
}
```

Output:

```
{
 "pipeline": {
  "artifactStore": {
    "location": "codepipeline-us-east-1-11EXAMPLE11",
    "type": "S3"
  },
  "name": "MyFirstPipeline",
  "roleArn": "arn:aws:iam::111111111111:role/AWS-CodePipeline-Service",
  "stages": [
    {
      "actions": [
        {
          "actionTypeId": {
            "__type": "ActionTypeId",
            "category": "Source",
            "owner": "AWS",
            "provider": "S3",
            "version": "1"
          },
```

```
          "configuration": {
            "S3Bucket": "awscodepipeline-demo-bucket2",
            "S3ObjectKey": "aws-codepipeline-s3-aws-codedeploy_linux.zip"
          },
          "inputArtifacts": [],
          "name": "Source",
          "outputArtifacts": [
            {
              "name": "MyApp"
            }
          ],
          "runOrder": 1
        }
      ],
      "name": "Source"
    },
    {
      "actions": [
        {
          "actionTypeId": {
            "__type": "ActionTypeId",
            "category": "Deploy",
            "owner": "AWS",
            "provider": "CodeDeploy",
            "version": "1"
          },
          "configuration": {
            "ApplicationName": "CodePipelineDemoApplication",
            "DeploymentGroupName": "CodePipelineDemoFleet"
          },
          "inputArtifacts": [
            {
              "name": "MyApp"
            }
          ],
          "name": "CodePipelineDemoFleet",
          "outputArtifacts": [],
          "runOrder": 1
        }
      ],
      "name": "Beta"
    }
  ],
  "version": 3
```

```
 }
}
```

- Per i dettagli sull'API, consulta Command Reference. UpdatePipelineAWS CLI

AWS CodeStar Esempi di notifiche utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with AWS CodeStar Notifications.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`create-notification-rule`

Il seguente esempio di codice mostra come utilizzare`create-notification-rule`.

AWS CLI

Come creare una regola di notifica

L'`create-notification-rule`esempio seguente utilizza un file JSON denominato `rule.json` per creare una regola di notifica denominata `MyNotificationRule` per un repository denominato `MyDemoRepo` nell'account specificato AWS . Le notifiche con il tipo di dettaglio `FULL` vengono inviate all'argomento Amazon SNS di destinazione specificato quando vengono creati ramificazioni e tag.

```
aws codestar-notifications create-notification-rule \
    --cli-input-json file://rule.json
```

Contenuto di `rule.json`:

```
{
    "Name": "MyNotificationRule",
    "EventTypeIds": [
        "codecommit-repository-branches-and-tags-created"
    ],
    "Resource": "arn:aws:codecommit:us-east-1:123456789012:MyDemoRepo",
    "Targets": [
        {
            "TargetType": "SNS",
            "TargetAddress": "arn:aws:sns:us-
east-1:123456789012:MyNotificationTopic"
        }
    ],
    "Status": "ENABLED",
    "DetailType": "FULL"
}
```

Output:

```
{
    "Arn": "arn:aws:codestar-notifications:us-east-1:123456789012:notificationrule/
dc82df7a-EXAMPLE"
}
```

Per ulteriori informazioni, consulta Creazione di una regola di notifica nella Guida per l'utente della console degli strumenti per sviluppatori AWS .

- Per i dettagli sull'API, consulta AWS CLI Command CreateNotificationRuleReference.

`delete-notification-rule`

Il seguente esempio di codice mostra come utilizzare`delete-notification-rule`.

AWS CLI

Come eliminare una regola di notifica

L'esempio `delete-notification-rule` seguente elimina la regola di notifica specificata.

```
aws codestar-notifications delete-notification-rule \
```

```
    --arn arn:aws:codestar-notifications:us-east-1:123456789012:notificationrule/
dc82df7a-EXAMPLE
```

Output:

```
{
    "Arn": "arn:aws:codestar-notifications:us-east-1:123456789012:notificationrule/
dc82df7a-EXAMPLE"
}
```

Per ulteriori informazioni, consulta Eliminazione di una regola di notifica nella Guida per l'utente della console degli strumenti per sviluppatori AWS .

- Per i dettagli sull'API, consulta DeleteNotificationRule AWS CLICommand Reference.

`delete-target`

Il seguente esempio di codice mostra come utilizzare`delete-target`.

AWS CLI

Come eliminare una destinazione delle regole di notifica

L'esempio `delete-target` seguente rimuove la destinazione da tutte le regole di notifica configurate per l'utilizzo come destinazione, quindi elimina la destinazione.

```
aws codestar-notifications  delete-target \
    --target-address arn:aws:sns:us-east-1:123456789012:MyNotificationTopic \
    --force-unsubscribe-all
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Eliminazione di una destinazione delle regole di notifica nella Guida per l'utente della console degli strumenti per sviluppatori AWS .

- Per i dettagli sull'API, consulta DeleteTarget AWS CLICommand Reference.

`describe-notification-rule`

Il seguente esempio di codice mostra come utilizzare`describe-notification-rule`.

AWS CLI

Come recuperare i dettagli di una regola di notifica

L'esempio `describe-notification-rule` seguente recupera i dettagli della regola di notifica specificata.

```
aws codestar-notifications describe-notification-rule \
    --arn arn:aws:codestar-notifications:us-west-2:123456789012:notificationrule/
dc82df7a-EXAMPLE
```

Output:

```
{
    "LastModifiedTimestamp": 1569199844.857,
    "EventTypes": [
        {
            "ServiceName": "CodeCommit",
            "EventTypeName": "Branches and tags: Created",
            "ResourceType": "Repository",
            "EventTypeId": "codecommit-repository-branches-and-tags-created"
        }
    ],
    "Status": "ENABLED",
    "DetailType": "FULL",
    "Resource": "arn:aws:codecommit:us-west-2:123456789012:MyDemoRepo",
    "Arn": "arn:aws:codestar-notifications:us-west-w:123456789012:notificationrule/
dc82df7a-EXAMPLE",
    "Targets": [
        {
            "TargetStatus": "ACTIVE",
            "TargetAddress": "arn:aws:sns:us-
west-2:123456789012:MyNotificationTopic",
            "TargetType": "SNS"
        }
    ],
    "Name": "MyNotificationRule",
    "CreatedTimestamp": 1569199844.857,
    "CreatedBy": "arn:aws:iam::123456789012:user/Mary_Major"
}
```

Per ulteriori informazioni, consulta Visualizzazione delle regole di notifica nella Guida per l'utente della console degli strumenti per sviluppatori AWS .

- Per i dettagli sull'API, consulta DescribeNotificationRule AWS CLICommand Reference.

`list-event-types`

Il seguente esempio di codice mostra come utilizzare`list-event-types`.

AWS CLI

Come un elenco dei tipi di eventi per le regole di notifica

L'`list-event-types`esempio seguente recupera un elenco filtrato di tutti i tipi di eventi di notifica disponibili per CodeDeploy le applicazioni. Se invece non si utilizza alcun filtro, il comando restituisce tutti i tipi di eventi di notifica per tutti i tipi di risorse.

```
aws codestar-notifications list-event-types \
    --filters Name=SERVICE_NAME,Value=CodeDeploy
```

Output:

```
{
    "EventTypes": [
        {
            "EventTypeId": "codedeploy-application-deployment-succeeded",
            "ServiceName": "CodeDeploy",
            "EventTypeName": "Deployment: Succeeded",
            "ResourceType": "Application"
        },
        {
            "EventTypeId": "codedeploy-application-deployment-failed",
            "ServiceName": "CodeDeploy",
            "EventTypeName": "Deployment: Failed",
            "ResourceType": "Application"
        },
        {
            "EventTypeId": "codedeploy-application-deployment-started",
            "ServiceName": "CodeDeploy",
            "EventTypeName": "Deployment: Started",
            "ResourceType": "Application"
        }
    ]
}
```

Per ulteriori informazioni, consulta Creazione di una regola di notifica nella Guida per l'utente della console degli strumenti per sviluppatori AWS .

- Per i dettagli sull'API, vedere ListEventTypesin AWS CLI Command Reference.

`list-notification-rules`

Il seguente esempio di codice mostra come utilizzare`list-notification-rules`.

AWS CLI

Come recuperare un elenco di regole di notifica

L'`list-notification-rules`esempio seguente recupera un elenco di tutte le regole di notifica nella AWS regione specificata.

```
aws codestar-notifications list-notification-rules --region us-east-1
```

Output:

```
{
    "NotificationRules": [
        {
            "Id": "dc82df7a-EXAMPLE",
            "Arn": "arn:aws:codestar-notifications:us-
east-1:123456789012:notificationrule/dc82df7a-EXAMPLE"
        },
        {
            "Id": "8d1f0983-EXAMPLE",
            "Arn": "arn:aws:codestar-notifications:us-
east-1:123456789012:notificationrule/8d1f0983-EXAMPLE"
        }
    ]
}
```

Per ulteriori informazioni, consulta Visualizzazione delle regole di notifica nella Guida per l'utente della console degli strumenti per sviluppatori AWS .

- Per i dettagli sull'API, vedere ListNotificationRulesin AWS CLI Command Reference.

`list-tags-for-resource`

Il seguente esempio di codice mostra come utilizzare`list-tags-for-resource`.

AWS CLI

Come ottenere un elenco di tag allegati a una regola di notifica

L'esempio `list-tags-for-resource` seguente recupera un elenco di tutti i tag associati alla regola di notifica specificata. In questo esempio, alla regola di notifica non sono attualmente associati tag.

```
aws codestar-notifications list-tags-for-resource \
    --arn arn:aws:codestar-notifications:us-east-1:123456789012:notificationrule/
fe1efd35-EXAMPLE
```

Output:

```
{
    "Tags": {}
}
```

Per ulteriori informazioni, consulta Creazione di una regola di notifica nella Guida per l'utente della console degli strumenti per sviluppatori AWS .

- Per i dettagli sull'API, consulta ListTagsForResource AWS CLICommand Reference.

`list-targets`

Il seguente esempio di codice mostra come utilizzare`list-targets`.

AWS CLI

Come recuperare un elenco di destinazioni di regole di notifica

L'`list-targets`esempio seguente recupera un elenco di tutti gli obiettivi delle regole di notifica nella AWS regione specificata.

```
aws codestar-notifications list-targets \
    --region us-east-1
```

Output:

```
{
    "Targets": [
        {
            "TargetAddress": "arn:aws:sns:us-
east-1:123456789012:MySNSTopicForNotificationRules",
            "TargetType": "SNS",
            "TargetStatus": "ACTIVE"
        },
        {
            "TargetAddress": "arn:aws:sns:us-
east-1:123456789012:MySNSTopicForNotificationsAboutMyDemoRepo",
            "TargetType": "SNS",
            "TargetStatus": "ACTIVE"
        }
    ]
}
```

Per ulteriori informazioni, consulta Visualizzazione delle destinazioni delle regole di notifica nella Guida per l'utente della console degli strumenti per sviluppatori AWS .

- Per i dettagli sull'API, vedere ListTargetsin AWS CLI Command Reference.

`subscribe`

Il seguente esempio di codice mostra come utilizzare`subscribe`.

AWS CLI

Come aggiungere una destinazione a una regola di notifica

L'esempio `subscribe` seguente aggiunge un argomento Amazon SNS come destinazione per la regola di notifica specificata.

```
aws codestar-notifications subscribe \
    --arn arn:aws:codestar-notifications:us-east-1:123456789012:notificationrule/
dc82df7a-EXAMPLE \
    --target TargetType=SNS,TargetAddress=arn:aws:sns:us-
east-1:123456789012:MyNotificationTopic
```

Output:

```
{
    "Arn": "arn:aws:codestar-notifications:us-east-1:123456789012:notificationrule/
dc82df7a-EXAMPLE"
}
```

Per ulteriori informazioni, consulta Aggiungere o rimuovere un argomento Amazon SNS come una destinazione per una regola di notifica nella Guida per l'utente della console degli strumenti per sviluppatori AWS .

- Per informazioni dettagliate sull'API, consulta Subscribe nella documentazione di riferimento dei comandi della AWS CLI .

`tag-resource`

Il seguente esempio di codice mostra come utilizzare`tag-resource`.

AWS CLI

Come aggiungere un tag a una regola di notifica

L'esempio `tag-resource` seguente aggiunge un tag con il nome di chiave `Team` e il valore `Li_Juan` alla regole di notifica specificata.

```
aws codestar-notifications tag-resource \
    --arn arn:aws:codestar-notifications:us-east-1:123456789012:notificationrule/
fe1efd35-EXAMPLE \
    --tags Team=Li_Juan
```

Output:

```
{
    "Tags": {
        "Team": "Li_Juan"
    }
}
```

Per ulteriori informazioni, consulta Creazione di una regola di notifica nella Guida per l'utente della console degli strumenti per sviluppatori AWS .

- Per i dettagli sull'API, consulta TagResource AWS CLICommand Reference.

`unsubscribe`

Il seguente esempio di codice mostra come utilizzare`unsubscribe`.

AWS CLI

Come rimuovere una destinazione da una regola di notifica

L'esempio `unsubscribe` seguente rimuove un argomento Amazon SNS come destinazione dalla regola di notifica specificata.

```
aws codestar-notifications unsubscribe \
    --arn arn:aws:codestar-notifications:us-east-1:123456789012:notificationrule/
dc82df7a-EXAMPLE \
    --target TargetType=SNS,TargetAddress=arn:aws:sns:us-
east-1:123456789012:MyNotificationTopic
```

Output:

```
{
    "Arn": "arn:aws:codestar-notifications:us-east-1:123456789012:notificationrule/
dc82df7a-EXAMPLE"
    "TargetAddress": "arn:aws:sns:us-east-1:123456789012:MyNotificationTopic"
}
```

Per ulteriori informazioni, consulta Aggiungere o rimuovere un argomento Amazon SNS come una destinazione per una regola di notifica nella Guida per l'utente della console degli strumenti per sviluppatori AWS .

- Per informazioni dettagliate sull'API, consulta Unsubscribe nella documentazione di riferimento dei comandi della AWS CLI .

`untag-resource`

Il seguente esempio di codice mostra come utilizzare`untag-resource`.

AWS CLI

Come rimuovere un tag da una regola di destinazione

L'esempio `untag-resource` seguente rimuove il tag con il nome di chiave `Team` dalla regola di notifica specificata.

```
aws codestar-notifications untag-resource \
    --arn arn:aws:codestar-notifications:us-east-1:123456789012:notificationrule/
fe1efd35-EXAMPLE \
    --tag-keys Team
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Modifica di una regola di notifica nella Guida per l'utente della console degli strumenti per sviluppatori AWS .

- Per i dettagli sull'API, consulta UntagResource AWS CLICommand Reference.

update-notification-rule

Il seguente esempio di codice mostra come utilizzareupdate-notification-rule.

AWS CLI

Come aggiornare una regola di notifica

L'update-notification-ruleesempio seguente aggiorna una regola di notifica denominata MyNotificationRule nell' AWS account 123456789012 utilizzando un file JSON denominatoupdate.json.

```
aws codestar-notifications update-notification-rule \
    --cli-input-json file://update.json
```

Contenuto di update.json:

```
{
    "Name": "MyUpdatedNotificationRule",
    "EventTypeIds": [
        "codecommit-repository-branches-and-tags-created"
    ],
    "Resource": "arn:aws:codecommit:us-east-1:123456789012:MyDemoRepo",
    "Targets": [
        {
            "TargetType": "SNS",
            "TargetAddress": "arn:aws:sns:us-
east-1:123456789012:MyNotificationTopic"
        }
    ],
```

```
    "Status": "ENABLED",
    "DetailType": "FULL"
}
```

Output:

```
{
    "Arn": "arn:aws:codestar-notifications:us-east-1:123456789012:notificationrule/
dc82df7a-EXAMPLE"
}
```

Per ulteriori informazioni, consulta Modifica di una regola di notifica nella Guida per l'utente della console degli strumenti per sviluppatori AWS .

- Per i dettagli sull'API, vedere UpdateNotificationRulein AWS CLI Command Reference.

CodeConnections esempi utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with CodeConnections.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`create-connection`

Il seguente esempio di codice mostra come utilizzare`create-connection`.

AWS CLI

Come creare una connessione

L'esempio `create-connection` seguente crea una connessione a un repository di terze parti. Questo esempio crea una connessione in cui il provider di terze parti è Bitbucket.

Una connessione creata tramite la AWS CLI o AWS CloudFormation è in sospeso per impostazione predefinita. Dopo aver creato una connessione con la CLI o AWS CloudFormation, utilizza la console per modificare la connessione e renderne lo stato Disponibile.

```
aws codestar-connections create-connection \
    --provider-type Bitbucket \
    --connection-name MyConnection
```

Output:

```
{
    "ConnectionArn": "arn:aws:codestar-connections:us-
east-1:123456789012:connection/aEXAMPLE-8aad-4d5d-8878-dfcab0bc441f"
}
```

Per ulteriori informazioni, consulta Creazione di una connessione nella Guida per l'utente della console degli strumenti per sviluppatori.

- Per i dettagli sull'API, consulta CreateConnection AWS CLICommand Reference.

`create-host`

Il seguente esempio di codice mostra come utilizzare`create-host`.

AWS CLI

Come creare un host

L'esempio `create-host` seguente mostra come creare un host per rappresentare l'endpoint per l'infrastruttura in cui è installato il provider di terze parti. Questo esempio crea un host in cui il provider installato da terze parti è GitHub Enterprise Server.

Per impostazione predefinita, un host creato tramite la AWS CLI è in sospeso. Dopo aver creato un host con la CLI, utilizza la console o la CLI per configurare l'host e impostarne lo stato su Disponibile.

```
aws codestar-connections create-host \
    --name MyHost \
    --provider-type GitHubEnterpriseServer \
```

```
    --provider-endpoint "https://my-instance.dev"
```

Output:

```
{
    "HostArn": "arn:aws:codestar-connections:us-east-1:123456789012:host/My-
Host-28aef605"
}
```

Per ulteriori informazioni, consulta Creazione di un host (CLI) nella Guida per l'utente della console degli strumenti per sviluppatori.

- Per i dettagli sull'API, consulta AWS CLI Command CreateHostReference.

`delete-connection`

Il seguente esempio di codice mostra come utilizzare`delete-connection`.

AWS CLI

Come eliminare una connessione

L'esempio `delete-connection` seguente mostra come eliminare una connessione.

```
aws codestar-connections delete-connection \
    --connection-arn arn:aws:codestar-connections:us-west-2:123456789012:connection/
aEXAMPLE-8aad-4d5d-8878-dfcab0bc441f
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Eliminazione di una connessione (CLI) nella Guida per l'utente della console degli strumenti per sviluppatori.

- Per i dettagli sull'API, consulta DeleteConnection AWS CLICommand Reference.

`delete-host`

Il seguente esempio di codice mostra come utilizzare`delete-host`.

AWS CLI

Come eliminare un host

L'esempio `delete-host` seguente mostra come eliminare un host. Per eliminare un host, è necessario eliminare tutte le connessioni associate all'host.

```
aws codestar-connections delete-host \
    --host-arn "arn:aws:codestar-connections:us-east-1 :123456789012:host/My-Host-28aef605"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Eliminare un host (CLI) nella Guida per l'utente della console degli strumenti per sviluppatori.

- Per i dettagli sull'API, consulta DeleteHost AWS CLICommand Reference.

`get-connection`

Il seguente esempio di codice mostra come utilizzare`get-connection`.

AWS CLI

Come ottenere informazioni su una connessione

L'esempio `get-connection` seguente mostra i dettagli di una connessione.

```
aws codestar-connections get-connection \
    --connection-arn arn:aws:codestar-connections:us-east-1:123456789012:connection/aEXAMPLE-8aad-4d5d-8878-dfcab0bc441f
```

Output:

```
{
    "Connection": {
        "ConnectionName": "MyConnection",
        "ConnectionArn": "arn:aws:codestar-connections:us-east-1:123456789012:connection/aEXAMPLE-8aad-4d5d-8878-dfcab0bc441f",
        "ProviderType": "Bitbucket",
        "OwnerAccountId": "123456789012",
        "ConnectionStatus": "AVAILABLE"
    }
}
```

Per ulteriori informazioni, consulta Visualizzare i dettagli di connessione nella Guida per l'utente della console degli strumenti per sviluppatori.

- Per i dettagli sull'API, consulta GetConnection AWS CLICommand Reference.

`get-host`

Il seguente esempio di codice mostra come utilizzare`get-host`.

AWS CLI

Come ottenere informazioni su un host

L'esempio `get-host` seguente mostra i dettagli di un host.

```
aws codestar-connections get-host \
    --host-arn arn:aws:codestar-connections:us-east-1:123456789012:host/
MyHost-28aef605
```

Output:

```
{
    "Name": "MyHost",
    "Status": "AVAILABLE",
    "ProviderType": "GitHubEnterpriseServer",
    "ProviderEndpoint": "https://test-instance-1.dev/"
}
```

Per ulteriori informazioni, consulta Visualizzare i dettagli dell'host (CLI) nella Guida per l'utente della console degli strumenti per sviluppatori.

- Per i dettagli sull'API, consulta GetHost AWS CLICommand Reference.

`list-connections`

Il seguente esempio di codice mostra come utilizzare`list-connections`.

AWS CLI

Come elencare le connessioni

L'esempio `list-connections` seguente recupera un elenco di tutte le connessioni nell'account per il tipo di provider Bitbucket:

```
aws codestar-connections list-connections \
--provider-type Bitbucket \
--max-results 5 \
--next-token: next-token
```

Output:

```
{
    "Connections": [
        {
            "ConnectionName": "my-connection",
            "ProviderType": "Bitbucket",
            "Status": "PENDING",
            "ARN": "arn:aws:codestar-connections:us-east-1:123456789012:connection/
aEXAMPLE-8aad-4d5d-8878-dfcab0bc441f",
            "OwnerAccountId": "123456789012"
        },
        {
            "ConnectionName": "my-other-connection",
            "ProviderType": "Bitbucket",
            "Status": "AVAILABLE",
            "ARN": "arn:aws:codestar-connections:us-east-1:123456789012:connection/
aEXAMPLE-8aad-4d5d-8878-dfcab0bc441f",
            "OwnerAccountId": "123456789012"
        },
    ],
    "NextToken": "next-token"
}
```

Per ulteriori informazioni, consulta Elenco delle connessioni (CLI) nella Guida per l'utente della console degli strumenti per sviluppatori.

- Per i dettagli sull'API, consulta ListConnections AWS CLICommand Reference.

`list-hosts`

Il seguente esempio di codice mostra come utilizzare`list-hosts`.

AWS CLI

Come elencare gli host

L'esempio `list-hosts` seguente recupera un elenco di tutti gli host nell'account in uso.

```
aws codestar-connections list-hosts
```

Output:

```
{
    "Hosts": [
        {
            "Name": "My-Host",
            "HostArn": "arn:aws:codestar-connections:us-east-1:123456789012:host/My-Host-28aef605",
            "ProviderType": "GitHubEnterpriseServer",
            "ProviderEndpoint": "https://my-instance.test.dev",
            "Status": "AVAILABLE"
        }
    ]
}
```

Per ulteriori informazioni, consulta Elencare gli host (CLI) nella Guida per l'utente della console degli strumenti per sviluppatori.

- Per i dettagli sull'API, consulta ListHosts AWS CLICommand Reference.

`list-tags-for-resource`

Il seguente esempio di codice mostra come utilizzare`list-tags-for-resource`.

AWS CLI

Come elencare i tag

L'esempio `list-tags-for-resource` seguente recupera un elenco di tutti i tag associati alla risorsa di connessione specificata.

```
aws codestar-connections list-tags-for-resource \
    --resource-arn arn:aws:codestar-connections:us-east-1:123456789012:connection/aEXAMPLE-8aad-4d5d-8878-dfcab0bc441f
```

Output:

```
{
```

```
    "Tags": [
        {
            "Key": "Project",
            "Value": "ProjectA"
        },
        {
            "Key": "ReadOnly",
            "Value": "true"
        }
    ]
}
```

Per ulteriori informazioni, consulta Visualizzazione dei tag per una risorsa di connessione nella Guida per l'utente della console degli strumenti per sviluppatori.

- Per i dettagli sull'API, consulta ListTagsForResource AWS CLICommand Reference.

`tag-resource`

Il seguente esempio di codice mostra come utilizzare`tag-resource`.

AWS CLI

Come taggare a una risorsa

L'esempio `tag-resource` seguente associa un set di tag forniti a una connessione. Utilizza questo comando per aggiungere o modificare tag.

```
aws codestar-connections tag-resource \
    --resource-arn arn:aws:codestar-connections:us-east-1:123456789012:connection/aEXAMPLE-8aad-4d5d-8878-dfcab0bc441f \
    --tags Key=Project,Value=ProjectA Key=IscontainerBased,Value=true
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Aggiunta dei tag a una risorsa di connessione nella Guida per l'utente della console degli strumenti per sviluppatori.

- Per i dettagli sull'API, consulta TagResource AWS CLICommand Reference.

`untag-resource`

Il seguente esempio di codice mostra come utilizzare`untag-resource`.

AWS CLI

Per rimuovere i AWS tag da una risorsa di connessione

L'esempio `untag-resource` seguente rimuove un tag dalla risorsa specificata.

```
aws codestar-connections untag-resource \
    --resource-arn arn:aws:codestar-connections:us-east-1:123456789012:connection/aEXAMPLE-8aad-4d5d-8878-dfcab0bc441f \
    --tag-keys Project ReadOnly
```

Output:

```
{
    "Tags": []
}
```

Per ulteriori informazioni, consulta Rimozione dei tag da una risorsa di connessione nella Guida per l'utente della console degli strumenti per sviluppatori.

- Per i dettagli sull'API, consulta UntagResource AWS CLICommand Reference.

Esempi di identità di Amazon Cognito utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando Amazon Cognito Identity. AWS Command Line Interface

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`create-identity-pool`

Il seguente esempio di codice mostra come usare`create-identity-pool`.

AWS CLI

Come creare un pool di identità con il provider del pool di identità Cognito

Questo esempio crea un pool di identità denominato MyIdentityPool. Dispone di un provider del pool di identità Cognito. Le identità non autenticate non sono consentite.

Comando:

```
aws cognito-identity create-identity-pool --identity-pool-
name MyIdentityPool --no-allow-unauthenticated-identities --cognito-
identity-providers ProviderName="cognito-idp.us-west-2.amazonaws.com/us-
west-2_aaaaaaaaa",ClientId="3n4b5urk1ft4fl3mg5e62d9ado",ServerSideTokenCheck=false
```

Output:

```
{
  "IdentityPoolId": "us-west-2:11111111-1111-1111-1111-111111111111",
  "IdentityPoolName": "MyIdentityPool",
  "AllowUnauthenticatedIdentities": false,
  "CognitoIdentityProviders": [
      {
          "ProviderName": "cognito-idp.us-west-2.amazonaws.com/us-west-2_111111111",
          "ClientId": "3n4b5urk1ft4fl3mg5e62d9ado",
          "ServerSideTokenCheck": false
      }
  ]
}
```

- Per i dettagli sull'API, vedere CreateIdentityPoolin AWS CLI Command Reference.

`delete-identities`

Il seguente esempio di codice mostra come utilizzaredelete-identities.

AWS CLI

Come eliminare un pool di identità

Questo esempio elimina un pool di identità.

Comando:

```
aws cognito-identity delete-identity-pool --identity-ids-to-delete "us-west-2:11111111-1111-1111-1111-111111111111"
```

Output:

```
{
  "UnprocessedIdentityIds": []
}
```

- Per i dettagli sull'API, consulta DeleteIdentities AWS CLICommand Reference.

`delete-identity-pool`

Il seguente esempio di codice mostra come utilizzare`delete-identity-pool`.

AWS CLI

Come eliminare un pool di identità

L'esempio `delete-identity-pool` seguente elimina il pool di identità specificato.

Comando:

```
aws cognito-identity delete-identity-pool \
    --identity-pool-id "us-west-2:11111111-1111-1111-1111-111111111111"
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta DeleteIdentityPool AWS CLICommand Reference.

`describe-identity-pool`

Il seguente esempio di codice mostra come utilizzare`describe-identity-pool`.

AWS CLI

Come descrivere un pool di identità

Questo esempio descrive un pool di identità.

Comando:

```
aws cognito-identity describe-identity-pool --identity-pool-id "us-west-2:11111111-1111-1111-1111-111111111111"
```

Output:

```
{
  "IdentityPoolId": "us-west-2:11111111-1111-1111-1111-111111111111",
  "IdentityPoolName": "MyIdentityPool",
  "AllowUnauthenticatedIdentities": false,
  "CognitoIdentityProviders": [
      {
          "ProviderName": "cognito-idp.us-west-2.amazonaws.com/us-west-2_111111111",
          "ClientId": "3n4b5urk1ft4fl3mg5e62d9ado",
          "ServerSideTokenCheck": false
      }
  ]
}
```

- Per i dettagli sull'API, consulta DescribeIdentityPool AWS CLICommand Reference.

`get-identity-pool-roles`

Il seguente esempio di codice mostra come utilizzare`get-identity-pool-roles`.

AWS CLI

Come ottenere i ruoli del pool di identità

Questo esempio elenca i ruoli del pool di identità.

Comando:

```
aws cognito-identity get-identity-pool-roles --identity-pool-id "us-west-2:11111111-1111-1111-1111-111111111111"
```

Output:

```
{
  "IdentityPoolId": "us-west-2:11111111-1111-1111-1111-111111111111",
  "Roles": {
      "authenticated": "arn:aws:iam::111111111111:role/
Cognito_MyIdentityPoolAuth_Role",
      "unauthenticated": "arn:aws:iam::111111111111:role/
Cognito_MyIdentityPoolUnauth_Role"
  }
}
```

- Per i dettagli sull'API, consulta GetIdentityPoolRoles AWS CLICommand Reference.

`list-identity-pools`

Il seguente esempio di codice mostra come utilizzare`list-identity-pools`.

AWS CLI

Come elencare pool di identità

In questo esempio vengono elencati i pool di identità. Sono elencate un massimo di 20 identità.

Comando:

```
aws cognito-identity list-identity-pools --max-results 20
```

Output:

```
{
  "IdentityPools": [
      {
          "IdentityPoolId": "us-west-2:11111111-1111-1111-1111-111111111111",
          "IdentityPoolName": "MyIdentityPool"
      },
      {
          "IdentityPoolId": "us-west-2:11111111-1111-1111-1111-111111111111",
          "IdentityPoolName": "AnotherIdentityPool"
      },
      {
```

```
            "IdentityPoolId": "us-west-2:11111111-1111-1111-1111-111111111111",
            "IdentityPoolName": "IdentityPoolRegionA"
        }
    ]
}
```

- Per i dettagli sull'API, consulta ListIdentityPools AWS CLICommand Reference.

`set-identity-pool-roles`

Il seguente esempio di codice mostra come utilizzare`set-identity-pool-roles`.

AWS CLI

Come ottenere i ruoli del pool di identità

L'esempio `set-identity-pool-roles` seguente imposta il ruolo di un pool di identità.

```
aws cognito-identity set-identity-pool-roles \
    --identity-pool-id "us-west-2:11111111-1111-1111-1111-111111111111" \
    --roles authenticated="arn:aws:iam::111111111111:role/
Cognito_MyIdentityPoolAuth_Role"
```

- Per i dettagli sull'API, consulta SetIdentityPoolRoles AWS CLICommand Reference.

`update-identity-pool`

Il seguente esempio di codice mostra come utilizzare`update-identity-pool`.

AWS CLI

Come aggiornare un pool di identità

Questo esempio aggiorna un pool di identità. Imposta il nome su MyIdentityPool. Aggiunge Cognito come gestore dell'identità digitale. Non ammette identità non autenticate.

Comando:

```
aws cognito-identity update-identity-pool --identity-pool-id "us-
west-2:11111111-1111-1111-1111-111111111111" --identity-pool-
name "MyIdentityPool" --no-allow-unauthenticated-identities --cognito-
```

```
identity-providers ProviderName="cognito-idp.us-west-2.amazonaws.com/us-
west-2_111111111",ClientId="3n4b5urk1ft4fl3mg5e62d9ado",ServerSideTokenCheck=false
```

Output:

```
{
  "IdentityPoolId": "us-west-2:11111111-1111-1111-1111-111111111111",
  "IdentityPoolName": "MyIdentityPool",
  "AllowUnauthenticatedIdentities": false,
  "CognitoIdentityProviders": [
      {
          "ProviderName": "cognito-idp.us-west-2.amazonaws.com/us-west-2_111111111",
          "ClientId": "3n4b5urk1ft4fl3mg5e62d9ado",
          "ServerSideTokenCheck": false
      }
  ]
}
```

- Per i dettagli sull'API, consulta UpdateIdentityPool AWS CLICommand Reference.

Esempi di Amazon Cognito Identity Provider utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando Amazon Cognito Identity Provider. AWS Command Line Interface

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`add-custom-attributes`

Il seguente esempio di codice mostra come usare`add-custom-attributes`.

AWS CLI

Come aggiungere un attributo personalizzato

Questo esempio aggiunge un attributo personalizzato CustomAttr 1 a un pool di utenti. È un tipo stringa e richiede un minimo di 1 carattere e un massimo di 15. Non è obbligatorio.

Comando:

```
aws cognito-idp add-custom-attributes --user-pool-id us-west-2_aaaaaaaaa --custom-
attributes
 Name="CustomAttr1",AttributeDataType="String",DeveloperOnlyAttribute=false,Required=false,S
```

- Per i dettagli sull'API, vedere AddCustomAttributesin AWS CLI Command Reference.

`admin-add-user-to-group`

Il seguente esempio di codice mostra come utilizzare`admin-add-user-to-group`.

AWS CLI

Come aggiungere un utente a un gruppo

Questo esempio aggiunge l'utente Jane al gruppo MyGroup.

Comando:

```
aws cognito-idp admin-add-user-to-group --user-pool-id us-west-2_aaaaaaaaa --
username Jane --group-name MyGroup
```

- Per i dettagli sull'API, consulta AdminAddUserToGroup AWS CLICommand Reference.

`admin-confirm-sign-up`

Il seguente esempio di codice mostra come utilizzare`admin-confirm-sign-up`.

AWS CLI

Come confermare la registrazione dell'utente

Questo esempio conferma l'utente jane@example.com.

Comando:

```
aws cognito-idp admin-confirm-sign-up --user-pool-id us-west-2_aaaaaaaaa --
username jane@example.com
```

- Per i dettagli sull'API, consulta AdminConfirmSignUp AWS CLICommand Reference.

`admin-create-user`

Il seguente esempio di codice mostra come utilizzare`admin-create-user`.

AWS CLI

Come creare un utente

L'esempio `admin-create-user` seguente crea un utente con le impostazioni specificate per l'indirizzo e-mail e il numero di telefono.

```
aws cognito-idp admin-create-user \
    --user-pool-id us-west-2_aaaaaaaaa \
    --username diego \
    --user-attributes Name=email,Value=diego@example.com
 Name=phone_number,Value="+15555551212" \
    --message-action SUPPRESS
```

Output:

```
{
    "User": {
        "Username": "diego",
        "Attributes": [
            {
                "Name": "sub",
                "Value": "7325c1de-b05b-4f84-b321-9adc6e61f4a2"
            },
            {
                "Name": "phone_number",
                "Value": "+15555551212"
            },
            {
                "Name": "email",
```

```
                "Value": "diego@example.com"
            }
        ],
        "UserCreateDate": 1548099495.428,
        "UserLastModifiedDate": 1548099495.428,
        "Enabled": true,
        "UserStatus": "FORCE_CHANGE_PASSWORD"
    }
}
```

- Per i dettagli sull'API, consulta AdminCreateUser AWS CLICommand Reference.

`admin-delete-user-attributes`

Il seguente esempio di codice mostra come utilizzare`admin-delete-user-attributes`.

AWS CLI

Come eliminare un attributo utente

Questo esempio elimina un attributo personalizzato CustomAttr 1 per l'utente diego@example.com.

Comando:

```
aws cognito-idp admin-delete-user-attributes --user-pool-id us-west-2_aaaaaaaaa --
username diego@example.com --user-attribute-names "custom:CustomAttr1"
```

- Per i dettagli sull'API, vedere AdminDeleteUserAttributesin AWS CLI Command Reference.

`admin-delete-user`

Il seguente esempio di codice mostra come utilizzare`admin-delete-user`.

AWS CLI

Come eliminare un utente

Questo esempio elimina un utente.

Comando:

```
aws cognito-idp admin-delete-user --user-pool-id us-west-2_aaaaaaaaa --
username diego@example.com
```

- Per i dettagli sull'API, consulta AdminDeleteUser AWS CLICommand Reference.

admin-disable-provider-for-user

Il seguente esempio di codice mostra come utilizzareadmin-disable-provider-for-user.

AWS CLI

Come scollegare un utente federato da un profilo utente locale

L'esempio admin-disable-provider-for-user seguente disconnette un utente Google dal profilo locale collegato.

```
aws cognito-idp admin-disable-provider-for-user \
    --user-pool-id us-west-2_EXAMPLE \
    --
user ProviderAttributeName=Cognito_Subject,ProviderAttributeValue=0000000000000000,ProviderN
```

Per ulteriori informazioni, consulta Collegamento di utenti federati a un profilo utente esistente nella Guida per gli sviluppatori di Amazon Cognito.

- Per i dettagli sull'API, consulta AdminDisableProviderForUser AWS CLICommand Reference.

admin-disable-user

Il seguente esempio di codice mostra come utilizzareadmin-disable-user.

AWS CLI

Come impedire l'accesso da parte di un utente

L'esempio admin-disable-user seguente impedisce l'accesso da parte dell'utente diego@example.com.

```
aws cognito-idp admin-disable-user \
    --user-pool-id us-west-2_EXAMPLE \
    --username diego@example.com
```

Per ulteriori informazioni, consulta Gestione degli utenti nella Guida per gli sviluppatori di Amazon Cognito.

- Per i dettagli sull'API, consulta AdminDisableUser AWS CLICommand Reference.

`admin-enable-user`

Il seguente esempio di codice mostra come utilizzare`admin-enable-user`.

AWS CLI

Come abilitare l'accesso da parte di un utente

L'esempio `admin-enable-user` seguente abilita l'accesso da parte dell'utente diego@example.com.

```
aws cognito-idp admin-enable-user \
    --user-pool-id us-west-2_EXAMPLE \
    --username diego@example.com
```

Per ulteriori informazioni, consulta Gestione degli utenti nella Guida per gli sviluppatori di Amazon Cognito.

- Per i dettagli sull'API, consulta AdminEnableUser AWS CLICommand Reference.

`admin-forget-device`

Il seguente esempio di codice mostra come utilizzare`admin-forget-device`.

AWS CLI

Come dimenticare un dispositivo

Questo esempio mostra come dimenticare il dispositivo per il nome utente jane@example.com

Comando:

```
aws cognito-idp admin-forget-device --user-pool-id us-west-2_aaaaaaaaa --
username jane@example.com --device-key us-west-2_abcd_1234-5678
```

- Per i dettagli sull'API, consulta AdminForgetDevice AWS CLICommand Reference.

admin-get-device

Il seguente esempio di codice mostra come utilizzare`admin-get-device`.

AWS CLI

Come ottenere un dispositivo

L'esempio `admin-get-device` seguente visualizza un dispositivo per l'utente `diego`.

```
aws cognito-idp admin-get-device \
    --user-pool-id us-west-2_EXAMPLE \
    --username diego \
    --device-key us-west-2_a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Output:

```
{
    "Device": {
        "DeviceKey": "us-west-2_a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
        "DeviceAttributes": [
            {
                "Name": "device_status",
                "Value": "valid"
            },
            {
                "Name": "device_name",
                "Value": "MyDevice"
            },
            {
                "Name": "dev:device_arn",
                "Value": "arn:aws:cognito-idp:us-west-2:123456789012:owner/diego.us-
west-2_EXAMPLE/device/us-west-2_a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
            },
            {
                "Name": "dev:device_owner",
                "Value": "diego.us-west-2_EXAMPLE"
            },
            {
                "Name": "last_ip_used",
                "Value": "192.0.2.1"
            },
            {
```

```
                "Name": "dev:device_remembered_status",
                "Value": "remembered"
            },
            {
                "Name": "dev:device_sdk",
                "Value": "aws-sdk"
            }
        ],
        "DeviceCreateDate": 1715100742.022,
        "DeviceLastModifiedDate": 1723233651.167,
        "DeviceLastAuthenticatedDate": 1715100742.0
    }
}
```

Per ulteriori informazioni, consulta Utilizzo dei dispositivi utente nel pool di utenti nella Guida per gli sviluppatori di Amazon Cognito.

- Per i dettagli sull'API, consulta AdminGetDevice AWS CLICommand Reference.

`admin-get-user`

Il seguente esempio di codice mostra come utilizzare`admin-get-user`.

AWS CLI

Come ottenere un utente

Questo esempio mostra come ottenere informazioni sul nome utente jane@example.com.

Comando:

```
aws cognito-idp admin-get-user --user-pool-id us-west-2_aaaaaaaaa --
username jane@example.com
```

Output:

```
{
  "Username": "4320de44-2322-4620-999b-5e2e1c8df013",
  "Enabled": true,
  "UserStatus": "FORCE_CHANGE_PASSWORD",
  "UserCreateDate": 1548108509.537,
  "UserAttributes": [
```

```
        {
            "Name": "sub",
            "Value": "4320de44-2322-4620-999b-5e2e1c8df013"
        },
        {
            "Name": "email_verified",
            "Value": "true"
        },
        {
            "Name": "phone_number_verified",
            "Value": "true"
        },
        {
            "Name": "phone_number",
            "Value": "+01115551212"
        },
        {
            "Name": "email",
            "Value": "jane@example.com"
        }
    ],
    "UserLastModifiedDate": 1548108509.537
}
```

- Per i dettagli sull'API, consulta AdminGetUser AWS CLICommand Reference.

`admin-initiate-auth`

Il seguente esempio di codice mostra come utilizzare`admin-initiate-auth`.

AWS CLI

Come accedere come utente admin

L'esempio `admin-initiate-auth` seguente accede all'utente diego@example.com. Questo esempio include anche metadati per la protezione dalle minacce e ClientMetadata per i trigger Lambda. L'utente è configurato per l'autenticazione a più fattori (MFA) basata su password monouso e riceve una richiesta di verifica che richiede l'immissione di un codice nell'app di autenticazione in uso prima di poter completare l'autenticazione.

```
aws cognito-idp admin-initiate-auth \
    --user-pool-id us-west-2_EXAMPLE \
```

```
    --client-id 1example23456789 \
    --auth-flow ADMIN_USER_PASSWORD_AUTH \
    --auth-parameters USERNAME=diego@example.com,PASSWORD="My@Example
$Password3!",SECRET_HASH=ExampleEncodedClientIdSecretAndUsername= \
    --context-data="{\"EncodedData\":\"abc123example\",\"HttpHeaders\":
[{\"headerName\":\"UserAgent\",\"headerValue\":\"Mozilla/5.0 (Windows NT 6.1;
 Win64; x64; rv:47.0) Gecko/20100101 Firefox/47.0\"}],\"IpAddress\":\"192.0.2.1\",
\"ServerName\":\"example.com\",\"ServerPath\":\"/login\"}" \
    --client-metadata="{\"MyExampleKey\": \"MyExampleValue\"}"
```

Output:

```
{
    "ChallengeName": "SOFTWARE_TOKEN_MFA",
    "Session": "AYABeExample...",
    "ChallengeParameters": {
        "FRIENDLY_DEVICE_NAME": "MyAuthenticatorApp",
        "USER_ID_FOR_SRP": "diego@example.com"
    }
}
```

Per ulteriori informazioni, consulta Flusso di autenticazione amministratore nella Guida per gli sviluppatori di Amazon Cognito.

- Per i dettagli sull'API, consulta Command AdminInitiateAuthReference AWS CLI .

admin-link-provider-for-user

Il seguente esempio di codice mostra come utilizzare`admin-link-provider-for-user`.

AWS CLI

Come collegare un utente locale a un utente federato

L'esempio `admin-link-provider-for-user` seguente collega l'utente locale diego a un utente che eseguirà l'accesso federato con Google.

```
aws cognito-idp admin-link-provider-for-user \
    --user-pool-id us-west-2_EXAMPLE \
    --destination-user ProviderName=Cognito,ProviderAttributeValue=diego \
    --source-
user ProviderAttributeName=Cognito_Subject,ProviderAttributeValue=0000000000000000,ProviderN
```

Per ulteriori informazioni, consulta Collegamento di utenti federati a un profilo utente esistente nella Guida per gli sviluppatori di Amazon Cognito.

- Per i dettagli sull'API, consulta AdminLinkProviderForUser AWS CLICommand Reference.

`admin-list-devices`

Il seguente esempio di codice mostra come utilizzare`admin-list-devices`.

AWS CLI

Come elencare i dispositivi per un utente

L'esempio `admin-list-devices` seguente elenca i dispositivi per l'utente diego.

```
aws cognito-idp admin-list-devices \
    --user-pool-id us-west-2_EXAMPLE \
    --username diego \
    --limit 1
```

Output:

```
{
    "Devices": [
        {
            "DeviceKey": "us-west-2_a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
            "DeviceAttributes": [
                {
                    "Name": "device_status",
                    "Value": "valid"
                },
                {
                    "Name": "device_name",
                    "Value": "MyDevice"
                },
                {
                    "Name": "dev:device_arn",
                    "Value": "arn:aws:cognito-idp:us-west-2:123456789012:owner/
diego.us-west-2_EXAMPLE/device/us-west-2_a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
                },
                {
                    "Name": "dev:device_owner",
                    "Value": "diego.us-west-2_EXAMPLE"
```

```
                },
                {
                    "Name": "last_ip_used",
                    "Value": "192.0.2.1"
                },
                {
                    "Name": "dev:device_remembered_status",
                    "Value": "remembered"
                },
                {
                    "Name": "dev:device_sdk",
                    "Value": "aws-sdk"
                }
            ],
            "DeviceCreateDate": 1715100742.022,
            "DeviceLastModifiedDate": 1723233651.167,
            "DeviceLastAuthenticatedDate": 1715100742.0
        }
    ]
}
```

Per ulteriori informazioni, consulta Utilizzo dei dispositivi utente nel pool di utenti nella Guida per gli sviluppatori di Amazon Cognito.

- Per i dettagli sull'API, consulta AdminListDevices AWS CLICommand Reference.

`admin-list-groups-for-user`

Il seguente esempio di codice mostra come utilizzare`admin-list-groups-for-user`.

AWS CLI

Come elencare i gruppi per un utente

Questo esempio elenca i gruppi per il nome utente jane@example.com.

Comando:

```
aws cognito-idp admin-list-groups-for-user --user-pool-id us-west-2_aaaaaaaaa --
username diego@example.com
```

Output:

```
{
  "Groups": [
      {
          "Description": "Sample group",
          "Precedence": 1,
          "LastModifiedDate": 1548097827.125,
          "RoleArn": "arn:aws:iam::111111111111:role/SampleRole",
          "GroupName": "SampleGroup",
          "UserPoolId": "us-west-2_aaaaaaaaa",
          "CreationDate": 1548097827.125
      }
  ]
}
```

- Per i dettagli sull'API, consulta AdminListGroupsForUser AWS CLICommand Reference.

`admin-list-user-auth-events`

Il seguente esempio di codice mostra come utilizzare`admin-list-user-auth-events`.

AWS CLI

Come elencare gli eventi di autorizzazione per un utente

L'esempio `admin-list-user-auth-events` seguente elenca l'evento del log delle attività dell'utente più recenti per l'utente diego.

```
aws cognito-idp admin-list-user-auth-events \
    --user-pool-id us-west-2_ywDJHlIfU \
    --username brcotter+050123 \
    --max-results 1
```

Output:

```
{
    "AuthEvents": [
        {
            "EventId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
            "EventType": "SignIn",
            "CreationDate": 1726694203.495,
            "EventResponse": "InProgress",
```

```
            "EventRisk": {
                "RiskDecision": "AccountTakeover",
                "RiskLevel": "Medium",
                "CompromisedCredentialsDetected": false
            },
            "ChallengeResponses": [
                {
                    "ChallengeName": "Password",
                    "ChallengeResponse": "Success"
                }
            ],
            "EventContextData": {
                "IpAddress": "192.0.2.1",
                "City": "Seattle",
                "Country": "United States"
            }
        }
    ],
    "NextToken": "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222#2024-09-18T21:16:43.495Z"
}
```

Per ulteriori informazioni, consulta Visualizzazione ed esportazione della cronologia degli eventi dell'utente nella Guida per gli sviluppatori di Amazon Cognito.

- Per i dettagli sull'API, consulta AdminListUserAuthEvents AWS CLICommand Reference.

`admin-remove-user-from-group`

Il seguente esempio di codice mostra come utilizzare`admin-remove-user-from-group`.

AWS CLI

Come rimuovere un utente da un gruppo

Questo esempio rimuove jane@example.com da SampleGroup.

Comando:

```
aws cognito-idp admin-remove-user-from-group --user-pool-id us-west-2_aaaaaaaaa --
username jane@example.com --group-name SampleGroup
```

- Per i dettagli sull'API, consulta AdminRemoveUserFromGroup AWS CLICommand Reference.

`admin-reset-user-password`

Il seguente esempio di codice mostra come utilizzare`admin-reset-user-password`.

AWS CLI

Come reimpostare la password di un utente

Questo esempio reimposta la password per diego@example.com.

Comando:

```
aws cognito-idp admin-reset-user-password --user-pool-id us-west-2_aaaaaaaaa --
username diego@example.com
```

- Per i dettagli sull'API, consulta AdminResetUserPassword AWS CLICommand Reference.

`admin-respond-to-auth-challenge`

Il seguente esempio di codice mostra come utilizzare`admin-respond-to-auth-challenge`.

AWS CLI

Come rispondere a una richiesta di autenticazione

Esistono molti modi per rispondere alle diverse richieste di autenticazione, a seconda del flusso di autenticazione, della configurazione del pool di utenti e delle impostazioni utente. L'esempio `admin-respond-to-auth-challenge` seguente fornisce un codice MFA TOTP per diego@example.com e completa l'accesso. In questo pool di utenti è attivata la memorizzazione dei dispositivi, quindi il risultato dell'autenticazione restituisce anche una nuova chiave del dispositivo.

```
aws cognito-idp admin-respond-to-auth-challenge \
    --user-pool-id us-west-2_EXAMPLE \
    --client-id 1example23456789 \
    --challenge-name SOFTWARE_TOKEN_MFA \
    --challenge-responses USERNAME=diego@example.com,SOFTWARE_TOKEN_MFA_CODE=000000
 \
    --session AYABeExample...
```

Output:

```
{
    "ChallengeParameters": {},
    "AuthenticationResult": {
        "AccessToken": "eyJra456defEXAMPLE",
        "ExpiresIn": 3600,
        "TokenType": "Bearer",
        "RefreshToken": "eyJra123abcEXAMPLE",
        "IdToken": "eyJra789ghiEXAMPLE",
        "NewDeviceMetadata": {
            "DeviceKey": "us-west-2_a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
            "DeviceGroupKey": "-ExAmPlE1"
        }
    }
}
```

Per ulteriori informazioni, consulta Flusso di autenticazione amministratore nella Guida per gli sviluppatori di Amazon Cognito.

- Per i dettagli sull'API, consulta AdminRespondToAuthChallenge AWS CLICommand Reference.

`admin-set-user-mfa-preference`

Il seguente esempio di codice mostra come utilizzare`admin-set-user-mfa-preference`.

AWS CLI

Come impostare la preferenza MFA dell'utente

Questo esempio imposta la preferenza SMS MFA per il nome utente diego@example.com.

Comando:

```
aws cognito-idp admin-set-user-mfa-preference --user-pool-id us-west-2_aaaaaaaaa --
username diego@example.com --sms-mfa-settings Enabled=false,PreferredMfa=false
```

- Per i dettagli sull'API, consulta AdminSetUserMfaPreference AWS CLICommand Reference.

`admin-set-user-password`

Il seguente esempio di codice mostra come utilizzare`admin-set-user-password`.

AWS CLI

Come impostare una password utente come amministratore

L'esempio `admin-set-user-password` seguente imposta in modo permanente la password per diego@example.com.

```
aws cognito-idp admin-set-user-password \
    --user-pool-id us-west-2_EXAMPLE \
    --username diego@example.com \
    --password MyExamplePassword1! \
    --permanent
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Password, recupero delle password e policy delle password nella Guida per gli sviluppatori di Amazon Cognito.

- Per i dettagli sull'API, consulta AdminSetUserPassword AWS CLICommand Reference.

`admin-set-user-settings`

Il seguente esempio di codice mostra come utilizzare`admin-set-user-settings`.

AWS CLI

Come configurare le impostazioni utente

Questo esempio imposta la preferenza di consegna MFA per il nome utente diego@example.com su EMAIL.

Comando:

```
aws cognito-idp admin-set-user-settings --user-pool-id us-west-2_aaaaaaaaa --
username diego@example.com --mfa-options DeliveryMedium=EMAIL
```

- Per i dettagli sull'API, consulta AdminSetUserSettings AWS CLICommand Reference.

`admin-update-auth-event-feedback`

Il seguente esempio di codice mostra come utilizzare`admin-update-auth-event-feedback`.

AWS CLI

Come fornire un feedback per un evento di autorizzazione

Questo esempio imposta il valore di feedback per un evento di autorizzazione identificato da event-id su Valido.

Comando:

```
aws cognito-idp admin-update-auth-event-feedback --user-pool-id us-west-2_aaaaaaaaa
 --username diego@example.com --event-id c2c2cf89-c0d3-482d-aba6-99d78a5b0bfe --
feedback-value Valid
```

- Per i dettagli sull'API, consulta AdminUpdateAuthEventFeedback AWS CLICommand Reference.

`admin-update-device-status`

Il seguente esempio di codice mostra come utilizzare`admin-update-device-status`.

AWS CLI

Come aggiornare lo stato del dispositivo

Questo esempio imposta lo stato memorizzato del dispositivo per il dispositivo identificato da device-key su not_remembered.

Comando:

```
aws cognito-idp admin-update-device-status --user-pool-id us-west-2_aaaaaaaaa
 --username diego@example.com --device-key xxxx  --device-remembered-
status not_remembered
```

- Per i dettagli sull'API, consulta AdminUpdateDeviceStatus AWS CLICommand Reference.

`admin-update-user-attributes`

Il seguente esempio di codice mostra come utilizzare`admin-update-user-attributes`.

AWS CLI

Come aggiornare gli attributi utente

Questo esempio aggiorna un attributo utente personalizzato CustomAttr 1 per l'utente diego@example.com.

Comando:

```
aws cognito-idp admin-update-user-attributes --user-pool-id us-
west-2_aaaaaaaaa --username diego@example.com --user-attributes
 Name="custom:CustomAttr1",Value="Purple"
```

- Per i dettagli sull'API, vedere AdminUpdateUserAttributesin AWS CLI Command Reference.

`admin-user-global-sign-out`

Il seguente esempio di codice mostra come utilizzare`admin-user-global-sign-out`.

AWS CLI

Come disconnettere un utente come amministratore

L'esempio `admin-user-global-sign-out` seguente disconnette l'utente diego@example.com.

```
aws cognito-idp admin-user-global-sign-out \
    --user-pool-id us-west-2_EXAMPLE \
    --username diego@example.com
```

Per ulteriori informazioni, consulta Autenticazione con un pool di utenti nella Guida per gli sviluppatori di Amazon Cognito.

- Per i dettagli sull'API, consulta AdminUserGlobalSignOut AWS CLICommand Reference.

`associate-software-token`

Il seguente esempio di codice mostra come utilizzare`associate-software-token`.

AWS CLI

Come generare una chiave segreta per un'app di autenticazione MFA

L'esempio `associate-software-token` seguente genera una chiave privata TOTP per un utente che ha effettuato l'accesso e ha ricevuto un token di accesso. La chiave privata risultante

può essere immessa manualmente in un'app di autenticazione oppure le applicazioni possono visualizzarla come codice QR che l'utente può scansionare.

```
aws cognito-idp associate-software-token \
    --access-token eyJra456defEXAMPLE
```

Output:

```
{
    "SecretCode": "QWERTYUIOP123456EXAMPLE"
}
```

Per ulteriori informazioni, consulta Autenticazione MFA con token di software TOTP nella Guida per gli sviluppatori di Amazon Cognito.

- Per i dettagli sull'API, consulta AssociateSoftwareToken AWS CLICommand Reference.

`change-password`

Il seguente esempio di codice mostra come utilizzare`change-password`.

AWS CLI

Come modificare una password

Questo esempio modifica una password.

Comando:

```
aws cognito-idp change-password --previous-password OldPassword --proposed-
password NewPassword --access-token ACCESS_TOKEN
```

- Per i dettagli sull'API, consulta ChangePassword AWS CLICommand Reference.

`confirm-device`

Il seguente esempio di codice mostra come utilizzare`confirm-device`.

AWS CLI

Come confermare un dispositivo utente

L'esempio `confirm-device` seguente aggiunge un nuovo dispositivo memorizzato per l'utente corrente.

```
aws cognito-idp confirm-device \
   --access-token eyJra456defEXAMPLE \
   --device-key us-west-2_a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 \
   --device-secret-verifier-
config PasswordVerifier=TXlWZXJpZmllclN0cmluZw,Salt=TXlTUlBTYWx0
```

Output:

```
{
     "UserConfirmationNecessary": false
}
```

Per ulteriori informazioni, consulta Utilizzo dei dispositivi utente nel pool di utenti nella Guida per gli sviluppatori di Amazon Cognito.

- Per i dettagli sull'API, consulta ConfirmDevice AWS CLICommand Reference.

`confirm-forgot-password`

Il seguente esempio di codice mostra come utilizzare`confirm-forgot-password`.

AWS CLI

Come confermare una password dimenticata

Questo esempio conferma una password dimenticata per diego@example.com.

Comando:

```
aws cognito-idp confirm-forgot-password --client-id 3n4b5urk1ft4fl3mg5e62d9ado --
username=diego@example.com --password PASSWORD --confirmation-code CONF_CODE
```

- Per i dettagli sull'API, consulta ConfirmForgotPassword AWS CLICommand Reference.

`confirm-sign-up`

Il seguente esempio di codice mostra come utilizzare`confirm-sign-up`.

AWS CLI

Come confermare la registrazione

Questo esempio conferma la registrazione per il nome utente diego@example.com.

Comando:

```
aws cognito-idp confirm-sign-up --client-id 3n4b5urk1ft4fl3mg5e62d9ado --
username=diego@example.com --confirmation-code CONF_CODE
```

- Per i dettagli sull'API, consulta ConfirmSignUp AWS CLICommand Reference.

`create-group`

Il seguente esempio di codice mostra come utilizzare`create-group`.

AWS CLI

Come creare un gruppo

Questo esempio crea un gruppo con una descrizione.

Comando:

```
aws cognito-idp create-group --user-pool-id us-west-2_aaaaaaaaa --group-
name MyNewGroup --description "New group."
```

Output:

```
{
  "Group": {
      "GroupName": "MyNewGroup",
      "UserPoolId": "us-west-2_aaaaaaaaa",
      "Description": "New group.",
      "LastModifiedDate": 1548270073.795,
      "CreationDate": 1548270073.795
  }
}
```

Come creare un gruppo con un ruolo e una precedenza

Questo esempio crea un gruppo con una descrizione. Include anche un ruolo e una precedenza.

Comando:

```
aws cognito-idp create-group --user-pool-id us-west-2_aaaaaaaaa --group-
name MyNewGroupWithRole --description "New group with a role." --role-
arn arn:aws:iam::111111111111:role/MyNewGroupRole --precedence 2
```

Output:

```
{
  "Group": {
      "GroupName": "MyNewGroupWithRole",
      "UserPoolId": "us-west-2_aaaaaaaaa",
      "Description": "New group with a role.",
      "RoleArn": "arn:aws:iam::111111111111:role/MyNewGroupRole",
      "Precedence": 2,
      "LastModifiedDate": 1548270211.761,
      "CreationDate": 1548270211.761
  }
}
```

- Per i dettagli sull'API, consulta CreateGroup AWS CLICommand Reference.

`create-identity-provider`

Il seguente esempio di codice mostra come utilizzare`create-identity-provider`.

AWS CLI

Esempio 1: come creare un gestore dell'identità digitale SAML del pool di utenti con un URL dei metadati

L'esempio `create-identity-provider` seguente crea un nuovo IdP SAML con metadati da un URL pubblico, mappatura degli attributi e due identificatori.

```
aws cognito-idp create-identity-provider \
    --user-pool-id us-west-2_EXAMPLE \
    --provider-name MySAML \
    --provider-type SAML \
```

```
    --provider-
details IDPInit=true,IDPSignout=true,EncryptedResponses=true,MetadataURL=https://
auth.example.com/sso/saml/metadata,RequestSigningAlgorithm=rsa-sha256 \
    --attribute-mapping email=emailaddress,phone_number=phone,custom:111=department
 \
    --idp-identifiers CorpSAML WestSAML
```

Output:

```
{
    "IdentityProvider": {
        "UserPoolId": "us-west-2_EXAMPLE",
        "ProviderName": "MySAML",
        "ProviderType": "SAML",
        "ProviderDetails": {
            "ActiveEncryptionCertificate": "MIICvTCCAaEXAMPLE",
            "EncryptedResponses": "true",
            "IDPInit": "true",
            "IDPSignout": "true",
            "MetadataURL": "https://auth.example.com/sso/saml/metadata",
            "RequestSigningAlgorithm": "rsa-sha256",
            "SLORedirectBindingURI": "https://auth.example.com/slo/saml",
            "SSORedirectBindingURI": "https://auth.example.com/sso/saml"
        },
        "AttributeMapping": {
            "custom:111": "department",
            "emailaddress": "email",
            "phone": "phone_number"
        },
        "IdpIdentifiers": [
            "CorpSAML",
            "WestSAML"
        ],
        "LastModifiedDate": 1726853833.977,
        "CreationDate": 1726853833.977
    }
}
```

Per ulteriori informazioni, consulta Aggiunta di un accesso al pool di utenti tramite terze parti nella Guida per gli sviluppatori di Amazon Cognito.

Esempio 2: come creare un gestore dell'identità digitale SAML del pool di utenti con un file dei metadati

L'esempio `create-identity-provider` seguente crea un nuovo IdP SAML con metadati da un file, mappatura degli attributi e due identificatori. La sintassi dei file può variare tra i diversi sistemi operativi nel parametro `--provider-details`. Il modo più semplice per eseguire questa operazione è creare un file di input JSON:

```
aws cognito-idp create-identity-provider \
    --cli-input-json file://.\SAML-identity-provider.json
```

Contenuto di `SAML-identity-provider.json`:

```
{
    "AttributeMapping": {
        "email" : "idp_email",
        "email_verified" : "idp_email_verified"
    },
    "IdpIdentifiers": [ "platform" ],
    "ProviderDetails": {
        "MetadataFile": "<md:EntityDescriptor xmlns:md=
\"urn:oasis:names:tc:SAML:2.0:metadata\" entityID=\"http://www.example.com/
sso\"><md:IDPSSODescriptor WantAuthnRequestsSigned=\"false\"
 protocolSupportEnumeration=\"urn:oasis:names:tc:SAML:2.0:protocol
\"><md:KeyDescriptor use=\"signing\"><ds:KeyInfo xmlns:ds=\"http://
www.w3.org/2000/09/xmldsig#
\"><ds:X509Data><ds:X509Certificate>[IDP_CERTIFICATE_DATA]</ds:X509Certificate></
ds:X509Data></ds:KeyInfo></md:KeyDescriptor><md:SingleLogoutService
 Binding=\"urn:oasis:names:tc:SAML:2.0:bindings:HTTP-POST\" Location=
\"https://www.example.com/slo/saml\"/><md:SingleLogoutService
 Binding=\"urn:oasis:names:tc:SAML:2.0:bindings:HTTP-
Redirect\" Location=\"https://www.example.com/slo/saml\"/
><md:NameIDFormat>urn:oasis:names:tc:SAML:1.1:nameid-format:unspecified</
md:NameIDFormat><md:NameIDFormat>urn:oasis:names:tc:SAML:1.1:nameid-
format:emailAddress</md:NameIDFormat><md:SingleSignOnService
 Binding=\"urn:oasis:names:tc:SAML:2.0:bindings:HTTP-POST\" Location=
\"https://www.example.com/sso/saml\"/><md:SingleSignOnService Binding=
\"urn:oasis:names:tc:SAML:2.0:bindings:HTTP-Redirect\" Location=\"https://
www.example.com/sso/saml\"/></md:IDPSSODescriptor></md:EntityDescriptor>",
        "IDPSignout" : "true",
        "RequestSigningAlgorithm" : "rsa-sha256",
        "EncryptedResponses" : "true",
        "IDPInit" : "true"
    },
    "ProviderName": "MySAML2",
```

```
    "ProviderType": "SAML",
    "UserPoolId": "us-west-2_EXAMPLE"
}
```

Output:

```
{
    "IdentityProvider": {
        "UserPoolId": "us-west-2_EXAMPLE",
        "ProviderName": "MySAML2",
        "ProviderType": "SAML",
        "ProviderDetails": {
            "ActiveEncryptionCertificate":
 "[USER_POOL_ENCRYPTION_CERTIFICATE_DATA]",
            "EncryptedResponses": "true",
            "IDPInit": "true",
            "IDPSignout": "true",
            "MetadataFile": "<md:EntityDescriptor xmlns:md=
\"urn:oasis:names:tc:SAML:2.0:metadata\" entityID=\"http://www.example.com/
sso\"><md:IDPSSODescriptor WantAuthnRequestsSigned=\"false\"
 protocolSupportEnumeration=\"urn:oasis:names:tc:SAML:2.0:protocol
\"><md:KeyDescriptor use=\"signing\"><ds:KeyInfo xmlns:ds=\"http://
www.w3.org/2000/09/xmldsig#
\"><ds:X509Data><ds:X509Certificate>[IDP_CERTIFICATE_DATA]</ds:X509Certificate></
ds:X509Data></ds:KeyInfo></md:KeyDescriptor><md:SingleLogoutService
 Binding=\"urn:oasis:names:tc:SAML:2.0:bindings:HTTP-POST\" Location=
\"https://www.example.com/slo/saml\"/><md:SingleLogoutService
 Binding=\"urn:oasis:names:tc:SAML:2.0:bindings:HTTP-
Redirect\" Location=\"https://www.example.com/slo/saml\"/
><md:NameIDFormat>urn:oasis:names:tc:SAML:1.1:nameid-format:unspecified</
md:NameIDFormat><md:NameIDFormat>urn:oasis:names:tc:SAML:1.1:nameid-
format:emailAddress</md:NameIDFormat><md:SingleSignOnService
 Binding=\"urn:oasis:names:tc:SAML:2.0:bindings:HTTP-POST\" Location=
\"https://www.example.com/sso/saml\"/><md:SingleSignOnService Binding=
\"urn:oasis:names:tc:SAML:2.0:bindings:HTTP-Redirect\" Location=\"https://
www.example.com/sso/saml\"/></md:IDPSSODescriptor></md:EntityDescriptor>",
            "RequestSigningAlgorithm": "rsa-sha256",
            "SLORedirectBindingURI": "https://www.example.com/slo/saml",
            "SSORedirectBindingURI": "https://www.example.com/sso/saml"
        },
        "AttributeMapping": {
            "email": "idp_email",
            "email_verified": "idp_email_verified"
```

```
        },
        "IdpIdentifiers": [
            "platform"
        ],
        "LastModifiedDate": 1726855290.731,
        "CreationDate": 1726855290.731
    }
}
```

Per ulteriori informazioni, consulta Aggiunta di un accesso al pool di utenti tramite terze parti nella Guida per gli sviluppatori di Amazon Cognito.

- Per i dettagli sull'API, consulta CreateIdentityProvider AWS CLICommand Reference.

`create-resource-server`

Il seguente esempio di codice mostra come utilizzare`create-resource-server`.

AWS CLI

Come creare un client del pool di utenti

L'esempio `create-resource-server` seguente crea un nuovo server di risorse con ambiti personalizzati.

```
aws cognito-idp create-resource-server \
    --user-pool-id us-west-2_EXAMPLE \
    --identifier solar-system-data \
    --name "Solar system object tracker" \
    --scopes ScopeName=sunproximity.read,ScopeDescription="Distance in AU from Sol"
 ScopeName=asteroids.add,ScopeDescription="Enter a new asteroid"
```

Output:

```
{
    "ResourceServer": {
        "UserPoolId": "us-west-2_EXAMPLE",
        "Identifier": "solar-system-data",
        "Name": "Solar system object tracker",
        "Scopes": [
            {
                "ScopeName": "sunproximity.read",
```

```
                "ScopeDescription": "Distance in AU from Sol"
            },
            {
                "ScopeName": "asteroids.add",
                "ScopeDescription": "Enter a new asteroid"
            }
        ]
    }
}
```

Per ulteriori informazioni, consulta Scopes, M2M e APIs with resource servers nella Amazon Cognito Developer Guide.

- Per i dettagli sull'API, consulta Command CreateResourceServerReference AWS CLI .

`create-user-import-job`

Il seguente esempio di codice mostra come utilizzare`create-user-import-job`.

AWS CLI

Come creare il processo di importazione degli utenti

Questo esempio crea un processo di importazione utente denominato MyImportJob.

Per ulteriori informazioni sull'importazione di utenti, consulta Importazione di utenti nel pool di utenti da un file CSV.

Comando:

```
aws cognito-idp create-user-import-job --user-pool-id us-west-2_aaaaaaaaa --
job-name MyImportJob --cloud-watch-logs-role-arn arn:aws:iam::111111111111:role/
CognitoCloudWatchLogsRole
```

Output:

```
{
  "UserImportJob": {
      "JobName": "MyImportJob",
      "JobId": "import-qQ0DCt2fRh",
      "UserPoolId": "us-west-2_aaaaaaaaa",
      "PreSignedUrl": "PRE_SIGNED_URL",
```

```
      "CreationDate": 1548271795.471,
      "Status": "Created",
      "CloudWatchLogsRoleArn": "arn:aws:iam::111111111111:role/
CognitoCloudWatchLogsRole",
      "ImportedUsers": 0,
      "SkippedUsers": 0,
      "FailedUsers": 0
  }
}
```

Caricare il file .csv con curl utilizzando l'URL pre-firmato:

Comando:

```
curl -v -T "PATH_TO_CSV_FILE" -H "x-amz-server-side-encryption:aws:kms"
 "PRE_SIGNED_URL"
```

- Per i dettagli sull'API, vedere CreateUserImportJobin AWS CLI Command Reference.

`create-user-pool-client`

Il seguente esempio di codice mostra come utilizzare`create-user-pool-client`.

AWS CLI

Come creare un client del pool di utenti

L'`create-user-pool-client`esempio seguente crea un nuovo client con pool di utenti con un client secret, attributi di lettura e scrittura espliciti, accesso con flussi username-password e SRP, accesso con tre, accesso a un sottoinsieme di OAuth ambiti IdPs, PinPoint analisi e una validità estesa della sessione di autenticazione.

```
aws cognito-idp create-user-pool-client \
    --user-pool-id us-west-2_EXAMPLE \
    --client-name MyTestClient \
    --generate-secret \
    --refresh-token-validity 10 \
    --access-token-validity 60 \
    --id-token-validity 60 \
    --token-validity-units AccessToken=minutes,IdToken=minutes,RefreshToken=days \
    --read-attributes email phone_number email_verified phone_number_verified \
    --write-attributes email phone_number \
```

```
    --explicit-auth-
flows ALLOW_USER_PASSWORD_AUTH ALLOW_USER_SRP_AUTH ALLOW_REFRESH_TOKEN_AUTH \
    --supported-identity-providers Google Facebook MyOIDC \
    --callback-urls https://www.amazon.com https://example.com http://
localhost:8001 myapp://example \
    --allowed-o-auth-flows code implicit \
    --allowed-o-auth-scopes openid profile aws.cognito.signin.user.admin solar-
system-data/asteroids.add \
    --allowed-o-auth-flows-user-pool-client \
    --analytics-configuration ApplicationArn=arn:aws:mobiletargeting:us-
west-2:767671399759:apps/thisisanexamplepinpointapplicationid,UserDataShared=TRUE \
    --prevent-user-existence-errors ENABLED \
    --enable-token-revocation \
    --enable-propagate-additional-user-context-data \
    --auth-session-validity 4
```

Output:

```
{
    "UserPoolClient": {
        "UserPoolId": "us-west-2_EXAMPLE",
        "ClientName": "MyTestClient",
        "ClientId": "123abc456defEXAMPLE",
        "ClientSecret": "this1234is5678my91011example1213client1415secret",
        "LastModifiedDate": 1726788459.464,
        "CreationDate": 1726788459.464,
        "RefreshTokenValidity": 10,
        "AccessTokenValidity": 60,
        "IdTokenValidity": 60,
        "TokenValidityUnits": {
            "AccessToken": "minutes",
            "IdToken": "minutes",
            "RefreshToken": "days"
        },
        "ReadAttributes": [
            "email_verified",
            "phone_number_verified",
            "phone_number",
            "email"
        ],
        "WriteAttributes": [
            "phone_number",
            "email"
```

```
        ],
        "ExplicitAuthFlows": [
            "ALLOW_USER_PASSWORD_AUTH",
            "ALLOW_USER_SRP_AUTH",
            "ALLOW_REFRESH_TOKEN_AUTH"
        ],
        "SupportedIdentityProviders": [
            "Google",
            "MyOIDC",
            "Facebook"
        ],
        "CallbackURLs": [
            "https://example.com",
            "https://www.amazon.com",
            "myapp://example",
            "http://localhost:8001"
        ],
        "AllowedOAuthFlows": [
            "implicit",
            "code"
        ],
        "AllowedOAuthScopes": [
            "aws.cognito.signin.user.admin",
            "openid",
            "profile",
            "solar-system-data/asteroids.add"
        ],
        "AllowedOAuthFlowsUserPoolClient": true,
        "AnalyticsConfiguration": {
            "ApplicationArn": "arn:aws:mobiletargeting:us-west-2:123456789012:apps/
thisisanexamplepinpointapplicationid",
            "RoleArn": "arn:aws:iam::123456789012:role/aws-service-role/cognito-
idp.amazonaws.com/AWSServiceRoleForAmazonCognitoIdp",
            "UserDataShared": true
        },
        "PreventUserExistenceErrors": "ENABLED",
        "EnableTokenRevocation": true,
        "EnablePropagateAdditionalUserContextData": true,
        "AuthSessionValidity": 4
    }
}
```

Per ulteriori informazioni, consulta Impostazioni specifiche dell'applicazione con i client dell'app nella Guida per gli sviluppatori di Amazon Cognito.

- Per i dettagli sull'API, consulta Command Reference. CreateUserPoolClientAWS CLI

`create-user-pool-domain`

Il seguente esempio di codice mostra come utilizzare`create-user-pool-domain`.

AWS CLI

Esempio 1: come creare un dominio del pool di utenti

L'esempio `create-user-pool-domain` seguente crea un nuovo dominio personalizzato.

```
aws cognito-idp create-user-pool-domain \
    --user-pool-id us-west-2_EXAMPLE \
    --domain auth.example.com \
    --custom-domain-config CertificateArn=arn:aws:acm:us-
east-1:123456789012:certificate/a1b2c3d4-5678-90ab-cdef-EXAMPLE22222
```

Output:

```
{
    "CloudFrontDomain": "example1domain.cloudfront.net"
}
```

Per ulteriori informazioni, consulta Configurazione di un dominio del pool di utenti nella Guida per gli sviluppatori di Amazon Cognito.

Esempio 2: come creare un dominio del pool di utenti

L'esempio `create-user-pool-domain` seguente crea un nuovo dominio con un prefisso di proprietà del servizio.

```
aws cognito-idp create-user-pool-domain \
    --user-pool-id us-west-2_EXAMPLE2 \
    --domain mydomainprefix
```

Per ulteriori informazioni, consulta Configurazione di un dominio del pool di utenti nella Guida per gli sviluppatori di Amazon Cognito.

- Per i dettagli sull'API, consulta CreateUserPoolDomain AWS CLICommand Reference.

`create-user-pool`

Il seguente esempio di codice mostra come utilizzare`create-user-pool`.

AWS CLI

Come creare un pool di utenti con configurazione minima

Questo esempio crea un pool di utenti denominato MyUserPool utilizzando valori predefiniti. Non ci sono attributi obbligatori né client di applicazioni. MFA e sicurezza avanzata sono disattivate.

Comando:

```
aws cognito-idp create-user-pool --pool-name MyUserPool
```

Output:

```
{
  "UserPool": {
      "SchemaAttributes": [
          {
              "Name": "sub",
              "StringAttributeConstraints": {
                  "MinLength": "1",
                  "MaxLength": "2048"
              },
              "DeveloperOnlyAttribute": false,
              "Required": true,
              "AttributeDataType": "String",
              "Mutable": false
          },
          {
              "Name": "name",
              "StringAttributeConstraints": {
                  "MinLength": "0",
                  "MaxLength": "2048"
              },
              "DeveloperOnlyAttribute": false,
              "Required": false,
              "AttributeDataType": "String",
              "Mutable": true
          },
          {
              "Name": "given_name",
```

```
            "StringAttributeConstraints": {
                "MinLength": "0",
                "MaxLength": "2048"
            },
            "DeveloperOnlyAttribute": false,
            "Required": false,
            "AttributeDataType": "String",
            "Mutable": true
        },
        {
            "Name": "family_name",
            "StringAttributeConstraints": {
                "MinLength": "0",
                "MaxLength": "2048"
            },
            "DeveloperOnlyAttribute": false,
            "Required": false,
            "AttributeDataType": "String",
            "Mutable": true
        },
        {
            "Name": "middle_name",
            "StringAttributeConstraints": {
                "MinLength": "0",
                "MaxLength": "2048"
            },
            "DeveloperOnlyAttribute": false,
            "Required": false,
            "AttributeDataType": "String",
            "Mutable": true
        },
        {
            "Name": "nickname",
            "StringAttributeConstraints": {
                "MinLength": "0",
                "MaxLength": "2048"
            },
            "DeveloperOnlyAttribute": false,
            "Required": false,
            "AttributeDataType": "String",
            "Mutable": true
        },
        {
            "Name": "preferred_username",
```

```
            "StringAttributeConstraints": {
                "MinLength": "0",
                "MaxLength": "2048"
            },
            "DeveloperOnlyAttribute": false,
            "Required": false,
            "AttributeDataType": "String",
            "Mutable": true
        },
        {
            "Name": "profile",
            "StringAttributeConstraints": {
                "MinLength": "0",
                "MaxLength": "2048"
            },
            "DeveloperOnlyAttribute": false,
            "Required": false,
            "AttributeDataType": "String",
            "Mutable": true
        },
        {
            "Name": "picture",
            "StringAttributeConstraints": {
                "MinLength": "0",
                "MaxLength": "2048"
            },
            "DeveloperOnlyAttribute": false,
            "Required": false,
            "AttributeDataType": "String",
            "Mutable": true
        },
        {
            "Name": "website",
            "StringAttributeConstraints": {
                "MinLength": "0",
                "MaxLength": "2048"
            },
            "DeveloperOnlyAttribute": false,
            "Required": false,
            "AttributeDataType": "String",
            "Mutable": true
        },
        {
            "Name": "email",
```

```
            "StringAttributeConstraints": {
                "MinLength": "0",
                "MaxLength": "2048"
            },
            "DeveloperOnlyAttribute": false,
            "Required": false,
            "AttributeDataType": "String",
            "Mutable": true
        },
        {
            "AttributeDataType": "Boolean",
            "DeveloperOnlyAttribute": false,
            "Required": false,
            "Name": "email_verified",
            "Mutable": true
        },
        {
            "Name": "gender",
            "StringAttributeConstraints": {
                "MinLength": "0",
                "MaxLength": "2048"
            },
            "DeveloperOnlyAttribute": false,
            "Required": false,
            "AttributeDataType": "String",
            "Mutable": true
        },
        {
            "Name": "birthdate",
            "StringAttributeConstraints": {
                "MinLength": "10",
                "MaxLength": "10"
            },
            "DeveloperOnlyAttribute": false,
            "Required": false,
            "AttributeDataType": "String",
            "Mutable": true
        },
        {
            "Name": "zoneinfo",
            "StringAttributeConstraints": {
                "MinLength": "0",
                "MaxLength": "2048"
            },
```

```
            "DeveloperOnlyAttribute": false,
            "Required": false,
            "AttributeDataType": "String",
            "Mutable": true
        },
        {
            "Name": "locale",
            "StringAttributeConstraints": {
                "MinLength": "0",
                "MaxLength": "2048"
            },
            "DeveloperOnlyAttribute": false,
            "Required": false,
            "AttributeDataType": "String",
            "Mutable": true
        },
        {
            "Name": "phone_number",
            "StringAttributeConstraints": {
                "MinLength": "0",
                "MaxLength": "2048"
            },
            "DeveloperOnlyAttribute": false,
            "Required": false,
            "AttributeDataType": "String",
            "Mutable": true
        },
        {
            "AttributeDataType": "Boolean",
            "DeveloperOnlyAttribute": false,
            "Required": false,
            "Name": "phone_number_verified",
            "Mutable": true
        },
        {
            "Name": "address",
            "StringAttributeConstraints": {
                "MinLength": "0",
                "MaxLength": "2048"
            },
            "DeveloperOnlyAttribute": false,
            "Required": false,
            "AttributeDataType": "String",
            "Mutable": true
```

```
            },
            {
                "Name": "updated_at",
                "NumberAttributeConstraints": {
                    "MinValue": "0"
                },
                "DeveloperOnlyAttribute": false,
                "Required": false,
                "AttributeDataType": "Number",
                "Mutable": true
            }
        ],
        "MfaConfiguration": "OFF",
        "Name": "MyUserPool",
        "LastModifiedDate": 1547833345.777,
        "AdminCreateUserConfig": {
            "UnusedAccountValidityDays": 7,
            "AllowAdminCreateUserOnly": false
        },
        "EmailConfiguration": {},
        "Policies": {
            "PasswordPolicy": {
                "RequireLowercase": true,
                "RequireSymbols": true,
                "RequireNumbers": true,
                "MinimumLength": 8,
                "RequireUppercase": true
            }
        },
        "CreationDate": 1547833345.777,
        "EstimatedNumberOfUsers": 0,
        "Id": "us-west-2_aaaaaaaaa",
        "LambdaConfig": {}
    }
}
```

Come creare un pool di utenti con due attributi richiesti

Questo esempio crea un pool di utenti MyUserPool. Il pool è configurato per accettare l'e-mail come attributo del nome utente. Inoltre, imposta l'indirizzo e-mail di origine su un indirizzo convalidato utilizzando Amazon Simple Email Service (Amazon SES).

Comando:

```
aws cognito-idp create-user-pool --pool-name MyUserPool --username-
attributes "email" --email-configuration=SourceArn="arn:aws:ses:us-
east-1:111111111111:identity/
jane@example.com",ReplyToEmailAddress="jane@example.com"
```

Output:

```
{
  "UserPool": {
      "SchemaAttributes": [
          {
              "Name": "sub",
              "StringAttributeConstraints": {
                  "MinLength": "1",
                  "MaxLength": "2048"
              },
              "DeveloperOnlyAttribute": false,
              "Required": true,
              "AttributeDataType": "String",
              "Mutable": false
          },
          {
              "Name": "name",
              "StringAttributeConstraints": {
                  "MinLength": "0",
                  "MaxLength": "2048"
              },
              "DeveloperOnlyAttribute": false,
              "Required": false,
              "AttributeDataType": "String",
              "Mutable": true
          },
          {
              "Name": "given_name",
              "StringAttributeConstraints": {
                  "MinLength": "0",
                  "MaxLength": "2048"
              },
              "DeveloperOnlyAttribute": false,
              "Required": false,
              "AttributeDataType": "String",
              "Mutable": true
          },
```

```
            {
                "Name": "family_name",
                "StringAttributeConstraints": {
                    "MinLength": "0",
                    "MaxLength": "2048"
                },
                "DeveloperOnlyAttribute": false,
                "Required": false,
                "AttributeDataType": "String",
                "Mutable": true
            },
            {
                "Name": "middle_name",
                "StringAttributeConstraints": {
                    "MinLength": "0",
                    "MaxLength": "2048"
                },
                "DeveloperOnlyAttribute": false,
                "Required": false,
                "AttributeDataType": "String",
                "Mutable": true
            },
            {
                "Name": "nickname",
                "StringAttributeConstraints": {
                    "MinLength": "0",
                    "MaxLength": "2048"
                },
                "DeveloperOnlyAttribute": false,
                "Required": false,
                "AttributeDataType": "String",
                "Mutable": true
            },
            {
                "Name": "preferred_username",
                "StringAttributeConstraints": {
                    "MinLength": "0",
                    "MaxLength": "2048"
                },
                "DeveloperOnlyAttribute": false,
                "Required": false,
                "AttributeDataType": "String",
                "Mutable": true
            },
```

```
            {
                "Name": "profile",
                "StringAttributeConstraints": {
                    "MinLength": "0",
                    "MaxLength": "2048"
                },
                "DeveloperOnlyAttribute": false,
                "Required": false,
                "AttributeDataType": "String",
                "Mutable": true
            },
            {
                "Name": "picture",
                "StringAttributeConstraints": {
                    "MinLength": "0",
                    "MaxLength": "2048"
                },
                "DeveloperOnlyAttribute": false,
                "Required": false,
                "AttributeDataType": "String",
                "Mutable": true
            },
            {
                "Name": "website",
                "StringAttributeConstraints": {
                    "MinLength": "0",
                    "MaxLength": "2048"
                },
                "DeveloperOnlyAttribute": false,
                "Required": false,
                "AttributeDataType": "String",
                "Mutable": true
            },
            {
                "Name": "email",
                "StringAttributeConstraints": {
                    "MinLength": "0",
                    "MaxLength": "2048"
                },
                "DeveloperOnlyAttribute": false,
                "Required": false,
                "AttributeDataType": "String",
                "Mutable": true
            },
```

```
            {
                "AttributeDataType": "Boolean",
                "DeveloperOnlyAttribute": false,
                "Required": false,
                "Name": "email_verified",
                "Mutable": true
            },
            {
                "Name": "gender",
                "StringAttributeConstraints": {
                    "MinLength": "0",
                    "MaxLength": "2048"
                },
                "DeveloperOnlyAttribute": false,
                "Required": false,
                "AttributeDataType": "String",
                "Mutable": true
            },
            {
                "Name": "birthdate",
                "StringAttributeConstraints": {
                    "MinLength": "10",
                    "MaxLength": "10"
                },
                "DeveloperOnlyAttribute": false,
                "Required": false,
                "AttributeDataType": "String",
                "Mutable": true
            },
            {
                "Name": "zoneinfo",
                "StringAttributeConstraints": {
                    "MinLength": "0",
                    "MaxLength": "2048"
                },
                "DeveloperOnlyAttribute": false,
                "Required": false,
                "AttributeDataType": "String",
                "Mutable": true
            },
            {
                "Name": "locale",
                "StringAttributeConstraints": {
                    "MinLength": "0",
```

```
                "MaxLength": "2048"
            },
            "DeveloperOnlyAttribute": false,
            "Required": false,
            "AttributeDataType": "String",
            "Mutable": true
        },
        {
            "Name": "phone_number",
            "StringAttributeConstraints": {
                "MinLength": "0",
                "MaxLength": "2048"
            },
            "DeveloperOnlyAttribute": false,
            "Required": false,
            "AttributeDataType": "String",
            "Mutable": true
        },
        {
            "AttributeDataType": "Boolean",
            "DeveloperOnlyAttribute": false,
            "Required": false,
            "Name": "phone_number_verified",
            "Mutable": true
        },
        {
            "Name": "address",
            "StringAttributeConstraints": {
                "MinLength": "0",
                "MaxLength": "2048"
            },
            "DeveloperOnlyAttribute": false,
            "Required": false,
            "AttributeDataType": "String",
            "Mutable": true
        },
        {
            "Name": "updated_at",
            "NumberAttributeConstraints": {
                "MinValue": "0"
            },
            "DeveloperOnlyAttribute": false,
            "Required": false,
            "AttributeDataType": "Number",
```

```
                "Mutable": true
            }
        ],
        "MfaConfiguration": "OFF",
        "Name": "MyUserPool",
        "LastModifiedDate": 1547837788.189,
        "AdminCreateUserConfig": {
            "UnusedAccountValidityDays": 7,
            "AllowAdminCreateUserOnly": false
        },
        "EmailConfiguration": {
            "ReplyToEmailAddress": "jane@example.com",
            "SourceArn": "arn:aws:ses:us-east-1:111111111111:identity/
jane@example.com"
        },
        "Policies": {
            "PasswordPolicy": {
                "RequireLowercase": true,
                "RequireSymbols": true,
                "RequireNumbers": true,
                "MinimumLength": 8,
                "RequireUppercase": true
            }
        },
        "UsernameAttributes": [
            "email"
        ],
        "CreationDate": 1547837788.189,
        "EstimatedNumberOfUsers": 0,
        "Id": "us-west-2_aaaaaaaaa",
        "LambdaConfig": {}
    }
}
```

- Per i dettagli sull'API, consulta CreateUserPool AWS CLICommand Reference.

`delete-group`

Il seguente esempio di codice mostra come utilizzare`delete-group`.

AWS CLI

Come eliminare un gruppo

Questo esempio elimina un gruppo.

Comando:

```
aws cognito-idp delete-group --user-pool-id us-west-2_aaaaaaaaa  --group-
name MyGroupName
```

- Per i dettagli sull'API, consulta DeleteGroup AWS CLICommand Reference.

`delete-identity-provider`

Il seguente esempio di codice mostra come utilizzare`delete-identity-provider`.

AWS CLI

Come eliminare un gestore dell'identità

Questo esempio elimina un gestore dell'identità digitale.

Comando:

```
aws cognito-idp delete-identity-provider --user-pool-id us-west-2_aaaaaaaaa --
provider-name Facebook
```

- Per i dettagli sull'API, consulta DeleteIdentityProvider AWS CLICommand Reference.

`delete-resource-server`

Il seguente esempio di codice mostra come utilizzare`delete-resource-server`.

AWS CLI

Come eliminare un server di risorse

Questo esempio elimina un server di risorse denominato weather.example.com.

Comando:

```
aws cognito-idp delete-resource-server --user-pool-id us-west-2_aaaaaaaaa --
identifier weather.example.com
```

- Per i dettagli sull'API, consulta DeleteResourceServer AWS CLICommand Reference.

`delete-user-attributes`

Il seguente esempio di codice mostra come utilizzare`delete-user-attributes`.

AWS CLI

Come eliminare un attributo utente

L'esempio `delete-user-attributes` seguente elimina l'attributo personalizzato "custom:attribute" dall'utente che ha effettuato l'accesso.

```
aws cognito-idp delete-user-attributes \
    --access-token ACCESS_TOKEN \
    --user-attribute-names "custom:department"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Utilizzo degli attributi utente nella Guida per gli sviluppatori di Amazon Cognito.

- Per i dettagli sull'API, consulta DeleteUserAttributes AWS CLICommand Reference.

`delete-user-pool-client`

Il seguente esempio di codice mostra come utilizzare`delete-user-pool-client`.

AWS CLI

Come eliminare un client del pool di utenti

Questo esempio elimina un client del pool di utenti.

Comando:

```
aws cognito-idp delete-user-pool-client --user-pool-id us-west-2_aaaaaaaaa --client-
id 38fjsnc484p94kpqsnet7mpld0
```

- Per i dettagli sull'API, consulta DeleteUserPoolClient AWS CLICommand Reference.

`delete-user-pool-domain`

Il seguente esempio di codice mostra come utilizzare`delete-user-pool-domain`.

AWS CLI

Come eliminare un dominio del pool di utenti

L'esempio `delete-user-pool-domain` seguente elimina un dominio del pool di utenti denominato `my-domain`

```
aws cognito-idp delete-user-pool-domain \
    --user-pool-id us-west-2_aaaaaaaaa \
    --domain my-domain
```

- Per i dettagli sull'API, consulta DeleteUserPoolDomain AWS CLICommand Reference.

`delete-user-pool`

Il seguente esempio di codice mostra come utilizzare`delete-user-pool`.

AWS CLI

Come eliminare un pool di utenti

Questo esempio elimina un pool di utenti utilizzando l'ID del pool di utenti us-west-2_aaaaaaaaa.

Comando:

```
aws cognito-idp delete-user-pool --user-pool-id us-west-2_aaaaaaaaa
```

- Per i dettagli sull'API, consulta DeleteUserPool AWS CLICommand Reference.

`delete-user`

Il seguente esempio di codice mostra come utilizzare`delete-user`.

AWS CLI

Come eliminare un utente

Questo esempio elimina un utente.

Comando:

```
aws cognito-idp delete-user --access-token ACCESS_TOKEN
```

- Per i dettagli sull'API, consulta DeleteUser AWS CLICommand Reference.

`describe-identity-provider`

Il seguente esempio di codice mostra come utilizzare`describe-identity-provider`.

AWS CLI

Come descrivere un gestore dell'identità

Questo esempio descrive un gestore dell'identità digitale denominato Facebook.

Comando:

```
aws cognito-idp describe-identity-provider --user-pool-id us-west-2_aaaaaaaaa --
provider-name Facebook
```

Output:

```
{
  "IdentityProvider": {
      "UserPoolId": "us-west-2_aaaaaaaaa",
      "ProviderName": "Facebook",
      "ProviderType": "Facebook",
      "ProviderDetails": {
          "attributes_url": "https://graph.facebook.com/me?fields=",
          "attributes_url_add_attributes": "true",
          "authorize_scopes": myscope",
          "authorize_url": "https://www.facebook.com/v2.9/dialog/oauth",
          "client_id": "11111",
          "client_secret": "11111",
          "token_request_method": "GET",
          "token_url": "https://graph.facebook.com/v2.9/oauth/access_token"
      },
      "AttributeMapping": {
          "username": "id"
      },
      "IdpIdentifiers": [],
      "LastModifiedDate": 1548105901.736,
      "CreationDate": 1548105901.736
  }
}
```

- Per i dettagli sull'API, consulta DescribeIdentityProvider AWS CLICommand Reference.

`describe-resource-server`

Il seguente esempio di codice mostra come utilizzare`describe-resource-server`.

AWS CLI

Come descrivere un server di risorse

Questo esempio descrive il server di risorse weather.example.com.

Comando:

```
aws cognito-idp describe-resource-server --user-pool-id us-west-2_aaaaaaaaa --
identifier weather.example.com
```

Output:

```
{
  "ResourceServer": {
      "UserPoolId": "us-west-2_aaaaaaaaa",
      "Identifier": "weather.example.com",
      "Name": "Weather",
      "Scopes": [
          {
              "ScopeName": "weather.update",
              "ScopeDescription": "Update weather forecast"
          },
          {
              "ScopeName": "weather.read",
              "ScopeDescription": "Read weather forecasts"
          },
          {
              "ScopeName": "weather.delete",
              "ScopeDescription": "Delete a weather forecast"
          }
      ]
  }
}
```

- Per i dettagli sull'API, consulta DescribeResourceServer AWS CLICommand Reference.

`describe-risk-configuration`

Il seguente esempio di codice mostra come utilizzare`describe-risk-configuration`.

AWS CLI

Come descrivere una configurazione di rischio

Questo esempio descrive la configurazione di rischio associata al pool us-west-2_aaaaaaaaa.

Comando:

```
aws cognito-idp describe-risk-configuration --user-pool-id us-west-2_aaaaaaaaa
```

Output:

```
{
  "RiskConfiguration": {
      "UserPoolId": "us-west-2_aaaaaaaaa",
      "CompromisedCredentialsRiskConfiguration": {
          "EventFilter": [
              "SIGN_IN",
              "SIGN_UP",
              "PASSWORD_CHANGE"
          ],
          "Actions": {
              "EventAction": "BLOCK"
          }
      },
      "AccountTakeoverRiskConfiguration": {
          "NotifyConfiguration": {
              "From": "diego@example.com",
              "ReplyTo": "diego@example.com",
              "SourceArn": "arn:aws:ses:us-east-1:111111111111:identity/
diego@example.com",
              "BlockEmail": {
                  "Subject": "Blocked sign-in attempt",
                  "HtmlBody": "<!DOCTYPE html>\n<html>\n<head>\n\t<title>HTML
 email context</title>\n\t<meta charset=\"utf-8\">\n</head>\n<body>\n<pre>We
 blocked an unrecognized sign-in to your account with this information:\n<ul>
\n<li>Time: {login-time}</li>\n<li>Device: {device-name}</li>\n<li>Location: {city},
 {country}</li>\n</ul>\nIf this sign-in was not by you, you should change your
 password and notify us by clicking on <a href={one-click-link-invalid}>this link</
```

```
a>\nIf this sign-in was by you, you can follow <a href={one-click-link-valid}>this
 link</a> to let us know</pre>\n</body>\n</html>",
                    "TextBody": "We blocked an unrecognized sign-in to your account
 with this information:\nTime: {login-time}\nDevice: {device-name}\nLocation:
 {city}, {country}\nIf this sign-in was not by you, you should change your password
 and notify us by clicking on {one-click-link-invalid}\nIf this sign-in was by you,
 you can follow {one-click-link-valid} to let us know"
                },
                "NoActionEmail": {
                    "Subject": "New sign-in attempt",
                    "HtmlBody": "<!DOCTYPE html>\n<html>\n<head>\n\t<title>HTML
 email context</title>\n\t<meta charset=\"utf-8\">\n</head>\n<body>\n<pre>We
 observed an unrecognized sign-in to your account with this information:\n<ul>
\n<li>Time: {login-time}</li>\n<li>Device: {device-name}</li>\n<li>Location: {city},
 {country}</li>\n</ul>\nIf this sign-in was not by you, you should change your
 password and notify us by clicking on <a href={one-click-link-invalid}>this link</
a>\nIf this sign-in was by you, you can follow <a href={one-click-link-valid}>this
 link</a> to let us know</pre>\n</body>\n</html>",
                    "TextBody": "We observed an unrecognized sign-in to your account
 with this information:\nTime: {login-time}\nDevice: {device-name}\nLocation:
 {city}, {country}\nIf this sign-in was not by you, you should change your password
 and notify us by clicking on {one-click-link-invalid}\nIf this sign-in was by you,
 you can follow {one-click-link-valid} to let us know"
                },
                "MfaEmail": {
                    "Subject": "New sign-in attempt",
                    "HtmlBody": "<!DOCTYPE html>\n<html>\n<head>\n\t<title>HTML email
 context</title>\n\t<meta charset=\"utf-8\">\n</head>\n<body>\n<pre>We required
 you to use multi-factor authentication for the following sign-in attempt:\n<ul>
\n<li>Time: {login-time}</li>\n<li>Device: {device-name}</li>\n<li>Location: {city},
 {country}</li>\n</ul>\nIf this sign-in was not by you, you should change your
 password and notify us by clicking on <a href={one-click-link-invalid}>this link</
a>\nIf this sign-in was by you, you can follow <a href={one-click-link-valid}>this
 link</a> to let us know</pre>\n</body>\n</html>",
                    "TextBody": "We required you to use multi-factor authentication
 for the following sign-in attempt:\nTime: {login-time}\nDevice: {device-
name}\nLocation: {city}, {country}\nIf this sign-in was not by you, you should
 change your password and notify us by clicking on {one-click-link-invalid}\nIf this
 sign-in was by you, you can follow {one-click-link-valid} to let us know"
                }
            },
            "Actions": {
                "LowAction": {
                    "Notify": true,
```

```
                "EventAction": "NO_ACTION"
            },
            "MediumAction": {
                "Notify": true,
                "EventAction": "MFA_IF_CONFIGURED"
            },
            "HighAction": {
                "Notify": true,
                "EventAction": "MFA_IF_CONFIGURED"
            }
        }
    }
}
```

- Per i dettagli sull'API, consulta DescribeRiskConfiguration AWS CLICommand Reference.

`describe-user-import-job`

Il seguente esempio di codice mostra come utilizzare`describe-user-import-job`.

AWS CLI

Come descrivere il processo di importazione degli utenti

Questo esempio descrive un processo di importazione degli utenti.

Per ulteriori informazioni sull'importazione di utenti, consulta Importazione di utenti nel pool di utenti da un file CSV.

Comando:

```
aws cognito-idp describe-user-import-job --user-pool-id us-west-2_aaaaaaaaa --job-id import-TZqNQvDRnW
```

Output:

```
{
    "UserImportJob": {
        "JobName": "import-Test1",
        "JobId": "import-TZqNQvDRnW",
        "UserPoolId": "us-west-2_aaaaaaaaa",
        "PreSignedUrl": "PRE_SIGNED URL",
```

```
      "CreationDate": 1548271708.512,
      "Status": "Created",
      "CloudWatchLogsRoleArn": "arn:aws:iam::111111111111:role/
CognitoCloudWatchLogsRole",
      "ImportedUsers": 0,
      "SkippedUsers": 0,
      "FailedUsers": 0
  }
}
```

- Per i dettagli sull'API, consulta DescribeUserImportJob AWS CLICommand Reference.

`describe-user-pool-client`

Il seguente esempio di codice mostra come utilizzare`describe-user-pool-client`.

AWS CLI

Come descrivere un client del pool di utenti

Questo esempio descrive un client del pool di utenti.

Comando:

```
aws cognito-idp describe-user-pool-client --user-pool-id us-west-2_aaaaaaaaa --
client-id 38fjsnc484p94kpqsnet7mpld0
```

Output:

```
{
  "UserPoolClient": {
      "UserPoolId": "us-west-2_aaaaaaaaa",
      "ClientName": "MyApp",
      "ClientId": "38fjsnc484p94kpqsnet7mpld0",
      "ClientSecret": "CLIENT_SECRET",
      "LastModifiedDate": 1548108676.163,
      "CreationDate": 1548108676.163,
      "RefreshTokenValidity": 30,
      "ReadAttributes": [
          "address",
          "birthdate",
          "custom:CustomAttr1",
          "custom:CustomAttr2",
```

```
            "email",
            "email_verified",
            "family_name",
            "gender",
            "given_name",
            "locale",
            "middle_name",
            "name",
            "nickname",
            "phone_number",
            "phone_number_verified",
            "picture",
            "preferred_username",
            "profile",
            "updated_at",
            "website",
            "zoneinfo"
        ],
        "WriteAttributes": [
            "address",
            "birthdate",
            "custom:CustomAttr1",
            "custom:CustomAttr2",
            "email",
            "family_name",
            "gender",
            "given_name",
            "locale",
            "middle_name",
            "name",
            "nickname",
            "phone_number",
            "picture",
            "preferred_username",
            "profile",
            "updated_at",
            "website",
            "zoneinfo"
        ],
        "ExplicitAuthFlows": [
            "ADMIN_NO_SRP_AUTH",
            "USER_PASSWORD_AUTH"
        ],
        "AllowedOAuthFlowsUserPoolClient": false
```

```
  }
}
```

- Per i dettagli sull'API, consulta DescribeUserPoolClient AWS CLICommand Reference.

`describe-user-pool-domain`

Il seguente esempio di codice mostra come utilizzare`describe-user-pool-domain`.

AWS CLI

Come descrivere un client del pool di utenti

Questo esempio descrive un dominio del pool di utenti denominato my-domain.

Comando:

```
aws cognito-idp describe-user-pool-domain --domain my-domain
```

Output:

```
{
  "DomainDescription": {
      "UserPoolId": "us-west-2_aaaaaaaaa",
      "AWSAccountId": "111111111111",
      "Domain": "my-domain",
      "S3Bucket": "aws-cognito-prod-pdx-assets",
      "CloudFrontDistribution": "aaaaaaaaaaaaa.cloudfront.net",
      "Version": "20190128175402",
      "Status": "ACTIVE",
      "CustomDomainConfig": {}
  }
}
```

- Per i dettagli sull'API, consulta DescribeUserPoolDomain AWS CLICommand Reference.

`describe-user-pool`

Il seguente esempio di codice mostra come utilizzare`describe-user-pool`.

AWS CLI

Come descrivere un pool di utenti

L'esempio seguente descrive un pool di utenti con l'ID del pool di utenti US-West-2_example.

```
aws cognito-idp describe-user-pool \
    --user-pool-id us-west-2_EXAMPLE
```

Output:

```
{
    "UserPool": {
        "Id": "us-west-2_EXAMPLE",
        "Name": "MyUserPool",
        "Policies": {
            "PasswordPolicy": {
                "MinimumLength": 8,
                "RequireUppercase": true,
                "RequireLowercase": true,
                "RequireNumbers": true,
                "RequireSymbols": true,
                "TemporaryPasswordValidityDays": 1
            }
        },
        "DeletionProtection": "ACTIVE",
        "LambdaConfig": {
            "PreSignUp": "arn:aws:lambda:us-
west-2:123456789012:function:MyPreSignUpFunction",
            "CustomMessage": "arn:aws:lambda:us-
west-2:123456789012:function:MyCustomMessageFunction",
            "PostConfirmation": "arn:aws:lambda:us-
west-2:123456789012:function:MyPostConfirmationFunction",
            "PreAuthentication": "arn:aws:lambda:us-
west-2:123456789012:function:MyPreAuthenticationFunction",
            "PostAuthentication": "arn:aws:lambda:us-
west-2:123456789012:function:MyPostAuthenticationFunction",
            "DefineAuthChallenge": "arn:aws:lambda:us-
west-2:123456789012:function:MyDefineAuthChallengeFunction",
            "CreateAuthChallenge": "arn:aws:lambda:us-
west-2:123456789012:function:MyCreateAuthChallengeFunction",
            "VerifyAuthChallengeResponse": "arn:aws:lambda:us-
west-2:123456789012:function:MyVerifyAuthChallengeFunction",
```

```
            "PreTokenGeneration": "arn:aws:lambda:us-
west-2:123456789012:function:MyPreTokenGenerationFunction",
            "UserMigration": "arn:aws:lambda:us-
west-2:123456789012:function:MyMigrateUserFunction",
            "PreTokenGenerationConfig": {
                "LambdaVersion": "V2_0",
                "LambdaArn": "arn:aws:lambda:us-
west-2:123456789012:function:MyPreTokenGenerationFunction"
            },
            "CustomSMSSender": {
                "LambdaVersion": "V1_0",
                "LambdaArn": "arn:aws:lambda:us-
west-2:123456789012:function:MyCustomSMSSenderFunction"
            },
            "CustomEmailSender": {
                "LambdaVersion": "V1_0",
                "LambdaArn": "arn:aws:lambda:us-
west-2:123456789012:function:MyCustomEmailSenderFunction"
            },
            "KMSKeyID": "arn:aws:kms:us-west-2:123456789012:key/a1b2c3d4-5678-90ab-
cdef-EXAMPLE22222"
        },
        "LastModifiedDate": 1726784814.598,
        "CreationDate": 1602103465.273,
        "SchemaAttributes": [
            {
                "Name": "sub",
                "AttributeDataType": "String",
                "DeveloperOnlyAttribute": false,
                "Mutable": false,
                "Required": true,
                "StringAttributeConstraints": {
                    "MinLength": "1",
                    "MaxLength": "2048"
                }
            },
            {
                "Name": "name",
                "AttributeDataType": "String",
                "DeveloperOnlyAttribute": false,
                "Mutable": true,
                "Required": false,
                "StringAttributeConstraints": {
                    "MinLength": "0",
```

```
                "MaxLength": "2048"
            }
        },
        {
            "Name": "given_name",
            "AttributeDataType": "String",
            "DeveloperOnlyAttribute": false,
            "Mutable": true,
            "Required": false,
            "StringAttributeConstraints": {
                "MinLength": "0",
                "MaxLength": "2048"
            }
        },
        {
            "Name": "family_name",
            "AttributeDataType": "String",
            "DeveloperOnlyAttribute": false,
            "Mutable": true,
            "Required": false,
            "StringAttributeConstraints": {
                "MinLength": "0",
                "MaxLength": "2048"
            }
        },
        {
            "Name": "middle_name",
            "AttributeDataType": "String",
            "DeveloperOnlyAttribute": false,
            "Mutable": true,
            "Required": false,
            "StringAttributeConstraints": {
                "MinLength": "0",
                "MaxLength": "2048"
            }
        },
        {
            "Name": "nickname",
            "AttributeDataType": "String",
            "DeveloperOnlyAttribute": false,
            "Mutable": true,
            "Required": false,
            "StringAttributeConstraints": {
                "MinLength": "0",
```

```
                "MaxLength": "2048"
            }
        },
        {
            "Name": "preferred_username",
            "AttributeDataType": "String",
            "DeveloperOnlyAttribute": false,
            "Mutable": true,
            "Required": false,
            "StringAttributeConstraints": {
                "MinLength": "0",
                "MaxLength": "2048"
            }
        },
        {
            "Name": "profile",
            "AttributeDataType": "String",
            "DeveloperOnlyAttribute": false,
            "Mutable": true,
            "Required": false,
            "StringAttributeConstraints": {
                "MinLength": "0",
                "MaxLength": "2048"
            }
        },
        {
            "Name": "picture",
            "AttributeDataType": "String",
            "DeveloperOnlyAttribute": false,
            "Mutable": true,
            "Required": false,
            "StringAttributeConstraints": {
                "MinLength": "0",
                "MaxLength": "2048"
            }
        },
        {
            "Name": "website",
            "AttributeDataType": "String",
            "DeveloperOnlyAttribute": false,
            "Mutable": true,
            "Required": false,
            "StringAttributeConstraints": {
                "MinLength": "0",
```

```
                "MaxLength": "2048"
            }
        },
        {
            "Name": "email",
            "AttributeDataType": "String",
            "DeveloperOnlyAttribute": false,
            "Mutable": true,
            "Required": true,
            "StringAttributeConstraints": {
                "MinLength": "0",
                "MaxLength": "2048"
            }
        },
        {
            "Name": "email_verified",
            "AttributeDataType": "Boolean",
            "DeveloperOnlyAttribute": false,
            "Mutable": true,
            "Required": false
        },
        {
            "Name": "gender",
            "AttributeDataType": "String",
            "DeveloperOnlyAttribute": false,
            "Mutable": true,
            "Required": false,
            "StringAttributeConstraints": {
                "MinLength": "0",
                "MaxLength": "2048"
            }
        },
        {
            "Name": "birthdate",
            "AttributeDataType": "String",
            "DeveloperOnlyAttribute": false,
            "Mutable": true,
            "Required": false,
            "StringAttributeConstraints": {
                "MinLength": "10",
                "MaxLength": "10"
            }
        },
        {
```

```
                "Name": "zoneinfo",
                "AttributeDataType": "String",
                "DeveloperOnlyAttribute": false,
                "Mutable": true,
                "Required": false,
                "StringAttributeConstraints": {
                    "MinLength": "0",
                    "MaxLength": "2048"
                }
            },
            {
                "Name": "locale",
                "AttributeDataType": "String",
                "DeveloperOnlyAttribute": false,
                "Mutable": true,
                "Required": false,
                "StringAttributeConstraints": {
                    "MinLength": "0",
                    "MaxLength": "2048"
                }
            },
            {
                "Name": "phone_number",
                "AttributeDataType": "String",
                "DeveloperOnlyAttribute": false,
                "Mutable": true,
                "Required": false,
                "StringAttributeConstraints": {
                    "MinLength": "0",
                    "MaxLength": "2048"
                }
            },
            {
                "Name": "phone_number_verified",
                "AttributeDataType": "Boolean",
                "DeveloperOnlyAttribute": false,
                "Mutable": true,
                "Required": false
            },
            {
                "Name": "address",
                "AttributeDataType": "String",
                "DeveloperOnlyAttribute": false,
                "Mutable": true,
```

```
                "Required": false,
                "StringAttributeConstraints": {
                    "MinLength": "0",
                    "MaxLength": "2048"
                }
            },
            {
                "Name": "updated_at",
                "AttributeDataType": "Number",
                "DeveloperOnlyAttribute": false,
                "Mutable": true,
                "Required": false,
                "NumberAttributeConstraints": {
                    "MinValue": "0"
                }
            },
            {
                "Name": "identities",
                "AttributeDataType": "String",
                "DeveloperOnlyAttribute": false,
                "Mutable": true,
                "Required": false,
                "StringAttributeConstraints": {}
            },
            {
                "Name": "custom:111",
                "AttributeDataType": "String",
                "DeveloperOnlyAttribute": false,
                "Mutable": true,
                "Required": false,
                "StringAttributeConstraints": {
                    "MinLength": "1",
                    "MaxLength": "256"
                }
            },
            {
                "Name": "dev:custom:222",
                "AttributeDataType": "String",
                "DeveloperOnlyAttribute": true,
                "Mutable": true,
                "Required": false,
                "StringAttributeConstraints": {
                    "MinLength": "1",
                    "MaxLength": "421"
```

```
                }
            },
            {
                "Name": "custom:accesstoken",
                "AttributeDataType": "String",
                "DeveloperOnlyAttribute": false,
                "Mutable": true,
                "Required": false,
                "StringAttributeConstraints": {
                    "MaxLength": "2048"
                }
            },
            {
                "Name": "custom:idtoken",
                "AttributeDataType": "String",
                "DeveloperOnlyAttribute": false,
                "Mutable": true,
                "Required": false,
                "StringAttributeConstraints": {
                    "MaxLength": "2048"
                }
            }
        ],
        "AutoVerifiedAttributes": [
            "email"
        ],
        "SmsVerificationMessage": "Your verification code is {####}. ",
        "EmailVerificationMessage": "Your verification code is {####}. ",
        "EmailVerificationSubject": "Your verification code",
        "VerificationMessageTemplate": {
            "SmsMessage": "Your verification code is {####}. ",
            "EmailMessage": "Your verification code is {####}. ",
            "EmailSubject": "Your verification code",
            "EmailMessageByLink": "Please click the link below to verify your email
address. <b>{##Verify Your Email##}</b>\n this is from us-west-2_ywDJHlIfU",
            "EmailSubjectByLink": "Your verification link",
            "DefaultEmailOption": "CONFIRM_WITH_LINK"
        },
        "SmsAuthenticationMessage": "Your verification code is {####}. ",
        "UserAttributeUpdateSettings": {
            "AttributesRequireVerificationBeforeUpdate": []
        },
        "MfaConfiguration": "OPTIONAL",
        "DeviceConfiguration": {
```

```
            "ChallengeRequiredOnNewDevice": true,
            "DeviceOnlyRememberedOnUserPrompt": false
        },
        "EstimatedNumberOfUsers": 166,
        "EmailConfiguration": {
            "SourceArn": "arn:aws:ses:us-west-2:123456789012:identity/
admin@example.com",
            "EmailSendingAccount": "DEVELOPER"
        },
        "SmsConfiguration": {
            "SnsCallerArn": "arn:aws:iam::123456789012:role/service-role/userpool-
SMS-Role",
            "ExternalId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
            "SnsRegion": "us-west-2"
        },
        "UserPoolTags": {},
        "Domain": "myCustomDomain",
        "CustomDomain": "auth.example.com",
        "AdminCreateUserConfig": {
            "AllowAdminCreateUserOnly": false,
            "UnusedAccountValidityDays": 1,
            "InviteMessageTemplate": {
                "SMSMessage": "Your username is {username} and temporary password is
 {####}. ",
                "EmailMessage": "Your username is {username} and temporary password
 is {####}. ",
                "EmailSubject": "Your temporary password"
            }
        },
        "UserPoolAddOns": {
            "AdvancedSecurityMode": "ENFORCED",
            "AdvancedSecurityAdditionalFlows": {}
        },
        "Arn": "arn:aws:cognito-idp:us-west-2:123456789012:userpool/us-
west-2_EXAMPLE",
        "AccountRecoverySetting": {
            "RecoveryMechanisms": [
                {
                    "Priority": 1,
                    "Name": "verified_email"
                }
            ]
        }
    }
```

```
}
```

Per ulteriori informazioni, consulta Pool di utenti di Amazon Cognito nella Guida per gli sviluppatori di Amazon Cognito.

- Per i dettagli sull'API, consulta DescribeUserPool AWS CLICommand Reference.

`forget-device`

Il seguente esempio di codice mostra come utilizzare`forget-device`.

AWS CLI

Come dimenticare un dispositivo

Questo esempio mostra come dimenticare un dispositivo.

Comando:

```
aws cognito-idp forget-device --device-key us-west-2_abcd_1234-5678
```

- Per i dettagli sull'API, consulta ForgetDevice AWS CLICommand Reference.

`forgot-password`

Il seguente esempio di codice mostra come utilizzare`forgot-password`.

AWS CLI

Come forzare la modifica della password

L'esempio `forgot-password` seguente invia un messaggio a jane@example.com per la modifica della password.

```
aws cognito-idp forgot-password --client-id 38fjsnc484p94kpqsnet7mpld0 --
username jane@example.com
```

Output:

```
{
    "CodeDeliveryDetails": {
        "Destination": "j***@e***.com",
```

```
        "DeliveryMedium": "EMAIL",
        "AttributeName": "email"
    }
}
```

- Per i dettagli sull'API, consulta ForgotPassword AWS CLICommand Reference.

`get-csv-header`

Il seguente esempio di codice mostra come utilizzare`get-csv-header`.

AWS CLI

Come creare un'intestazione CSV

Questo esempio crea un'intestazione CSV.

Per ulteriori informazioni sull'importazione di utenti, consulta Importazione di utenti nel pool di utenti da un file CSV.

Comando:

```
aws cognito-idp get-csv-header --user-pool-id us-west-2_aaaaaaaaa
```

Output:

```
{
  "UserPoolId": "us-west-2_aaaaaaaaa",
  "CSVHeader": [
      "name",
      "given_name",
      "family_name",
      "middle_name",
      "nickname",
      "preferred_username",
      "profile",
      "picture",
      "website",
      "email",
      "email_verified",
      "gender",
      "birthdate",
```

```
      "zoneinfo",
      "locale",
      "phone_number",
      "phone_number_verified",
      "address",
      "updated_at",
      "cognito:mfa_enabled",
      "cognito:username"
  ]
}
```

... Importazione di utenti in pool di utenti da un file CSV: https://docs.aws.amazon.com/cognito/latest/developerguide/cognito - -tool.html user-pools-using-import

- Per i dettagli sull'API, consulta AWS CLI Command GetCsvHeaderReference.

`get-device`

Il seguente esempio di codice mostra come utilizzare`get-device`.

AWS CLI

Come ottenere un dispositivo

L'esempio `get-device` seguente visualizza un dispositivo per l'utente che ha effettuato l'accesso.

```
aws cognito-idp get-device \
    --access-token eyJra456defEXAMPLE \
    --device-key us-west-2_a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Output:

```
{
    "Device": {
        "DeviceKey": "us-west-2_a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
        "DeviceAttributes": [
            {
                "Name": "device_status",
                "Value": "valid"
            },
            {
```

```
                "Name": "device_name",
                "Value": "MyDevice"
            },
            {
                "Name": "dev:device_arn",
                "Value": "arn:aws:cognito-idp:us-west-2:123456789012:owner/diego.us-
west-2_EXAMPLE/device/us-west-2_a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
            },
            {
                "Name": "dev:device_owner",
                "Value": "diego.us-west-2_EXAMPLE"
            },
            {
                "Name": "last_ip_used",
                "Value": "192.0.2.1"
            },
            {
                "Name": "dev:device_remembered_status",
                "Value": "remembered"
            },
            {
                "Name": "dev:device_sdk",
                "Value": "aws-sdk"
            }
        ],
        "DeviceCreateDate": 1715100742.022,
        "DeviceLastModifiedDate": 1723233651.167,
        "DeviceLastAuthenticatedDate": 1715100742.0
    }
}
```

Per ulteriori informazioni, consulta Utilizzo dei dispositivi utente nel pool di utenti nella Guida per gli sviluppatori di Amazon Cognito.

- Per i dettagli sull'API, consulta GetDevice AWS CLICommand Reference.

`get-group`

Il seguente esempio di codice mostra come utilizzare`get-group`.

AWS CLI

Come ottenere informazioni relative a un gruppo

L'esempio `get-group` seguente elenca le proprietà del gruppo di utenti denominato `MyGroup`. A questo gruppo sono associati una precedenza e un ruolo IAM.

```
aws cognito-idp get-group \
    --user-pool-id us-west-2_EXAMPLE \
    --group-name MyGroup
```

Output:

```
{
    "Group": {
        "GroupName": "MyGroup",
        "UserPoolId": "us-west-2_EXAMPLE",
        "RoleArn": "arn:aws:iam::123456789012:role/example-cognito-role",
        "Precedence": 7,
        "LastModifiedDate": 1697211218.305,
        "CreationDate": 1611685503.954
    }
}
```

Per ulteriori informazioni, consulta Aggiunta di gruppi a un pool di utenti nella Guida per gli sviluppatori di Amazon Cognito.

- Per i dettagli sull'API, consulta GetGroup AWS CLICommand Reference.

`get-identity-provider-by-identifier`

Il seguente esempio di codice mostra come utilizzare`get-identity-provider-by-identifier`.

AWS CLI

Come ottenere la configurazione di un gestore dell'identità dall'identificatore IdP

L'esempio `get-identity-provider-by-identifier` seguente restituisce la configurazione del gestore dell'identità digitale con l'identificatore `mysso`.

```
aws cognito-idp get-identity-provider-by-identifier \
    --user-pool-id us-west-2_EXAMPLE \
    --idp-identifier mysso
```

Output:

```
{
    "IdentityProvider": {
        "UserPoolId": "us-west-2_EXAMPLE",
        "ProviderName": "MYSAML",
        "ProviderType": "SAML",
        "ProviderDetails": {
            "ActiveEncryptionCertificate": "[Certificate contents]",
            "IDPSignout": "false",
            "MetadataURL": "https://auth.example.com/saml/metadata/",
            "SLORedirectBindingURI": "https://auth.example.com/saml/logout/",
            "SSORedirectBindingURI": "https://auth.example.com/saml/assertion/"
        },
        "AttributeMapping": {
            "email": "email"
        },
        "IdpIdentifiers": [
            "mysso",
            "mysamlsso"
        ],
        "LastModifiedDate": 1705616729.188,
        "CreationDate": 1643734622.919
    }
}
```

Per ulteriori informazioni, consulta Accesso tramite IdP di terze parti nella Guida per gli sviluppatori di Amazon Cognito.

- Per i dettagli sull'API, consulta GetIdentityProviderByIdentifier AWS CLICommand Reference.

`get-log-delivery-configuration`

Il seguente esempio di codice mostra come utilizzare`get-log-delivery-configuration`.

AWS CLI

Come visualizzare la configurazione di consegna dei log

L'esempio `get-log-delivery-configuration` seguente visualizza le impostazioni di esportazione log del pool di utenti richiesto.

```
aws cognito-idp get-log-delivery-configuration \
    --user-pool-id us-west-2_EXAMPLE
```

Output:

```
{
    "LogDeliveryConfiguration": {
        "UserPoolId": "us-west-2_EXAMPLE",
        "LogConfigurations": [
            {
                "LogLevel": "INFO",
                "EventSource": "userAuthEvents",
                "FirehoseConfiguration": {
                    "StreamArn": "arn:aws:firehose:us-
west-2:123456789012:deliverystream/my-test-deliverystream"
                }
            },
            {
                "LogLevel": "ERROR",
                "EventSource": "userNotification",
                "CloudWatchLogsConfiguration": {
                    "LogGroupArn": "arn:aws:logs:us-west-2:123456789012:log-
group:my-message-delivery-logs"
                }
            }
        ]
    }
}
```

Per ulteriori informazioni, consulta Esportazione dei log dei pool di utenti nella Guida per gli sviluppatori di Amazon Cognito.

- Per i dettagli sull'API, consulta GetLogDeliveryConfiguration AWS CLICommand Reference.

`get-signing-certificate`

Il seguente esempio di codice mostra come utilizzare`get-signing-certificate`.

AWS CLI

Come visualizzare il certificato di firma SAML

L'esempio `get-signing-certificate` seguente visualizza il certificato di firma SAML 2.0 per il pool di utenti della richiesta.

```
aws cognito-idp get-signing-certificate \
```

```
    --user-pool-id us-west-2_EXAMPLE
```

Output:

```
{
    "Certificate": "[Certificate content]"
}
```

Per ulteriori informazioni, consulta Firma e crittografia SAML nella Guida per gli sviluppatori di Amazon Cognito.

- Per i dettagli sull'API, consulta GetSigningCertificate AWS CLICommand Reference.

`get-ui-customization`

Il seguente esempio di codice mostra come utilizzare`get-ui-customization`.

AWS CLI

Come visualizzare le impostazioni di personalizzazione dell'interfaccia utente ospitata classica per un client dell'app

L'esempio `get-ui-customization` seguente mostra le impostazioni di personalizzazione dell'interfaccia utente ospitata classica per un client dell'app che non eredita le impostazioni dal pool di utenti.

```
aws cognito-idp get-ui-customization \
    --user-pool-id us-west-2_EXAMPLE \
    --client-id 1example23456789
```

Output:

```
{
    "UICustomization": {
        "UserPoolId": "us-west-2_EXAMPLE",
        "ClientId": "1example23456789",
        "ImageUrl": "https://example.cloudfront.net/us-
west-2_EXAMPLE/1example23456789/20250115191928/assets/images/image.jpg",
        "CSS": "\n.logo-customizable {\n  max-width: 80%;\n  max-height: 30%;
\n}\n\n.banner-customizable {\n  padding: 25px 0px 25px 0px;\n  background-color:
 lightgray;\n}\n\n.label-customizable {\n  font-weight: 400;\n}\n\n.textDescription-
customizable {\n  padding-top: 100px;\n  padding-bottom: 10px;\n  display: block;
```

```
\n  font-size: 12px;\n}\n\n.idpDescription-customizable {\n  padding-top: 10px;\n
  padding-bottom: 10px;\n  display: block;\n  font-size: 16px;\n}\n\n.legalText-
customizable {\n  color: #747474;\n  font-size: 11px;\n}\n\n.submitButton-
customizable {\n  font-size: 14px;\n  font-weight: bold;\n  margin: 20px 0px
 10px 0px;\n  height: 50px;\n  width: 100%;\n  color: #fff;\n  background-color:
 #337ab7;\n}\n\n.submitButton-customizable:hover {\n  color: #fff;\n  background-
color: #286090;\n}\n\n.errorMessage-customizable {\n  padding: 5px;\n  font-size:
 12px;\n  width: 100%;\n  background: #F5F5F5;\n  border: 2px solid #D64958;\n
  color: #D64958;\n}\n\n.inputField-customizable {\n  width: 100%;\n  height:
 34px;\n  color: #555;\n  background-color: #fff;\n  border: 1px solid #ccc;\n}\n
\n.inputField-customizable:focus {\n  border-color: #66afe9;\n  outline: 0;\n}\n
\n.idpButton-customizable {\n  height: 40px;\n  width: 100%;\n  width: 100%;\n
 text-align: center;\n  margin-bottom: 15px;\n  color: #fff;\n  background-color:
 #5bc0de;\n  border-color: #46b8da;\n}\n\n.idpButton-customizable:hover {\n
 color: #fff;\n  background-color: #31b0d5;\n}\n\n.socialButton-customizable {\n
  border-radius: 2px;\n  height: 60px;\n  margin-bottom: 15px;\n  padding: 1px;
\n  text-align: left;\n  width: 100%;\n}\n\n.redirect-customizable {\n  text-
align: center;\n}\n\n.passwordCheck-notValid-customizable {\n  color: #DF3312;
\n}\n\n.passwordCheck-valid-customizable {\n  color: #19BF00;\n}\n\n.background-
customizable {\n  background-color: #fff;\n}\n",
        "CSSVersion": "20250115191928"
    }
}
```

Per ulteriori informazioni, consulta Branding dell'interfaccia utente ospitata (classica) nella Guida per gli sviluppatori di Amazon Cognito.

- Per i dettagli sull'API, consulta GetUiCustomization AWS CLICommand Reference.

`get-user-attribute-verification-code`

Il seguente esempio di codice mostra come utilizzare`get-user-attribute-verification-code`.

AWS CLI

Come inviare un codice di verifica degli attributi all'utente corrente

L'esempio `get-user-attribute-verification-code` seguente invia un codice di verifica degli attributi all'indirizzo e-mail dell'utente che ha effettuato l'accesso.

```
aws cognito-idp get-user-attribute-verification-code \
    --access-token eyJra456defEXAMPLE \
```

```
    --attribute-name email
```

Output:

```
{
    "CodeDeliveryDetails": {
        "Destination": "a***@e***",
        "DeliveryMedium": "EMAIL",
        "AttributeName": "email"
    }
}
```

Per ulteriori informazioni, consulta Registrazione e conferma degli account utente nella Guida per gli sviluppatori di Amazon Cognito.

- Per i dettagli sull'API, consulta GetUserAttributeVerificationCode AWS CLICommand Reference.

`get-user-auth-factors`

Il seguente esempio di codice mostra come utilizzare`get-user-auth-factors`.

AWS CLI

Come elencare i fattori di autenticazione disponibili per l'utente corrente

L'esempio `get-user-auth-factors` seguente elenca i fattori di autenticazione disponibili per l'utente che ha effettuato l'accesso.

```
aws cognito-idp get-user-auth-factors \
    --access-token eyJra456defEXAMPLE
```

Output:

```
{
    "Username": "testuser",
    "ConfiguredUserAuthFactors": [
        "PASSWORD",
        "EMAIL_OTP",
        "SMS_OTP",
        "WEB_AUTHN"
    ]
```

```
}
```

Per ulteriori informazioni, consulta Autenticazione nella Guida per gli sviluppatori di Amazon Cognito.

- Per i dettagli sull'API, consulta GetUserAuthFactors AWS CLICommand Reference.

`get-user-pool-mfa-config`

Il seguente esempio di codice mostra come utilizzare`get-user-pool-mfa-config`.

AWS CLI

Per visualizzare l'autenticazione a più fattori e WebAuthn le impostazioni di un pool di utenti

L'`get-user-pool-mfa-config`esempio seguente visualizza l'MFA e la WebAuthn configurazione del pool di utenti richiesto.

```
aws cognito-idp get-user-pool-mfa-config \
    --user-pool-id us-west-2_EXAMPLE
```

Output:

```
{
    "SmsMfaConfiguration": {
        "SmsAuthenticationMessage": "Your OTP for MFA or sign-in: use {####}.",
        "SmsConfiguration": {
            "SnsCallerArn": "arn:aws:iam::123456789012:role/service-role/my-SMS-
Role",
            "ExternalId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
            "SnsRegion": "us-west-2"
        }
    },
    "SoftwareTokenMfaConfiguration": {
        "Enabled": true
    },
    "EmailMfaConfiguration": {
        "Message": "Your OTP for MFA or sign-in: use {####}",
        "Subject": "OTP test"
    },
    "MfaConfiguration": "OPTIONAL",
    "WebAuthnConfiguration": {
        "RelyingPartyId": "auth.example.com",
```

```
        "UserVerification": "preferred"
    }
}
```

Per ulteriori informazioni, consulta Aggiunta di MFA nella Guida per gli sviluppatori di Amazon Cognito.

- Per i dettagli sull'API, vedere GetUserPoolMfaConfigin AWS CLI Command Reference.

`get-user`

Il seguente esempio di codice mostra come utilizzare`get-user`.

AWS CLI

Come ottenere i dettagli dell'utente corrente

L'esempio `get-user` seguente mostra il profilo dell'utente che ha effettuato l'accesso.

```
aws cognito-idp get-user \
    --access-token eyJra456defEXAMPLE
```

Output:

```
{
    "Username": "johndoe",
    "UserAttributes": [
        {
            "Name": "sub",
            "Value": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
        },
        {
            "Name": "identities",
            "Value": "[{\"userId\":\"a1b2c3d4-5678-90ab-cdef-EXAMPLE22222\",
\"providerName\":\"SignInWithApple\",\"providerType\":\"SignInWithApple\",\"issuer
\":null,\"primary\":false,\"dateCreated\":1701125599632}]"
        },
        {
            "Name": "email_verified",
            "Value": "true"
        },
        {
            "Name": "custom:state",
```

```
            "Value": "Maine"
        },
        {
            "Name": "name",
            "Value": "John Doe"
        },
        {
            "Name": "phone_number_verified",
            "Value": "true"
        },
        {
            "Name": "phone_number",
            "Value": "+12065551212"
        },
        {
            "Name": "preferred_username",
            "Value": "jamesdoe"
        },
        {
            "Name": "locale",
            "Value": "EMEA"
        },
        {
            "Name": "email",
            "Value": "jamesdoe@example.com"
        }
    ]
}
```

Per ulteriori informazioni, consulta Gestione degli utenti nella Guida per gli sviluppatori di Amazon Cognito.

- Per i dettagli sull'API, consulta GetUser AWS CLICommand Reference.

`global-sign-out`

Il seguente esempio di codice mostra come utilizzare`global-sign-out`.

AWS CLI

Come disconnettere l'utente corrente

L'esempio `global-sign-out` seguente disconnette l'utente corrente.

```
aws cognito-idp global-sign-out \
    --access-token eyJra456defEXAMPLE
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Gestione degli utenti nella Guida per gli sviluppatori di Amazon Cognito.

- Per i dettagli sull'API, consulta GlobalSignOut AWS CLICommand Reference.

`initiate-auth`

Il seguente esempio di codice mostra come utilizzare`initiate-auth`.

AWS CLI

Come effettuare l'accesso per un utente

L'esempio `initiate-auth` seguente accede a un utente con il flusso nome utente-password di base e senza richieste di verifica aggiuntive.

```
aws cognito-idp initiate-auth \
    --auth-flow USER_PASSWORD_AUTH \
    --client-id 1example23456789 \
    --analytics-metadata AnalyticsEndpointId=d70b2ba36a8c4dc5a04a0451aEXAMPLE \
    --auth-parameters USERNAME=testuser,PASSWORD=[Password] --user-context-
data EncodedData=mycontextdata --client-metadata MyTestKey=MyTestValue
```

Output:

```
{
    "AuthenticationResult": {
        "AccessToken": "eyJra456defEXAMPLE",
        "ExpiresIn": 3600,
        "TokenType": "Bearer",
        "RefreshToken": "eyJra123abcEXAMPLE",
        "IdToken": "eyJra789ghiEXAMPLE",
        "NewDeviceMetadata": {
            "DeviceKey": "us-west-2_a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
            "DeviceGroupKey": "-v7w9UcY6"
        }
    }
```

```
}
```

Per ulteriori informazioni, consulta Autenticazione nella Guida per gli sviluppatori di Amazon Cognito.

- Per i dettagli sull'API, consulta InitiateAuth AWS CLICommand Reference.

`list-devices`

Il seguente esempio di codice mostra come utilizzare`list-devices`.

AWS CLI

Come elencare i dispositivi di un utente

L'esempio `list-devices` seguente elenca i dispositivi che sono stati registrati dall'utente corrente.

```
aws cognito-idp list-devices \
    --access-token eyJra456defEXAMPLE
```

Output:

```
{
    "Devices": [
        {
            "DeviceAttributes": [
                {
                    "Name": "device_status",
                    "Value": "valid"
                },
                {
                    "Name": "device_name",
                    "Value": "Dart-device"
                },
                {
                    "Name": "last_ip_used",
                    "Value": "192.0.2.1"
                }
            ],
            "DeviceCreateDate": 1715100742.022,
            "DeviceKey": "us-west-2_a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
            "DeviceLastAuthenticatedDate": 1715100742.0,
```

```
            "DeviceLastModifiedDate": 1723233651.167
        },
        {
            "DeviceAttributes": [
                {
                    "Name": "device_status",
                    "Value": "valid"
                },
                {
                    "Name": "last_ip_used",
                    "Value": "192.0.2.2"
                }
            ],
            "DeviceCreateDate": 1726856147.993,
            "DeviceKey": "us-west-2_a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
            "DeviceLastAuthenticatedDate": 1726856147.0,
            "DeviceLastModifiedDate": 1726856147.993
        }
    ]
}
```

Per ulteriori informazioni, consulta Utilizzo dei dispositivi nella Guida per gli sviluppatori di Amazon Cognito.

- Per i dettagli sull'API, consulta ListDevices AWS CLICommand Reference.

`list-groups`

Il seguente esempio di codice mostra come utilizzare`list-groups`.

AWS CLI

Come elencare i gruppi in un pool di utenti

L'esempio `list-groups` seguente elenca i primi due gruppi del pool di utenti richiesto.

```
aws cognito-idp list-groups \
    --user-pool-id us-west-2_EXAMPLE \
    --max-items 2
```

Output:

```
{
```

```
    "Groups": [
        {
            "CreationDate": 1681760899.633,
            "Description": "My test group",
            "GroupName": "testgroup",
            "LastModifiedDate": 1681760899.633,
            "Precedence": 1,
            "UserPoolId": "us-west-2_EXAMPLE"
        },
        {
            "CreationDate": 1642632749.051,
            "Description": "Autogenerated group for users who sign in using
 Facebook",
            "GroupName": "us-west-2_EXAMPLE_Facebook",
            "LastModifiedDate": 1642632749.051,
            "UserPoolId": "us-west-2_EXAMPLE"
        }
    ],
    "NextToken": "[Pagination token]"
}
```

Per ulteriori informazioni, consulta Aggiunta di gruppi a un pool di utenti nella Guida per gli sviluppatori di Amazon Cognito.

- Per i dettagli sull'API, consulta ListGroups AWS CLICommand Reference.

`list-identity-providers`

Il seguente esempio di codice mostra come utilizzare`list-identity-providers`.

AWS CLI

Come elencare gestore dell'identità

L'esempio `list-identity-providers` seguente elenca i primi due gestori dell'identità digitale nel pool di utenti richiesto.

```
aws cognito-idp list-identity-providers \
    --user-pool-id us-west-2_EXAMPLE \
    --max-items 2
```

Output:

```
{
    "Providers": [
        {
            "CreationDate": 1619477386.504,
            "LastModifiedDate": 1703798328.142,
            "ProviderName": "Azure",
            "ProviderType": "SAML"
        },
        {
            "CreationDate": 1642698776.175,
            "LastModifiedDate": 1642699086.453,
            "ProviderName": "LoginWithAmazon",
            "ProviderType": "LoginWithAmazon"
        }
    ],
    "NextToken": "[Pagination token]"
}
```

Per ulteriori informazioni, consulta Accesso tramite IdP di terze parti nella Guida per gli sviluppatori di Amazon Cognito.

- Per i dettagli sull'API, consulta ListIdentityProviders AWS CLICommand Reference.

`list-resource-servers`

Il seguente esempio di codice mostra come utilizzare`list-resource-servers`.

AWS CLI

Come elencare i server di risorse

L'esempio `list-resource-servers` seguente elenca i primi due server di risorse nel pool di utenti richiesto.

```
aws cognito-idp list-resource-servers \
    --user-pool-id us-west-2_EXAMPLE \
    --max-results 2
```

Output:

```
{
```

```
    "ResourceServers": [
        {
            "Identifier": "myapi.example.com",
            "Name": "Example API with custom access control scopes",
            "Scopes": [
                {
                    "ScopeDescription": "International customers",
                    "ScopeName": "international.read"
                },
                {
                    "ScopeDescription": "Domestic customers",
                    "ScopeName": "domestic.read"
                }
            ],
            "UserPoolId": "us-west-2_EXAMPLE"
        },
        {
            "Identifier": "myapi2.example.com",
            "Name": "Another example API for access control",
            "Scopes": [
                {
                    "ScopeDescription": "B2B customers",
                    "ScopeName": "b2b.read"
                }
            ],
            "UserPoolId": "us-west-2_EXAMPLE"
        }
    ],
    "NextToken": "[Pagination token]"
}
```

Per ulteriori informazioni, consulta Controllo dell'accesso con i server delle risorse nella Guida per gli sviluppatori di Amazon Cognito.

- Per i dettagli sull'API, consulta ListResourceServers AWS CLICommand Reference.

`list-tags-for-resource`

Il seguente esempio di codice mostra come utilizzare`list-tags-for-resource`.

AWS CLI

Come elencare i tag del pool di utenti

L'esempio `list-tags-for-resource` seguente elenca i tag assegnati al pool di utenti con l'ARN richiesto.

```
aws cognito-idp list-tags-for-resource \
    --resource-arn arn:aws:cognito-idp:us-west-2:123456789012:userpool/us-west-2_EXAMPLE
```

Output:

```
{
    "Tags": {
        "administrator": "Jie",
        "tenant": "ExampleCorp"
    }
}
```

Per ulteriori informazioni, consulta Assegnazione di tag alle risorse Amazon Cognito nella Guida per gli sviluppatori di Amazon Cognito.

- Per i dettagli sull'API, consulta ListTagsForResource AWS CLICommand Reference.

`list-user-import-jobs`

Il seguente esempio di codice mostra come utilizzare`list-user-import-jobs`.

AWS CLI

Come elencare i processi e gli stati di importazione degli utenti

L'esempio `list-user-import-jobs` seguente elenca i primi tre processi di importazione utente e i relativi dettagli nel pool di utenti richiesto.

```
aws cognito-idp list-user-import-jobs \
    --user-pool-id us-west-2_EXAMPLE \
    --max-results 3
```

Output:

```
{
    "PaginationToken": "us-west-2_EXAMPLE#import-example3#1667948397084",
    "UserImportJobs": [
        {
```

```
            "CloudWatchLogsRoleArn": "arn:aws:iam::123456789012:role/service-role/
Cognito-UserImport-Role",
            "CompletionDate": 1735329786.142,
            "CompletionMessage": "The user import job has expired.",
            "CreationDate": 1735241621.022,
            "FailedUsers": 0,
            "ImportedUsers": 0,
            "JobId": "import-example1",
            "JobName": "Test-import-job-1",
            "PreSignedUrl": "https://aws-cognito-idp-user-import-pdx.s3.us-
west-2.amazonaws.com/123456789012/us-west-2_EXAMPLE/import-mAgUtd8PMm?
X-Amz-Security-Token=[token]&X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-
Date=20241226T193341Z&X-Amz-SignedHeaders=host%3Bx-amz-server-side-encryption&X-Amz-
Expires=899&X-Amz-Credential=[credential]&X-Amz-Signature=[signature]",
            "SkippedUsers": 0,
            "Status": "Expired",
            "UserPoolId": "us-west-2_EXAMPLE"
        },
        {
            "CloudWatchLogsRoleArn": "arn:aws:iam::123456789012:role/service-role/
Cognito-UserImport-Role",
            "CompletionDate": 1681509058.408,
            "CompletionMessage": "Too many users have failed or been skipped during
 the import.",
            "CreationDate": 1681509001.477,
            "FailedUsers": 1,
            "ImportedUsers": 0,
            "JobId": "import-example2",
            "JobName": "Test-import-job-2",
            "PreSignedUrl": "https://aws-cognito-idp-user-import-pdx.s3.us-
west-2.amazonaws.com/123456789012/us-west-2_EXAMPLE/import-mAgUtd8PMm?
X-Amz-Security-Token=[token]&X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-
Date=20241226T193341Z&X-Amz-SignedHeaders=host%3Bx-amz-server-side-encryption&X-Amz-
Expires=899&X-Amz-Credential=[credential]&X-Amz-Signature=[signature]",
            "SkippedUsers": 0,
            "StartDate": 1681509057.965,
            "Status": "Failed",
            "UserPoolId": "us-west-2_EXAMPLE"
        },
        {
            "CloudWatchLogsRoleArn": "arn:aws:iam::123456789012:role/service-role/
Cognito-UserImport-Role",
            "CompletionDate": 1.667864578676E9,
            "CompletionMessage": "Import Job Completed Successfully.",
```

```
            "CreationDate": 1.667864480281E9,
            "FailedUsers": 0,
            "ImportedUsers": 6,
            "JobId": "import-example3",
            "JobName": "Test-import-job-3",
            "PreSignedUrl": "https://aws-cognito-idp-user-import-pdx.s3.us-
west-2.amazonaws.com/123456789012/us-west-2_EXAMPLE/import-mAgUtd8PMm?
X-Amz-Security-Token=[token]&X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-
Date=20241226T193341Z&X-Amz-SignedHeaders=host%3Bx-amz-server-side-encryption&X-Amz-
Expires=899&X-Amz-Credential=[credential]&X-Amz-Signature=[signature]",
            "SkippedUsers": 0,
            "StartDate": 1.667864578167E9,
            "Status": "Succeeded",
            "UserPoolId": "us-west-2_EXAMPLE"
        }
    ]
}
```

Per ulteriori informazioni, consulta Importazione di utenti da un file CSV nella Guida per gli sviluppatori di Amazon Cognito.

- Per i dettagli sull'API, consulta ListUserImportJobs AWS CLICommand Reference.

`list-user-pool-clients`

Il seguente esempio di codice mostra come utilizzare`list-user-pool-clients`.

AWS CLI

Come elencare i client dell'app

L'esempio `list-user-pool-clients` seguente elenca i primi tre client dell'app nel pool di utenti richiesto.

```
aws cognito-idp list-user-pool-clients \
    --user-pool-id us-west-2_EXAMPLE \
    --max-results 3
```

Output:

```
{
    "NextToken": "[Pagination token]",
    "UserPoolClients": [
```

```
        {
            "ClientId": "1example23456789",
            "ClientName": "app-client-1",
            "UserPoolId": "us-west-2_EXAMPLE"
        },
        {
            "ClientId": "2example34567890",
            "ClientName": "app-client-2",
            "UserPoolId": "us-west-2_EXAMPLE"
        },
        {
            "ClientId": "3example45678901",
            "ClientName": "app-client-3",
            "UserPoolId": "us-west-2_EXAMPLE"
        }
    ]
}
```

Per ulteriori informazioni, consulta Client dell'app nella Guida per gli sviluppatori di Amazon Cognito.

- Per i dettagli sull'API, consulta ListUserPoolClients AWS CLICommand Reference.

`list-user-pools`

Il seguente esempio di codice mostra come utilizzare`list-user-pools`.

AWS CLI

Come elencare i pool di utenti

L'`list-user-pools`esempio seguente elenca 3 dei pool di utenti disponibili nell' AWS account delle credenziali CLI correnti.

```
aws cognito-idp list-user-pools \
    --max-results 3
```

Output:

```
{
    "NextToken": "[Pagination token]",
    "UserPools": [
```

```
        {
            "CreationDate": 1681502497.741,
            "Id": "us-west-2_EXAMPLE1",
            "LambdaConfig": {
                "CustomMessage": "arn:aws:lambda:us-
east-1:123456789012:function:MyFunction",
                "PreSignUp": "arn:aws:lambda:us-
east-1:123456789012:function:MyFunction",
                "PreTokenGeneration": "arn:aws:lambda:us-
east-1:123456789012:function:MyFunction",
                "PreTokenGenerationConfig": {
                    "LambdaArn": "arn:aws:lambda:us-
east-1:123456789012:function:MyFunction",
                    "LambdaVersion": "V1_0"
                }
            },
            "LastModifiedDate": 1681502497.741,
            "Name": "user pool 1"
        },
        {
            "CreationDate": 1686064178.717,
            "Id": "us-west-2_EXAMPLE2",
            "LambdaConfig": {
            },
            "LastModifiedDate": 1686064178.873,
            "Name": "user pool 2"
        },
        {
            "CreationDate": 1627681712.237,
            "Id": "us-west-2_EXAMPLE3",
            "LambdaConfig": {
                "UserMigration": "arn:aws:lambda:us-
east-1:123456789012:function:MyFunction"
            },
            "LastModifiedDate": 1678486942.479,
            "Name": "user pool 3"
        }
    ]
}
```

Per ulteriori informazioni, consulta Pool di utenti di Amazon Cognito nella Guida per gli sviluppatori di Amazon Cognito.

- Per i dettagli sull'API, vedere ListUserPoolsin AWS CLI Command Reference.

`list-users-in-group`

Il seguente esempio di codice mostra come utilizzare`list-users-in-group`.

AWS CLI

Come elencare gli utenti in un gruppo

Questo esempio elenca gli utenti in gruppo MyGroup.

Comando:

```
aws cognito-idp list-users-in-group --user-pool-id us-west-2_aaaaaaaaa --group-
name MyGroup
```

Output:

```
{
  "Users": [
      {
          "Username": "acf10624-80bb-401a-ac61-607bee2110ec",
          "Attributes": [
              {
                  "Name": "sub",
                  "Value": "acf10624-80bb-401a-ac61-607bee2110ec"
              },
              {
                  "Name": "custom:CustomAttr1",
                  "Value": "New Value!"
              },
              {
                  "Name": "email",
                  "Value": "jane@example.com"
              }
          ],
          "UserCreateDate": 1548102770.284,
          "UserLastModifiedDate": 1548103204.893,
          "Enabled": true,
          "UserStatus": "CONFIRMED"
      },
      {
          "Username": "22704aa3-fc10-479a-97eb-2af5806bd327",
          "Attributes": [
              {
```

```
                "Name": "sub",
                "Value": "22704aa3-fc10-479a-97eb-2af5806bd327"
            },
            {
                "Name": "email_verified",
                "Value": "true"
            },
            {
                "Name": "email",
                "Value": "diego@example.com"
            }
        ],
        "UserCreateDate": 1548089817.683,
        "UserLastModifiedDate": 1548089817.683,
        "Enabled": true,
        "UserStatus": "FORCE_CHANGE_PASSWORD"
    }
  ]
}
```

- Per i dettagli sull'API, consulta ListUsersInGroup AWS CLICommand Reference.

`list-users`

Il seguente esempio di codice mostra come utilizzare`list-users`.

AWS CLI

Esempio 1: come restituire l'elenco degli utenti con un filtro lato server

L'esempio `list-users` seguente elenca 3 utenti nel pool di utenti richiesto i cui indirizzi e-mail iniziano con `testuser`.

```
aws cognito-idp list-users \
    --user-pool-id us-west-2_EXAMPLE \
    --filter email^=\"testuser\" \
    --max-items 3
```

Output:

```
{
    "PaginationToken": "efgh5678EXAMPLE",
```

```
    "Users": [
        {
            "Attributes": [
                {
                    "Name": "sub",
                    "Value": "eaad0219-2117-439f-8d46-4db20e59268f"
                },
                {
                    "Name": "email",
                    "Value": "testuser@example.com"
                }
            ],
            "Enabled": true,
            "UserCreateDate": 1682955829.578,
            "UserLastModifiedDate": 1689030181.63,
            "UserStatus": "CONFIRMED",
            "Username": "testuser"
        },
        {
            "Attributes": [
                {
                    "Name": "sub",
                    "Value": "3b994cfd-0b07-4581-be46-3c82f9a70c90"
                },
                {
                    "Name": "email",
                    "Value": "testuser2@example.com"
                }
            ],
            "Enabled": true,
            "UserCreateDate": 1684427979.201,
            "UserLastModifiedDate": 1684427979.201,
            "UserStatus": "UNCONFIRMED",
            "Username": "testuser2"
        },
        {
            "Attributes": [
                {
                    "Name": "sub",
                    "Value": "5929e0d1-4c34-42d1-9b79-a5ecacfe66f7"
                },
                {
                    "Name": "email",
                    "Value": "testuser3@example.com"
```

```
                }
            ],
            "Enabled": true,
            "UserCreateDate": 1684427823.641,
            "UserLastModifiedDate": 1684427823.641,
            "UserStatus": "UNCONFIRMED",
            "Username": "testuser3@example.com"
        }
    ]
}
```

Per ulteriori informazioni, consulta Gestione e ricerca degli utenti nella Guida per gli sviluppatori di Amazon Cognito.

Esempio 2: come restituire l'elenco degli utenti con un filtro lato client

L'esempio `list-users` seguente elenca gli attributi di tre utenti che dispongono di un attributo (in questo caso l'indirizzo e-mail contenente il dominio e-mail "@example .com"). Verranno visualizzati anche eventuali altri attributi contenuti in questa stringa. Il secondo utente non dispone di attributi corrispondenti alla query e pertanto è escluso dall'output visualizzato, ma non dalla risposta del server.

```
aws cognito-idp list-users \
    --user-pool-id us-west-2_EXAMPLE \
    --max-items 3
    --query Users\[\*\].Attributes\[\?Value\.contains\(\@\,\'@example.com\'\)\]
```

Output:

```
[
    [
        {
            "Name": "email",
            "Value": "admin@example.com"
        }
    ],
    [],
    [
        {
            "Name": "email",
            "Value": "operator@example.com"
        }
```

```
    ]
]
```

Per ulteriori informazioni, consulta Gestione e ricerca degli utenti nella Guida per gli sviluppatori di Amazon Cognito.

- Per i dettagli sull'API, consulta ListUsers AWS CLICommand Reference.

`list-web-authn-credentials`

Il seguente esempio di codice mostra come utilizzare`list-web-authn-credentials`.

AWS CLI

Come elencare le credenziali passkey

L'`list-web-authn-credentials`esempio seguente elenca la passkey, o le credenziali WebAuthn, per l'utente corrente. Hanno un dispositivo registrato.

```
aws cognito-idp list-web-authn-credentials \
    --access-token eyJra456defEXAMPLE
```

Output:

```
{
    "Credentials": [
        {
            "AuthenticatorAttachment": "cross-platform",
            "CreatedAt": 1736293876.115,
            "CredentialId": "8LApgk4-lNUFHbhm2w6Und7-
uxcc8coJGsPxiogvHoItc64xWQc3r4CEXAMPLE",
            "FriendlyCredentialName": "Roaming passkey",
            "RelyingPartyId": "auth.example.com"
        }
    ]
}
```

Per ulteriori informazioni, consulta Accesso con passkey nella Guida per gli sviluppatori di Amazon Cognito.

- Per i dettagli sull'API, vedere ListWebAuthnCredentialsin AWS CLI Command Reference.

`resend-confirmation-code`

Il seguente esempio di codice mostra come utilizzare`resend-confirmation-code`.

AWS CLI

Come inviare nuovamente un codice di conferma

L'esempio `resend-confirmation-code` seguente invia un codice di conferma all'utente `jane`.

```
aws cognito-idp resend-confirmation-code \
    --client-id 12a3b456c7de890f11g123hijk \
    --username jane
```

Output:

```
{
    "CodeDeliveryDetails": {
        "Destination": "j***@e***.com",
        "DeliveryMedium": "EMAIL",
        "AttributeName": "email"
    }
}
```

Per ulteriori informazioni, consulta Registrazione e conferma degli account utente nella Guida per gli sviluppatori di Amazon Cognito.

- Per i dettagli sull'API, consulta ResendConfirmationCode AWS CLICommand Reference.

`respond-to-auth-challenge`

Il seguente esempio di codice mostra come utilizzare`respond-to-auth-challenge`.

AWS CLI

Esempio 1: come rispondere a una richiesta di verifica NEW_PASSWORD_REQUIRED

L'esempio `respond-to-auth-challenge` seguente risponde alla richiesta di verifica NEW_PASSWORD_REQUIRED restituita da initiate-auth. Imposta una password per l'utente `jane@example.com`.

```
aws cognito-idp respond-to-auth-challenge \
    --client-id 1example23456789 \
```

```
    --challenge-name NEW_PASSWORD_REQUIRED \
    --challenge-responses USERNAME=jane@example.com,NEW_PASSWORD=[Password] \
    --session AYABeEv5HklEXAMPLE
```

Output:

```
{
    "ChallengeParameters": {},
    "AuthenticationResult": {
        "AccessToken": "ACCESS_TOKEN",
        "ExpiresIn": 3600,
        "TokenType": "Bearer",
        "RefreshToken": "REFRESH_TOKEN",
        "IdToken": "ID_TOKEN",
        "NewDeviceMetadata": {
            "DeviceKey": "us-west-2_a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
            "DeviceGroupKey": "-wt2ha1Zd"
        }
    }
}
```

Per ulteriori informazioni, consulta Autenticazione nella Guida per gli sviluppatori di Amazon Cognito.

Esempio 2: come rispondere a una richiesta di verifica SELECT_MFA_TYPE

L'esempio `respond-to-auth-challenge` seguente sceglie TOTP MFA come opzione MFA per l'utente corrente. All'utente viene richiesto di selezionare un tipo di autenticazione MFA e quindi di inserire il proprio codice MFA.

```
aws cognito-idp respond-to-auth-challenge \
    --client-id 1example23456789
    --session AYABeEv5HklEXAMPLE
    --challenge-name SELECT_MFA_TYPE
    --challenge-responses USERNAME=testuser,ANSWER=SOFTWARE_TOKEN_MFA
```

Output:

```
{
    "ChallengeName": "SOFTWARE_TOKEN_MFA",
    "Session": "AYABeEv5HklEXAMPLE",
    "ChallengeParameters": {
```

```
        "FRIENDLY_DEVICE_NAME": "transparent"
    }
}
```

Per ulteriori informazioni, consulta Aggiunta di MFA nella Guida per gli sviluppatori di Amazon Cognito.

Esempio 3: come rispondere a una richiesta di verifica SOFTWARE_TOKEN_MFA

L'esempio `respond-to-auth-challenge` seguente fornisce un codice MFA TOTP e completa l'accesso.

```
aws cognito-idp respond-to-auth-challenge \
    --client-id 1example23456789 \
    --session AYABeEv5HklEXAMPLE \
    --challenge-name SOFTWARE_TOKEN_MFA \
    --challenge-responses USERNAME=testuser,SOFTWARE_TOKEN_MFA_CODE=123456
```

Output:

```
{
    "AuthenticationResult": {
        "AccessToken": "eyJra456defEXAMPLE",
        "ExpiresIn": 3600,
        "TokenType": "Bearer",
        "RefreshToken": "eyJra123abcEXAMPLE",
        "IdToken": "eyJra789ghiEXAMPLE",
        "NewDeviceMetadata": {
            "DeviceKey": "us-west-2_a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
            "DeviceGroupKey": "-v7w9UcY6"
        }
    }
}
```

Per ulteriori informazioni, consulta Aggiunta di MFA nella Guida per gli sviluppatori di Amazon Cognito.

- Per i dettagli sull'API, consulta RespondToAuthChallenge AWS CLICommand Reference.

`revoke-token`

Il seguente esempio di codice mostra come utilizzare`revoke-token`.

AWS CLI

Come revocare un token di aggiornamento

L'esempio `revoke-token` seguente revoca il token di aggiornamento richiesto e i token di accesso associati.

```
aws cognito-idp revoke-token \
    --token eyJjd123abcEXAMPLE \
    --client-id 1example23456789
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Revoca dei token nella Guida per gli sviluppatori di Amazon Cognito.

- Per i dettagli sull'API, consulta RevokeToken AWS CLICommand Reference.

`set-log-delivery-configuration`

Il seguente esempio di codice mostra come utilizzare`set-log-delivery-configuration`.

AWS CLI

Come configurare l'esportazione di log da un pool di utenti

L'esempio `set-log-delivery-configuration` seguente configura il pool di utenti richiesto con la registrazione degli errori di notifica utente in un gruppo di log e la registrazione delle informazioni di autenticazione utente in un bucket S3.

```
aws cognito-idp set-log-delivery-configuration \
    --user-pool-id us-west-2_EXAMPLE \
    --log-
configurations LogLevel=ERROR,EventSource=userNotification,CloudWatchLogsConfiguration={LogG
west-2:123456789012:log-group:cognito-
exported} LogLevel=INFO,EventSource=userAuthEvents,S3Configuration={BucketArn=arn:aws:s3:::a
s3-demo-bucket1}
```

Output:

```
{
```

```
   "LogDeliveryConfiguration": {
        "LogConfigurations": [
            {
                "CloudWatchLogsConfiguration": {
                    "LogGroupArn": "arn:aws:logs:us-west-2:123456789012:log-
group:cognito-exported"
                },
                "EventSource": "userNotification",
                "LogLevel": "ERROR"
            },
            {
                "EventSource": "userAuthEvents",
                "LogLevel": "INFO",
                "S3Configuration": {
                    "BucketArn": "arn:aws:s3:::amzn-s3-demo-bucket1"
                }
            }
        ],
        "UserPoolId": "us-west-2_EXAMPLE"
   }
}
```

Per ulteriori informazioni, consulta Esportazione dei log dei pool di utenti nella Guida per gli sviluppatori di Amazon Cognito.

- Per i dettagli sull'API, consulta SetLogDeliveryConfiguration AWS CLICommand Reference.

`set-risk-configuration`

Il seguente esempio di codice mostra come utilizzare`set-risk-configuration`.

AWS CLI

Come impostare la configurazione del rischio di protezione dalle minacce

L'esempio `set-risk-configuration` seguente configura i messaggi e le azioni di protezione dalle minacce, le credenziali compromesse e le eccezioni degli indirizzi IP nel client dell'app richiesto. A causa della complessità dell' NotifyConfiguration oggetto, l'input JSON è una procedura consigliata per questo comando.

```
aws cognito-idp set-risk-configuration \
    --cli-input-json file://set-risk-configuration.json
```

Contenuto di `set-risk-configuration.json`:

```
{
    "AccountTakeoverRiskConfiguration": {
        "Actions": {
            "HighAction": {
                "EventAction": "MFA_REQUIRED",
                "Notify": true
            },
            "LowAction": {
                "EventAction": "NO_ACTION",
                "Notify": true
            },
            "MediumAction": {
                "EventAction": "MFA_IF_CONFIGURED",
                "Notify": true
            }
        },
        "NotifyConfiguration": {
            "BlockEmail": {
                "HtmlBody": "<!DOCTYPE html>\n<html>\n<head>\n\t<title>HTML
 email context</title>\n\t<meta charset=\"utf-8\">\n</head>\n<body>\n<pre>We
 blocked an unrecognized sign-in to your account with this information:\n<ul>
\n<li>Time: {login-time}</li>\n<li>Device: {device-name}</li>\n<li>Location: {city},
 {country}</li>\n</ul>\nIf this sign-in was not by you, you should change your
 password and notify us by clicking on <a href={one-click-link-invalid}>this link</
a>\nIf this sign-in was by you, you can follow <a href={one-click-link-valid}>this
 link</a> to let us know</pre>\n</body>\n</html>",
                "Subject": "Blocked sign-in attempt",
                "TextBody": "We blocked an unrecognized sign-in to your account with
 this information:\nTime: {login-time}\nDevice: {device-name}\nLocation: {city},
 {country}\nIf this sign-in was not by you, you should change your password and
 notify us by clicking on {one-click-link-invalid}\nIf this sign-in was by you, you
 can follow {one-click-link-valid} to let us know"
            },
            "From": "admin@example.com",
            "MfaEmail": {
                "HtmlBody": "<!DOCTYPE html>\n<html>\n<head>\n\t<title>HTML email
 context</title>\n\t<meta charset=\"utf-8\">\n</head>\n<body>\n<pre>We required
 you to use multi-factor authentication for the following sign-in attempt:\n<ul>
\n<li>Time: {login-time}</li>\n<li>Device: {device-name}</li>\n<li>Location: {city},
 {country}</li>\n</ul>\nIf this sign-in was not by you, you should change your
 password and notify us by clicking on <a href={one-click-link-invalid}>this link</
```

```
a>\nIf this sign-in was by you, you can follow <a href={one-click-link-valid}>this
 link</a> to let us know</pre>\n</body>\n</html>",
                "Subject": "New sign-in attempt",
                "TextBody": "We required you to use multi-factor authentication
 for the following sign-in attempt:\nTime: {login-time}\nDevice: {device-
name}\nLocation: {city}, {country}\nIf this sign-in was not by you, you should
 change your password and notify us by clicking on {one-click-link-invalid}\nIf this
 sign-in was by you, you can follow {one-click-link-valid} to let us know"
            },
            "NoActionEmail": {
                "HtmlBody": "<!DOCTYPE html>\n<html>\n<head>\n\t<title>HTML
 email context</title>\n\t<meta charset=\"utf-8\">\n</head>\n<body>\n<pre>We
 observed an unrecognized sign-in to your account with this information:\n<ul>
\n<li>Time: {login-time}</li>\n<li>Device: {device-name}</li>\n<li>Location: {city},
 {country}</li>\n</ul>\nIf this sign-in was not by you, you should change your
 password and notify us by clicking on <a href={one-click-link-invalid}>this link</
a>\nIf this sign-in was by you, you can follow <a href={one-click-link-valid}>this
 link</a> to let us know</pre>\n</body>\n</html>",
                "Subject": "New sign-in attempt",
                "TextBody": "We observed an unrecognized sign-in to your account
 with this information:\nTime: {login-time}\nDevice: {device-name}\nLocation:
 {city}, {country}\nIf this sign-in was not by you, you should change your password
 and notify us by clicking on {one-click-link-invalid}\nIf this sign-in was by you,
 you can follow {one-click-link-valid} to let us know"
            },
            "ReplyTo": "admin@example.com",
            "SourceArn": "arn:aws:ses:us-west-2:123456789012:identity/
admin@example.com"
        }
    },
    "ClientId": "1example23456789",
    "CompromisedCredentialsRiskConfiguration": {
        "Actions": {
            "EventAction": "BLOCK"
        },
        "EventFilter": [
            "PASSWORD_CHANGE",
            "SIGN_UP",
            "SIGN_IN"
        ]
    },
    "RiskExceptionConfiguration": {
        "BlockedIPRangeList": [
            "192.0.2.1/32",
```

```
            "192.0.2.2/32"
        ],
        "SkippedIPRangeList": [
            "203.0.113.1/32",
            "203.0.113.2/32"
        ]
    },
    "UserPoolId": "us-west-2_EXAMPLE"
}
```

Output:

```
{
    "RiskConfiguration": {
        "AccountTakeoverRiskConfiguration": {
            "Actions": {
                "HighAction": {
                    "EventAction": "MFA_REQUIRED",
                    "Notify": true
                },
                "LowAction": {
                    "EventAction": "NO_ACTION",
                    "Notify": true
                },
                "MediumAction": {
                    "EventAction": "MFA_IF_CONFIGURED",
                    "Notify": true
                }
            },
            "NotifyConfiguration": {
                "BlockEmail": {
                    "HtmlBody": "<!DOCTYPE html>\n<html>\n<head>\n\t<title>HTML
 email context</title>\n\t<meta charset=\"utf-8\">\n</head>\n<body>\n<pre>We
 blocked an unrecognized sign-in to your account with this information:\n<ul>
\n<li>Time: {login-time}</li>\n<li>Device: {device-name}</li>\n<li>Location: {city},
 {country}</li>\n</ul>\nIf this sign-in was not by you, you should change your
 password and notify us by clicking on <a href={one-click-link-invalid}>this link</
a>\nIf this sign-in was by you, you can follow <a href={one-click-link-valid}>this
 link</a> to let us know</pre>\n</body>\n</html>",
                    "Subject": "Blocked sign-in attempt",
                    "TextBody": "We blocked an unrecognized sign-in to your account
 with this information:\nTime: {login-time}\nDevice: {device-name}\nLocation:
 {city}, {country}\nIf this sign-in was not by you, you should change your password
```

```
 and notify us by clicking on {one-click-link-invalid}\nIf this sign-in was by you,
 you can follow {one-click-link-valid} to let us know"
                },
                "From": "admin@example.com",
                "MfaEmail": {
                    "HtmlBody": "<!DOCTYPE html>\n<html>\n<head>\n\t<title>HTML
 email context</title>\n\t<meta charset=\"utf-8\">\n</head>\n<body>\n<pre>We
 required you to use multi-factor authentication for the following sign-in attempt:
\n<ul>\n<li>Time: {login-time}</li>\n<li>Device: {device-name}</li>\n<li>Location:
 {city}, {country}</li>\n</ul>\nIf this sign-in was not by you, you should change
 your password and notify us by clicking on <a href={one-click-link-invalid}>this
 link</a>\nIf this sign-in was by you, you can follow <a href={one-click-link-
valid}>this link</a> to let us know</pre>\n</body>\n</html>",
                    "Subject": "New sign-in attempt",
                    "TextBody": "We required you to use multi-factor authentication
 for the following sign-in attempt:\nTime: {login-time}\nDevice: {device-
name}\nLocation: {city}, {country}\nIf this sign-in was not by you, you should
 change your password and notify us by clicking on {one-click-link-invalid}\nIf this
 sign-in was by you, you can follow {one-click-link-valid} to let us know"
                },
                "NoActionEmail": {
                    "HtmlBody": "<!DOCTYPE html>\n<html>\n<head>\n\t<title>HTML
 email context</title>\n\t<meta charset=\"utf-8\">\n</head>\n<body>\n<pre>We
 observed an unrecognized sign-in to your account with this information:\n<ul>
\n<li>Time: {login-time}</li>\n<li>Device: {device-name}</li>\n<li>Location: {city},
 {country}</li>\n</ul>\nIf this sign-in was not by you, you should change your
 password and notify us by clicking on <a href={one-click-link-invalid}>this link</
a>\nIf this sign-in was by you, you can follow <a href={one-click-link-valid}>this
 link</a> to let us know</pre>\n</body>\n</html>",
                    "Subject": "New sign-in attempt",
                    "TextBody": "We observed an unrecognized sign-in to your account
 with this information:\nTime: {login-time}\nDevice: {device-name}\nLocation:
 {city}, {country}\nIf this sign-in was not by you, you should change your password
 and notify us by clicking on {one-click-link-invalid}\nIf this sign-in was by you,
 you can follow {one-click-link-valid} to let us know"
                },
                "ReplyTo": "admin@example.com",
                "SourceArn": "arn:aws:ses:us-west-2:123456789012:identity/
admin@example.com"
            }
        },
        "ClientId": "1example23456789",
        "CompromisedCredentialsRiskConfiguration": {
            "Actions": {
```

```
                "EventAction": "BLOCK"
            },
            "EventFilter": [
                "PASSWORD_CHANGE",
                "SIGN_UP",
                "SIGN_IN"
            ]
        },
        "RiskExceptionConfiguration": {
            "BlockedIPRangeList": [
                "192.0.2.1/32",
                "192.0.2.2/32"
            ],
            "SkippedIPRangeList": [
                "203.0.113.1/32",
                "203.0.113.2/32"
            ]
        },
        "UserPoolId": "us-west-2_EXAMPLE"
    }
}
```

Per ulteriori informazioni, consulta Protezione dalle minacce nella Guida per gli sviluppatori di Amazon Cognito.

- Per i dettagli sull'API, consulta SetRiskConfiguration AWS CLICommand Reference.

`set-ui-customization`

Il seguente esempio di codice mostra come utilizzare`set-ui-customization`.

AWS CLI

Esempio 1: come personalizzare l'interfaccia utente ospitata classica per un client dell'app

L'esempio `set-ui-customization` seguente configura il client dell'app richiesto con alcuni CSS personalizzati e con il logo Amazon Cognito come logo dell'applicazione.

```
aws cognito-idp set-ui-customization \
    --user-pool-id us-west-2_ywDJHlIfU \
    --client-id 14pq32c5q2uq2q7keorloqvb23 \
    --css ".logo-customizable {\n\tmax-width: 60%;\n\tmax-height: 30%;\n}\n.banner-
customizable {\n\tpadding: 25px 0px 25px 0px;\n\tbackground-color: lightgray;
```

```
\n}\n.label-customizable {\n\tfont-weight: 400;\n}\n.textDescription-customizable
 {\n\tpadding-top: 10px;\n\tpadding-bottom: 10px;\n\tdisplay: block;\n\tfont-
size: 16px;\n}\n.idpDescription-customizable {\n\tpadding-top: 10px;\n\tpadding-
bottom: 10px;\n\tdisplay: block;\n\tfont-size: 16px;\n}\n.legalText-customizable
 {\n\tcolor: #747474;\n\tfont-size: 11px;\n}\n.submitButton-customizable {\n\tfont-
size: 11px;\n\tfont-weight: normal;\n\tmargin: 20px -15px 10px -13px;\n\theight:
 40px;\n\twidth: 108%;\n\tcolor: #fff;\n\tbackground-color: #337ab7;\n\ttext-align:
 center;\n}\n.submitButton-customizable:hover {\n\tcolor: #fff;\n\tbackground-color:
 #286090;\n}\n.errorMessage-customizable {\n\tpadding: 5px;\n\tfont-size: 14px;
\n\twidth: 100%;\n\tbackground: #F5F5F5;\n\tborder: 2px solid #D64958;\n\tcolor:
 #D64958;\n}\n.inputField-customizable {\n\twidth: 100%;\n\theight: 34px;\n\tcolor:
 #555;\n\tbackground-color: #fff;\n\tborder: 1px solid #ccc;\n\tborder-radius:
 0px;\n}\n.inputField-customizable:focus {\n\tborder-color: #66afe9;\n\toutline:
 0;\n}\n.idpButton-customizable {\n\theight: 40px;\n\twidth: 100%;\n\twidth: 100%;
\n\ttext-align: center;\n\tmargin-bottom: 15px;\n\tcolor: #fff;\n\tbackground-
color: #5bc0de;\n\tborder-color: #46b8da;\n}\n.idpButton-customizable:hover {\n
\tcolor: #fff;\n\tbackground-color: #31b0d5;\n}\n.socialButton-customizable {\n
\tborder-radius: 2px;\n\theight: 40px;\n\tmargin-bottom: 15px;\n\tpadding: 1px;
\n\ttext-align: left;\n\twidth: 100%;\n}\n.redirect-customizable {\n\ttext-
align: center;\n}\n.passwordCheck-notValid-customizable {\n\tcolor: #DF3312;
\n}\n.passwordCheck-valid-customizable {\n\tcolor: #19BF00;\n}\n.background-
customizable {\n\tbackground-color: #fff;\n}\n" \
    --image-
file iVBORw0KGgoAAAANSUhEUgAAAFAAAABQCAMAAAC5zwKfAAAAAXNSR0IArs4c6QAAAARnQU1BAACxjwv8YQUAAAA
Cmsfvm6f3y9P////fM0uqAj
+yNmu6ZpvnZ3eNabuFNYuZneehzhPKzvPTAxwAAAOiMMlkAAAASdFJOU/////////////////////
AOK/vxIAAAAJcEhZcwAADsMAAA7DAcdvqGQAAAKDSURBVFhH7ZfpkoMgEISDHKuEw/
d/2u2BQWMiBrG29o+fVsKatdPMAeZxc3Nz8w+ISekzmB++sYIw/I/
tjHzrPpO2Tx62EbR2PNxFac+jVuKxRaV50IzXkUe76NOCoUuwlvnQKei02gNF0ykotOLRBq/
nboeWRxAISx2EbsHFoRhK6Igk2JJlwScfQjgt06dOaWWiTbEDAe/
iq8N9kqCw2uCbHkHlYkaXEF8EYeL9RDqT4FhC6XMIIEifdcUwCc4leNyhabadWU6OlKYJE1Oac3NSPhB5rlaXlSgmr/1
F0L6Q5pZiSG0SfZTSTCOUhxOCH1AdIoCpTTIjtd
+VpEjUDDytQH/0Fpc661Aisas/4qmyUItD557pSCOSQQzlx27J
+meyDGc5zZgfhWuXE1lGgmVOMwmWdeGdzhjqZV14x5vSj7vsC5JDz/Cl0Vhp56n2NQt1wQIpury1EPbwyaYm
+IhmAQKoajkH51wg4cMZ1wQ3QG9efKWWOaDhYWnU6jXjCMdRmm21PArI
+Pb5DYoH93hq0ZCPlxeGJho/DI15C6sQc/L2sTC47UFBKZGHT6k+zlXg7WebA0Nr0HTcLMfk/
Y4Rc65D3iG6WDd7YLSlVqk87bVhUwhnClrx11RsVQwlAA818Mn
+QEs71BhSFU6orsUfKhHp72XMGYXi4q9c64RXRvzkWurRfG2vI2be/VaNcNgpX0Evb/
vio7nPMmj5qujkpQgSaPd1UcVqciHFDNZpOcGlcOPyi+AamCbIL9fitxAGeFN2Dl
+3vZubm5u/4fH4Bd14HhIPdwZPAAAAAElFTkSuQmCC
```

Output:

```
{
    "UICustomization": {
        "UserPoolId": "us-west-2_ywDJHlIfU",
        "ClientId": "14pq32c5q2uq2q7keorloqvb23",
        "ImageUrl": "https://
cf.thewrong.club/14pq32c5q2uq2q7keorloqvb23/20250117005911/assets/images/image.jpg",
        "CSS": ".logo-customizable {\n\tmax-width: 60%;\n\tmax-height: 30%;
\n}\n.banner-customizable {\n\tpadding: 25px 0px 25px 0px;\n\tbackground-color:
 lightgray;\n}\n.label-customizable {\n\tfont-weight: 400;\n}\n.textDescription-
customizable {\n\tpadding-top: 10px;\n\tpadding-bottom: 10px;\n\tdisplay: block;
\n\tfont-size: 16px;\n}\n.idpDescription-customizable {\n\tpadding-top: 10px;\n
\tpadding-bottom: 10px;\n\tdisplay: block;\n\tfont-size: 16px;\n}\n.legalText-
customizable {\n\tcolor: #747474;\n\tfont-size: 11px;\n}\n.submitButton-customizable
 {\n\tfont-size: 11px;\n\tfont-weight: normal;\n\tmargin: 20px -15px 10px -13px;
\n\theight: 40px;\n\twidth: 108%;\n\tcolor: #fff;\n\tbackground-color: #337ab7;
\n\ttext-align: center;\n}\n.submitButton-customizable:hover {\n\tcolor: #fff;
\n\tbackground-color: #286090;\n}\n.errorMessage-customizable {\n\tpadding:
 5px;\n\tfont-size: 14px;\n\twidth: 100%;\n\tbackground: #F5F5F5;\n\tborder:
 2px solid #D64958;\n\tcolor: #D64958;\n}\n.inputField-customizable {\n\twidth:
 100%;\n\theight: 34px;\n\tcolor: #555;\n\tbackground-color: #fff;\n\tborder:
 1px solid #ccc;\n\tborder-radius: 0px;\n}\n.inputField-customizable:focus {\n
\tborder-color: #66afe9;\n\toutline: 0;\n}\n.idpButton-customizable {\n\theight:
 40px;\n\twidth: 100%;\n\twidth: 100%;\n\ttext-align: center;\n\tmargin-bottom:
 15px;\n\tcolor: #fff;\n\tbackground-color: #5bc0de;\n\tborder-color: #46b8da;
\n}\n.idpButton-customizable:hover {\n\tcolor: #fff;\n\tbackground-color: #31b0d5;
\n}\n.socialButton-customizable {\n\tborder-radius: 2px;\n\theight: 40px;\n\tmargin-
bottom: 15px;\n\tpadding: 1px;\n\ttext-align: left;\n\twidth: 100%;\n}\n.redirect-
customizable {\n\ttext-align: center;\n}\n.passwordCheck-notValid-customizable
 {\n\tcolor: #DF3312;\n}\n.passwordCheck-valid-customizable {\n\tcolor: #19BF00;
\n}\n.background-customizable {\n\tbackground-color: #fff;\n}\n",
        "CSSVersion": "20250117005911"
    }
}
```

Esempio 2: come impostare la personalizzazione dell'interfaccia utente predefinita per tutti i client dell'app

L'esempio `set-ui-customization` seguente configura il pool di utenti richiesto per tutti i client dell'app che non dispongono di una configurazione specifica per il client. Il comando applica alcuni CSS personalizzati e utilizza il logo Amazon Cognito come logo dell'applicazione.

```
aws cognito-idp set-ui-customization \
```

```
--user-pool-id us-west-2_ywDJHlIfU \
--client-id ALL \
--css ".logo-customizable {\n\tmax-width: 60%;\n\tmax-height: 30%;\n}\n.banner-
customizable {\n\tpadding: 25px 0px 25px 0px;\n\tbackground-color: lightgray;
\n}\n.label-customizable {\n\tfont-weight: 400;\n}\n.textDescription-customizable
 {\n\tpadding-top: 10px;\n\tpadding-bottom: 10px;\n\tdisplay: block;\n\tfont-
size: 16px;\n}\n.idpDescription-customizable {\n\tpadding-top: 10px;\n\tpadding-
bottom: 10px;\n\tdisplay: block;\n\tfont-size: 16px;\n}\n.legalText-customizable
 {\n\tcolor: #747474;\n\tfont-size: 11px;\n}\n.submitButton-customizable {\n\tfont-
size: 11px;\n\tfont-weight: normal;\n\tmargin: 20px -15px 10px -13px;\n\theight:
 40px;\n\twidth: 108%;\n\tcolor: #fff;\n\tbackground-color: #337ab7;\n\ttext-align:
 center;\n}\n.submitButton-customizable:hover {\n\tcolor: #fff;\n\tbackground-color:
 #286090;\n}\n.errorMessage-customizable {\n\tpadding: 5px;\n\tfont-size: 14px;
\n\twidth: 100%;\n\tbackground: #F5F5F5;\n\tborder: 2px solid #D64958;\n\tcolor:
 #D64958;\n}\n.inputField-customizable {\n\twidth: 100%;\n\theight: 34px;\n\tcolor:
 #555;\n\tbackground-color: #fff;\n\tborder: 1px solid #ccc;\n\tborder-radius:
 0px;\n}\n.inputField-customizable:focus {\n\tborder-color: #66afe9;\n\toutline:
 0;\n}\n.idpButton-customizable {\n\theight: 40px;\n\twidth: 100%;\n\twidth: 100%;
\n\ttext-align: center;\n\tmargin-bottom: 15px;\n\tcolor: #fff;\n\tbackground-
color: #5bc0de;\n\tborder-color: #46b8da;\n}\n.idpButton-customizable:hover {\n
\tcolor: #fff;\n\tbackground-color: #31b0d5;\n}\n.socialButton-customizable {\n
\tborder-radius: 2px;\n\theight: 40px;\n\tmargin-bottom: 15px;\n\tpadding: 1px;
\n\ttext-align: left;\n\twidth: 100%;\n}\n.redirect-customizable {\n\ttext-
align: center;\n}\n.passwordCheck-notValid-customizable {\n\tcolor: #DF3312;
\n}\n.passwordCheck-valid-customizable {\n\tcolor: #19BF00;\n}\n.background-
customizable {\n\tbackground-color: #fff;\n}\n" \
--image-
file iVBORw0KGgoAAAANSUhEUgAAAFAAAABQCAMAAAC5zwKfAAAAAXNSR0IArs4c6QAAAARnQU1BAACxjwv8YQUAAAA
Cmsfvm6f3y9P////fM0uqAj
+yNmu6ZpvnZ3eNabuFNYuZneehzhPKzvPTAxwAAAOiMMlkAAAASdFJOU//////////////////////
AOK/vxIAAAAJcEhZcwAADsMAAA7DAcdvqGQAAAKDSURBVFhH7ZfpkoMgEISDHKuEw/
d/2u2BQWMiBrG29o+fVsKatdPMAeZxc3Nz8w+ISekzmB++sYIw/I/
tjHzrPpO2Tx62EbR2PNxFac+jVuKxRaV50IzXkUe76NOCoUuwlvnQKei02gNF0ykotOLRBq/
nboeWRxAISx2EbsHFoRhK6Igk2JJlwScfQjgt06dOaWWiTbEDAe/
iq8N9kqCw2uCbHkHlYkaXEF8EYeL9RDqT4FhC6XMIIEifdcUwCc4leNyhabadWU6OlKYJE1Oac3NSPhB5rlaXlSgmr/1
F0L6Q5pZiSG0SfZTSTCOUhxOCH1AdIoCpTTIjtd
+VpEjUDDytQH/0Fpc661Aisas/4qmyUItD557pSCOSQQzlx27J
+meyDGc5zZgfhWuXE1lGgmVOMwmWdeGdzhjqZV14x5vSj7vsC5JDz/Cl0Vhp56n2NQt1wQIpury1EPbwyaYm
+IhmAQKoajkH51wg4cMZ1wQ3QG9efKWWOaDhYWnU6jXjCMdRmm21PArI
+Pb5DYoH93hq0ZCPlxeGJho/DI15C6sQc/L2sTC47UFBKZGHT6k+zlXg7WebA0Nr0HTcLMfk/
Y4Rc65D3iG6WDd7YLSlVqk87bVhUwhnClrx11RsVQwlAA818Mn
+QEs71BhSFU6orsUfKhHp72XMGYXi4q9c64RXRvzkWurRfG2vI2be/VaNcNgpX0Evb/
```

```
vio7nPMmj5qujkpQgSaPd1UcVqciHFDNZpOcGlcOPyi+AamCbIL9fitxAGeFN2Dl
+3vZubm5u/4fH4Bd14HhIPdwZPAAAAAElFTkSuQmCC
```

Output:

```
{
    "UICustomization": {
        "UserPoolId": "us-west-2_ywDJHlIfU",
        "ClientId": "14pq32c5q2uq2q7keorloqvb23",
        "ImageUrl": "https://
cf.thewrong.club/14pq32c5q2uq2q7keorloqvb23/20250117005911/assets/images/image.jpg",
        "CSS": ".logo-customizable {\n\tmax-width: 60%;\n\tmax-height: 30%;
\n}\n.banner-customizable {\n\tpadding: 25px 0px 25px 0px;\n\tbackground-color:
 lightgray;\n}\n.label-customizable {\n\tfont-weight: 400;\n}\n.textDescription-
customizable {\n\tpadding-top: 10px;\n\tpadding-bottom: 10px;\n\tdisplay: block;
\n\tfont-size: 16px;\n}\n.idpDescription-customizable {\n\tpadding-top: 10px;\n
\tpadding-bottom: 10px;\n\tdisplay: block;\n\tfont-size: 16px;\n}\n.legalText-
customizable {\n\tcolor: #747474;\n\tfont-size: 11px;\n}\n.submitButton-customizable
 {\n\tfont-size: 11px;\n\tfont-weight: normal;\n\tmargin: 20px -15px 10px -13px;
\n\theight: 40px;\n\twidth: 108%;\n\tcolor: #fff;\n\tbackground-color: #337ab7;
\n\ttext-align: center;\n}\n.submitButton-customizable:hover {\n\tcolor: #fff;
\n\tbackground-color: #286090;\n}\n.errorMessage-customizable {\n\tpadding:
 5px;\n\tfont-size: 14px;\n\twidth: 100%;\n\tbackground: #F5F5F5;\n\tborder:
 2px solid #D64958;\n\tcolor: #D64958;\n}\n.inputField-customizable {\n\twidth:
 100%;\n\theight: 34px;\n\tcolor: #555;\n\tbackground-color: #fff;\n\tborder:
 1px solid #ccc;\n\tborder-radius: 0px;\n}\n.inputField-customizable:focus {\n
\tborder-color: #66afe9;\n\toutline: 0;\n}\n.idpButton-customizable {\n\theight:
 40px;\n\twidth: 100%;\n\twidth: 100%;\n\ttext-align: center;\n\tmargin-bottom:
 15px;\n\tcolor: #fff;\n\tbackground-color: #5bc0de;\n\tborder-color: #46b8da;
\n}\n.idpButton-customizable:hover {\n\tcolor: #fff;\n\tbackground-color: #31b0d5;
\n}\n.socialButton-customizable {\n\tborder-radius: 2px;\n\theight: 40px;\n\tmargin-
bottom: 15px;\n\tpadding: 1px;\n\ttext-align: left;\n\twidth: 100%;\n}\n.redirect-
customizable {\n\ttext-align: center;\n}\n.passwordCheck-notValid-customizable
 {\n\tcolor: #DF3312;\n}\n.passwordCheck-valid-customizable {\n\tcolor: #19BF00;
\n}\n.background-customizable {\n\tbackground-color: #fff;\n}\n",
        "CSSVersion": "20250117005911"
    }
}
```

Per ulteriori informazioni, consulta Branding dell'interfaccia utente ospitata (classica) nella Guida per gli sviluppatori di Amazon Cognito.

- Per i dettagli sull'API, consulta SetUiCustomization AWS CLICommand Reference.

`set-user-mfa-preference`

Il seguente esempio di codice mostra come utilizzare`set-user-mfa-preference`.

AWS CLI

Come impostare la preferenza MFA di un utente

L'esempio `set-user-mfa-preference` seguente configura l'utente corrente per utilizzare MFA TOTP e disabilita tutti gli altri fattori MFA.

```
aws cognito-idp set-user-mfa-preference \
    --access-token eyJra456defEXAMPLE \
    --software-token-mfa-settings Enabled=true,PreferredMfa=true \
    --sms-mfa-settings Enabled=false,PreferredMfa=false \
    --email-mfa-settings Enabled=false,PreferredMfa=false
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Aggiunta di MFA nella Guida per gli sviluppatori di Amazon Cognito.

- Per i dettagli sull'API, consulta SetUserMfaPreference AWS CLICommand Reference.

`set-user-pool-mfa-config`

Il seguente esempio di codice mostra come utilizzare`set-user-pool-mfa-config`.

AWS CLI

Per configurare l'MFA del pool di utenti e WebAuthn

L'`set-user-pool-mfa-config`esempio seguente configura il pool di utenti richiesto con l'MFA opzionale con tutti i metodi MFA disponibili e imposta la configurazione. WebAuthn

```
aws cognito-idp set-user-pool-mfa-config \
    --user-pool-id us-west-2_EXAMPLE \
    --sms-mfa-configuration "SmsAuthenticationMessage=\"Your OTP for MFA or sign-
in: use {####}.\",SmsConfiguration={SnsCallerArn=arn:aws:iam::123456789012:role/
service-role/test-SMS-Role,ExternalId=a1b2c3d4-5678-90ab-cdef-
EXAMPLE11111,SnsRegion=us-west-2}" \
    --software-token-mfa-configuration Enabled=true \
```

```
    --email-mfa-configuration "Message=\"Your OTP for MFA or sign-in: use
 {####}\",Subject=\"OTP test\"" \
    --mfa-configuration OPTIONAL \
    --web-authn-
configuration RelyingPartyId=auth.example.com,UserVerification=preferred
```

Output:

```
{
    "EmailMfaConfiguration": {
        "Message": "Your OTP for MFA or sign-in: use {####}",
        "Subject": "OTP test"
    },
    "MfaConfiguration": "OPTIONAL",
    "SmsMfaConfiguration": {
        "SmsAuthenticationMessage": "Your OTP for MFA or sign-in: use {####}.",
        "SmsConfiguration": {
            "ExternalId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
            "SnsCallerArn": "arn:aws:iam::123456789012:role/service-role/test-SMS-
Role",
            "SnsRegion": "us-west-2"
        }
    },
    "SoftwareTokenMfaConfiguration": {
        "Enabled": true
    },
    "WebAuthnConfiguration": {
        "RelyingPartyId": "auth.example.com",
        "UserVerification": "preferred"
    }
}
```

Per ulteriori informazioni, consulta Aggiunta di MFA e Accesso passkey nella Guida per gli sviluppatori di Amazon Cognito.

- Per i dettagli sull'API, vedere SetUserPoolMfaConfigin AWS CLI Command Reference.

`set-user-settings`

Il seguente esempio di codice mostra come utilizzare`set-user-settings`.

AWS CLI

Come configurare le impostazioni utente

Questo esempio imposta la preferenza di consegna MFA su EMAIL.

Comando:

```
aws cognito-idp set-user-settings --access-token ACCESS_TOKEN --mfa-
options DeliveryMedium=EMAIL
```

- Per i dettagli sull'API, consulta SetUserSettings AWS CLICommand Reference.

`sign-up`

Il seguente esempio di codice mostra come utilizzare`sign-up`.

AWS CLI

Come registrare un utente

In questo esempio viene registrato jane@example.com.

Comando:

```
aws cognito-idp sign-up --client-id 3n4b5urk1ft4fl3mg5e62d9ado --
username jane@example.com --password PASSWORD --user-attributes
 Name="email",Value="jane@example.com" Name="name",Value="Jane"
```

Output:

```
{
  "UserConfirmed": false,
  "UserSub": "e04d60a6-45dc-441c-a40b-e25a787d4862"
}
```

- Per i dettagli sull'API, consulta SignUp AWS CLICommand Reference.

`start-user-import-job`

Il seguente esempio di codice mostra come utilizzare`start-user-import-job`.

AWS CLI

Come avviare un processo di importazione

L'esempio `start-user-import-job` seguente avvia il processo di importazione richiesto nel pool di utenti richiesto.

```
aws cognito-idp start-user-import-job \
    --user-pool-id us-west-2_EXAMPLE \
    --job-id import-mAgUtd8PMm
```

Output:

```
{
    "UserImportJob": {
        "CloudWatchLogsRoleArn": "arn:aws:iam::123456789012:role/example-cloudwatch-
logs-role",
        "CreationDate": 1736442975.904,
        "FailedUsers": 0,
        "ImportedUsers": 0,
        "JobId": "import-mAgUtd8PMm",
        "JobName": "Customer import",
        "PreSignedUrl": "https://aws-cognito-idp-user-import-pdx.s3.us-
west-2.amazonaws.com/123456789012/us-west-2_EXAMPLE/import-mAgUtd8PMm?
X-Amz-Security-Token=[token]&X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-
Date=20241226T193341Z&X-Amz-SignedHeaders=host%3Bx-amz-server-side-encryption&X-Amz-
Expires=899&X-Amz-Credential=[credential]&X-Amz-Signature=[signature]",
        "SkippedUsers": 0,
        "StartDate": 1736443020.081,
        "Status": "Pending",
        "UserPoolId": "us-west-2_EXAMPLE"
    }
}
```

Per ulteriori informazioni, consulta Importazione di utenti in un pool di utenti nella Guida per gli sviluppatori di Amazon Cognito.

- Per i dettagli sull'API, consulta StartUserImportJob AWS CLICommand Reference.

`start-web-authn-registration`

Il seguente esempio di codice mostra come utilizzare`start-web-authn-registration`.

AWS CLI

Come ottenere informazioni sulla registrazione della passkey per un utente che ha effettuato l'accesso

L'`start-web-authn-registration`esempio seguente genera opzioni di WebAuthn registrazione per l'utente corrente.

```
aws cognito-idp start-web-authn-registration \
    --access-token eyJra456defEXAMPLE
```

Output:

```
{
    "CredentialCreationOptions": {
        "authenticatorSelection": {
            "requireResidentKey": true,
            "residentKey": "required",
            "userVerification": "preferred"
        },
        "challenge": "wxvbDicyqQqvF2EXAMPLE",
        "excludeCredentials": [
            {
                "id": "8LApgk4-lNUFHbhm2w6Und7-
uxcc8coJGsPxiogvHoItc64xWQc3r4CEXAMPLE",
                "type": "public-key"
            }
        ],
        "pubKeyCredParams": [
            {
                "alg": -7,
                "type": "public-key"
            },
            {
                "alg": -257,
                "type": "public-key"
            }
        ],
        "rp": {
            "id": "auth.example.com",
            "name": "auth.example.com"
        },
        "timeout": 60000,
```

```
        "user": {
            "displayName": "testuser",
            "id": "ZWFhZDAyMTktMjExNy00MzlmLThkNDYtNGRiMjBlNEXAMPLE",
            "name": "testuser"
        }
    }
}
```

Per ulteriori informazioni, consulta Accesso con passkey nella Guida per gli sviluppatori di Amazon Cognito.

- Per i dettagli sull'API, vedere StartWebAuthnRegistrationin AWS CLI Command Reference.

`stop-user-import-job`

Il seguente esempio di codice mostra come utilizzare`stop-user-import-job`.

AWS CLI

Come interrompere un processo di importazione

L'esempio `stop-user-import-job` seguente interrompe il processo di importazione degli utenti richiesto nel pool di utenti richiesto.

```
aws cognito-idp stop-user-import-job \
    --user-pool-id us-west-2_EXAMPLE \
    --job-id import-mAgUtd8PMm
```

Output:

```
{
    "UserImportJob": {
        "CloudWatchLogsRoleArn": "arn:aws:iam::123456789012:role/example-cloudwatch-
logs-role",
        "CompletionDate": 1736443496.379,
        "CompletionMessage": "The Import Job was stopped by the developer.",
        "CreationDate": 1736443471.781,
        "FailedUsers": 0,
        "ImportedUsers": 0,
        "JobId": "import-mAgUtd8PMm",
        "JobName": "Customer import",
        "PreSignedUrl": "https://aws-cognito-idp-user-import-pdx.s3.us-
west-2.amazonaws.com/123456789012/us-west-2_EXAMPLE/import-mAgUtd8PMm?
```

```
X-Amz-Security-Token=[token]&X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-
Date=20241226T193341Z&X-Amz-SignedHeaders=host%3Bx-amz-server-side-encryption&X-Amz-
Expires=899&X-Amz-Credential=[credential]&X-Amz-Signature=[signature]",
        "SkippedUsers": 0,
        "StartDate": 1736443494.154,
        "Status": "Stopped",
        "UserPoolId": "us-west-2_EXAMPLE"
    }
}
```

Per ulteriori informazioni, consulta Importazione di utenti in un pool di utenti nella Guida per gli sviluppatori di Amazon Cognito.

- Per i dettagli sull'API, consulta StopUserImportJob AWS CLICommand Reference.

`tag-resource`

Il seguente esempio di codice mostra come utilizzare`tag-resource`.

AWS CLI

Come applicare tag a un pool di utenti

L'esempio `tag-resource` seguente applica i tag `administrator` e `department` al pool di utenti richiesto.

```
aws cognito-idp tag-resource \
    --resource-arn arn:aws:cognito-idp:us-west-2:123456789012:userpool/us-
west-2_EXAMPLE \
    --tags administrator=Jie,tenant=ExampleCorp
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Assegnazione di tag alle risorse Amazon Cognito nella Guida per gli sviluppatori di Amazon Cognito.

- Per i dettagli sull'API, consulta TagResource AWS CLICommand Reference.

`untag-resource`

Il seguente esempio di codice mostra come utilizzare`untag-resource`.

AWS CLI

Come rimuovere i tag da un pool di utenti

L'esempio `untag-resource` seguente rimuove i tag `administrator` e `department` dal pool di utenti richiesto.

```
aws cognito-idp untag-resource \
    --resource-arn arn:aws:cognito-idp:us-west-2:767671399759:userpool/us-west-2_l5cxwdm2K \
    --tag-keys administrator tenant
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Assegnazione di tag alle risorse Amazon Cognito nella Guida per gli sviluppatori di Amazon Cognito.

- Per i dettagli sull'API, consulta UntagResource AWS CLICommand Reference.

`update-auth-event-feedback`

Il seguente esempio di codice mostra come utilizzare`update-auth-event-feedback`.

AWS CLI

Come aggiornare il feedback sugli eventi di autenticazione

Questo esempio aggiorna il feedback sugli eventi di autorizzazione. L'evento viene contrassegnato come "Valido".

Comando:

```
aws cognito-idp update-auth-event-feedback --user-pool-id us-west-2_aaaaaaaaa --username diego@example.com --event-id EVENT_ID --feedback-token FEEDBACK_TOKEN --feedback-value "Valid"
```

- Per i dettagli sull'API, consulta UpdateAuthEventFeedback AWS CLICommand Reference.

`update-device-status`

Il seguente esempio di codice mostra come utilizzare`update-device-status`.

AWS CLI

Come aggiornare lo stato del dispositivo

Questo esempio aggiorna lo stato di un dispositivo a "not_remembered".

Comando:

```
aws cognito-idp update-device-status --access-token ACCESS_TOKEN --device-
key DEVICE_KEY --device-remembered-status "not_remembered"
```

- Per i dettagli sull'API, consulta UpdateDeviceStatus AWS CLICommand Reference.

`update-group`

Il seguente esempio di codice mostra come utilizzare`update-group`.

AWS CLI

Come aggiornare un gruppo

Questo esempio aggiorna la descrizione e la precedenza di MyGroup.

Comando:

```
aws cognito-idp update-group --user-pool-id us-west-2_aaaaaaaaa --group-name MyGroup
 --description "New description" --precedence 2
```

Output:

```
{
  "Group": {
      "GroupName": "MyGroup",
      "UserPoolId": "us-west-2_aaaaaaaaa",
      "Description": "New description",
      "RoleArn": "arn:aws:iam::111111111111:role/MyRole",
      "Precedence": 2,
      "LastModifiedDate": 1548800862.812,
      "CreationDate": 1548097827.125
  }
```

```
}
```

- Per i dettagli sull'API, vedere UpdateGroupin AWS CLI Command Reference.

`update-identity-provider`

Il seguente esempio di codice mostra come utilizzareu`pdate-identity-provider`.

AWS CLI

Come aggiornare un gestore dell'identità del pool di utenti

L'`update-identity-provider`esempio seguente aggiorna il provider OIDC «My OIDCId P» nel pool di utenti richiesto.

```
aws cognito-idp update-identity-provider \
    --cli-input-json file://update-identity-provider.json
```

Contenuto di `update-identity-provider.json`:

```
{
   "AttributeMapping": {
            "email": "idp_email",
            "email_verified": "idp_email_verified",
            "username": "sub"
    },
    "CreationDate": 1.701129701653E9,
    "IdpIdentifiers": [
        "corp",
        "dev"
    ],
    "LastModifiedDate": 1.701129701653E9,
    "ProviderDetails": {
        "attributes_request_method": "GET",
        "attributes_url": "https://example.com/userInfo",
        "attributes_url_add_attributes": "false",
        "authorize_scopes": "openid profile",
        "authorize_url": "https://example.com/authorize",
        "client_id": "idpexampleclient123",
        "client_secret": "idpexamplesecret456",
        "jwks_uri": "https://example.com/.well-known/jwks.json",
        "oidc_issuer": "https://example.com",
```

```
        "token_url": "https://example.com/token"
    },
    "ProviderName": "MyOIDCIdP",
    "UserPoolId": "us-west-2_EXAMPLE"
}
```

Output:

```
{
    "IdentityProvider": {
        "AttributeMapping": {
            "email": "idp_email",
            "email_verified": "idp_email_verified",
            "username": "sub"
        },
        "CreationDate": 1701129701.653,
        "IdpIdentifiers": [
            "corp",
            "dev"
        ],
        "LastModifiedDate": 1736444278.211,
        "ProviderDetails": {
            "attributes_request_method": "GET",
            "attributes_url": "https://example.com/userInfo",
            "attributes_url_add_attributes": "false",
            "authorize_scopes": "openid profile",
            "authorize_url": "https://example.com/authorize",
            "client_id": "idpexampleclient123",
            "client_secret": "idpexamplesecret456",
            "jwks_uri": "https://example.com/.well-known/jwks.json",
            "oidc_issuer": "https://example.com",
            "token_url": "https://example.com/token"
        },
        "ProviderName": "MyOIDCIdP",
        "ProviderType": "OIDC",
        "UserPoolId": "us-west-2_EXAMPLE"
    }
}
```

Per ulteriori informazioni, consulta Configurazione di un dominio nella Guida per gli sviluppatori di Amazon Cognito.

- Per i dettagli sull'API, vedere UpdateIdentityProviderin AWS CLI Command Reference.

update-managed-login-branding

Il seguente esempio di codice mostra come utilizzareupdate-managed-login-branding.

AWS CLI

Come aggiornare lo stile di branding dell'accesso gestito

L'esempio update-managed-login-branding seguente aggiorna lo stile di branding del client dell'app richiesto.

```
aws cognito-idp update-managed-login-branding \
    --cli-input-json file://update-managed-login-branding.json
```

Contenuto di update-managed-login-branding.json:

```
{
    "Assets": [
        {
            "Bytes":
 "PHN2ZyB3aWR0aD0iMjAwMDAiIGhlaWdodD0iNDAwIiB2aWV3Qm94PSIwIDAgMjAwMDAgNDAwIiBmaWxsPSJub25lIi
+CjxyZWN0IHdpZHRoPSIyMDAwMCIgaGVpZ2h0PSI0MDAiIGZpbGw9InVybCgjcGFpbnQwX2xpbmVhcl8xNzI1OV8yMzY
+CjxsaW5lYXJHcmFkaWVudCBpZD0icGFpbnQwX2xpbmVhcl8xNzI1OV8yMzY2NzQiIHgxPSItODk0LjI0OSIgeTE9IjE
+Cjwvc3ZnPgo=",
            "Category": "PAGE_FOOTER_BACKGROUND",
            "ColorMode": "DARK",
            "Extension": "SVG"
        }
    ],
    "ManagedLoginBrandingId": "63f30090-6b1f-4278-b885-2bbb81f8e545",
    "Settings": {
        "categories": {
            "auth": {
                "authMethodOrder": [
                    [
                        {
                            "display": "BUTTON",
                            "type": "FEDERATED"
                        },
                        {
                            "display": "INPUT",
                            "type": "USERNAME_PASSWORD"
                        }
```

```
                ]
            ],
            "federation": {
                "interfaceStyle": "BUTTON_LIST",
                "order": [
                ]
            }
        },
        "form": {
            "displayGraphics": true,
            "instructions": {
                "enabled": false
            },
            "languageSelector": {
                "enabled": false
            },
            "location": {
                "horizontal": "CENTER",
                "vertical": "CENTER"
            },
            "sessionTimerDisplay": "NONE"
        },
        "global": {
            "colorSchemeMode": "LIGHT",
            "pageFooter": {
                "enabled": false
            },
            "pageHeader": {
                "enabled": false
            },
            "spacingDensity": "REGULAR"
        },
        "signUp": {
            "acceptanceElements": [
                {
                    "enforcement": "NONE",
                    "textKey": "en"
                }
            ]
        }
    },
    "componentClasses": {
        "buttons": {
            "borderRadius": 8.0
```

```
            },
            "divider": {
                "darkMode": {
                    "borderColor": "232b37ff"
                },
                "lightMode": {
                    "borderColor": "ebebf0ff"
                }
            },
            "dropDown": {
                "borderRadius": 8.0,
                "darkMode": {
                    "defaults": {
                        "itemBackgroundColor": "192534ff"
                    },
                    "hover": {
                        "itemBackgroundColor": "081120ff",
                        "itemBorderColor": "5f6b7aff",
                        "itemTextColor": "e9ebedff"
                    },
                    "match": {
                        "itemBackgroundColor": "d1d5dbff",
                        "itemTextColor": "89bdeeff"
                    }
                },
                "lightMode": {
                    "defaults": {
                        "itemBackgroundColor": "ffffffff"
                    },
                    "hover": {
                        "itemBackgroundColor": "f4f4f4ff",
                        "itemBorderColor": "7d8998ff",
                        "itemTextColor": "000716ff"
                    },
                    "match": {
                        "itemBackgroundColor": "414d5cff",
                        "itemTextColor": "0972d3ff"
                    }
                }
            },
            "focusState": {
                "darkMode": {
                    "borderColor": "539fe5ff"
                },
```

```
                "lightMode": {
                    "borderColor": "0972d3ff"
                }
            },
            "idpButtons": {
                "icons": {
                    "enabled": true
                }
            },
            "input": {
                "borderRadius": 8.0,
                "darkMode": {
                    "defaults": {
                        "backgroundColor": "0f1b2aff",
                        "borderColor": "5f6b7aff"
                    },
                    "placeholderColor": "8d99a8ff"
                },
                "lightMode": {
                    "defaults": {
                        "backgroundColor": "ffffffff",
                        "borderColor": "7d8998ff"
                    },
                    "placeholderColor": "5f6b7aff"
                }
            },
            "inputDescription": {
                "darkMode": {
                    "textColor": "8d99a8ff"
                },
                "lightMode": {
                    "textColor": "5f6b7aff"
                }
            },
            "inputLabel": {
                "darkMode": {
                    "textColor": "d1d5dbff"
                },
                "lightMode": {
                    "textColor": "000716ff"
                }
            },
            "link": {
                "darkMode": {
```

```
                    "defaults": {
                        "textColor": "539fe5ff"
                    },
                    "hover": {
                        "textColor": "89bdeeff"
                    }
                },
                "lightMode": {
                    "defaults": {
                        "textColor": "0972d3ff"
                    },
                    "hover": {
                        "textColor": "033160ff"
                    }
                }
            },
            "optionControls": {
                "darkMode": {
                    "defaults": {
                        "backgroundColor": "0f1b2aff",
                        "borderColor": "7d8998ff"
                    },
                    "selected": {
                        "backgroundColor": "539fe5ff",
                        "foregroundColor": "000716ff"
                    }
                },
                "lightMode": {
                    "defaults": {
                        "backgroundColor": "ffffffff",
                        "borderColor": "7d8998ff"
                    },
                    "selected": {
                        "backgroundColor": "0972d3ff",
                        "foregroundColor": "ffffffff"
                    }
                }
            },
            "statusIndicator": {
                "darkMode": {
                    "error": {
                        "backgroundColor": "1a0000ff",
                        "borderColor": "eb6f6fff",
                        "indicatorColor": "eb6f6fff"
```

```
                },
                "pending": {
                    "indicatorColor": "AAAAAAAA"
                },
                "success": {
                    "backgroundColor": "001a02ff",
                    "borderColor": "29ad32ff",
                    "indicatorColor": "29ad32ff"
                },
                "warning": {
                    "backgroundColor": "1d1906ff",
                    "borderColor": "e0ca57ff",
                    "indicatorColor": "e0ca57ff"
                }
            },
            "lightMode": {
                "error": {
                    "backgroundColor": "fff7f7ff",
                    "borderColor": "d91515ff",
                    "indicatorColor": "d91515ff"
                },
                "pending": {
                    "indicatorColor": "AAAAAAAA"
                },
                "success": {
                    "backgroundColor": "f2fcf3ff",
                    "borderColor": "037f0cff",
                    "indicatorColor": "037f0cff"
                },
                "warning": {
                    "backgroundColor": "fffce9ff",
                    "borderColor": "8d6605ff",
                    "indicatorColor": "8d6605ff"
                }
            }
        }
    },
    "components": {
        "alert": {
            "borderRadius": 12.0,
            "darkMode": {
                "error": {
                    "backgroundColor": "1a0000ff",
                    "borderColor": "eb6f6fff"
```

```
                }
            },
            "lightMode": {
                "error": {
                    "backgroundColor": "fff7f7ff",
                    "borderColor": "d91515ff"
                }
            }
        },
        "favicon": {
            "enabledTypes": [
                "ICO",
                "SVG"
            ]
        },
        "form": {
            "backgroundImage": {
                "enabled": false
            },
            "borderRadius": 8.0,
            "darkMode": {
                "backgroundColor": "0f1b2aff",
                "borderColor": "424650ff"
            },
            "lightMode": {
                "backgroundColor": "ffffffff",
                "borderColor": "c6c6cdff"
            },
            "logo": {
                "enabled": false,
                "formInclusion": "IN",
                "location": "CENTER",
                "position": "TOP"
            }
        },
        "idpButton": {
            "custom": {
            },
            "standard": {
                "darkMode": {
                    "active": {
                        "backgroundColor": "354150ff",
                        "borderColor": "89bdeeff",
                        "textColor": "89bdeeff"
```

```
                    },
                    "defaults": {
                        "backgroundColor": "0f1b2aff",
                        "borderColor": "c6c6cdff",
                        "textColor": "c6c6cdff"
                    },
                    "hover": {
                        "backgroundColor": "192534ff",
                        "borderColor": "89bdeeff",
                        "textColor": "89bdeeff"
                    }
                },
                "lightMode": {
                    "active": {
                        "backgroundColor": "d3e7f9ff",
                        "borderColor": "033160ff",
                        "textColor": "033160ff"
                    },
                    "defaults": {
                        "backgroundColor": "ffffffff",
                        "borderColor": "424650ff",
                        "textColor": "424650ff"
                    },
                    "hover": {
                        "backgroundColor": "f2f8fdff",
                        "borderColor": "033160ff",
                        "textColor": "033160ff"
                    }
                }
            }
        },
        "pageBackground": {
            "darkMode": {
                "color": "0f1b2aff"
            },
            "image": {
                "enabled": true
            },
            "lightMode": {
                "color": "ffffffff"
            }
        },
        "pageFooter": {
            "backgroundImage": {
```

```
                    "enabled": false
                },
                "darkMode": {
                    "background": {
                        "color": "0f141aff"
                    },
                    "borderColor": "424650ff"
                },
                "lightMode": {
                    "background": {
                        "color": "fafafaff"
                    },
                    "borderColor": "d5dbdbff"
                },
                "logo": {
                    "enabled": false,
                    "location": "START"
                }
            },
            "pageHeader": {
                "backgroundImage": {
                    "enabled": false
                },
                "darkMode": {
                    "background": {
                        "color": "0f141aff"
                    },
                    "borderColor": "424650ff"
                },
                "lightMode": {
                    "background": {
                        "color": "fafafaff"
                    },
                    "borderColor": "d5dbdbff"
                },
                "logo": {
                    "enabled": false,
                    "location": "START"
                }
            },
            "pageText": {
                "darkMode": {
                    "bodyColor": "b6bec9ff",
                    "descriptionColor": "b6bec9ff",
```

```
                    "headingColor": "d1d5dbff"
                },
                "lightMode": {
                    "bodyColor": "414d5cff",
                    "descriptionColor": "414d5cff",
                    "headingColor": "000716ff"
                }
            },
            "phoneNumberSelector": {
                "displayType": "TEXT"
            },
            "primaryButton": {
                "darkMode": {
                    "active": {
                        "backgroundColor": "539fe5ff",
                        "textColor": "000716ff"
                    },
                    "defaults": {
                        "backgroundColor": "539fe5ff",
                        "textColor": "000716ff"
                    },
                    "disabled": {
                        "backgroundColor": "ffffffff",
                        "borderColor": "ffffffff"
                    },
                    "hover": {
                        "backgroundColor": "89bdeeff",
                        "textColor": "000716ff"
                    }
                },
                "lightMode": {
                    "active": {
                        "backgroundColor": "033160ff",
                        "textColor": "ffffffff"
                    },
                    "defaults": {
                        "backgroundColor": "0972d3ff",
                        "textColor": "ffffffff"
                    },
                    "disabled": {
                        "backgroundColor": "ffffffff",
                        "borderColor": "ffffffff"
                    },
                    "hover": {
```

```
                    "backgroundColor": "033160ff",
                    "textColor": "ffffffff"
                }
            }
        },
        "secondaryButton": {
            "darkMode": {
                "active": {
                    "backgroundColor": "354150ff",
                    "borderColor": "89bdeeff",
                    "textColor": "89bdeeff"
                },
                "defaults": {
                    "backgroundColor": "0f1b2aff",
                    "borderColor": "539fe5ff",
                    "textColor": "539fe5ff"
                },
                "hover": {
                    "backgroundColor": "192534ff",
                    "borderColor": "89bdeeff",
                    "textColor": "89bdeeff"
                }
            },
            "lightMode": {
                "active": {
                    "backgroundColor": "d3e7f9ff",
                    "borderColor": "033160ff",
                    "textColor": "033160ff"
                },
                "defaults": {
                    "backgroundColor": "ffffffff",
                    "borderColor": "0972d3ff",
                    "textColor": "0972d3ff"
                },
                "hover": {
                    "backgroundColor": "f2f8fdff",
                    "borderColor": "033160ff",
                    "textColor": "033160ff"
                }
            }
        }
    },
    "UseCognitoProvidedValues": false,
```

```
    "UserPoolId": "ca-central-1_EXAMPLE"
}
```

Output:

```
{
    "ManagedLoginBranding": {
        "Assets": [
            {
                "Bytes":
 "PHN2ZyB3aWR0aD0iMjAwMDAiIGhlaWdodD0iNDAwIiB2aWV3Qm94PSIwIDAgMjAwMDAgNDAwIiBmaWxsPSJub25lIi
+CjxyZWN0IHdpZHRoPSIyMDAwMCIgaGVpZ2h0PSI0MDAiIGZpbGw9InVybCgjcGFpbnQwX2xpbmVhcl8xNzI1OV8yMzY
+CjxsaW5lYXJHcmFkaWVudCBpZD0icGFpbnQwX2xpbmVhcl8xNzI1OV8yMzY2NzQiIHgxPSItODk0LjI0OSIgeTE9IjE
+Cjwvc3ZnPgo=",
                "Category": "PAGE_FOOTER_BACKGROUND",
                "ColorMode": "DARK",
                "Extension": "SVG"
            }
        ],
        "CreationDate": 1732138490.642,
        "LastModifiedDate": 1732140420.301,
        "ManagedLoginBrandingId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
        "Settings": {
            "categories": {
                "auth": {
                    "authMethodOrder": [
                        [
                            {
                                "display": "BUTTON",
                                "type": "FEDERATED"
                            },
                            {
                                "display": "INPUT",
                                "type": "USERNAME_PASSWORD"
                            }
                        ]
                    ],
                    "federation": {
                        "interfaceStyle": "BUTTON_LIST",
                        "order": [
                        ]
                    }
                },
```

```
            "form": {
                "displayGraphics": true,
                "instructions": {
                    "enabled": false
                },
                "languageSelector": {
                    "enabled": false
                },
                "location": {
                    "horizontal": "CENTER",
                    "vertical": "CENTER"
                },
                "sessionTimerDisplay": "NONE"
            },
            "global": {
                "colorSchemeMode": "LIGHT",
                "pageFooter": {
                    "enabled": false
                },
                "pageHeader": {
                    "enabled": false
                },
                "spacingDensity": "REGULAR"
            },
            "signUp": {
                "acceptanceElements": [
                    {
                        "enforcement": "NONE",
                        "textKey": "en"
                    }
                ]
            }
        },
        "componentClasses": {
            "buttons": {
                "borderRadius": 8.0
            },
            "divider": {
                "darkMode": {
                    "borderColor": "232b37ff"
                },
                "lightMode": {
                    "borderColor": "ebebf0ff"
                }
```

```
                },
                "dropDown": {
                    "borderRadius": 8.0,
                    "darkMode": {
                        "defaults": {
                            "itemBackgroundColor": "192534ff"
                        },
                        "hover": {
                            "itemBackgroundColor": "081120ff",
                            "itemBorderColor": "5f6b7aff",
                            "itemTextColor": "e9ebedff"
                        },
                        "match": {
                            "itemBackgroundColor": "d1d5dbff",
                            "itemTextColor": "89bdeeff"
                        }
                    },
                    "lightMode": {
                        "defaults": {
                            "itemBackgroundColor": "ffffffff"
                        },
                        "hover": {
                            "itemBackgroundColor": "f4f4f4ff",
                            "itemBorderColor": "7d8998ff",
                            "itemTextColor": "000716ff"
                        },
                        "match": {
                            "itemBackgroundColor": "414d5cff",
                            "itemTextColor": "0972d3ff"
                        }
                    }
                },
                "focusState": {
                    "darkMode": {
                        "borderColor": "539fe5ff"
                    },
                    "lightMode": {
                        "borderColor": "0972d3ff"
                    }
                },
                "idpButtons": {
                    "icons": {
                        "enabled": true
                    }
```

```
                },
                "input": {
                    "borderRadius": 8.0,
                    "darkMode": {
                        "defaults": {
                            "backgroundColor": "0f1b2aff",
                            "borderColor": "5f6b7aff"
                        },
                        "placeholderColor": "8d99a8ff"
                    },
                    "lightMode": {
                        "defaults": {
                            "backgroundColor": "ffffffff",
                            "borderColor": "7d8998ff"
                        },
                        "placeholderColor": "5f6b7aff"
                    }
                },
                "inputDescription": {
                    "darkMode": {
                        "textColor": "8d99a8ff"
                    },
                    "lightMode": {
                        "textColor": "5f6b7aff"
                    }
                },
                "inputLabel": {
                    "darkMode": {
                        "textColor": "d1d5dbff"
                    },
                    "lightMode": {
                        "textColor": "000716ff"
                    }
                },
                "link": {
                    "darkMode": {
                        "defaults": {
                            "textColor": "539fe5ff"
                        },
                        "hover": {
                            "textColor": "89bdeeff"
                        }
                    },
                    "lightMode": {
```

```
                        "defaults": {
                            "textColor": "0972d3ff"
                        },
                        "hover": {
                            "textColor": "033160ff"
                        }
                    }
                },
                "optionControls": {
                    "darkMode": {
                        "defaults": {
                            "backgroundColor": "0f1b2aff",
                            "borderColor": "7d8998ff"
                        },
                        "selected": {
                            "backgroundColor": "539fe5ff",
                            "foregroundColor": "000716ff"
                        }
                    },
                    "lightMode": {
                        "defaults": {
                            "backgroundColor": "ffffffff",
                            "borderColor": "7d8998ff"
                        },
                        "selected": {
                            "backgroundColor": "0972d3ff",
                            "foregroundColor": "ffffffff"
                        }
                    }
                },
                "statusIndicator": {
                    "darkMode": {
                        "error": {
                            "backgroundColor": "1a0000ff",
                            "borderColor": "eb6f6fff",
                            "indicatorColor": "eb6f6fff"
                        },
                        "pending": {
                            "indicatorColor": "AAAAAAAA"
                        },
                        "success": {
                            "backgroundColor": "001a02ff",
                            "borderColor": "29ad32ff",
                            "indicatorColor": "29ad32ff"
```

```
                    },
                    "warning": {
                        "backgroundColor": "1d1906ff",
                        "borderColor": "e0ca57ff",
                        "indicatorColor": "e0ca57ff"
                    }
                },
                "lightMode": {
                    "error": {
                        "backgroundColor": "fff7f7ff",
                        "borderColor": "d91515ff",
                        "indicatorColor": "d91515ff"
                    },
                    "pending": {
                        "indicatorColor": "AAAAAAAA"
                    },
                    "success": {
                        "backgroundColor": "f2fcf3ff",
                        "borderColor": "037f0cff",
                        "indicatorColor": "037f0cff"
                    },
                    "warning": {
                        "backgroundColor": "fffce9ff",
                        "borderColor": "8d6605ff",
                        "indicatorColor": "8d6605ff"
                    }
                }
            }
        },
        "components": {
            "alert": {
                "borderRadius": 12.0,
                "darkMode": {
                    "error": {
                        "backgroundColor": "1a0000ff",
                        "borderColor": "eb6f6fff"
                    }
                },
                "lightMode": {
                    "error": {
                        "backgroundColor": "fff7f7ff",
                        "borderColor": "d91515ff"
                    }
                }
```

```
                },
                "favicon": {
                    "enabledTypes": [
                        "ICO",
                        "SVG"
                    ]
                },
                "form": {
                    "backgroundImage": {
                        "enabled": false
                    },
                    "borderRadius": 8.0,
                    "darkMode": {
                        "backgroundColor": "0f1b2aff",
                        "borderColor": "424650ff"
                    },
                    "lightMode": {
                        "backgroundColor": "ffffffff",
                        "borderColor": "c6c6cdff"
                    },
                    "logo": {
                        "enabled": false,
                        "formInclusion": "IN",
                        "location": "CENTER",
                        "position": "TOP"
                    }
                },
                "idpButton": {
                    "custom": {
                    },
                    "standard": {
                        "darkMode": {
                            "active": {
                                "backgroundColor": "354150ff",
                                "borderColor": "89bdeeff",
                                "textColor": "89bdeeff"
                            },
                            "defaults": {
                                "backgroundColor": "0f1b2aff",
                                "borderColor": "c6c6cdff",
                                "textColor": "c6c6cdff"
                            },
                            "hover": {
                                "backgroundColor": "192534ff",
```

```
                        "borderColor": "89bdeeff",
                        "textColor": "89bdeeff"
                    }
                },
                "lightMode": {
                    "active": {
                        "backgroundColor": "d3e7f9ff",
                        "borderColor": "033160ff",
                        "textColor": "033160ff"
                    },
                    "defaults": {
                        "backgroundColor": "ffffffff",
                        "borderColor": "424650ff",
                        "textColor": "424650ff"
                    },
                    "hover": {
                        "backgroundColor": "f2f8fdff",
                        "borderColor": "033160ff",
                        "textColor": "033160ff"
                    }
                }
            }
        },
        "pageBackground": {
            "darkMode": {
                "color": "0f1b2aff"
            },
            "image": {
                "enabled": true
            },
            "lightMode": {
                "color": "ffffffff"
            }
        },
        "pageFooter": {
            "backgroundImage": {
                "enabled": false
            },
            "darkMode": {
                "background": {
                    "color": "0f141aff"
                },
                "borderColor": "424650ff"
            },
```

```
                    "lightMode": {
                        "background": {
                            "color": "fafafaff"
                        },
                        "borderColor": "d5dbdbff"
                    },
                    "logo": {
                        "enabled": false,
                        "location": "START"
                    }
                },
                "pageHeader": {
                    "backgroundImage": {
                        "enabled": false
                    },
                    "darkMode": {
                        "background": {
                            "color": "0f141aff"
                        },
                        "borderColor": "424650ff"
                    },
                    "lightMode": {
                        "background": {
                            "color": "fafafaff"
                        },
                        "borderColor": "d5dbdbff"
                    },
                    "logo": {
                        "enabled": false,
                        "location": "START"
                    }
                },
                "pageText": {
                    "darkMode": {
                        "bodyColor": "b6bec9ff",
                        "descriptionColor": "b6bec9ff",
                        "headingColor": "d1d5dbff"
                    },
                    "lightMode": {
                        "bodyColor": "414d5cff",
                        "descriptionColor": "414d5cff",
                        "headingColor": "000716ff"
                    }
                },
```

```
                "phoneNumberSelector": {
                    "displayType": "TEXT"
                },
                "primaryButton": {
                    "darkMode": {
                        "active": {
                            "backgroundColor": "539fe5ff",
                            "textColor": "000716ff"
                        },
                        "defaults": {
                            "backgroundColor": "539fe5ff",
                            "textColor": "000716ff"
                        },
                        "disabled": {
                            "backgroundColor": "ffffffff",
                            "borderColor": "ffffffff"
                        },
                        "hover": {
                            "backgroundColor": "89bdeeff",
                            "textColor": "000716ff"
                        }
                    },
                    "lightMode": {
                        "active": {
                            "backgroundColor": "033160ff",
                            "textColor": "ffffffff"
                        },
                        "defaults": {
                            "backgroundColor": "0972d3ff",
                            "textColor": "ffffffff"
                        },
                        "disabled": {
                            "backgroundColor": "ffffffff",
                            "borderColor": "ffffffff"
                        },
                        "hover": {
                            "backgroundColor": "033160ff",
                            "textColor": "ffffffff"
                        }
                    }
                },
                "secondaryButton": {
                    "darkMode": {
                        "active": {
```

```
                            "backgroundColor": "354150ff",
                            "borderColor": "89bdeeff",
                            "textColor": "89bdeeff"
                        },
                        "defaults": {
                            "backgroundColor": "0f1b2aff",
                            "borderColor": "539fe5ff",
                            "textColor": "539fe5ff"
                        },
                        "hover": {
                            "backgroundColor": "192534ff",
                            "borderColor": "89bdeeff",
                            "textColor": "89bdeeff"
                        }
                    },
                    "lightMode": {
                        "active": {
                            "backgroundColor": "d3e7f9ff",
                            "borderColor": "033160ff",
                            "textColor": "033160ff"
                        },
                        "defaults": {
                            "backgroundColor": "ffffffff",
                            "borderColor": "0972d3ff",
                            "textColor": "0972d3ff"
                        },
                        "hover": {
                            "backgroundColor": "f2f8fdff",
                            "borderColor": "033160ff",
                            "textColor": "033160ff"
                        }
                    }
                }
            }
        },
        "UseCognitoProvidedValues": false,
        "UserPoolId": "ca-central-1_EXAMPLE"
    }
}
```

Per ulteriori informazioni, consulta Applicare il branding alle pagine di accesso gestite nella Guida per gli sviluppatori di Amazon Cognito.

- Per i dettagli sull'API, consulta UpdateManagedLoginBranding AWS CLICommand Reference.

`update-resource-server`

Il seguente esempio di codice mostra come utilizzare`update-resource-server`.

AWS CLI

Come aggiornare un server di risorse

Questo esempio aggiorna il server di risorse Weather. Aggiunge un nuovo ambito.

Comando:

```
aws cognito-idp update-resource-server --user-pool-id us-west-2_aaaaaaaaa
 --identifier weather.example.com --name Weather --scopes
 ScopeName=NewScope,ScopeDescription="New scope description"
```

Output:

```
{
  "ResourceServer": {
      "UserPoolId": "us-west-2_aaaaaaaaa",
      "Identifier": "weather.example.com",
      "Name": "Happy",
      "Scopes": [
          {
              "ScopeName": "NewScope",
              "ScopeDescription": "New scope description"
          }
      ]
  }
}
```

- Per i dettagli sull'API, consulta UpdateResourceServer AWS CLICommand Reference.

`update-user-attributes`

Il seguente esempio di codice mostra come utilizzare`update-user-attributes`.

AWS CLI

Come aggiornare gli attributi utente

Questo esempio aggiorna l'attributo utente "nickname".

Comando:

```
aws cognito-idp update-user-attributes --access-token ACCESS_TOKEN --user-attributes
 Name="nickname",Value="Dan"
```

- Per i dettagli sull'API, consulta UpdateUserAttributes AWS CLICommand Reference.

`update-user-pool-client`

Il seguente esempio di codice mostra come utilizzare`update-user-pool-client`.

AWS CLI

Come aggiornare un client dell'app

L'esempio `update-user-pool-client` seguente aggiorna la configurazione del client dell'app richiesto.

```
aws cognito-idp update-user-pool-client \
    --user-pool-id us-west-2_EXAMPLE \
    --client-id 1example23456789 \
    --client-name my-test-app \
    --refresh-token-validity 30 \
    --access-token-validity 60 \
    --id-token-validity 60 \
    --token-validity-units AccessToken=minutes,IdToken=minutes,RefreshToken=days \
    --read-
attributes "address" "birthdate" "email" "email_verified" "family_name" "gender" "locale" "m
 \
    --write-
attributes "address" "birthdate" "email" "family_name" "gender" "locale" "middle_name" "name
 \
    --explicit-auth-
flows "ALLOW_ADMIN_USER_PASSWORD_AUTH" "ALLOW_CUSTOM_AUTH" "ALLOW_REFRESH_TOKEN_AUTH" "ALLOW
 \
    --supported-identity-providers "MySAML" "COGNITO" "Google" \
    --callback-urls "https://www.example.com" "https://app2.example.com" \
    --logout-urls "https://auth.example.com/login?
client_id=1example23456789&response_type=code&redirect_uri=https%3A%2F
%2Fwww.example.com" "https://example.com/logout" \
```

```
    --default-redirect-uri "https://www.example.com" \
    --allowed-o-auth-flows "code" "implicit" \
    --allowed-o-auth-scopes "openid" "profile" "aws.cognito.signin.user.admin" \
    --allowed-o-auth-flows-user-pool-client \
    --prevent-user-existence-errors ENABLED \
    --enable-token-revocation \
    --no-enable-propagate-additional-user-context-data \
    --auth-session-validity 3
```

Output:

```
{
    "UserPoolClient": {
        "UserPoolId": "us-west-2_EXAMPLE",
        "ClientName": "my-test-app",
        "ClientId": "1example23456789",
        "LastModifiedDate": "2025-01-31T14:40:12.498000-08:00",
        "CreationDate": "2023-09-13T16:26:34.408000-07:00",
        "RefreshTokenValidity": 30,
        "AccessTokenValidity": 60,
        "IdTokenValidity": 60,
        "TokenValidityUnits": {
            "AccessToken": "minutes",
            "IdToken": "minutes",
            "RefreshToken": "days"
        },
        "ReadAttributes": [
            "website",
            "zoneinfo",
            "address",
            "birthdate",
            "email_verified",
            "gender",
            "profile",
            "phone_number_verified",
            "preferred_username",
            "locale",
            "middle_name",
            "picture",
            "updated_at",
            "name",
            "nickname",
            "phone_number",
```

```
            "family_name",
            "email"
        ],
        "WriteAttributes": [
            "website",
            "zoneinfo",
            "address",
            "birthdate",
            "gender",
            "profile",
            "preferred_username",
            "locale",
            "middle_name",
            "picture",
            "updated_at",
            "name",
            "nickname",
            "phone_number",
            "family_name",
            "email"
        ],
        "ExplicitAuthFlows": [
            "ALLOW_CUSTOM_AUTH",
            "ALLOW_USER_PASSWORD_AUTH",
            "ALLOW_ADMIN_USER_PASSWORD_AUTH",
            "ALLOW_USER_SRP_AUTH",
            "ALLOW_REFRESH_TOKEN_AUTH"
        ],
        "SupportedIdentityProviders": [
            "Google",
            "COGNITO",
            "MySAML"
        ],
        "CallbackURLs": [
            "https://www.example.com",
            "https://app2.example.com"
        ],
        "LogoutURLs": [
            "https://example.com/logout",
            "https://auth.example.com/login?
client_id=1example23456789&response_type=code&redirect_uri=https%3A%2F
%2Fwww.example.com"
        ],
        "DefaultRedirectURI": "https://www.example.com",
```

```
        "AllowedOAuthFlows": [
            "implicit",
            "code"
        ],
        "AllowedOAuthScopes": [
            "aws.cognito.signin.user.admin",
            "openid",
            "profile"
        ],
        "AllowedOAuthFlowsUserPoolClient": true,
        "PreventUserExistenceErrors": "ENABLED",
        "EnableTokenRevocation": true,
        "EnablePropagateAdditionalUserContextData": false,
        "AuthSessionValidity": 3
    }
}
```

Per ulteriori informazioni, consulta Impostazioni specifiche dell'applicazione con i client dell'app nella Guida per gli sviluppatori di Amazon Cognito.

- Per i dettagli sull'API, consulta UpdateUserPoolClient AWS CLICommand Reference.

`update-user-pool-domain`

Il seguente esempio di codice mostra come utilizzare`update-user-pool-domain`.

AWS CLI

Come aggiornare un dominio personalizzato

L'esempio `update-user-pool-domain` seguente configura la versione di branding e il certificato per il dominio personalizzato del pool di utenti richiesto.

```
aws cognito-idp update-user-pool-domain \
    --user-pool-id ca-central-1_EXAMPLE \
    --domain auth.example.com \
    --managed-login-version 2 \
    --custom-domain-config CertificateArn=arn:aws:acm:us-
east-1:123456789012:certificate/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Output:

```
{
```

```
    "CloudFrontDomain": "example.cloudfront.net",
    "ManagedLoginVersion": 2
}
```

Per ulteriori informazioni, consulta Accesso gestito e Configurazione di un dominio nella Guida per gli sviluppatori di Amazon Cognito.

- Per i dettagli sull'API, consulta UpdateUserPoolDomain AWS CLICommand Reference.

`update-user-pool`

Il seguente esempio di codice mostra come utilizzare`update-user-pool`.

AWS CLI

Come aggiornare un pool di utenti

L'esempio `update-user-pool` seguente modifica un pool di utenti con una sintassi di esempio per ciascuna delle opzioni di configurazione disponibili. Per aggiornare un pool di utenti, è necessario specificare tutte le opzioni configurate in precedenza. In caso contrario, verrà ripristinato il relativo valore predefinito.

```
aws cognito-idp update-user-pool --user-pool-id us-west-2_EXAMPLE \
    --policies PasswordPolicy=
\{MinimumLength=6,RequireUppercase=true,RequireLowercase=true,RequireNumbers=true,RequireSym
 \
    --deletion-protection ACTIVE \
    --lambda-config PreSignUp="arn:aws:lambda:us-
west-2:123456789012:function:cognito-test-presignup-
function",PreTokenGeneration="arn:aws:lambda:us-
west-2:123456789012:function:cognito-test-pretoken-function" \
    --auto-verified-attributes "phone_number" "email" \
    --verification-message-template \{\"SmsMessage\":\""Your code is
 {####}"\",\"EmailMessage\":\""Your code is {####}"\",\"EmailSubject\":\""Your
 verification code"\",\"EmailMessageByLink\":\""Click {##here##} to verify
 your email address."\",\"EmailSubjectByLink\":\""Your verification link"\",
\"DefaultEmailOption\":\"CONFIRM_WITH_LINK\"\} \
    --sms-authentication-message "Your code is {####}" \
    --user-attribute-update-settings
 AttributesRequireVerificationBeforeUpdate="email","phone_number" \
    --mfa-configuration "OPTIONAL" \
```

```
    --device-
configuration ChallengeRequiredOnNewDevice=true,DeviceOnlyRememberedOnUserPrompt=true
 \
    --email-configuration SourceArn="arn:aws:ses:us-
west-2:123456789012:identity/admin@example.com",ReplyToEmailAddress="amdin
+noreply@example.com",EmailSendingAccount=DEVELOPER,From="admin@amazon.com",ConfigurationSet
configuration-set" \
    --sms-configuration SnsCallerArn="arn:aws:iam::123456789012:role/service-role/
SNS-SMS-Role",ExternalId="12345",SnsRegion="us-west-2" \
    --admin-create-user-config AllowAdminCreateUserOnly=false,InviteMessageTemplate=
\{SMSMessage=\""Welcome {username}. Your confirmation code is
 {####}"\",EmailMessage=\""Welcome {username}. Your confirmation code is
 {####}"\",EmailSubject=\""Welcome to MyMobileGame"\"\} \
    --user-pool-tags "Function"="MyMobileGame","Developers"="Berlin" \
    --admin-create-user-config AllowAdminCreateUserOnly=false,InviteMessageTemplate=
\{SMSMessage=\""Welcome {username}. Your confirmation code is
 {####}"\",EmailMessage=\""Welcome {username}. Your confirmation code is
 {####}"\",EmailSubject=\""Welcome to MyMobileGame"\"\} \
    --user-pool-add-ons AdvancedSecurityMode="AUDIT" \
    --account-recovery-setting RecoveryMechanisms=
\[\{Priority=1,Name="verified_email"\},\{Priority=2,Name="verified_phone_number"\}\]
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Aggiornamento della configurazione del pool di utenti nella Guida per gli sviluppatori di Amazon Cognito.

- Per i dettagli sull'API, consulta UpdateUserPool AWS CLICommand Reference.

`verify-software-token`

Il seguente esempio di codice mostra come utilizzare`verify-software-token`.

AWS CLI

Come confermare la registrazione di un autenticatore TOTP

L'esempio `verify-software-token` seguente completa la registrazione TOTP per l'utente corrente.

```
aws cognito-idp verify-software-token \
    --access-token eyJra456defEXAMPLE \
    --user-code 123456
```

Output:

```
{
    "Status": "SUCCESS"
}
```

Per ulteriori informazioni, consulta Aggiunta dell'autenticazione MFA a un pool di utenti nella Guida per gli sviluppatori di Amazon Cognito.

- Per i dettagli sull'API, consulta VerifySoftwareToken AWS CLICommand Reference.

`verify-user-attribute`

Il seguente esempio di codice mostra come utilizzare`verify-user-attribute`.

AWS CLI

Come verificare una modifica degli attributi

L'esempio `verify-user-attribute` seguente verifica una modifica apportata all'attributo e-mail dell'utente corrente.

```
aws cognito-idp verify-user-attribute \
    --access-token eyJra456defEXAMPLE \
    --attribute-name email \
    --code 123456
```

Per ulteriori informazioni, consulta Configurazione della verifica di e-mail o telefono nella Guida per gli sviluppatori di Amazon Cognito.

- Per i dettagli sull'API, consulta VerifyUserAttribute AWS CLICommand Reference.

Esempi di Amazon Comprehend con AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando Amazon Comprehend. AWS Command Line Interface

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`batch-detect-dominant-language`

Il seguente esempio di codice mostra come usare`batch-detect-dominant-language`.

AWS CLI

Come rilevare la lingua dominante di più testi di input

L'esempio `batch-detect-dominant-language` seguente analizza più testi di input e restituisce la lingua dominante di ciascuno di essi. Per ogni previsione viene inoltre restituito il punteggio di attendibilità del modello pre-addestrato.

```
aws comprehend batch-detect-dominant-language \
    --text-list "Physics is the natural science that involves the study of matter
 and its motion and behavior through space and time, along with related concepts
 such as energy and force."
```

Output:

```
{
    "ResultList": [
        {
            "Index": 0,
            "Languages": [
                {
                    "LanguageCode": "en",
                    "Score": 0.9986501932144165
                }
            ]
        }
    ],
    "ErrorList": []
}
```

Per ulteriori informazioni, consulta Lingua dominante nella Guida per gli sviluppatori di Amazon Comprehend.

- Per i dettagli sull'API, consulta BatchDetectDominantLanguage AWS CLICommand Reference.

`batch-detect-entities`

Il seguente esempio di codice mostra come utilizzare`batch-detect-entities`.

AWS CLI

Come rilevare entità da più testi di input

L'esempio `batch-detect-entities` seguente analizza più testi di input e restituisce le entità denominate di ciascuno di essi. Per ogni previsione viene inoltre restituito il punteggio di attendibilità del modello pre-addestrato.

```
aws comprehend batch-detect-entities \
    --language-code en \
    --text-list "Dear Jane, Your AnyCompany Financial Services LLC credit card account 1111-XXXX-1111-XXXX has a minimum payment of $24.53 that is due by July 31st." "Please send customer feedback to Sunshine Spa, 123 Main St, Anywhere or to Alice at AnySpa@example.com."
```

Output:

```
{
    "ResultList": [
        {
            "Index": 0,
            "Entities": [
                {
                    "Score": 0.9985517859458923,
                    "Type": "PERSON",
                    "Text": "Jane",
                    "BeginOffset": 5,
                    "EndOffset": 9
                },
                {
                    "Score": 0.9767839312553406,
                    "Type": "ORGANIZATION",
                    "Text": "AnyCompany Financial Services, LLC",
                    "BeginOffset": 16,
```

```
                    "EndOffset": 50
                },
                {
                    "Score": 0.9856694936752319,
                    "Type": "OTHER",
                    "Text": "1111-XXXX-1111-XXXX",
                    "BeginOffset": 71,
                    "EndOffset": 90
                },
                {
                    "Score": 0.9652159810066223,
                    "Type": "QUANTITY",
                    "Text": ".53",
                    "BeginOffset": 116,
                    "EndOffset": 119
                },
                {
                    "Score": 0.9986667037010193,
                    "Type": "DATE",
                    "Text": "July 31st",
                    "BeginOffset": 135,
                    "EndOffset": 144
                }
            ]
        },
        {
            "Index": 1,
            "Entities": [
                {
                    "Score": 0.720084547996521,
                    "Type": "ORGANIZATION",
                    "Text": "Sunshine Spa",
                    "BeginOffset": 33,
                    "EndOffset": 45
                },
                {
                    "Score": 0.9865870475769043,
                    "Type": "LOCATION",
                    "Text": "123 Main St",
                    "BeginOffset": 47,
                    "EndOffset": 58
                },
                {
                    "Score": 0.5895616412162781,
```

```
                    "Type": "LOCATION",
                    "Text": "Anywhere",
                    "BeginOffset": 60,
                    "EndOffset": 68
                },
                {
                    "Score": 0.6809214353561401,
                    "Type": "PERSON",
                    "Text": "Alice",
                    "BeginOffset": 75,
                    "EndOffset": 80
                },
                {
                    "Score": 0.9979087114334106,
                    "Type": "OTHER",
                    "Text": "AnySpa@example.com",
                    "BeginOffset": 84,
                    "EndOffset": 99
                }
            ]
        }
    ],
    "ErrorList": []
}
```

Per ulteriori informazioni, consulta Entità nella Guida per gli sviluppatori di Amazon Comprehend.

- Per i dettagli sull'API, consulta BatchDetectEntities AWS CLICommand Reference.

`batch-detect-key-phrases`

Il seguente esempio di codice mostra come utilizzare`batch-detect-key-phrases`.

AWS CLI

Come rilevare le frasi chiave di più input di testo

L'esempio `batch-detect-key-phrases` seguente analizza più testi di input e restituisce le frasi nominali chiave di ciascuno di essi. Per ogni previsione viene inoltre restituito il punteggio di attendibilità del modello pre-addestrato.

```
aws comprehend batch-detect-key-phrases \
    --language-code en \
```

```
    --text-list "Hello Zhang Wei, I am John, writing to you about the trip for
next Saturday." "Dear Jane, Your AnyCompany Financial Services LLC credit card
account 1111-XXXX-1111-XXXX has a minimum payment of $24.53 that is due by July
31st." "Please send customer feedback to Sunshine Spa, 123 Main St, Anywhere or to
Alice at AnySpa@example.com."
```

Output:


```
{
    "ResultList": [
        {
            "Index": 0,
            "KeyPhrases": [
                {
                    "Score": 0.99700927734375,
                    "Text": "Zhang Wei",
                    "BeginOffset": 6,
                    "EndOffset": 15
                },
                {
                    "Score": 0.9929308891296387,
                    "Text": "John",
                    "BeginOffset": 22,
                    "EndOffset": 26
                },
                {
                    "Score": 0.9997230172157288,
                    "Text": "the trip",
                    "BeginOffset": 49,
                    "EndOffset": 57
                },
                {
                    "Score": 0.9999470114707947,
                    "Text": "next Saturday",
                    "BeginOffset": 62,
                    "EndOffset": 75
                }
            ]
        },
        {
            "Index": 1,
            "KeyPhrases": [
                {
```

```
                    "Score": 0.8358274102210999,
                    "Text": "Dear Jane",
                    "BeginOffset": 0,
                    "EndOffset": 9
                },
                {
                    "Score": 0.989359974861145,
                    "Text": "Your AnyCompany Financial Services",
                    "BeginOffset": 11,
                    "EndOffset": 45
                },
                {
                    "Score": 0.8812323808670044,
                    "Text": "LLC credit card account 1111-XXXX-1111-XXXX",
                    "BeginOffset": 47,
                    "EndOffset": 90
                },
                {
                    "Score": 0.9999381899833679,
                    "Text": "a minimum payment",
                    "BeginOffset": 95,
                    "EndOffset": 112
                },
                {
                    "Score": 0.9997439980506897,
                    "Text": ".53",
                    "BeginOffset": 116,
                    "EndOffset": 119
                },
                {
                    "Score": 0.996875524520874,
                    "Text": "July 31st",
                    "BeginOffset": 135,
                    "EndOffset": 144
                }
            ]
        },
        {
            "Index": 2,
            "KeyPhrases": [
                {
                    "Score": 0.9990295767784119,
                    "Text": "customer feedback",
                    "BeginOffset": 12,
```

```
                    "EndOffset": 29
                },
                {
                    "Score": 0.9994127750396729,
                    "Text": "Sunshine Spa",
                    "BeginOffset": 33,
                    "EndOffset": 45
                },
                {
                    "Score": 0.9892991185188293,
                    "Text": "123 Main St",
                    "BeginOffset": 47,
                    "EndOffset": 58
                },
                {
                    "Score": 0.9969810843467712,
                    "Text": "Alice",
                    "BeginOffset": 75,
                    "EndOffset": 80
                },
                {
                    "Score": 0.9703696370124817,
                    "Text": "AnySpa@example.com",
                    "BeginOffset": 84,
                    "EndOffset": 99
                }
            ]
        }
    ],
    "ErrorList": []
}
```

Per ulteriori informazioni, consulta Frasi chiave nella Guida per gli sviluppatori di Amazon Comprehend.

- Per i dettagli sull'API, consulta BatchDetectKeyPhrases AWS CLICommand Reference.

`batch-detect-sentiment`

Il seguente esempio di codice mostra come utilizzare`batch-detect-sentiment`.

AWS CLI

Come rilevare il sentiment prevalente di più testi in input

L'esempio `batch-detect-sentiment` seguente analizza più testi di input e restituisce il sentiment prevalente (`POSITIVE`, `NEUTRAL`, `MIXED` o `NEGATIVE` di ciascuno di essi).

```
aws comprehend batch-detect-sentiment \
    --text-list "That movie was very boring, I can't believe it was over four hours long." "It is a beautiful day for hiking today." "My meal was okay, I'm excited to try other restaurants." \
    --language-code en
```

Output:

```
{
    "ResultList": [
        {
            "Index": 0,
            "Sentiment": "NEGATIVE",
            "SentimentScore": {
                "Positive": 0.00011316669406369328,
                "Negative": 0.9995445609092712,
                "Neutral": 0.00014722718333359808,
                "Mixed": 0.00019498742767609656
            }
        },
        {
            "Index": 1,
            "Sentiment": "POSITIVE",
            "SentimentScore": {
                "Positive": 0.9981263279914856,
                "Negative": 0.00015240783977787942,
                "Neutral": 0.0013876151060685515,
                "Mixed": 0.00033366199932061136
            }
        },
        {
            "Index": 2,
            "Sentiment": "MIXED",
            "SentimentScore": {
                "Positive": 0.15930435061454773,
                "Negative": 0.11471917480230331,
```

```
                "Neutral": 0.26897063851356506,
                "Mixed": 0.45700588822364807
            }
        }
    ],
    "ErrorList": []
}
```

Per ulteriori informazioni, consulta Sentiment nella Guida per gli sviluppatori di Amazon Comprehend.

- Per i dettagli sull'API, consulta BatchDetectSentiment AWS CLICommand Reference.

`batch-detect-syntax`

Il seguente esempio di codice mostra come utilizzare`batch-detect-syntax`.

AWS CLI

Come esaminare la sintassi e le parti del discorso delle parole in più testi di input

L'esempio `batch-detect-syntax` seguente analizza la sintassi di più testi di input e restituisce le diverse parti del discorso. Per ogni previsione viene inoltre restituito il punteggio di attendibilità del modello pre-addestrato.

```
aws comprehend batch-detect-syntax \
    --text-list "It is a beautiful day." "Can you please pass the salt?" "Please pay
 the bill before the 31st." \
    --language-code en
```

Output:

```
{
    "ResultList": [
        {
            "Index": 0,
            "SyntaxTokens": [
                {
                    "TokenId": 1,
                    "Text": "It",
                    "BeginOffset": 0,
                    "EndOffset": 2,
                    "PartOfSpeech": {
```

```
                "Tag": "PRON",
                "Score": 0.9999740719795227
            }
        },
        {
            "TokenId": 2,
            "Text": "is",
            "BeginOffset": 3,
            "EndOffset": 5,
            "PartOfSpeech": {
                "Tag": "VERB",
                "Score": 0.999937117099762
            }
        },
        {
            "TokenId": 3,
            "Text": "a",
            "BeginOffset": 6,
            "EndOffset": 7,
            "PartOfSpeech": {
                "Tag": "DET",
                "Score": 0.9999926686286926
            }
        },
        {
            "TokenId": 4,
            "Text": "beautiful",
            "BeginOffset": 8,
            "EndOffset": 17,
            "PartOfSpeech": {
                "Tag": "ADJ",
                "Score": 0.9987891912460327
            }
        },
        {
            "TokenId": 5,
            "Text": "day",
            "BeginOffset": 18,
            "EndOffset": 21,
            "PartOfSpeech": {
                "Tag": "NOUN",
                "Score": 0.9999778866767883
            }
        },
```

```
                {
                    "TokenId": 6,
                    "Text": ".",
                    "BeginOffset": 21,
                    "EndOffset": 22,
                    "PartOfSpeech": {
                        "Tag": "PUNCT",
                        "Score": 0.9999974966049194
                    }
                }
            ]
        },
        {
            "Index": 1,
            "SyntaxTokens": [
                {
                    "TokenId": 1,
                    "Text": "Can",
                    "BeginOffset": 0,
                    "EndOffset": 3,
                    "PartOfSpeech": {
                        "Tag": "AUX",
                        "Score": 0.9999770522117615
                    }
                },
                {
                    "TokenId": 2,
                    "Text": "you",
                    "BeginOffset": 4,
                    "EndOffset": 7,
                    "PartOfSpeech": {
                        "Tag": "PRON",
                        "Score": 0.9999986886978149
                    }
                },
                {
                    "TokenId": 3,
                    "Text": "please",
                    "BeginOffset": 8,
                    "EndOffset": 14,
                    "PartOfSpeech": {
                        "Tag": "INTJ",
                        "Score": 0.9681622385978699
                    }
```

```
                },
                {
                    "TokenId": 4,
                    "Text": "pass",
                    "BeginOffset": 15,
                    "EndOffset": 19,
                    "PartOfSpeech": {
                        "Tag": "VERB",
                        "Score": 0.9999874830245972
                    }
                },
                {
                    "TokenId": 5,
                    "Text": "the",
                    "BeginOffset": 20,
                    "EndOffset": 23,
                    "PartOfSpeech": {
                        "Tag": "DET",
                        "Score": 0.9999827146530151
                    }
                },
                {
                    "TokenId": 6,
                    "Text": "salt",
                    "BeginOffset": 24,
                    "EndOffset": 28,
                    "PartOfSpeech": {
                        "Tag": "NOUN",
                        "Score": 0.9995040893554688
                    }
                },
                {
                    "TokenId": 7,
                    "Text": "?",
                    "BeginOffset": 28,
                    "EndOffset": 29,
                    "PartOfSpeech": {
                        "Tag": "PUNCT",
                        "Score": 0.999998152256012
                    }
                }
            ]
        },
        {
```

```
            "Index": 2,
            "SyntaxTokens": [
                {
                    "TokenId": 1,
                    "Text": "Please",
                    "BeginOffset": 0,
                    "EndOffset": 6,
                    "PartOfSpeech": {
                        "Tag": "INTJ",
                        "Score": 0.9997857809066772
                    }
                },
                {
                    "TokenId": 2,
                    "Text": "pay",
                    "BeginOffset": 7,
                    "EndOffset": 10,
                    "PartOfSpeech": {
                        "Tag": "VERB",
                        "Score": 0.9999252557754517
                    }
                },
                {
                    "TokenId": 3,
                    "Text": "the",
                    "BeginOffset": 11,
                    "EndOffset": 14,
                    "PartOfSpeech": {
                        "Tag": "DET",
                        "Score": 0.9999842643737793
                    }
                },
                {
                    "TokenId": 4,
                    "Text": "bill",
                    "BeginOffset": 15,
                    "EndOffset": 19,
                    "PartOfSpeech": {
                        "Tag": "NOUN",
                        "Score": 0.9999588131904602
                    }
                },
                {
                    "TokenId": 5,
```

```
                    "Text": "before",
                    "BeginOffset": 20,
                    "EndOffset": 26,
                    "PartOfSpeech": {
                        "Tag": "ADP",
                        "Score": 0.9958304762840271
                    }
                },
                {
                    "TokenId": 6,
                    "Text": "the",
                    "BeginOffset": 27,
                    "EndOffset": 30,
                    "PartOfSpeech": {
                        "Tag": "DET",
                        "Score": 0.9999947547912598
                    }
                },
                {
                    "TokenId": 7,
                    "Text": "31st",
                    "BeginOffset": 31,
                    "EndOffset": 35,
                    "PartOfSpeech": {
                        "Tag": "NOUN",
                        "Score": 0.9924124479293823
                    }
                },
                {
                    "TokenId": 8,
                    "Text": ".",
                    "BeginOffset": 35,
                    "EndOffset": 36,
                    "PartOfSpeech": {
                        "Tag": "PUNCT",
                        "Score": 0.9999955892562866
                    }
                }
            ]
        }
    ],
    "ErrorList": []
}
```

Per ulteriori informazioni, consulta Analisi della sintassi nella Guida per gli sviluppatori di Amazon Comprehend.

- Per i dettagli sull'API, consulta BatchDetectSyntax AWS CLICommand Reference.

`batch-detect-targeted-sentiment`

Il seguente esempio di codice mostra come utilizzare`batch-detect-targeted-sentiment`.

AWS CLI

Come rilevare il sentiment e ogni entità denominata per più testi di input

L'esempio `batch-detect-targeted-sentiment` seguente analizza più testi di input e restituisce le entità denominate insieme al sentiment prevalente di ciascuno di essi. Per ogni previsione viene inoltre restituito il punteggio di attendibilità del modello pre-addestrato.

```
aws comprehend batch-detect-targeted-sentiment \
    --language-code en \
    --text-list "That movie was really boring, the original was way more
 entertaining" "The trail is extra beautiful today." "My meal was just okay."
```

Output:

```
{
    "ResultList": [
        {
            "Index": 0,
            "Entities": [
                {
                    "DescriptiveMentionIndex": [
                        0
                    ],
                    "Mentions": [
                        {
                            "Score": 0.9999009966850281,
                            "GroupScore": 1.0,
                            "Text": "movie",
                            "Type": "MOVIE",
                            "MentionSentiment": {
                                "Sentiment": "NEGATIVE",
                                "SentimentScore": {
                                    "Positive": 0.13887299597263336,
```

```
                                "Negative": 0.8057460188865662,
                                "Neutral": 0.05525200068950653,
                                "Mixed": 0.00012799999967683107
                            }
                        },
                        "BeginOffset": 5,
                        "EndOffset": 10
                    }
                ]
            },
            {
                "DescriptiveMentionIndex": [
                    0
                ],
                "Mentions": [
                    {
                        "Score": 0.9921110272407532,
                        "GroupScore": 1.0,
                        "Text": "original",
                        "Type": "MOVIE",
                        "MentionSentiment": {
                            "Sentiment": "POSITIVE",
                            "SentimentScore": {
                                "Positive": 0.9999989867210388,
                                "Negative": 9.999999974752427e-07,
                                "Neutral": 0.0,
                                "Mixed": 0.0
                            }
                        },
                        "BeginOffset": 34,
                        "EndOffset": 42
                    }
                ]
            }
        ]
    },
    {
        "Index": 1,
        "Entities": [
            {
                "DescriptiveMentionIndex": [
                    0
                ],
                "Mentions": [
```

```
                    {
                        "Score": 0.7545599937438965,
                        "GroupScore": 1.0,
                        "Text": "trail",
                        "Type": "OTHER",
                        "MentionSentiment": {
                            "Sentiment": "POSITIVE",
                            "SentimentScore": {
                                "Positive": 1.0,
                                "Negative": 0.0,
                                "Neutral": 0.0,
                                "Mixed": 0.0
                            }
                        },
                        "BeginOffset": 4,
                        "EndOffset": 9
                    }
                ]
            },
            {
                "DescriptiveMentionIndex": [
                    0
                ],
                "Mentions": [
                    {
                        "Score": 0.9999960064888,
                        "GroupScore": 1.0,
                        "Text": "today",
                        "Type": "DATE",
                        "MentionSentiment": {
                            "Sentiment": "NEUTRAL",
                            "SentimentScore": {
                                "Positive": 9.000000318337698e-06,
                                "Negative": 1.9999999949504854e-06,
                                "Neutral": 0.9999859929084778,
                                "Mixed": 3.999999989900971e-06
                            }
                        },
                        "BeginOffset": 29,
                        "EndOffset": 34
                    }
                ]
            }
        ]
```

```
        },
        {
            "Index": 2,
            "Entities": [
                {
                    "DescriptiveMentionIndex": [
                        0
                    ],
                    "Mentions": [
                        {
                            "Score": 0.9999880194664001,
                            "GroupScore": 1.0,
                            "Text": "My",
                            "Type": "PERSON",
                            "MentionSentiment": {
                                "Sentiment": "NEUTRAL",
                                "SentimentScore": {
                                    "Positive": 0.0,
                                    "Negative": 0.0,
                                    "Neutral": 1.0,
                                    "Mixed": 0.0
                                }
                            },
                            "BeginOffset": 0,
                            "EndOffset": 2
                        }
                    ]
                },
                {
                    "DescriptiveMentionIndex": [
                        0
                    ],
                    "Mentions": [
                        {
                            "Score": 0.9995260238647461,
                            "GroupScore": 1.0,
                            "Text": "meal",
                            "Type": "OTHER",
                            "MentionSentiment": {
                                "Sentiment": "NEUTRAL",
                                "SentimentScore": {
                                    "Positive": 0.04695599898695946,
                                    "Negative": 0.003226999891921878,
                                    "Neutral": 0.6091709733009338,
```

```
                                "Mixed": 0.34064599871635437
                            }
                        },
                        "BeginOffset": 3,
                        "EndOffset": 7
                    }
                ]
            }
        ]
    }
    ],
    "ErrorList": []
}
```

Per ulteriori informazioni, consulta Sentiment mirato nella Guida per gli sviluppatori di Amazon Comprehend.

- Per i dettagli sull'API, consulta BatchDetectTargetedSentiment AWS CLICommand Reference.

`classify-document`

Il seguente esempio di codice mostra come utilizzare`classify-document`.

AWS CLI

Come classificare i documenti con un endpoint specifico del modello

L'esempio `classify-document` seguente classifica un documento con un endpoint di un modello personalizzato. Il modello in questo esempio è stato addestrato su un set di dati contenente messaggi SMS etichettati come spam o non spam oppure "ham" (legittimi).

```
aws comprehend classify-document \
    --endpoint-arn arn:aws:comprehend:us-west-2:111122223333:document-classifier-endpoint/example-classifier-endpoint \
    --text "CONGRATULATIONS! TXT 1235550100 to win $5000"
```

Output:

```
{
    "Classes": [
        {
```

```
            "Name": "spam",
            "Score": 0.9998599290847778
        },
        {
            "Name": "ham",
            "Score": 0.00014001205272506922
        }
    ]
}
```

Per ulteriori informazioni, consulta Classificazione personalizzata nella Guida per gli sviluppatori di Amazon Comprehend.

- Per i dettagli sull'API, consulta ClassifyDocument AWS CLICommand Reference.

`contains-pii-entities`

Il seguente esempio di codice mostra come utilizzare`contains-pii-entities`.

AWS CLI

Come analizzare il testo di input per verificare la presenza di informazioni PII

L'esempio `contains-pii-entities` seguente analizza il testo di input per verificare la presenza di informazioni di identificazione personale (PII) e restituisce le etichette dei tipi di entità PII identificati, come nome, indirizzo, numero di conto corrente bancario o numero di telefono.

```
aws comprehend contains-pii-entities \
    --language-code en \
    --text "Hello Zhang Wei, I am John. Your AnyCompany Financial Services, LLC
 credit card
       account 1111-XXXX-1111-XXXX has a minimum payment of $24.53 that is due by
 July 31st. Based on your autopay settings,
       we will withdraw your payment on the due date from your bank account number
 XXXXXX1111 with the routing number XXXXX0000.
       Customer feedback for Sunshine Spa, 100 Main St, Anywhere. Send comments to
 Alice at AnySpa@example.com."
```

Output:

```
{
    "Labels": [
```

```
        {
            "Name": "NAME",
            "Score": 1.0
        },
        {
            "Name": "EMAIL",
            "Score": 1.0
        },
        {
            "Name": "BANK_ACCOUNT_NUMBER",
            "Score": 0.9995794296264648
        },
        {
            "Name": "BANK_ROUTING",
            "Score": 0.9173126816749573
        },
        {
            "Name": "CREDIT_DEBIT_NUMBER",
            "Score": 1.0
        }
}
```

Per ulteriori informazioni, consulta Informazioni di identificazione personale (PII) nella Guida per gli sviluppatori di Amazon Comprehend.

- Per i dettagli sull'API, consulta ContainsPiiEntities AWS CLICommand Reference.

`create-dataset`

Il seguente esempio di codice mostra come utilizzare`create-dataset`.

AWS CLI

Come creare un set di dati del flywheel

L'esempio `create-dataset` seguente crea un set di dati per un flywheel. Questo set di dati verrà utilizzato come dati di addestramento aggiuntivi, come specificato dal tag `--dataset-type`.

```
aws comprehend create-dataset \
    --flywheel-arn arn:aws:comprehend:us-west-2:111122223333:flywheel/flywheel-entity \
```

```
    --dataset-name example-dataset \
    --dataset-type "TRAIN" \
    --input-data-config file://inputConfig.json
```

Contenuto di `file://inputConfig.json`:

```
{
    "DataFormat": "COMPREHEND_CSV",
    "DocumentClassifierInputDataConfig": {
        "S3Uri": "s3://amzn-s3-demo-bucket/training-data.csv"
    }
}
```

Output:

```
{
    "DatasetArn": "arn:aws:comprehend:us-west-2:111122223333:flywheel/flywheel-
entity/dataset/example-dataset"
}
```

Per ulteriori informazioni, consulta Panoramica del volano dei dati nella Guida per gli sviluppatori di Amazon Comprehend.

- Per i dettagli sull'API, consulta CreateDataset AWS CLICommand Reference.

`create-document-classifier`

Il seguente esempio di codice mostra come utilizzare`create-document-classifier`.

AWS CLI

Come creare un classificatore di documenti per suddividere i documenti in categorie

L'esempio `create-document-classifier` seguente inizia il processo di addestramento per un modello di classificatore di documenti. Il file dei dati di addestramento `training.csv` si trova nel tag `--input-data-config`. `training.csv` è un documento a due colonne in cui le etichette o le classificazioni sono fornite nella prima colonna e i documenti nella seconda colonna.

```
aws comprehend create-document-classifier \
    --document-classifier-name example-classifier \
```

```
    --data-access-arn arn:aws:comprehend:us-west-2:111122223333:pii-entities-
detection-job/123456abcdeb0e11022f22a11EXAMPLE \
    --input-data-config "S3Uri=s3://amzn-s3-demo-bucket/" \
    --language-code en
```

Output:

```
{
    "DocumentClassifierArn": "arn:aws:comprehend:us-west-2:111122223333:document-
classifier/example-classifier"
}
```

Per ulteriori informazioni, consulta Classificazione personalizzata nella Guida per gli sviluppatori di Amazon Comprehend.

- Per i dettagli sull'API, consulta CreateDocumentClassifier AWS CLICommand Reference.

`create-endpoint`

Il seguente esempio di codice mostra come utilizzare`create-endpoint`.

AWS CLI

Come creare un endpoint per un modello personalizzato

L'esempio `create-endpoint` seguente crea un endpoint per l'inferenza sincrona per un modello personalizzato addestrato in precedenza.

```
aws comprehend create-endpoint \
    --endpoint-name example-classifier-endpoint-1 \
    --model-arn arn:aws:comprehend:us-west-2:111122223333:document-classifier/
example-classifier \
    --desired-inference-units 1
```

Output:

```
{
    "EndpointArn": "arn:aws:comprehend:us-west-2:111122223333:document-classifier-
endpoint/example-classifier-endpoint-1"
}
```

Per ulteriori informazioni, consulta Gestione degli endpoint di Amazon Comprehend nella Guida per gli sviluppatori di Amazon Comprehend.

- Per i dettagli sull'API, consulta CreateEndpoint AWS CLICommand Reference.

`create-entity-recognizer`

Il seguente esempio di codice mostra come utilizzare`create-entity-recognizer`.

AWS CLI

Come creare un riconoscitore di entità personalizzato

L'esempio `create-entity-recognizer` seguente avvia il job di addestramento per un modello di riconoscimento di entità personalizzato. In questo esempio, un file CSV contenente documenti di addestramento, `raw_text.csv`, e un elenco di entità CSV, `entity_list.csv`, vengono utilizzati per addestrare il modello. `entity-list.csv` contiene le seguenti colonne: testo e tipo.

```
aws comprehend create-entity-recognizer \
    --recognizer-name example-entity-recognizer
    --data-access-role-arn arn:aws:iam::111122223333:role/service-role/
AmazonComprehendServiceRole-example-role \
    --input-data-config "EntityTypes=[{Type=DEVICE}],Documents={S3Uri=s3://amzn-s3-
demo-bucket/trainingdata/raw_text.csv},EntityList={S3Uri=s3://amzn-s3-demo-bucket/
trainingdata/entity_list.csv}"
    --language-code en
```

Output:

```
{
    "EntityRecognizerArn": "arn:aws:comprehend:us-west-2:111122223333:example-
entity-recognizer/entityrecognizer1"
}
```

Per ulteriori informazioni, consulta Riconoscimento delle entità personalizzate nella Guida per gli sviluppatori di Amazon Comprehend.

- Per i dettagli sull'API, consulta CreateEntityRecognizer AWS CLICommand Reference.

`create-flywheel`

Il seguente esempio di codice mostra come utilizzare`create-flywheel`.

AWS CLI

Come creare un flywheel

L'esempio `create-flywheel` seguente crea un flywheel per orchestrare l'addestramento continuo di un modello di classificazione dei documenti o di riconoscimento delle entità. Il flywheel in questo esempio viene creato per gestire un modello addestrato esistente specificato dal tag `--active-model-arn`. Quando si crea il flywheel, sul tag `--input-data-lake` viene creato un data lake.

```
aws comprehend create-flywheel \
    --flywheel-name example-flywheel \
    --active-model-arn arn:aws:comprehend:us-west-2:111122223333:document-classifier/example-model/version/1 \
    --data-access-role-arn arn:aws:iam::111122223333:role/service-role/AmazonComprehendServiceRole-example-role \
    --data-lake-s3-uri "s3://amzn-s3-demo-bucket"
```

Output:

```
{
    "FlywheelArn": "arn:aws:comprehend:us-west-2:111122223333:flywheel/example-flywheel"
}
```

Per ulteriori informazioni, consulta Panoramica del volano dei dati nella Guida per gli sviluppatori di Amazon Comprehend.

- Per i dettagli sull'API, consulta CreateFlywheel AWS CLICommand Reference.

`delete-document-classifier`

Il seguente esempio di codice mostra come utilizzare`delete-document-classifier`.

AWS CLI

Come eliminare un classificatore di documenti personalizzato

L'esempio `delete-document-classifier` seguente elimina un modello di classificatore di documenti personalizzato.

```
aws comprehend delete-document-classifier \
```

```
    --document-classifier-arn arn:aws:comprehend:us-west-2:111122223333:document-
classifier/example-classifier-1
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Gestione degli endpoint di Amazon Comprehend nella Guida per gli sviluppatori di Amazon Comprehend.

- Per i dettagli sull'API, consulta DeleteDocumentClassifier AWS CLICommand Reference.

`delete-endpoint`

Il seguente esempio di codice mostra come utilizzare`delete-endpoint`.

AWS CLI

Come eliminare un endpoint per un modello personalizzato

L'esempio `delete-endpoint` seguente elimina un endpoint specifico del modello. Per eliminare il modello, è necessario eliminare tutti gli endpoint.

```
aws comprehend delete-endpoint \
    --endpoint-arn arn:aws:comprehend:us-west-2:111122223333:document-classifier-
endpoint/example-classifier-endpoint-1
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Gestione degli endpoint di Amazon Comprehend nella Guida per gli sviluppatori di Amazon Comprehend.

- Per i dettagli sull'API, consulta DeleteEndpoint AWS CLICommand Reference.

`delete-entity-recognizer`

Il seguente esempio di codice mostra come utilizzare`delete-entity-recognizer`.

AWS CLI

Come eliminare un modello di riconoscimento delle entità personalizzato

L'esempio `delete-entity-recognizer` seguente elimina un modello di riconoscimento delle entità personalizzato.

```
aws comprehend delete-entity-recognizer \
    --entity-recognizer-arn arn:aws:comprehend:us-west-2:111122223333:entity-recognizer/example-entity-recognizer-1
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Gestione degli endpoint di Amazon Comprehend nella Guida per gli sviluppatori di Amazon Comprehend.

- Per i dettagli sull'API, consulta DeleteEntityRecognizer AWS CLICommand Reference.

`delete-flywheel`

Il seguente esempio di codice mostra come utilizzare`delete-flywheel`.

AWS CLI

Come eliminare un flywheel

L'esempio `delete-flywheel` seguente mostra come eliminare un flywheel. Il data lake o il modello associato al flywheel non vengono eliminati.

```
aws comprehend delete-flywheel \
    --flywheel-arn arn:aws:comprehend:us-west-2:111122223333:flywheel/example-flywheel-1
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Panoramica del volano dei dati nella Guida per gli sviluppatori di Amazon Comprehend.

- Per i dettagli sull'API, consulta DeleteFlywheel AWS CLICommand Reference.

`delete-resource-policy`

Il seguente esempio di codice mostra come utilizzare`delete-resource-policy`.

AWS CLI

Come visualizzare una policy basata su risorse

L'esempio `delete-resource-policy` seguente elimina una policy basata su risorse da una risorsa Amazon Comprehend.

```
aws comprehend delete-resource-policy \
    --resource-arn arn:aws:comprehend:us-west-2:111122223333:document-classifier/
example-classifier-1/version/1
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Copiare modelli personalizzati tra AWS account nella Amazon Comprehend Developer Guide.

- Per i dettagli sull'API, consulta AWS CLI Command DeleteResourcePolicyReference.

`describe-dataset`

Il seguente esempio di codice mostra come utilizzare`describe-dataset`.

AWS CLI

Come descrivere un set di dati del flywheel

L'esempio `describe-dataset` seguente ottiene le proprietà del set di dati di un flywheel.

```
aws comprehend describe-dataset \
    --dataset-arn arn:aws:comprehend:us-west-2:111122223333:flywheel/flywheel-
entity/dataset/example-dataset
```

Output:

```
{
    "DatasetProperties": {
        "DatasetArn": "arn:aws:comprehend:us-west-2:111122223333:flywheel/flywheel-
entity/dataset/example-dataset",
        "DatasetName": "example-dataset",
        "DatasetType": "TRAIN",
        "DatasetS3Uri": "s3://amzn-s3-demo-bucket/flywheel-entity/
schemaVersion=1/12345678A123456Z/datasets/example-dataset/20230616T203710Z/",
        "Status": "CREATING",
        "CreationTime": "2023-06-16T20:37:10.400000+00:00"
    }
```

```
}
```

Per ulteriori informazioni, consulta Panoramica del volano dei dati nella Guida per gli sviluppatori di Amazon Comprehend.

- Per i dettagli sull'API, consulta DescribeDataset AWS CLICommand Reference.

`describe-document-classification-job`

Il seguente esempio di codice mostra come utilizzare`describe-document-classification-job`.

AWS CLI

Come descrivere un processo di classificazione dei documenti

L'esempio `describe-document-classification-job` seguente ottiene le proprietà di un processo asincrono di classificazione di documenti.

```
aws comprehend describe-document-classification-job \
    --job-id 123456abcdeb0e11022f22a11EXAMPLE
```

Output:

```
{
    "DocumentClassificationJobProperties": {
        "JobId": "123456abcdeb0e11022f22a11EXAMPLE",
        "JobArn": "arn:aws:comprehend:us-west-2:111122223333:document-
classification-job/123456abcdeb0e11022f22a11EXAMPLE",
        "JobName": "exampleclassificationjob",
        "JobStatus": "COMPLETED",
        "SubmitTime": "2023-06-14T17:09:51.788000+00:00",
        "EndTime": "2023-06-14T17:15:58.582000+00:00",
        "DocumentClassifierArn": "arn:aws:comprehend:us-
west-2:111122223333:document-classifier/mymodel/version/1",
        "InputDataConfig": {
            "S3Uri": "s3://amzn-s3-demo-bucket/jobdata/",
            "InputFormat": "ONE_DOC_PER_LINE"
        },
        "OutputDataConfig": {
            "S3Uri": "s3://amzn-s3-demo-destination-bucket/testfolder/111122223333-
CLN-123456abcdeb0e11022f22a11EXAMPLE/output/output.tar.gz"
```

```
        },
        "DataAccessRoleArn": "arn:aws:iam::111122223333:role/service-role/
AmazonComprehendServiceRole-servicerole"
    }
}
```

Per ulteriori informazioni, consulta Classificazione personalizzata nella Guida per gli sviluppatori di Amazon Comprehend.

- Per i dettagli sull'API, consulta DescribeDocumentClassificationJob AWS CLICommand Reference.

`describe-document-classifier`

Il seguente esempio di codice mostra come utilizzare`describe-document-classifier`.

AWS CLI

Come descrivere un classificatore di documenti

L'esempio `describe-document-classifier` seguente ottiene le proprietà di un modello di classificatore di documenti personalizzato.

```
aws comprehend describe-document-classifier \
    --document-classifier-arn arn:aws:comprehend:us-west-2:111122223333:document-
classifier/example-classifier-1
```

Output:

```
{
    "DocumentClassifierProperties": {
        "DocumentClassifierArn": "arn:aws:comprehend:us-
west-2:111122223333:document-classifier/example-classifier-1",
        "LanguageCode": "en",
        "Status": "TRAINED",
        "SubmitTime": "2023-06-13T19:04:15.735000+00:00",
        "EndTime": "2023-06-13T19:42:31.752000+00:00",
        "TrainingStartTime": "2023-06-13T19:08:20.114000+00:00",
        "TrainingEndTime": "2023-06-13T19:41:35.080000+00:00",
        "InputDataConfig": {
            "DataFormat": "COMPREHEND_CSV",
```

```
            "S3Uri": "s3://amzn-s3-demo-bucket/trainingdata"
        },
        "OutputDataConfig": {},
        "ClassifierMetadata": {
            "NumberOfLabels": 3,
            "NumberOfTrainedDocuments": 5016,
            "NumberOfTestDocuments": 557,
            "EvaluationMetrics": {
                "Accuracy": 0.9856,
                "Precision": 0.9919,
                "Recall": 0.9459,
                "F1Score": 0.9673,
                "MicroPrecision": 0.9856,
                "MicroRecall": 0.9856,
                "MicroF1Score": 0.9856,
                "HammingLoss": 0.0144
            }
        },
        "DataAccessRoleArn": "arn:aws:iam::111122223333:role/service-role/
AmazonComprehendServiceRole-example-role",
        "Mode": "MULTI_CLASS"
    }
}
```

Per ulteriori informazioni, consulta Creazione e gestione di modelli personalizzati nella Guida per gli sviluppatori di Amazon Comprehend.

- Per i dettagli sull'API, consulta DescribeDocumentClassifier AWS CLICommand Reference.

`describe-dominant-language-detection-job`

Il seguente esempio di codice mostra come utilizzare`describe-dominant-language-detection-job`.

AWS CLI

Come descrivere un processo di rilevamento della lingua dominante.

L'esempio `describe-dominant-language-detection-job` seguente ottiene le proprietà di un processo di rilevamento della lingua dominante asincrono.

```
aws comprehend describe-dominant-language-detection-job \
```

```
    --job-id 123456abcdeb0e11022f22a11EXAMPLE
```

Output:

```
{
    "DominantLanguageDetectionJobProperties": {
        "JobId": "123456abcdeb0e11022f22a11EXAMPLE",
        "JobArn": "arn:aws:comprehend:us-west-2:111122223333:dominant-language-
detection-job/123456abcdeb0e11022f22a11EXAMPLE",
        "JobName": "languageanalysis1",
        "JobStatus": "IN_PROGRESS",
        "SubmitTime": "2023-06-09T18:10:38.037000+00:00",
        "InputDataConfig": {
            "S3Uri": "s3://amzn-s3-demo-bucket",
            "InputFormat": "ONE_DOC_PER_LINE"
        },
        "OutputDataConfig": {
            "S3Uri": "s3://amzn-s3-demo-destination-bucket/testfolder/111122223333-
LANGUAGE-123456abcdeb0e11022f22a11EXAMPLE/output/output.tar.gz"
        },
        "DataAccessRoleArn": "arn:aws:iam::111122223333:role/service-role/
AmazonComprehendServiceRole-example-role"
    }
}
```

Per ulteriori informazioni, consulta Analisi asincrona degli approfondimenti di Amazon Comprehend nella Guida per gli sviluppatori di Amazon Comprehend.

- Per i dettagli sull'API, consulta DescribeDominantLanguageDetectionJob AWS CLICommand Reference.

`describe-endpoint`

Il seguente esempio di codice mostra come utilizzare`describe-endpoint`.

AWS CLI

Come descrivere un endpoint specifico

L'esempio `describe-endpoint` seguente ottiene le proprietà di un endpoint specifico del modello.

```
aws comprehend describe-endpoint \
```

```
    --endpoint-arn arn:aws:comprehend:us-west-2:111122223333:document-classifier-
endpoint/example-classifier-endpoint
```

Output:

```
{
    "EndpointProperties": {
        "EndpointArn": "arn:aws:comprehend:us-west-2:111122223333:document-
classifier-endpoint/example-classifier-endpoint,
        "Status": "IN_SERVICE",
        "ModelArn": "arn:aws:comprehend:us-west-2:111122223333:document-classifier/
exampleclassifier1",
        "DesiredModelArn": "arn:aws:comprehend:us-west-2:111122223333:document-
classifier/exampleclassifier1",
        "DesiredInferenceUnits": 1,
        "CurrentInferenceUnits": 1,
        "CreationTime": "2023-06-13T20:32:54.526000+00:00",
        "LastModifiedTime": "2023-06-13T20:32:54.526000+00:00"
    }
}
```

Per ulteriori informazioni, consulta Gestione degli endpoint di Amazon Comprehend nella Guida per gli sviluppatori di Amazon Comprehend.

- Per i dettagli sull'API, consulta DescribeEndpoint AWS CLICommand Reference.

`describe-entities-detection-job`

Il seguente esempio di codice mostra come utilizzare`describe-entities-detection-job`.

AWS CLI

Come descrivere un processo di rilevamento delle entità

L'esempio `describe-entities-detection-job` seguente ottiene le proprietà di un processo di rilevamento delle entità asincrono.

```
aws comprehend describe-entities-detection-job \
    --job-id 123456abcdeb0e11022f22a11EXAMPLE
```

Output:

```
{
    "EntitiesDetectionJobProperties": {
        "JobId": "123456abcdeb0e11022f22a11EXAMPLE",
        "JobArn": "arn:aws:comprehend:us-west-2:111122223333:entities-detection-
job/123456abcdeb0e11022f22a11EXAMPLE",
        "JobName": "example-entity-detector",
        "JobStatus": "COMPLETED",
        "SubmitTime": "2023-06-08T21:30:15.323000+00:00",
        "EndTime": "2023-06-08T21:40:23.509000+00:00",
        "InputDataConfig": {
            "S3Uri": "s3://amzn-s3-demo-bucket/AsyncBatchJobs/",
            "InputFormat": "ONE_DOC_PER_LINE"
        },
        "OutputDataConfig": {
            "S3Uri": "s3://amzn-s3-demo-bucket/thefolder/111122223333-
NER-123456abcdeb0e11022f22a11EXAMPLE/output/output.tar.gz"
        },
        "LanguageCode": "en",
        "DataAccessRoleArn": "arn:aws:iam::12345678012:role/service-role/
AmazonComprehendServiceRole-example-role"
    }
}
```

Per ulteriori informazioni, consulta Analisi asincrona degli approfondimenti di Amazon Comprehend nella Guida per gli sviluppatori di Amazon Comprehend.

- Per i dettagli sull'API, consulta DescribeEntitiesDetectionJob AWS CLICommand Reference.

`describe-entity-recognizer`

Il seguente esempio di codice mostra come utilizzare`describe-entity-recognizer`.

AWS CLI

Come descrivere un riconoscitore di entità

L'esempio `describe-entity-recognizer` seguente ottiene le proprietà di un modello di riconoscimento delle entità personalizzato.

```
aws comprehend describe-entity-recognizer \
    entity-recognizer-arn arn:aws:comprehend:us-west-2:111122223333:entity-
recognizer/business-recongizer-1/version/1
```

Output:

```
{
    "EntityRecognizerProperties": {
        "EntityRecognizerArn": "arn:aws:comprehend:us-west-2:111122223333:entity-
recognizer/business-recongizer-1/version/1",
        "LanguageCode": "en",
        "Status": "TRAINED",
        "SubmitTime": "2023-06-14T20:44:59.631000+00:00",
        "EndTime": "2023-06-14T20:59:19.532000+00:00",
        "TrainingStartTime": "2023-06-14T20:48:52.811000+00:00",
        "TrainingEndTime": "2023-06-14T20:58:11.473000+00:00",
        "InputDataConfig": {
            "DataFormat": "COMPREHEND_CSV",
            "EntityTypes": [
                {
                    "Type": "BUSINESS"
                }
            ],
            "Documents": {
                "S3Uri": "s3://amzn-s3-demo-bucket/trainingdata/dataset/",
                "InputFormat": "ONE_DOC_PER_LINE"
            },
            "EntityList": {
                "S3Uri": "s3://amzn-s3-demo-bucket/trainingdata/entity.csv"
            }
        },
        "RecognizerMetadata": {
            "NumberOfTrainedDocuments": 1814,
            "NumberOfTestDocuments": 486,
            "EvaluationMetrics": {
                "Precision": 100.0,
                "Recall": 100.0,
                "F1Score": 100.0
            },
            "EntityTypes": [
                {
                    "Type": "BUSINESS",
                    "EvaluationMetrics": {
                        "Precision": 100.0,
                        "Recall": 100.0,
                        "F1Score": 100.0
                    },
                    "NumberOfTrainMentions": 1520
```

```
                }
            ]
        },
        "DataAccessRoleArn": "arn:aws:iam::111122223333:role/service-role/
AmazonComprehendServiceRole-example-role",
        "VersionName": "1"
    }
}
```

Per ulteriori informazioni, consulta Riconoscimento delle entità personalizzate nella Guida per gli sviluppatori di Amazon Comprehend.

- Per i dettagli sull'API, consulta DescribeEntityRecognizer AWS CLICommand Reference.

`describe-events-detection-job`

Il seguente esempio di codice mostra come utilizzare`describe-events-detection-job`.

AWS CLI

Come descrivere un processo di rilevamento degli eventi.

L'esempio `describe-events-detection-job` seguente ottiene le proprietà di un processo di rilevamento degli eventi asincrono.

```
aws comprehend describe-events-detection-job \
    --job-id 123456abcdeb0e11022f22a11EXAMPLE
```

Output:

```
{
    "EventsDetectionJobProperties": {
        "JobId": "123456abcdeb0e11022f22a11EXAMPLE",
        "JobArn": "arn:aws:comprehend:us-west-2:111122223333:events-detection-
job/123456abcdeb0e11022f22a11EXAMPLE",
        "JobName": "events_job_1",
        "JobStatus": "IN_PROGRESS",
        "SubmitTime": "2023-06-12T18:45:56.054000+00:00",
        "InputDataConfig": {
            "S3Uri": "s3://amzn-s3-demo-bucket/EventsData",
            "InputFormat": "ONE_DOC_PER_LINE"
        },
        "OutputDataConfig": {
```

```
            "S3Uri": "s3://amzn-s3-demo-destination-bucket/testfolder/111122223333-
EVENTS-123456abcdeb0e11022f22a11EXAMPLE/output/"
        },
        "LanguageCode": "en",
        "DataAccessRoleArn": "arn:aws:iam::111122223333:role/service-role/
AmazonComprehendServiceRole-example-role",
        "TargetEventTypes": [
            "BANKRUPTCY",
            "EMPLOYMENT",
            "CORPORATE_ACQUISITION",
            "CORPORATE_MERGER",
            "INVESTMENT_GENERAL"
        ]
    }
}
```

Per ulteriori informazioni, consulta Analisi asincrona degli approfondimenti di Amazon Comprehend nella Guida per gli sviluppatori di Amazon Comprehend.

- Per i dettagli sull'API, consulta DescribeEventsDetectionJob AWS CLICommand Reference.

`describe-flywheel-iteration`

Il seguente esempio di codice mostra come utilizzare`describe-flywheel-iteration`.

AWS CLI

Come descrivere un'iterazione del flywheel

L'esempio `describe-flywheel-iteration` seguente ottiene le proprietà dell'iterazione di un flywheel.

```
aws comprehend describe-flywheel-iteration \
    --flywheel-arn arn:aws:comprehend:us-west-2:111122223333:flywheel/example-
flywheel \
    --flywheel-iteration-id 20232222AEXAMPLE
```

Output:

```
{
    "FlywheelIterationProperties": {
        "FlywheelArn": "arn:aws:comprehend:us-west-2:111122223333:flywheel/flywheel-
entity",
```

```
        "FlywheelIterationId": "20232222AEXAMPLE",
        "CreationTime": "2023-06-16T21:10:26.385000+00:00",
        "EndTime": "2023-06-16T23:33:16.827000+00:00",
        "Status": "COMPLETED",
        "Message": "FULL_ITERATION: Flywheel iteration performed all functions
 successfully.",
        "EvaluatedModelArn": "arn:aws:comprehend:us-west-2:111122223333:document-
classifier/example-classifier/version/1",
        "EvaluatedModelMetrics": {
            "AverageF1Score": 0.7742663922375772,
            "AveragePrecision": 0.8287636394041166,
            "AverageRecall": 0.7427084833645399,
            "AverageAccuracy": 0.8795394154118689
        },
        "TrainedModelArn": "arn:aws:comprehend:us-west-2:111122223333:document-
classifier/example-classifier/version/Comprehend-Generated-v1-bb52d585",
        "TrainedModelMetrics": {
            "AverageF1Score": 0.9767700253081214,
            "AveragePrecision": 0.9767700253081214,
            "AverageRecall": 0.9767700253081214,
            "AverageAccuracy": 0.9858281665190434
        },
        "EvaluationManifestS3Prefix": "s3://amzn-s3-demo-destination-bucket/
flywheel-entity/schemaVersion=1/20230616T200543Z/evaluation/20230616T211026Z/"
    }
}
```

Per ulteriori informazioni, consulta Panoramica del volano dei dati nella Guida per gli sviluppatori di Amazon Comprehend.

- Per i dettagli sull'API, consulta DescribeFlywheelIteration AWS CLICommand Reference.

`describe-flywheel`

Il seguente esempio di codice mostra come utilizzare`describe-flywheel`.

AWS CLI

Come descrivere un flywheel

L'esempio `describe-flywheel` seguente ottiene le proprietà di un flywheel. In questo esempio, il modello associato al flywheel è un modello di classificazione personalizzato addestrato a classificare i documenti come spam o non spam oppure come "ham" (legittimi).

```
aws comprehend describe-flywheel \
    --flywheel-arn arn:aws:comprehend:us-west-2:111122223333:flywheel/example-flywheel
```

Output:

```
{
    "FlywheelProperties": {
        "FlywheelArn": "arn:aws:comprehend:us-west-2:111122223333:flywheel/example-flywheel",
        "ActiveModelArn": "arn:aws:comprehend:us-west-2:111122223333:document-classifier/example-model/version/1",
        "DataAccessRoleArn": "arn:aws:iam::111122223333:role/service-role/AmazonComprehendServiceRole-example-role",
        "TaskConfig": {
            "LanguageCode": "en",
            "DocumentClassificationConfig": {
                "Mode": "MULTI_CLASS",
                "Labels": [
                    "ham",
                    "spam"
                ]
            }
        },
        "DataLakeS3Uri": "s3://amzn-s3-demo-bucket/example-flywheel/schemaVersion=1/20230616T200543Z/",
        "DataSecurityConfig": {},
        "Status": "ACTIVE",
        "ModelType": "DOCUMENT_CLASSIFIER",
        "CreationTime": "2023-06-16T20:05:43.242000+00:00",
        "LastModifiedTime": "2023-06-16T20:21:43.567000+00:00"
    }
}
```

Per ulteriori informazioni, consulta Panoramica del volano dei dati nella Guida per gli sviluppatori di Amazon Comprehend.

- Per i dettagli sull'API, consulta DescribeFlywheel AWS CLICommand Reference.

`describe-key-phrases-detection-job`

Il seguente esempio di codice mostra come utilizzare`describe-key-phrases-detection-job`.

AWS CLI

Come descrivere un processo di rilevamento di frasi chiave

L'esempio `describe-key-phrases-detection-job` seguente ottiene le proprietà di un processo di rilevamento asincrono delle frasi chiave.

```
aws comprehend describe-key-phrases-detection-job \
    --job-id 123456abcdeb0e11022f22a11EXAMPLE
```

Output:

```
{
    "KeyPhrasesDetectionJobProperties": {
        "JobId": "69aa080c00fc68934a6a98f10EXAMPLE",
        "JobArn": "arn:aws:comprehend:us-west-2:111122223333:key-phrases-detection-
job/69aa080c00fc68934a6a98f10EXAMPLE",
        "JobName": "example-key-phrases-detection-job",
        "JobStatus": "COMPLETED",
        "SubmitTime": 1686606439.177,
        "EndTime": 1686606806.157,
        "InputDataConfig": {
            "S3Uri": "s3://dereksbucket1001/EventsData/",
            "InputFormat": "ONE_DOC_PER_LINE"
        },
        "OutputDataConfig": {
            "S3Uri": "s3://dereksbucket1002/testfolder/111122223333-
KP-69aa080c00fc68934a6a98f10EXAMPLE/output/output.tar.gz"
        },
        "LanguageCode": "en",
        "DataAccessRoleArn": "arn:aws:iam::111122223333:role/service-role/
AmazonComprehendServiceRole-testrole"
    }
}
```

Per ulteriori informazioni, consulta Analisi asincrona degli approfondimenti di Amazon Comprehend nella Guida per gli sviluppatori di Amazon Comprehend.

- Per i dettagli sull'API, consulta DescribeKeyPhrasesDetectionJob AWS CLICommand Reference.

`describe-pii-entities-detection-job`

Il seguente esempio di codice mostra come utilizzare`describe-pii-entities-detection-job`.

AWS CLI

Come descrivere un processo di rilevamento di entità PII

L'esempio `describe-pii-entities-detection-job` seguente ottiene le proprietà di un processo di rilevamento asincrono delle entità pii.

```
aws comprehend describe-pii-entities-detection-job \
    --job-id 123456abcdeb0e11022f22a11EXAMPLE
```

Output:

```
{
    "PiiEntitiesDetectionJobProperties": {
        "JobId": "123456abcdeb0e11022f22a11EXAMPLE",
        "JobArn": "arn:aws:comprehend:us-west-2:111122223333:pii-entities-detection-
job/123456abcdeb0e11022f22a11EXAMPLE",
        "JobName": "example-pii-entities-job",
        "JobStatus": "IN_PROGRESS",
        "SubmitTime": "2023-06-08T21:30:15.323000+00:00",
        "EndTime": "2023-06-08T21:40:23.509000+00:00",
        "InputDataConfig": {
            "S3Uri": "s3://amzn-s3-demo-bucket/AsyncBatchJobs/",
            "InputFormat": "ONE_DOC_PER_LINE"
        },
        "OutputDataConfig": {
            "S3Uri": "s3://amzn-s3-demo-bucket/thefolder/111122223333-
NER-123456abcdeb0e11022f22a11EXAMPLE/output/output.tar.gz"
        },
        "LanguageCode": "en",
        "DataAccessRoleArn": "arn:aws:iam::12345678012:role/service-role/
AmazonComprehendServiceRole-example-role"
    }
}
```

Per ulteriori informazioni, consulta Analisi asincrona degli approfondimenti di Amazon Comprehend nella Guida per gli sviluppatori di Amazon Comprehend.

- Per i dettagli sull'API, consulta DescribePiiEntitiesDetectionJob AWS CLICommand Reference.

`describe-resource-policy`

Il seguente esempio di codice mostra come utilizzare`describe-resource-policy`.

AWS CLI

Come descrivere una policy delle risorse allegata a un modello

L'esempio `describe-resource-policy` seguente ottiene le proprietà di una policy basata su risorse collegata a un modello.

```
aws comprehend describe-resource-policy \
    --resource-arn arn:aws:comprehend:us-west-2:111122223333:document-classifier/
example-classifier/version/1
```

Output:

```
{
    "ResourcePolicy": "{\"Version\":\"2012-10-17\",\"Statement\":[{\"Effect\":
\"Allow\",\"Principal\":{\"AWS\":\"arn:aws:iam::444455556666:root\"},\"Action\":
\"comprehend:ImportModel\",\"Resource\":\"*\"}]}",
    "CreationTime": "2023-06-19T18:44:26.028000+00:00",
    "LastModifiedTime": "2023-06-19T18:53:02.002000+00:00",
    "PolicyRevisionId": "baa675d069d07afaa2aa3106ae280f61"
}
```

Per ulteriori informazioni, consulta Copiare modelli personalizzati tra AWS account nella Amazon Comprehend Developer Guide.

- Per i dettagli sull'API, consulta AWS CLI Command DescribeResourcePolicyReference.

`describe-sentiment-detection-job`

Il seguente esempio di codice mostra come utilizzare`describe-sentiment-detection-job`.

AWS CLI

Come descrivere un processo di rilevamento del sentiment

L'esempio `describe-sentiment-detection-job` seguente ottiene le proprietà di un processo di rilevamento del sentiment asincrono.

```
aws comprehend describe-sentiment-detection-job \
```

```
    --job-id 123456abcdeb0e11022f22a11EXAMPLE
```

Output:

```
{
    "SentimentDetectionJobProperties": {
        "JobId": "123456abcdeb0e11022f22a11EXAMPLE",
        "JobArn": "arn:aws:comprehend:us-west-2:111122223333:sentiment-detection-
job/123456abcdeb0e11022f22a11EXAMPLE",
        "JobName": "movie_review_analysis",
        "JobStatus": "IN_PROGRESS",
        "SubmitTime": "2023-06-09T23:16:15.956000+00:00",
        "InputDataConfig": {
            "S3Uri": "s3://amzn-s3-demo-bucket/MovieData",
            "InputFormat": "ONE_DOC_PER_LINE"
        },
        "OutputDataConfig": {
            "S3Uri": "s3://amzn-s3-demo-destination-bucket/testfolder/111122223333-
TS-123456abcdeb0e11022f22a11EXAMPLE/output/output.tar.gz"
        },
        "LanguageCode": "en",
        "DataAccessRoleArn": "arn:aws:iam::111122223333:role/service-role/
AmazonComprehendServiceRole-servicerole"
    }
}
```

Per ulteriori informazioni, consulta Analisi asincrona degli approfondimenti di Amazon Comprehend nella Guida per gli sviluppatori di Amazon Comprehend.

- Per i dettagli sull'API, consulta DescribeSentimentDetectionJob AWS CLICommand Reference.

`describe-targeted-sentiment-detection-job`

Il seguente esempio di codice mostra come utilizzare`describe-targeted-sentiment-detection-job`.

AWS CLI

Come descrivere un processo di rilevamento del sentiment mirato

L'esempio `describe-targeted-sentiment-detection-job` seguente ottiene le proprietà di un processo di rilevamento del sentiment mirato asincrono.

```
aws comprehend describe-targeted-sentiment-detection-job \
    --job-id 123456abcdeb0e11022f22a11EXAMPLE
```

Output:

```
{
    "TargetedSentimentDetectionJobProperties": {
        "JobId": "123456abcdeb0e11022f22a11EXAMPLE",
        "JobArn": "arn:aws:comprehend:us-west-2:111122223333:targeted-sentiment-
detection-job/123456abcdeb0e11022f22a11EXAMPLE",
        "JobName": "movie_review_analysis",
        "JobStatus": "IN_PROGRESS",
        "SubmitTime": "2023-06-09T23:16:15.956000+00:00",
        "InputDataConfig": {
            "S3Uri": "s3://amzn-s3-demo-bucket/MovieData",
            "InputFormat": "ONE_DOC_PER_LINE"
        },
        "OutputDataConfig": {
            "S3Uri": "s3://amzn-s3-demo-destination-bucket/testfolder/111122223333-
TS-123456abcdeb0e11022f22a11EXAMPLE/output/output.tar.gz"
        },
        "LanguageCode": "en",
        "DataAccessRoleArn": "arn:aws:iam::111122223333:role/service-role/
AmazonComprehendServiceRole-servicerole"
    }
}
```

Per ulteriori informazioni, consulta Analisi asincrona degli approfondimenti di Amazon Comprehend nella Guida per gli sviluppatori di Amazon Comprehend.

- Per i dettagli sull'API, consulta DescribeTargetedSentimentDetectionJob AWS CLICommand Reference.

`describe-topics-detection-job`

Il seguente esempio di codice mostra come utilizzare`describe-topics-detection-job`.

AWS CLI

Come descrivere un processo di rilevamento di argomenti

L'esempio `describe-topics-detection-job` seguente ottiene le proprietà di un processo asincrono di rilevamento degli argomenti.

```
aws comprehend describe-topics-detection-job \
    --job-id 123456abcdeb0e11022f22a11EXAMPLE
```

Output:

```
{
    "TopicsDetectionJobProperties": {
        "JobId": "123456abcdeb0e11022f22a11EXAMPLE",
        "JobArn": "arn:aws:comprehend:us-west-2:111122223333:topics-detection-
job/123456abcdeb0e11022f22a11EXAMPLE",
        "JobName": "example_topics_detection",
        "JobStatus": "IN_PROGRESS",
        "SubmitTime": "2023-06-09T18:44:43.414000+00:00",
        "InputDataConfig": {
            "S3Uri": "s3://amzn-s3-demo-bucket",
            "InputFormat": "ONE_DOC_PER_LINE"
        },
        "OutputDataConfig": {
            "S3Uri": "s3://amzn-s3-demo-destination-bucket/testfolder/111122223333-
TOPICS-123456abcdeb0e11022f22a11EXAMPLE/output/output.tar.gz"
        },
        "NumberOfTopics": 10,
        "DataAccessRoleArn": "arn:aws:iam::111122223333:role/service-role/
AmazonComprehendServiceRole-examplerole"
    }
}
```

Per ulteriori informazioni, consulta Analisi asincrona degli approfondimenti di Amazon Comprehend nella Guida per gli sviluppatori di Amazon Comprehend.

- Per i dettagli sull'API, consulta DescribeTopicsDetectionJob AWS CLICommand Reference.

`detect-dominant-language`

Il seguente esempio di codice mostra come utilizzare`detect-dominant-language`.

AWS CLI

Come rilevare la lingua dominante del testo di input

Il comando `detect-dominant-language` seguente analizza il testo di input e identifica la lingua dominante. Viene inoltre restituito il punteggio di attendibilità del modello pre-addestrato.

```
aws comprehend detect-dominant-language \
    --text "It is a beautiful day in Seattle."
```

Output:

```
{
    "Languages": [
        {
            "LanguageCode": "en",
            "Score": 0.9877256155014038
        }
    ]
}
```

Per ulteriori informazioni, consulta Lingua dominante nella Guida per gli sviluppatori di Amazon Comprehend.

- Per i dettagli sull'API, consulta DetectDominantLanguage AWS CLICommand Reference.

`detect-entities`

Il seguente esempio di codice mostra come utilizzare`detect-entities`.

AWS CLI

Come rilevare entità denominate nel testo di input

L'esempio `detect-entities` seguente analizza il testo di input e restituisce le entità denominate. Per ogni previsione viene inoltre restituito il punteggio di attendibilità del modello pre-addestrato.

```
aws comprehend detect-entities \
    --language-code en \
    --text "Hello Zhang Wei, I am John. Your AnyCompany Financial Services, LLC
 credit card \
    account 1111-XXXX-1111-XXXX has a minimum payment of $24.53 that is due by July
 31st. Based on your autopay settings, \
    we will withdraw your payment on the due date from your bank account number
 XXXXXX1111 with the routing number XXXXX0000. \
```

```
    Customer feedback for Sunshine Spa, 123 Main St, Anywhere. Send comments to
 Alice at AnySpa@example.com."
```

Output:

```
{
    "Entities": [
        {
            "Score": 0.9994556307792664,
            "Type": "PERSON",
            "Text": "Zhang Wei",
            "BeginOffset": 6,
            "EndOffset": 15
        },
        {
            "Score": 0.9981022477149963,
            "Type": "PERSON",
            "Text": "John",
            "BeginOffset": 22,
            "EndOffset": 26
        },
        {
            "Score": 0.9986887574195862,
            "Type": "ORGANIZATION",
            "Text": "AnyCompany Financial Services, LLC",
            "BeginOffset": 33,
            "EndOffset": 67
        },
        {
            "Score": 0.9959119558334351,
            "Type": "OTHER",
            "Text": "1111-XXXX-1111-XXXX",
            "BeginOffset": 88,
            "EndOffset": 107
        },
        {
            "Score": 0.9708039164543152,
            "Type": "QUANTITY",
            "Text": ".53",
            "BeginOffset": 133,
            "EndOffset": 136
        },
        {
```

```
            "Score": 0.9987268447875977,
            "Type": "DATE",
            "Text": "July 31st",
            "BeginOffset": 152,
            "EndOffset": 161
        },
        {
            "Score": 0.9858865737915039,
            "Type": "OTHER",
            "Text": "XXXXXX1111",
            "BeginOffset": 271,
            "EndOffset": 281
        },
        {
            "Score": 0.9700471758842468,
            "Type": "OTHER",
            "Text": "XXXXX0000",
            "BeginOffset": 306,
            "EndOffset": 315
        },
        {
            "Score": 0.9591118693351746,
            "Type": "ORGANIZATION",
            "Text": "Sunshine Spa",
            "BeginOffset": 340,
            "EndOffset": 352
        },
        {
            "Score": 0.9797496795654297,
            "Type": "LOCATION",
            "Text": "123 Main St",
            "BeginOffset": 354,
            "EndOffset": 365
        },
        {
            "Score": 0.994929313659668,
            "Type": "PERSON",
            "Text": "Alice",
            "BeginOffset": 394,
            "EndOffset": 399
        },
        {
            "Score": 0.9949769377708435,
            "Type": "OTHER",
```

```
            "Text": "AnySpa@example.com",
            "BeginOffset": 403,
            "EndOffset": 418
        }
    ]
}
```

Per ulteriori informazioni, consulta Entità nella Guida per gli sviluppatori di Amazon Comprehend.

- Per i dettagli sull'API, consulta DetectEntities AWS CLICommand Reference.

`detect-key-phrases`

Il seguente esempio di codice mostra come utilizzare`detect-key-phrases`.

AWS CLI

Come rilevare le frasi chiave nel testo di input

L'esempio `detect-key-phrases` seguente analizza il testo di input e identifica le frasi nominali chiave. Per ogni previsione viene inoltre restituito il punteggio di attendibilità del modello pre-addestrato.

```
aws comprehend detect-key-phrases \
    --language-code en \
    --text "Hello Zhang Wei, I am John. Your AnyCompany Financial Services, LLC
 credit card \
       account 1111-XXXX-1111-XXXX has a minimum payment of $24.53 that is due by
 July 31st. Based on your autopay settings, \
       we will withdraw your payment on the due date from your bank account number
 XXXXXX1111 with the routing number XXXXX0000. \
       Customer feedback for Sunshine Spa, 123 Main St, Anywhere. Send comments to
 Alice at AnySpa@example.com."
```

Output:

```
{
    "KeyPhrases": [
        {
            "Score": 0.8996376395225525,
            "Text": "Zhang Wei",
            "BeginOffset": 6,
            "EndOffset": 15
```

```
        },
        {
            "Score": 0.9992469549179077,
            "Text": "John",
            "BeginOffset": 22,
            "EndOffset": 26
        },
        {
            "Score": 0.988385021686554,
            "Text": "Your AnyCompany Financial Services",
            "BeginOffset": 28,
            "EndOffset": 62
        },
        {
            "Score": 0.8740853071212769,
            "Text": "LLC credit card account 1111-XXXX-1111-XXXX",
            "BeginOffset": 64,
            "EndOffset": 107
        },
        {
            "Score": 0.9999437928199768,
            "Text": "a minimum payment",
            "BeginOffset": 112,
            "EndOffset": 129
        },
        {
            "Score": 0.9998900890350342,
            "Text": ".53",
            "BeginOffset": 133,
            "EndOffset": 136
        },
        {
            "Score": 0.9979453086853027,
            "Text": "July 31st",
            "BeginOffset": 152,
            "EndOffset": 161
        },
        {
            "Score": 0.9983011484146118,
            "Text": "your autopay settings",
            "BeginOffset": 172,
            "EndOffset": 193
        },
        {
```

```
            "Score": 0.9996572136878967,
            "Text": "your payment",
            "BeginOffset": 211,
            "EndOffset": 223
        },
        {
            "Score": 0.9995037317276001,
            "Text": "the due date",
            "BeginOffset": 227,
            "EndOffset": 239
        },
        {
            "Score": 0.9702621698379517,
            "Text": "your bank account number XXXXXX1111",
            "BeginOffset": 245,
            "EndOffset": 280
        },
        {
            "Score": 0.9179925918579102,
            "Text": "the routing number XXXXX0000.Customer feedback",
            "BeginOffset": 286,
            "EndOffset": 332
        },
        {
            "Score": 0.9978160858154297,
            "Text": "Sunshine Spa",
            "BeginOffset": 337,
            "EndOffset": 349
        },
        {
            "Score": 0.9706913232803345,
            "Text": "123 Main St",
            "BeginOffset": 351,
            "EndOffset": 362
        },
        {
            "Score": 0.9941995143890381,
            "Text": "comments",
            "BeginOffset": 379,
            "EndOffset": 387
        },
        {
            "Score": 0.9759287238121033,
            "Text": "Alice",
```

```
            "BeginOffset": 391,
            "EndOffset": 396
        },
        {
            "Score": 0.8376792669296265,
            "Text": "AnySpa@example.com",
            "BeginOffset": 400,
            "EndOffset": 415
        }
    ]
}
```

Per ulteriori informazioni, consulta Frasi chiave nella Guida per gli sviluppatori di Amazon Comprehend.

- Per i dettagli sull'API, consulta DetectKeyPhrases AWS CLICommand Reference.

`detect-pii-entities`

Il seguente esempio di codice mostra come utilizzare`detect-pii-entities`.

AWS CLI

Come rilevare entità PII nel testo di input

L'esempio `detect-pii-entities` seguente analizza il testo di input e identifica le entità che contengono informazioni di identificazione personale (PII). Per ogni previsione viene inoltre restituito il punteggio di attendibilità del modello pre-addestrato.

```
aws comprehend detect-pii-entities \
    --language-code en \
    --text "Hello Zhang Wei, I am John. Your AnyCompany Financial Services, LLC
 credit card \
        account 1111-XXXX-1111-XXXX has a minimum payment of $24.53 that is due by
 July 31st. Based on your autopay settings, \
        we will withdraw your payment on the due date from your bank account number
 XXXXXX1111 with the routing number XXXXX0000. \
        Customer feedback for Sunshine Spa, 123 Main St, Anywhere. Send comments to
 Alice at AnySpa@example.com."
```

Output:

```
{
```

```
    "Entities": [
        {
            "Score": 0.9998322129249573,
            "Type": "NAME",
            "BeginOffset": 6,
            "EndOffset": 15
        },
        {
            "Score": 0.9998878240585327,
            "Type": "NAME",
            "BeginOffset": 22,
            "EndOffset": 26
        },
        {
            "Score": 0.9994089603424072,
            "Type": "CREDIT_DEBIT_NUMBER",
            "BeginOffset": 88,
            "EndOffset": 107
        },
        {
            "Score": 0.9999760985374451,
            "Type": "DATE_TIME",
            "BeginOffset": 152,
            "EndOffset": 161
        },
        {
            "Score": 0.9999449253082275,
            "Type": "BANK_ACCOUNT_NUMBER",
            "BeginOffset": 271,
            "EndOffset": 281
        },
        {
            "Score": 0.9999847412109375,
            "Type": "BANK_ROUTING",
            "BeginOffset": 306,
            "EndOffset": 315
        },
        {
            "Score": 0.999925434589386,
            "Type": "ADDRESS",
            "BeginOffset": 354,
            "EndOffset": 365
        },
        {
```

```
            "Score": 0.9989161491394043,
            "Type": "NAME",
            "BeginOffset": 394,
            "EndOffset": 399
        },
        {
            "Score": 0.9994171857833862,
            "Type": "EMAIL",
            "BeginOffset": 403,
            "EndOffset": 418
        }
    ]
}
```

Per ulteriori informazioni, consulta Informazioni di identificazione personale (PII) nella Guida per gli sviluppatori di Amazon Comprehend.

- Per i dettagli sull'API, consulta DetectPiiEntities AWS CLICommand Reference.

`detect-sentiment`

Il seguente esempio di codice mostra come utilizzare`detect-sentiment`.

AWS CLI

Come rilevare il sentiment di un testo di input

L'esempio `detect-sentiment` seguente analizza il testo di input e restituisce un'inferenza del sentiment prevalente (`POSITIVE`, `NEUTRAL`, `MIXED` o `NEGATIVE`).

```
aws comprehend detect-sentiment \
    --language-code en \
    --text "It is a beautiful day in Seattle"
```

Output:

```
{
    "Sentiment": "POSITIVE",
    "SentimentScore": {
        "Positive": 0.9976957440376282,
        "Negative": 9.653854067437351e-05,
        "Neutral": 0.002169104292988777,
        "Mixed": 3.857641786453314e-05
```

```
    }
}
```

Per ulteriori informazioni, consulta Sentiment nella Guida per gli sviluppatori di Amazon Comprehend

- Per i dettagli sull'API, consulta DetectSentiment AWS CLICommand Reference.

`detect-syntax`

Il seguente esempio di codice mostra come utilizzare`detect-syntax`.

AWS CLI

Come rilevare le parti del discorso in un testo di input

L'esempio `detect-syntax` seguente analizza la sintassi del testo di input e restituisce le diverse parti del discorso. Per ogni previsione viene inoltre restituito il punteggio di attendibilità del modello pre-addestrato.

```
aws comprehend detect-syntax \
    --language-code en \
    --text "It is a beautiful day in Seattle."
```

Output:

```
{
    "SyntaxTokens": [
        {
            "TokenId": 1,
            "Text": "It",
            "BeginOffset": 0,
            "EndOffset": 2,
            "PartOfSpeech": {
                "Tag": "PRON",
                "Score": 0.9999740719795227
            }
        },
        {
            "TokenId": 2,
            "Text": "is",
            "BeginOffset": 3,
            "EndOffset": 5,
```

```
            "PartOfSpeech": {
                "Tag": "VERB",
                "Score": 0.999901294708252
            }
        },
        {
            "TokenId": 3,
            "Text": "a",
            "BeginOffset": 6,
            "EndOffset": 7,
            "PartOfSpeech": {
                "Tag": "DET",
                "Score": 0.9999938607215881
            }
        },
        {
            "TokenId": 4,
            "Text": "beautiful",
            "BeginOffset": 8,
            "EndOffset": 17,
            "PartOfSpeech": {
                "Tag": "ADJ",
                "Score": 0.9987351894378662
            }
        },
        {
            "TokenId": 5,
            "Text": "day",
            "BeginOffset": 18,
            "EndOffset": 21,
            "PartOfSpeech": {
                "Tag": "NOUN",
                "Score": 0.9999796748161316
            }
        },
        {
            "TokenId": 6,
            "Text": "in",
            "BeginOffset": 22,
            "EndOffset": 24,
            "PartOfSpeech": {
                "Tag": "ADP",
                "Score": 0.9998047947883606
            }
```

```
        },
        {
            "TokenId": 7,
            "Text": "Seattle",
            "BeginOffset": 25,
            "EndOffset": 32,
            "PartOfSpeech": {
                "Tag": "PROPN",
                "Score": 0.9940530061721802
            }
        }
    ]
}
```

Per ulteriori informazioni, consulta Analisi della sintassi nella Guida per gli sviluppatori di Amazon Comprehend.

- Per i dettagli sull'API, consulta DetectSyntax AWS CLICommand Reference.

`detect-targeted-sentiment`

Il seguente esempio di codice mostra come utilizzare`detect-targeted-sentiment`.

AWS CLI

Come rilevare il sentiment mirato delle entità denominate in un testo di input

L'esempio `detect-targeted-sentiment` seguente analizza il testo di input e restituisce le entità denominate insieme al sentiment mirato associato a ciascuna entità. Per ogni previsione viene inoltre restituito il punteggio di attendibilità del modello pre-addestrato.

```
aws comprehend detect-targeted-sentiment \
    --language-code en \
    --text "I do not enjoy January because it is too cold but August is the perfect
 temperature"
```

Output:

```
{
    "Entities": [
        {
            "DescriptiveMentionIndex": [
                0
```

```
            ],
            "Mentions": [
                {
                    "Score": 0.9999979734420776,
                    "GroupScore": 1.0,
                    "Text": "I",
                    "Type": "PERSON",
                    "MentionSentiment": {
                        "Sentiment": "NEUTRAL",
                        "SentimentScore": {
                            "Positive": 0.0,
                            "Negative": 0.0,
                            "Neutral": 1.0,
                            "Mixed": 0.0
                        }
                    },
                    "BeginOffset": 0,
                    "EndOffset": 1
                }
            ]
        },
        {
            "DescriptiveMentionIndex": [
                0
            ],
            "Mentions": [
                {
                    "Score": 0.9638869762420654,
                    "GroupScore": 1.0,
                    "Text": "January",
                    "Type": "DATE",
                    "MentionSentiment": {
                        "Sentiment": "NEGATIVE",
                        "SentimentScore": {
                            "Positive": 0.0031610000878572464,
                            "Negative": 0.9967250227928162,
                            "Neutral": 0.00011100000119768083,
                            "Mixed": 1.9999999949504854e-06
                        }
                    },
                    "BeginOffset": 15,
                    "EndOffset": 22
                }
            ]
```

```
        },
        {
            "DescriptiveMentionIndex": [
                0
            ],
            "Mentions": [
                {
                {
                    "Score": 0.9664419889450073,
                    "GroupScore": 1.0,
                    "Text": "August",
                    "Type": "DATE",
                    "MentionSentiment": {
                        "Sentiment": "POSITIVE",
                        "SentimentScore": {
                            "Positive": 0.9999549984931946,
                            "Negative": 3.999999989900971e-06,
                            "Neutral": 4.099999932805076e-05,
                            "Mixed": 0.0
                        }
                    },
                    "BeginOffset": 50,
                    "EndOffset": 56
                }
            ]
        },
        {
            "DescriptiveMentionIndex": [
                0
            ],
            "Mentions": [
                {
                    "Score": 0.9803199768066406,
                    "GroupScore": 1.0,
                    "Text": "temperature",
                    "Type": "ATTRIBUTE",
                    "MentionSentiment": {
                        "Sentiment": "POSITIVE",
                        "SentimentScore": {
                            "Positive": 1.0,
                            "Negative": 0.0,
                            "Neutral": 0.0,
                            "Mixed": 0.0
                        }
```

```
                    },
                    "BeginOffset": 77,
                    "EndOffset": 88
                }
            ]
        }
    ]
}
```

Per ulteriori informazioni, consulta Sentiment mirato nella Guida per gli sviluppatori di Amazon Comprehend.

- Per i dettagli sull'API, consulta DetectTargetedSentiment AWS CLICommand Reference.

`import-model`

Il seguente esempio di codice mostra come utilizzare`import-model`.

AWS CLI

Come importare un modello

L'`import-model`esempio seguente importa un modello da un AWS account diverso. Il modello di classificazione dei documenti nell'account `444455556666` dispone di una policy basata su risorse che consente all'account `111122223333` di importare il modello.

```
aws comprehend import-model \
    --source-model-arn arn:aws:comprehend:us-west-2:444455556666:document-classifier/example-classifier
```

Output:

```
{
    "ModelArn": "arn:aws:comprehend:us-west-2:111122223333:document-classifier/example-classifier"
}
```

Per ulteriori informazioni, consulta Copiare modelli personalizzati tra AWS account nella Amazon Comprehend Developer Guide.

- Per i dettagli sull'API, consulta AWS CLI Command ImportModelReference.

`list-datasets`

Il seguente esempio di codice mostra come utilizzare`list-datasets`.

AWS CLI

Come elencare tutti i set di dati del flywheel

L'esempio `list-datasets` seguente elenca tutti i set di dati associati a un flywheel.

```
aws comprehend list-datasets \
    --flywheel-arn arn:aws:comprehend:us-west-2:111122223333:flywheel/flywheel-
entity
```

Output:

```
{
    "DatasetPropertiesList": [
        {
            "DatasetArn": "arn:aws:comprehend:us-west-2:111122223333:flywheel/
flywheel-entity/dataset/example-dataset-1",
            "DatasetName": "example-dataset-1",
            "DatasetType": "TRAIN",
            "DatasetS3Uri": "s3://amzn-s3-demo-bucket/flywheel-entity/
schemaVersion=1/20230616T200543Z/datasets/example-dataset-1/20230616T203710Z/",
            "Status": "CREATING",
            "CreationTime": "2023-06-16T20:37:10.400000+00:00"
        },
        {
            "DatasetArn": "arn:aws:comprehend:us-west-2:111122223333:flywheel/
flywheel-entity/dataset/example-dataset-2",
            "DatasetName": "example-dataset-2",
            "DatasetType": "TRAIN",
            "DatasetS3Uri": "s3://amzn-s3-demo-bucket/flywheel-entity/
schemaVersion=1/20230616T200543Z/datasets/example-dataset-2/20230616T200607Z/",
            "Description": "TRAIN Dataset created by Flywheel creation.",
            "Status": "COMPLETED",
            "NumberOfDocuments": 5572,
            "CreationTime": "2023-06-16T20:06:07.722000+00:00"
        }
    ]
}
```

Per ulteriori informazioni, consulta Panoramica del volano dei dati nella Guida per gli sviluppatori di Amazon Comprehend.

- Per i dettagli sull'API, consulta ListDatasets AWS CLICommand Reference.

`list-document-classification-jobs`

Il seguente esempio di codice mostra come utilizzare`list-document-classification-jobs`.

AWS CLI

Come un elenco di tutti i processi di classificazione dei documenti

L'esempio `list-document-classification-jobs` seguente elenca tutti i processi di classificazione dei documenti.

```
aws comprehend list-document-classification-jobs
```

Output:

```
{
    "DocumentClassificationJobPropertiesList": [
        {
            "JobId": "123456abcdeb0e11022f22a11EXAMPLE",
            "JobArn": "arn:aws:comprehend:us-west-2:1234567890101:document-
classification-job/123456abcdeb0e11022f22a11EXAMPLE",
            "JobName": "exampleclassificationjob",
            "JobStatus": "COMPLETED",
            "SubmitTime": "2023-06-14T17:09:51.788000+00:00",
            "EndTime": "2023-06-14T17:15:58.582000+00:00",
            "DocumentClassifierArn": "arn:aws:comprehend:us-
west-2:1234567890101:document-classifier/mymodel/version/12",
            "InputDataConfig": {
                "S3Uri": "s3://amzn-s3-demo-bucket/jobdata/",
                "InputFormat": "ONE_DOC_PER_LINE"
            },
            "OutputDataConfig": {
                "S3Uri": "s3://amzn-s3-demo-destination-bucket/
thefolder/1234567890101-CLN-e758dd56b824aa717ceab551f11749fb/output/output.tar.gz"
            },
            "DataAccessRoleArn": "arn:aws:iam::1234567890101:role/service-role/
AmazonComprehendServiceRole-example-role"
        },
```

```
        {
            "JobId": "123456abcdeb0e11022f22a1EXAMPLE2",
            "JobArn": "arn:aws:comprehend:us-west-2:1234567890101:document-
classification-job/123456abcdeb0e11022f22a1EXAMPLE2",
            "JobName": "exampleclassificationjob2",
            "JobStatus": "COMPLETED",
            "SubmitTime": "2023-06-14T17:22:39.829000+00:00",
            "EndTime": "2023-06-14T17:28:46.107000+00:00",
            "DocumentClassifierArn": "arn:aws:comprehend:us-
west-2:1234567890101:document-classifier/mymodel/version/12",
            "InputDataConfig": {
                "S3Uri": "s3://amzn-s3-demo-bucket/jobdata/",
                "InputFormat": "ONE_DOC_PER_LINE"
            },
            "OutputDataConfig": {
                "S3Uri": "s3://amzn-s3-demo-destination-bucket/
thefolder/1234567890101-CLN-123456abcdeb0e11022f22a1EXAMPLE2/output/output.tar.gz"
            },
            "DataAccessRoleArn": "arn:aws:iam::1234567890101:role/service-role/
AmazonComprehendServiceRole-example-role"
        }
    ]
}
```

Per ulteriori informazioni, consulta Classificazione personalizzata nella Guida per gli sviluppatori di Amazon Comprehend.

- Per i dettagli sull'API, consulta ListDocumentClassificationJobs AWS CLICommand Reference.

`list-document-classifier-summaries`

Il seguente esempio di codice mostra come utilizzare`list-document-classifier-summaries`.

AWS CLI

Come elencare i riepiloghi di tutti i classificatori di documenti creati

L'esempio `list-document-classifier-summaries` seguente crea tutti i riepiloghi del classificatore di documenti.

```
aws comprehend list-document-classifier-summaries
```

Output:

```
{
    "DocumentClassifierSummariesList": [
        {
            "DocumentClassifierName": "example-classifier-1",
            "NumberOfVersions": 1,
            "LatestVersionCreatedAt": "2023-06-13T22:07:59.825000+00:00",
            "LatestVersionName": "1",
            "LatestVersionStatus": "TRAINED"
        },
        {
            "DocumentClassifierName": "example-classifier-2",
            "NumberOfVersions": 2,
            "LatestVersionCreatedAt": "2023-06-13T21:54:59.589000+00:00",
            "LatestVersionName": "2",
            "LatestVersionStatus": "TRAINED"
        }
    ]
}
```

Per ulteriori informazioni, consulta Creazione e gestione di modelli personalizzati nella Guida per gli sviluppatori di Amazon Comprehend.

- Per i dettagli sull'API, consulta ListDocumentClassifierSummaries AWS CLICommand Reference.

`list-document-classifiers`

Il seguente esempio di codice mostra come utilizzare`list-document-classifiers`.

AWS CLI

Come un elenco di tutti i classificatori dei documenti

L'esempio `list-document-classifiers` seguente elenca tutti i modelli di classificatori di documenti addestrati e in corso di addestramento.

```
aws comprehend list-document-classifiers
```

Output:

```
{
    "DocumentClassifierPropertiesList": [
```

```
        {
            "DocumentClassifierArn": "arn:aws:comprehend:us-
west-2:111122223333:document-classifier/exampleclassifier1",
            "LanguageCode": "en",
            "Status": "TRAINED",
            "SubmitTime": "2023-06-13T19:04:15.735000+00:00",
            "EndTime": "2023-06-13T19:42:31.752000+00:00",
            "TrainingStartTime": "2023-06-13T19:08:20.114000+00:00",
            "TrainingEndTime": "2023-06-13T19:41:35.080000+00:00",
            "InputDataConfig": {
                "DataFormat": "COMPREHEND_CSV",
                "S3Uri": "s3://amzn-s3-demo-bucket/trainingdata"
            },
            "OutputDataConfig": {},
            "ClassifierMetadata": {
                "NumberOfLabels": 3,
                "NumberOfTrainedDocuments": 5016,
                "NumberOfTestDocuments": 557,
                "EvaluationMetrics": {
                    "Accuracy": 0.9856,
                    "Precision": 0.9919,
                    "Recall": 0.9459,
                    "F1Score": 0.9673,
                    "MicroPrecision": 0.9856,
                    "MicroRecall": 0.9856,
                    "MicroF1Score": 0.9856,
                    "HammingLoss": 0.0144
                }
            },
            "DataAccessRoleArn": "arn:aws:iam::111122223333:role/service-role/
AmazonComprehendServiceRole-testorle",
            "Mode": "MULTI_CLASS"
        },
        {
            "DocumentClassifierArn": "arn:aws:comprehend:us-
west-2:111122223333:document-classifier/exampleclassifier2",
            "LanguageCode": "en",
            "Status": "TRAINING",
            "SubmitTime": "2023-06-13T21:20:28.690000+00:00",
            "InputDataConfig": {
                "DataFormat": "COMPREHEND_CSV",
                "S3Uri": "s3://amzn-s3-demo-bucket/trainingdata"
            },
            "OutputDataConfig": {},
```

```
            "DataAccessRoleArn": "arn:aws:iam::111122223333:role/service-role/
AmazonComprehendServiceRole-testorle",
            "Mode": "MULTI_CLASS"
        }
    ]
}
```

Per ulteriori informazioni, consulta Creazione e gestione di modelli personalizzati nella Guida per gli sviluppatori di Amazon Comprehend.

- Per i dettagli sull'API, consulta ListDocumentClassifiers AWS CLICommand Reference.

`list-dominant-language-detection-jobs`

Il seguente esempio di codice mostra come utilizzare`list-dominant-language-detection-jobs`.

AWS CLI

Come elencare tutti i processi di rilevamento della lingua dominante

L'esempio `list-dominant-language-detection-jobs` seguente elenca tutti i processi di rilevamento asincrono, in corso e completati, della lingua dominante.

```
aws comprehend list-dominant-language-detection-jobs
```

Output:

```
{
    "DominantLanguageDetectionJobPropertiesList": [
        {
            "JobId": "123456abcdeb0e11022f22a11EXAMPLE",
            "JobArn": "arn:aws:comprehend:us-west-2:111122223333:dominant-language-
detection-job/123456abcdeb0e11022f22a11EXAMPLE",
            "JobName": "languageanalysis1",
            "JobStatus": "COMPLETED",
            "SubmitTime": "2023-06-09T18:10:38.037000+00:00",
            "EndTime": "2023-06-09T18:18:45.498000+00:00",
            "InputDataConfig": {
                "S3Uri": "s3://amzn-s3-demo-bucket",
                "InputFormat": "ONE_DOC_PER_LINE"
            },
            "OutputDataConfig": {
```

```
                "S3Uri": "s3://amzn-s3-demo-destination-bucket/
testfolder/111122223333-LANGUAGE-123456abcdeb0e11022f22a11EXAMPLE/output/
output.tar.gz"
            },
            "DataAccessRoleArn": "arn:aws:iam::111122223333:role/service-role/
AmazonComprehendServiceRole-example-role"
        },
        {
            "JobId": "123456abcdeb0e11022f22a11EXAMPLE",
            "JobArn": "arn:aws:comprehend:us-west-2:111122223333:dominant-language-
detection-job/123456abcdeb0e11022f22a11EXAMPLE",
            "JobName": "languageanalysis2",
            "JobStatus": "STOPPED",
            "SubmitTime": "2023-06-09T18:16:33.690000+00:00",
            "EndTime": "2023-06-09T18:24:40.608000+00:00",
            "InputDataConfig": {
                "S3Uri": "s3://amzn-s3-demo-bucket",
                "InputFormat": "ONE_DOC_PER_LINE"
            },
            "OutputDataConfig": {
                "S3Uri": "s3://amzn-s3-demo-destination-bucket/
testfolder/111122223333-LANGUAGE-123456abcdeb0e11022f22a11EXAMPLE/output/
output.tar.gz"
            },
            "DataAccessRoleArn": "arn:aws:iam::111122223333:role/service-role/
AmazonComprehendServiceRole-example-role"
        }
    ]
}
```

Per ulteriori informazioni, consulta Analisi asincrona degli approfondimenti di Amazon Comprehend nella Guida per gli sviluppatori di Amazon Comprehend.

- Per i dettagli sull'API, consulta ListDominantLanguageDetectionJobs AWS CLICommand Reference.

`list-endpoints`

Il seguente esempio di codice mostra come utilizzare`list-endpoints`.

AWS CLI

Come elencare tutti gli endpoint

L'esempio `list-endpoints` seguente elenca tutti gli endpoint attivi specifici del modello.

```
aws comprehend list-endpoints
```

Output:

```
{
    "EndpointPropertiesList": [
        {
            "EndpointArn": "arn:aws:comprehend:us-west-2:111122223333:document-
classifier-endpoint/ExampleClassifierEndpoint",
            "Status": "IN_SERVICE",
            "ModelArn": "arn:aws:comprehend:us-west-2:111122223333:document-
classifier/exampleclassifier1",
            "DesiredModelArn": "arn:aws:comprehend:us-west-2:111122223333:document-
classifier/exampleclassifier1",
            "DesiredInferenceUnits": 1,
            "CurrentInferenceUnits": 1,
            "CreationTime": "2023-06-13T20:32:54.526000+00:00",
            "LastModifiedTime": "2023-06-13T20:32:54.526000+00:00"
        },
        {
            "EndpointArn": "arn:aws:comprehend:us-west-2:111122223333:document-
classifier-endpoint/ExampleClassifierEndpoint2",
            "Status": "IN_SERVICE",
            "ModelArn": "arn:aws:comprehend:us-west-2:111122223333:document-
classifier/exampleclassifier2",
            "DesiredModelArn": "arn:aws:comprehend:us-west-2:111122223333:document-
classifier/exampleclassifier2",
            "DesiredInferenceUnits": 1,
            "CurrentInferenceUnits": 1,
            "CreationTime": "2023-06-13T20:32:54.526000+00:00",
            "LastModifiedTime": "2023-06-13T20:32:54.526000+00:00"
        }
    ]
}
```

Per ulteriori informazioni, consulta Gestione degli endpoint di Amazon Comprehend nella Guida per gli sviluppatori di Amazon Comprehend.

- Per i dettagli sull'API, consulta ListEndpoints AWS CLICommand Reference.

`list-entities-detection-jobs`

Il seguente esempio di codice mostra come utilizzare`list-entities-detection-jobs`.

AWS CLI

Come elencare tutti i processi di rilevamento delle entità

L'esempio `list-entities-detection-jobs` seguente elenca tutti i processi di rilevamento delle entità asincroni.

```
aws comprehend list-entities-detection-jobs
```

Output:

```
{
    "EntitiesDetectionJobPropertiesList": [
        {
            "JobId": "468af39c28ab45b83eb0c4ab9EXAMPLE",
            "JobArn": "arn:aws:comprehend:us-west-2:111122223333:entities-detection-
job/468af39c28ab45b83eb0c4ab9EXAMPLE",
            "JobName": "example-entities-detection",
            "JobStatus": "COMPLETED",
            "SubmitTime": "2023-06-08T20:57:46.476000+00:00",
            "EndTime": "2023-06-08T21:05:53.718000+00:00",
            "InputDataConfig": {
                "S3Uri": "s3://amzn-s3-demo-bucket/AsyncBatchJobs/",
                "InputFormat": "ONE_DOC_PER_LINE"
            },
            "OutputDataConfig": {
                "S3Uri": "s3://amzn-s3-demo-destination-bucket/
thefolder/111122223333-NER-468af39c28ab45b83eb0c4ab9EXAMPLE/output/output.tar.gz"
            },
            "LanguageCode": "en",
            "DataAccessRoleArn": "arn:aws:iam::111122223333:role/service-role/
AmazonComprehendServiceRole-example-role"
        },
        {
            "JobId": "809691caeaab0e71406f80a28EXAMPLE",
            "JobArn": "arn:aws:comprehend:us-west-2:111122223333:entities-detection-
job/809691caeaab0e71406f80a28EXAMPLE",
            "JobName": "example-entities-detection-2",
            "JobStatus": "COMPLETED",
```

```
            "SubmitTime": "2023-06-08T21:30:15.323000+00:00",
            "EndTime": "2023-06-08T21:40:23.509000+00:00",
            "InputDataConfig": {
                "S3Uri": "s3://amzn-s3-demo-bucket/AsyncBatchJobs/",
                "InputFormat": "ONE_DOC_PER_LINE"
            },
            "OutputDataConfig": {
                "S3Uri": "s3://amzn-s3-demo-destination-bucket/
thefolder/111122223333-NER-809691caeaab0e71406f80a28EXAMPLE/output/output.tar.gz"
            },
            "LanguageCode": "en",
            "DataAccessRoleArn": "arn:aws:iam::111122223333:role/service-role/
AmazonComprehendServiceRole-example-role"
        },
        {
            "JobId": "e00597c36b448b91d70dea165EXAMPLE",
            "JobArn": "arn:aws:comprehend:us-west-2:111122223333:entities-detection-
job/e00597c36b448b91d70dea165EXAMPLE",
            "JobName": "example-entities-detection-3",
            "JobStatus": "STOPPED",
            "SubmitTime": "2023-06-08T22:19:28.528000+00:00",
            "EndTime": "2023-06-08T22:27:33.991000+00:00",
            "InputDataConfig": {
                "S3Uri": "s3://amzn-s3-demo-bucket/AsyncBatchJobs/",
                "InputFormat": "ONE_DOC_PER_LINE"
            },
            "OutputDataConfig": {
                "S3Uri": "s3://amzn-s3-demo-destination-bucket/
thefolder/111122223333-NER-e00597c36b448b91d70dea165EXAMPLE/output/output.tar.gz"
            },
            "LanguageCode": "en",
            "DataAccessRoleArn": "arn:aws:iam::111122223333:role/service-role/
AmazonComprehendServiceRole-example-role"
        }
    ]
}
```

Per ulteriori informazioni, consulta Entità nella Guida per gli sviluppatori di Amazon Comprehend.

- Per i dettagli sull'API, consulta ListEntitiesDetectionJobs AWS CLICommand Reference.

`list-entity-recognizer-summaries`

Il seguente esempio di codice mostra come utilizzare`list-entity-recognizer-summaries`.

AWS CLI

Come elencare i riepiloghi di tutti i riconoscitori di entità creati

L'esempio `list-entity-recognizer-summaries` seguente elenca tutti i riepiloghi del riconoscitore delle entità.

```
aws comprehend list-entity-recognizer-summaries
```

Output:

```
{
    "EntityRecognizerSummariesList": [
        {
            "RecognizerName": "entity-recognizer-3",
            "NumberOfVersions": 2,
            "LatestVersionCreatedAt": "2023-06-15T23:15:07.621000+00:00",
            "LatestVersionName": "2",
            "LatestVersionStatus": "STOP_REQUESTED"
        },
        {
            "RecognizerName": "entity-recognizer-2",
            "NumberOfVersions": 1,
            "LatestVersionCreatedAt": "2023-06-14T22:55:27.805000+00:00",
            "LatestVersionName": "2"
            "LatestVersionStatus": "TRAINED"
        },
        {
            "RecognizerName": "entity-recognizer-1",
            "NumberOfVersions": 1,
            "LatestVersionCreatedAt": "2023-06-14T20:44:59.631000+00:00",
            "LatestVersionName": "1",
            "LatestVersionStatus": "TRAINED"
        }
    ]
}
```

Per ulteriori informazioni, consulta Riconoscimento delle entità personalizzate nella Guida per gli sviluppatori di Amazon Comprehend.

- Per i dettagli sull'API, consulta ListEntityRecognizerSummaries AWS CLICommand Reference.

`list-entity-recognizers`

Il seguente esempio di codice mostra come utilizzare`list-entity-recognizers`.

AWS CLI

Come elencare tutti i riconoscitori di entità personalizzati

L'esempio `list-entity-recognizers` seguente elenca tutti i riconoscitori delle entità personalizzati creati.

```
aws comprehend list-entity-recognizers
```

Output:

```
{
    "EntityRecognizerPropertiesList": [
        {
            "EntityRecognizerArn": "arn:aws:comprehend:us-
west-2:111122223333:entity-recognizer/EntityRecognizer/version/1",
            "LanguageCode": "en",
            "Status": "TRAINED",
            "SubmitTime": "2023-06-14T20:44:59.631000+00:00",
            "EndTime": "2023-06-14T20:59:19.532000+00:00",
            "TrainingStartTime": "2023-06-14T20:48:52.811000+00:00",
            "TrainingEndTime": "2023-06-14T20:58:11.473000+00:00",
            "InputDataConfig": {
                "DataFormat": "COMPREHEND_CSV",
                "EntityTypes": [
                    {
                        "Type": "BUSINESS"
                    }
                ],
                "Documents": {
                    "S3Uri": "s3://amzn-s3-demo-bucket/trainingdata/dataset/",
                    "InputFormat": "ONE_DOC_PER_LINE"
                },
                "EntityList": {
                    "S3Uri": "s3://amzn-s3-demo-bucket/trainingdata/entity.csv"
                }
            },
```

```
            "RecognizerMetadata": {
                "NumberOfTrainedDocuments": 1814,
                "NumberOfTestDocuments": 486,
                "EvaluationMetrics": {
                    "Precision": 100.0,
                    "Recall": 100.0,
                    "F1Score": 100.0
                },
                "EntityTypes": [
                    {
                        "Type": "BUSINESS",
                        "EvaluationMetrics": {
                            "Precision": 100.0,
                            "Recall": 100.0,
                            "F1Score": 100.0
                        },
                        "NumberOfTrainMentions": 1520
                    }
                ]
            },
            "DataAccessRoleArn": "arn:aws:iam::111122223333:role/service-role/
AmazonComprehendServiceRole-servicerole",
            "VersionName": "1"
        },
        {
            "EntityRecognizerArn": "arn:aws:comprehend:us-
west-2:111122223333:entity-recognizer/entityrecognizer3",
            "LanguageCode": "en",
            "Status": "TRAINED",
            "SubmitTime": "2023-06-14T22:57:51.056000+00:00",
            "EndTime": "2023-06-14T23:14:13.894000+00:00",
            "TrainingStartTime": "2023-06-14T23:01:33.984000+00:00",
            "TrainingEndTime": "2023-06-14T23:13:02.984000+00:00",
            "InputDataConfig": {
                "DataFormat": "COMPREHEND_CSV",
                "EntityTypes": [
                    {
                        "Type": "DEVICE"
                    }
                ],
                "Documents": {
                    "S3Uri": "s3://amzn-s3-demo-bucket/trainingdata/raw_txt.csv",
                    "InputFormat": "ONE_DOC_PER_LINE"
                },
```

```
                "EntityList": {
                    "S3Uri": "s3://amzn-s3-demo-bucket/trainingdata/entity_list.csv"
                }
            },
            "RecognizerMetadata": {
                "NumberOfTrainedDocuments": 4616,
                "NumberOfTestDocuments": 3489,
                "EvaluationMetrics": {
                    "Precision": 98.54227405247813,
                    "Recall": 100.0,
                    "F1Score": 99.26578560939794
                },
                "EntityTypes": [
                    {
                        "Type": "DEVICE",
                        "EvaluationMetrics": {
                            "Precision": 98.54227405247813,
                            "Recall": 100.0,
                            "F1Score": 99.26578560939794
                        },
                        "NumberOfTrainMentions": 2764
                    }
                ]
            },
            "DataAccessRoleArn": "arn:aws:iam::111122223333:role/service-role/
AmazonComprehendServiceRole-servicerole"
        }
    ]
}
```

Per ulteriori informazioni, consulta Riconoscimento delle entità personalizzate nella Guida per gli sviluppatori di Amazon Comprehend.

- Per i dettagli sull'API, consulta ListEntityRecognizers AWS CLICommand Reference.

`list-events-detection-jobs`

Il seguente esempio di codice mostra come utilizzare`list-events-detection-jobs`.

AWS CLI

Come elencare tutti i processi di rilevamento degli eventi

L'esempio `list-events-detection-jobs` seguente elenca tutti i processi di rilevamento degli eventi asincroni.

```
aws comprehend list-events-detection-jobs
```

Output:

```
{
    "EventsDetectionJobPropertiesList": [
        {
            "JobId": "aa9593f9203e84f3ef032ce18EXAMPLE",
            "JobArn": "arn:aws:comprehend:us-west-2:1111222233333:events-detection-
job/aa9593f9203e84f3ef032ce18EXAMPLE",
            "JobName": "events_job_1",
            "JobStatus": "COMPLETED",
            "SubmitTime": "2023-06-12T19:14:57.751000+00:00",
            "EndTime": "2023-06-12T19:21:04.962000+00:00",
            "InputDataConfig": {
                "S3Uri": "s3://amzn-s3-demo-source-bucket/EventsData/",
                "InputFormat": "ONE_DOC_PER_LINE"
            },
            "OutputDataConfig": {
                "S3Uri": "s3://amzn-s3-demo-destination-bucket/
testfolder/1111222233333-EVENTS-aa9593f9203e84f3ef032ce18EXAMPLE/output/"
            },
            "LanguageCode": "en",
            "DataAccessRoleArn": "arn:aws:iam::1111222233333:role/service-role/
AmazonComprehendServiceRole-example-role",
            "TargetEventTypes": [
                "BANKRUPTCY",
                "EMPLOYMENT",
                "CORPORATE_ACQUISITION",
                "CORPORATE_MERGER",
                "INVESTMENT_GENERAL"
            ]
        },
        {
            "JobId": "4a990a2f7e82adfca6e171135EXAMPLE",
            "JobArn": "arn:aws:comprehend:us-west-2:1111222233333:events-detection-
job/4a990a2f7e82adfca6e171135EXAMPLE",
            "JobName": "events_job_2",
            "JobStatus": "COMPLETED",
            "SubmitTime": "2023-06-12T19:55:43.702000+00:00",
```

```
            "EndTime": "2023-06-12T20:03:49.893000+00:00",
            "InputDataConfig": {
                "S3Uri": "s3://amzn-s3-demo-source-bucket/EventsData/",
                "InputFormat": "ONE_DOC_PER_LINE"
            },
            "OutputDataConfig": {
                "S3Uri": "s3://amzn-s3-demo-destination-bucket/
testfolder/1111222233333-EVENTS-4a990a2f7e82adfca6e171135EXAMPLE/output/"
            },
            "LanguageCode": "en",
            "DataAccessRoleArn": "arn:aws:iam::1111222233333:role/service-role/
AmazonComprehendServiceRole-example-role",
            "TargetEventTypes": [
                "BANKRUPTCY",
                "EMPLOYMENT",
                "CORPORATE_ACQUISITION",
                "CORPORATE_MERGER",
                "INVESTMENT_GENERAL"
            ]
        }
    ]
}
```

Per ulteriori informazioni, consulta Analisi asincrona degli approfondimenti di Amazon Comprehend nella Guida per gli sviluppatori di Amazon Comprehend.

- Per i dettagli sull'API, consulta ListEventsDetectionJobs AWS CLICommand Reference.

`list-flywheel-iteration-history`

Il seguente esempio di codice mostra come utilizzare`list-flywheel-iteration-history`.

AWS CLI

Come elencare tutta la cronologia delle iterazioni del flywheel

L'esempio `list-flywheel-iteration-history` seguente elenca tutte le iterazioni di un flywheel.

```
aws comprehend list-flywheel-iteration-history
    --flywheel-arn arn:aws:comprehend:us-west-2:111122223333:flywheel/example-
flywheel
```

Output:


```
{
    "FlywheelIterationPropertiesList": [
        {
            "FlywheelArn": "arn:aws:comprehend:us-west-2:111122223333:flywheel/
example-flywheel",
            "FlywheelIterationId": "20230619TEXAMPLE",
            "CreationTime": "2023-06-19T04:00:32.594000+00:00",
            "EndTime": "2023-06-19T04:00:49.248000+00:00",
            "Status": "COMPLETED",
            "Message": "FULL_ITERATION: Flywheel iteration performed all functions
 successfully.",
            "EvaluatedModelArn": "arn:aws:comprehend:us-
west-2:111122223333:document-classifier/example-classifier/version/1",
            "EvaluatedModelMetrics": {
                "AverageF1Score": 0.7742663922375772,
                "AverageF1Score": 0.9876464664646313,
                "AveragePrecision": 0.9800000253081214,
                "AverageRecall": 0.9445600253081214,
                "AverageAccuracy": 0.9997281665190434
            },
            "EvaluationManifestS3Prefix": "s3://amzn-s3-demo-bucket/example-
flywheel/schemaVersion=1/20230619TEXAMPLE/evaluation/20230619TEXAMPLE/"
        },
        {
            "FlywheelArn": "arn:aws:comprehend:us-west-2:111122223333:flywheel/
example-flywheel-2",
            "FlywheelIterationId": "20230616TEXAMPLE",
            "CreationTime": "2023-06-16T21:10:26.385000+00:00",
            "EndTime": "2023-06-16T23:33:16.827000+00:00",
            "Status": "COMPLETED",
            "Message": "FULL_ITERATION: Flywheel iteration performed all functions
 successfully.",
            "EvaluatedModelArn": "arn:aws:comprehend:us-
west-2:111122223333:document-classifier/spamvshamclassify/version/1",
            "EvaluatedModelMetrics": {
                "AverageF1Score": 0.7742663922375772,
                "AverageF1Score": 0.9767700253081214,
                "AveragePrecision": 0.9767700253081214,
                "AverageRecall": 0.9767700253081214,
                "AverageAccuracy": 0.9858281665190434
            },
```

```
            "EvaluationManifestS3Prefix": "s3://amzn-s3-demo-bucket/example-
flywheel-2/schemaVersion=1/20230616TEXAMPLE/evaluation/20230616TEXAMPLE/"
        }
    ]
}
```

Per ulteriori informazioni, consulta Panoramica del volano dei dati nella Guida per gli sviluppatori di Amazon Comprehend.

- Per i dettagli sull'API, consulta ListFlywheelIterationHistory AWS CLICommand Reference.

`list-flywheels`

Il seguente esempio di codice mostra come utilizzare`list-flywheels`.

AWS CLI

Come elencare tutti i flywheel

L'esempio `list-flywheels` seguente elenca tutti i flywheel creati.

```
aws comprehend list-flywheels
```

Output:

```
{
    "FlywheelSummaryList": [
        {
            "FlywheelArn": "arn:aws:comprehend:us-west-2:111122223333:flywheel/
example-flywheel-1",
            "ActiveModelArn": "arn:aws:comprehend:us-west-2:111122223333:document-
classifier/exampleclassifier/version/1",
            "DataLakeS3Uri": "s3://amzn-s3-demo-bucket/example-flywheel-1/
schemaVersion=1/20230616T200543Z/",
            "Status": "ACTIVE",
            "ModelType": "DOCUMENT_CLASSIFIER",
            "CreationTime": "2023-06-16T20:05:43.242000+00:00",
            "LastModifiedTime": "2023-06-19T04:00:43.027000+00:00",
            "LatestFlywheelIteration": "20230619T040032Z"
        },
        {
            "FlywheelArn": "arn:aws:comprehend:us-west-2:111122223333:flywheel/
example-flywheel-2",
```

```
            "ActiveModelArn": "arn:aws:comprehend:us-west-2:111122223333:document-
classifier/exampleclassifier2/version/1",
            "DataLakeS3Uri": "s3://amzn-s3-demo-bucket/example-flywheel-2/
schemaVersion=1/20220616T200543Z/",
            "Status": "ACTIVE",
            "ModelType": "DOCUMENT_CLASSIFIER",
            "CreationTime": "2022-06-16T20:05:43.242000+00:00",
            "LastModifiedTime": "2022-06-19T04:00:43.027000+00:00",
            "LatestFlywheelIteration": "20220619T040032Z"
        }
    ]
}
```

Per ulteriori informazioni, consulta Panoramica del volano dei dati nella Guida per gli sviluppatori di Amazon Comprehend.

- Per i dettagli sull'API, consulta ListFlywheels AWS CLICommand Reference.

`list-key-phrases-detection-jobs`

Il seguente esempio di codice mostra come utilizzare`list-key-phrases-detection-jobs`.

AWS CLI

Come elencare tutti processi di rilevamento delle frasi chiave

L'esempio `list-key-phrases-detection-jobs` seguente elenca tutti i processi di rilevamento asincrono, in corso e completati, delle frasi chiave.

```
aws comprehend list-key-phrases-detection-jobs
```

Output:

```
{
    "KeyPhrasesDetectionJobPropertiesList": [
        {
            "JobId": "123456abcdeb0e11022f22a11EXAMPLE",
            "JobArn": "arn:aws:comprehend:us-west-2:111122223333:key-phrases-
detection-job/123456abcdeb0e11022f22a11EXAMPLE",
            "JobName": "keyphrasesanalysis1",
            "JobStatus": "COMPLETED",
            "SubmitTime": "2023-06-08T22:31:43.767000+00:00",
            "EndTime": "2023-06-08T22:39:52.565000+00:00",
```

```
            "InputDataConfig": {
                "S3Uri": "s3://amzn-s3-demo-source-bucket/AsyncBatchJobs/",
                "InputFormat": "ONE_DOC_PER_LINE"
            },
            "OutputDataConfig": {
                "S3Uri": "s3://amzn-s3-demo-destination-bucket/
testfolder/111122223333-KP-123456abcdeb0e11022f22a11EXAMPLE/output/output.tar.gz"
            },
            "LanguageCode": "en",
            "DataAccessRoleArn": "arn:aws:iam::111122223333:role/service-role/
AmazonComprehendServiceRole-example-role"
        },
        {
            "JobId": "123456abcdeb0e11022f22a33EXAMPLE",
            "JobArn": "arn:aws:comprehend:us-west-2:111122223333:key-phrases-
detection-job/123456abcdeb0e11022f22a33EXAMPLE",
            "JobName": "keyphrasesanalysis2",
            "JobStatus": "STOPPED",
            "SubmitTime": "2023-06-08T22:57:52.154000+00:00",
            "EndTime": "2023-06-08T23:05:48.385000+00:00",
            "InputDataConfig": {
                "S3Uri": "s3://amzn-s3-demo-bucket/AsyncBatchJobs/",
                "InputFormat": "ONE_DOC_PER_LINE"
            },
            "OutputDataConfig": {
                "S3Uri": "s3://amzn-s3-demo-destination-bucket/
testfolder/111122223333-KP-123456abcdeb0e11022f22a33EXAMPLE/output/output.tar.gz"
            },
            "LanguageCode": "en",
            "DataAccessRoleArn": "arn:aws:iam::111122223333:role/service-role/
AmazonComprehendServiceRole-example-role"
        },
        {
            "JobId": "123456abcdeb0e11022f22a44EXAMPLE",
            "JobArn": "arn:aws:comprehend:us-west-2:111122223333:key-phrases-
detection-job/123456abcdeb0e11022f22a44EXAMPLE",
            "JobName": "keyphrasesanalysis3",
            "JobStatus": "FAILED",
            "Message": "NO_READ_ACCESS_TO_INPUT: The provided data access role does
 not have proper access to the input data.",
            "SubmitTime": "2023-06-09T16:47:04.029000+00:00",
            "EndTime": "2023-06-09T16:47:18.413000+00:00",
            "InputDataConfig": {
                "S3Uri": "s3://amzn-s3-demo-bucket",
```

```
                "InputFormat": "ONE_DOC_PER_LINE"
            },
            "OutputDataConfig": {
                "S3Uri": "s3://amzn-s3-demo-destination-bucket/
testfolder/111122223333-KP-123456abcdeb0e11022f22a44EXAMPLE/output/output.tar.gz"
            },
            "LanguageCode": "en",
            "DataAccessRoleArn": "arn:aws:iam::111122223333:role/service-role/
AmazonComprehendServiceRole-example-role"
        }
    ]
}
```

Per ulteriori informazioni, consulta Analisi asincrona degli approfondimenti di Amazon Comprehend nella Guida per gli sviluppatori di Amazon Comprehend.

- Per i dettagli sull'API, consulta ListKeyPhrasesDetectionJobs AWS CLICommand Reference.

`list-pii-entities-detection-jobs`

Il seguente esempio di codice mostra come utilizzare`list-pii-entities-detection-jobs`.

AWS CLI

Come elencare tutti i processi di rilevamento delle entità pii

L'esempio `list-pii-entities-detection-jobs` seguente elenca tutti i processi di rilevamento asincrono, in corso e completati, delle pii.

```
aws comprehend list-pii-entities-detection-jobs
```

Output:

```
{
    "PiiEntitiesDetectionJobPropertiesList": [
        {
            "JobId": "6f9db0c42d0c810e814670ee4EXAMPLE",
            "JobArn": "arn:aws:comprehend:us-west-2:111122223333:pii-entities-
detection-job/6f9db0c42d0c810e814670ee4EXAMPLE",
            "JobName": "example-pii-detection-job",
            "JobStatus": "COMPLETED",
```

```
            "SubmitTime": "2023-06-09T21:02:46.241000+00:00",
            "EndTime": "2023-06-09T21:12:52.602000+00:00",
            "InputDataConfig": {
                "S3Uri": "s3://amzn-s3-demo-bucket/AsyncBatchJobs/",
                "InputFormat": "ONE_DOC_PER_LINE"
            },
            "OutputDataConfig": {
                "S3Uri": "s3://amzn-s3-demo-source-bucket/111122223333-
PII-6f9db0c42d0c810e814670ee4EXAMPLE/output/"
            },
            "LanguageCode": "en",
            "DataAccessRoleArn": "arn:aws:iam::111122223333:role/service-role/
AmazonComprehendServiceRole-example-role",
            "Mode": "ONLY_OFFSETS"
        },
        {
            "JobId": "d927562638cfa739331a99b3cEXAMPLE",
            "JobArn": "arn:aws:comprehend:us-west-2:111122223333:pii-entities-
detection-job/d927562638cfa739331a99b3cEXAMPLE",
            "JobName": "example-pii-detection-job-2",
            "JobStatus": "COMPLETED",
            "SubmitTime": "2023-06-09T21:20:58.211000+00:00",
            "EndTime": "2023-06-09T21:31:06.027000+00:00",
            "InputDataConfig": {
                "S3Uri": "s3://amzn-s3-demo-bucket/AsyncBatchJobs/",
                "InputFormat": "ONE_DOC_PER_LINE"
            },
            "OutputDataConfig": {
                "S3Uri": "s3://amzn-s3-demo-destination-bucket/
thefolder/111122223333-PII-d927562638cfa739331a99b3cEXAMPLE/output/"
            },
            "LanguageCode": "en",
            "DataAccessRoleArn": "arn:aws:iam::111122223333:role/service-role/
AmazonComprehendServiceRole-example-role",
            "Mode": "ONLY_OFFSETS"
        }
    ]
}
```

Per ulteriori informazioni, consulta Analisi asincrona degli approfondimenti di Amazon Comprehend nella Guida per gli sviluppatori di Amazon Comprehend.

- Per i dettagli sull'API, consulta ListPiiEntitiesDetectionJobs AWS CLICommand Reference.

`list-sentiment-detection-jobs`

Il seguente esempio di codice mostra come utilizzare`list-sentiment-detection-jobs`.

AWS CLI

Come elencare tutti i processi di rilevamento del sentiment

L'esempio `list-sentiment-detection-jobs` seguente elenca tutti i processi di rilevamento asincrono, in corso e completati, del sentiment.

```
aws comprehend list-sentiment-detection-jobs
```

Output:

```
{
    "SentimentDetectionJobPropertiesList": [
        {
            "JobId": "123456abcdeb0e11022f22a11EXAMPLE",
            "JobArn": "arn:aws:comprehend:us-west-2:111122223333:sentiment-
detection-job/123456abcdeb0e11022f22a11EXAMPLE",
            "JobName": "example-sentiment-detection-job",
            "JobStatus": "IN_PROGRESS",
            "SubmitTime": "2023-06-09T22:42:20.545000+00:00",
            "EndTime": "2023-06-09T22:52:27.416000+00:00",
            "InputDataConfig": {
                "S3Uri": "s3://amzn-s3-demo-bucket/MovieData",
                "InputFormat": "ONE_DOC_PER_LINE"
            },
            "OutputDataConfig": {
                "S3Uri": "s3://amzn-s3-demo-destination-bucket/
testfolder/111122223333-TS-123456abcdeb0e11022f22a11EXAMPLE/output/output.tar.gz"
            },
            "LanguageCode": "en",
            "DataAccessRoleArn": "arn:aws:iam::111122223333:role/service-role/
AmazonComprehendServiceRole-example-role"
        },
        {
            "JobId": "123456abcdeb0e11022f22a1EXAMPLE2",
            "JobArn": "arn:aws:comprehend:us-west-2:111122223333:sentiment-
detection-job/123456abcdeb0e11022f22a1EXAMPLE2",
            "JobName": "example-sentiment-detection-job-2",
            "JobStatus": "COMPLETED",
```

```
            "SubmitTime": "2023-06-09T23:16:15.956000+00:00",
            "EndTime": "2023-06-09T23:26:00.168000+00:00",
            "InputDataConfig": {
                "S3Uri": "s3://amzn-s3-demo-bucket/MovieData2",
                "InputFormat": "ONE_DOC_PER_LINE"
            },
            "OutputDataConfig": {
                "S3Uri": "s3://amzn-s3-demo-destination-bucket/
testfolder/111122223333-TS-123456abcdeb0e11022f22a1EXAMPLE2/output/output.tar.gz"
            },
            "LanguageCode": "en",
            "DataAccessRoleArn": "arn:aws:iam::111122223333:role/service-role/
AmazonComprehendServiceRole-example-role"
        }
    ]
}
```

Per ulteriori informazioni, consulta Analisi asincrona degli approfondimenti di Amazon Comprehend nella Guida per gli sviluppatori di Amazon Comprehend.

- Per i dettagli sull'API, consulta ListSentimentDetectionJobs AWS CLICommand Reference.

`list-tags-for-resource`

Il seguente esempio di codice mostra come utilizzare`list-tags-for-resource`.

AWS CLI

Come elencare i tag per le risorse

L'esempio `list-tags-for-resource` seguente elenca i tag per una risorsa Amazon Comprehend.

```
aws comprehend list-tags-for-resource \
    --resource-arn arn:aws:comprehend:us-west-2:111122223333:document-classifier/
example-classifier/version/1
```

Output:

```
{
    "ResourceArn": "arn:aws:comprehend:us-west-2:111122223333:document-classifier/
example-classifier/version/1",
    "Tags": [
```

```
        {
            "Key": "Department",
            "Value": "Finance"
        },
        {
            "Key": "location",
            "Value": "Seattle"
        }
    ]
}
```

Per ulteriori informazioni, consulta Tagging delle risorse nella Guida per gli sviluppatori di Amazon Comprehend.

- Per i dettagli sull'API, consulta ListTagsForResource AWS CLICommand Reference.

`list-targeted-sentiment-detection-jobs`

Il seguente esempio di codice mostra come utilizzare`list-targeted-sentiment-detection-jobs`.

AWS CLI

Come elencare tutti i processi di rilevamento del sentiment mirato

L'esempio `list-targeted-sentiment-detection-jobs` seguente elenca tutti i processi di rilevamento asincrono, in corso e completati, del sentiment mirato.

```
aws comprehend list-targeted-sentiment-detection-jobs
```

Output:

```
{
    "TargetedSentimentDetectionJobPropertiesList": [
        {
            "JobId": "123456abcdeb0e11022f22a11EXAMPLE",
            "JobArn": "arn:aws:comprehend:us-west-2:111122223333:targeted-sentiment-
detection-job/123456abcdeb0e11022f22a11EXAMPLE",
            "JobName": "example-targeted-sentiment-detection-job",
            "JobStatus": "COMPLETED",
            "SubmitTime": "2023-06-09T22:42:20.545000+00:00",
            "EndTime": "2023-06-09T22:52:27.416000+00:00",
            "InputDataConfig": {
```

```
                "S3Uri": "s3://amzn-s3-demo-bucket/MovieData",
                "InputFormat": "ONE_DOC_PER_LINE"
            },
            "OutputDataConfig": {
                "S3Uri": "s3://amzn-s3-demo-destination-bucket/
testfolder/111122223333-TS-123456abcdeb0e11022f22a11EXAMPLE/output/output.tar.gz"
            },
            "LanguageCode": "en",
            "DataAccessRoleArn": "arn:aws:iam::111122223333:role/service-role/
AmazonComprehendServiceRole-IOrole"
        },
        {
            "JobId": "123456abcdeb0e11022f22a1EXAMPLE2",
            "JobArn": "arn:aws:comprehend:us-west-2:111122223333:targeted-sentiment-
detection-job/123456abcdeb0e11022f22a1EXAMPLE2",
            "JobName": "example-targeted-sentiment-detection-job-2",
            "JobStatus": "COMPLETED",
            "SubmitTime": "2023-06-09T23:16:15.956000+00:00",
            "EndTime": "2023-06-09T23:26:00.168000+00:00",
            "InputDataConfig": {
                "S3Uri": "s3://amzn-s3-demo-bucket/MovieData2",
                "InputFormat": "ONE_DOC_PER_LINE"
            },
            "OutputDataConfig": {
                "S3Uri": "s3://amzn-s3-demo-destination-bucket/
testfolder/111122223333-TS-123456abcdeb0e11022f22a1EXAMPLE2/output/output.tar.gz"
            },
            "LanguageCode": "en",
            "DataAccessRoleArn": "arn:aws:iam::111122223333:role/service-role/
AmazonComprehendServiceRole-example-role"
        }
    ]
}
```

Per ulteriori informazioni, consulta Analisi asincrona degli approfondimenti di Amazon Comprehend nella Guida per gli sviluppatori di Amazon Comprehend.

- Per i dettagli sull'API, consulta ListTargetedSentimentDetectionJobs AWS CLICommand Reference.

`list-topics-detection-jobs`

Il seguente esempio di codice mostra come utilizzare`list-topics-detection-jobs`.

AWS CLI

Come elencare tutti i processi di rilevamento degli argomenti

L'esempio `list-topics-detection-jobs` seguente elenca tutti i processi asincroni di rilevamento di argomenti in corso e completati.

```
aws comprehend list-topics-detection-jobs
```

Output:

```
{
    "TopicsDetectionJobPropertiesList": [
        {
            "JobId": "123456abcdeb0e11022f22a11EXAMPLE",
            "JobArn": "arn:aws:comprehend:us-west-2:111122223333:topics-detection-
job/123456abcdeb0e11022f22a11EXAMPLE",
            "JobName" "topic-analysis-1"
            "JobStatus": "IN_PROGRESS",
            "SubmitTime": "2023-06-09T18:40:35.384000+00:00",
            "EndTime": "2023-06-09T18:46:41.936000+00:00",
            "InputDataConfig": {
                "S3Uri": "s3://amzn-s3-demo-bucket",
                "InputFormat": "ONE_DOC_PER_LINE"
            },
            "OutputDataConfig": {
                "S3Uri": "s3://amzn-s3-demo-destination-bucket/
thefolder/111122223333-TOPICS-123456abcdeb0e11022f22a11EXAMPLE/output/output.tar.gz"
            },
            "NumberOfTopics": 10,
            "DataAccessRoleArn": "arn:aws:iam::111122223333:role/service-role/
AmazonComprehendServiceRole-example-role"
        },
        {
            "JobId": "123456abcdeb0e11022f22a1EXAMPLE2",
            "JobArn": "arn:aws:comprehend:us-west-2:111122223333:topics-detection-
job/123456abcdeb0e11022f22a1EXAMPLE2",
            "JobName": "topic-analysis-2",
            "JobStatus": "COMPLETED",
            "SubmitTime": "2023-06-09T18:44:43.414000+00:00",
            "EndTime": "2023-06-09T18:50:50.872000+00:00",
            "InputDataConfig": {
                "S3Uri": "s3://amzn-s3-demo-bucket",
```

```
                "InputFormat": "ONE_DOC_PER_LINE"
            },
            "OutputDataConfig": {
                "S3Uri": "s3://amzn-s3-demo-destination-bucket/
thefolder/111122223333-TOPICS-123456abcdeb0e11022f22a1EXAMPLE2/output/output.tar.gz"
            },
            "NumberOfTopics": 10,
            "DataAccessRoleArn": "arn:aws:iam::111122223333:role/service-role/
AmazonComprehendServiceRole-example-role"
        },
        {
            "JobId": "123456abcdeb0e11022f22a1EXAMPLE3",
            "JobArn": "arn:aws:comprehend:us-west-2:111122223333:topics-detection-
job/123456abcdeb0e11022f22a1EXAMPLE3",
            "JobName": "topic-analysis-2",
            "JobStatus": "IN_PROGRESS",
            "SubmitTime": "2023-06-09T18:50:56.737000+00:00",
            "InputDataConfig": {
                "S3Uri": "s3://amzn-s3-demo-bucket",
                "InputFormat": "ONE_DOC_PER_LINE"
            },
            "OutputDataConfig": {
                "S3Uri": "s3://amzn-s3-demo-destination-bucket/
thefolder/111122223333-TOPICS-123456abcdeb0e11022f22a1EXAMPLE3/output/output.tar.gz"
            },
            "NumberOfTopics": 10,
            "DataAccessRoleArn": "arn:aws:iam::111122223333:role/service-role/
AmazonComprehendServiceRole-example-role"
        }
    ]
}
```

Per ulteriori informazioni, consulta Analisi asincrona degli approfondimenti di Amazon Comprehend nella Guida per gli sviluppatori di Amazon Comprehend.

- Per i dettagli sull'API, consulta ListTopicsDetectionJobs AWS CLICommand Reference.

`put-resource-policy`

Il seguente esempio di codice mostra come utilizzare`put-resource-policy`.

AWS CLI

Come allegare una policy basata su risorse

L'`put-resource-policy`esempio seguente allega una politica basata sulle risorse a un modello in modo che possa essere importato da un altro account. AWS La policy è collegata al modello nell'account `111122223333` e consente all'account `444455556666` l'importazione del modello.

```
aws comprehend put-resource-policy \
    --resource-arn arn:aws:comprehend:us-west-2:111122223333:document-classifier/example-classifier/version/1 \
    --resource-policy '{"Version":"2012-10-17",       "Statement": [{"Effect":"Allow","Action":"comprehend:ImportModel","Resource":"*","Principal": {"AWS":["arn:aws:iam::444455556666:root"]}}]}'
```

Ouput:

```
{
    "PolicyRevisionId": "aaa111d069d07afaa2aa3106aEXAMPLE"
}
```

Per ulteriori informazioni, consulta Copiare modelli personalizzati tra AWS account nella Amazon Comprehend Developer Guide.

- Per i dettagli sull'API, consulta AWS CLI Command PutResourcePolicyReference.

`start-document-classification-job`

Il seguente esempio di codice mostra come utilizzare`start-document-classification-job`.

AWS CLI

Come avviare un processo di classificazione dei documenti

L'esempio `start-document-classification-job` seguente avvia un processo di classificazione dei documenti con un modello personalizzato su tutti i file all'indirizzo specificato dal tag `--input-data-config`. In questo esempio, il bucket S3 di input contiene `SampleSMStext1.txt`, `SampleSMStext2.txt` e `SampleSMStext3.txt`. Il modello è stato precedentemente addestrato in base alla classificazione dei documenti dei messaggi SMS spam e non spam o "indesiderati". Al completamento del processo, `output.tar.gz` viene collocato

nella posizione specificata dal tag `--output-data-config`. `output.tar.gz` contiene `predictions.jsonl`, che riporta la classificazione di ogni documento. L'output Json viene stampato su una riga per file, ma di seguito è formattato per ottimizzarne la leggibilità.

```
aws comprehend start-document-classification-job \
    --job-name exampleclassificationjob \
    --input-data-config "S3Uri=s3://amzn-s3-demo-bucket-INPUT/jobdata/" \
    --output-data-config "S3Uri=s3://amzn-s3-demo-destination-bucket/testfolder/" \
    --data-access-role-arn arn:aws:iam::111122223333:role/service-role/
AmazonComprehendServiceRole-example-role \
    --document-classifier-arn arn:aws:comprehend:us-west-2:111122223333:document-
classifier/mymodel/version/12
```

Contenuto di `SampleSMStext1.txt`:

```
"CONGRATULATIONS! TXT 2155550100 to win $5000"
```

Contenuto di `SampleSMStext2.txt`:

```
"Hi, when do you want me to pick you up from practice?"
```

Contenuto di `SampleSMStext3.txt`:

```
"Plz send bank account # to 2155550100 to claim prize!!"
```

Output:

```
{
    "JobId": "e758dd56b824aa717ceab551fEXAMPLE",
    "JobArn": "arn:aws:comprehend:us-west-2:111122223333:document-classification-
job/e758dd56b824aa717ceab551fEXAMPLE",
    "JobStatus": "SUBMITTED"
}
```

Contenuto di `predictions.jsonl`:

```
{"File": "SampleSMSText1.txt", "Line": "0", "Classes": [{"Name": "spam", "Score":
 0.9999}, {"Name": "ham", "Score": 0.0001}]}
{"File": "SampleSMStext2.txt", "Line": "0", "Classes": [{"Name": "ham", "Score":
 0.9994}, {"Name": "spam", "Score": 0.0006}]}
```

```
{"File": "SampleSMSText3.txt", "Line": "0", "Classes": [{"Name": "spam", "Score":
 0.9999}, {"Name": "ham", "Score": 0.0001}]}
```

Per ulteriori informazioni, consulta Classificazione personalizzata nella Guida per gli sviluppatori di Amazon Comprehend.

- Per i dettagli sull'API, consulta StartDocumentClassificationJob AWS CLICommand Reference.

`start-dominant-language-detection-job`

Il seguente esempio di codice mostra come utilizzare`start-dominant-language-detection-job`.

AWS CLI

Come avviare un processo asincrono di rilevamento della lingua

L'esempio `start-dominant-language-detection-job` seguente avvia un processo di rilevamento della lingua asincrono per tutti i file che si trovano all'indirizzo specificato dal tag `--input-data-config`. Il bucket S3 in questo esempio contiene `Sampletext1.txt`. Al completamento del processo, la cartella `output` viene collocata nella posizione specificata dal tag `--output-data-config`. La cartella contiene `output.txt`, che a sua volta contiene la lingua dominante di ogni file di testo e il punteggio di affidabilità del modello pre-addestrato per ogni previsione.

```
aws comprehend start-dominant-language-detection-job \
    --job-name example_language_analysis_job \
    --language-code en \
    --input-data-config "S3Uri=s3://amzn-s3-demo-bucket/" \
    --output-data-config "S3Uri=s3://amzn-s3-demo-destination-bucket/testfolder/" \
    --data-access-role-arn arn:aws:iam::111122223333:role/service-role/
AmazonComprehendServiceRole-example-role \
    --language-code en
```

Contenuto di Sampletext1.txt:

```
"Physics is the natural science that involves the study of matter and its motion and
 behavior through space and time, along with related concepts such as energy and
 force."
```

Output:

```
{
    "JobId": "123456abcdeb0e11022f22a11EXAMPLE",
    "JobArn": "arn:aws:comprehend:us-west-2:111122223333:dominant-language-
detection-job/123456abcdeb0e11022f22a11EXAMPLE",
    "JobStatus": "SUBMITTED"
}
```

Contenuto di `output.txt`:

```
{"File": "Sampletext1.txt", "Languages": [{"LanguageCode": "en", "Score":
 0.9913753867149353}], "Line": 0}
```

Per ulteriori informazioni, consulta Analisi asincrona degli approfondimenti di Amazon Comprehend nella Guida per gli sviluppatori di Amazon Comprehend.

- Per i dettagli sull'API, consulta StartDominantLanguageDetectionJob AWS CLICommand Reference.

`start-entities-detection-job`

Il seguente esempio di codice mostra come utilizzare`start-entities-detection-job`.

AWS CLI

Esempio 1: come avviare un processo di rilevamento delle entità standard utilizzando il modello pre-addestrato

L'esempio `start-entities-detection-job` seguente avvia un processo di rilevamento delle entità asincrono per tutti i file che si trovano all'indirizzo specificato dal tag `--input-data-config`. Il bucket S3 in questo esempio contiene `Sampletext1.txt`, `Sampletext2.txt` e `Sampletext3.txt`. Al completamento del processo, la cartella `output` viene collocata nella posizione specificata dal tag `--output-data-config`. La cartella contiene `output.txt`, che elenca tutte le entità denominate rilevate in ogni file di testo e il punteggio di affidabilità del modello pre-addestrato per ogni previsione. L'output Json viene stampato su una riga per file, ma di seguito è formattato in modo da ottimizzarne la leggibilità.

```
aws comprehend start-entities-detection-job \
    --job-name entitiestest \
    --language-code en \
    --input-data-config "S3Uri=s3://amzn-s3-demo-bucket/" \
    --output-data-config "S3Uri=s3://amzn-s3-demo-destination-bucket/testfolder/" \
```

```
    --data-access-role-arn arn:aws:iam::111122223333:role/service-role/
AmazonComprehendServiceRole-example-role \
    --language-code en
```

Contenuto di `Sampletext1.txt`:

```
"Hello Zhang Wei, I am John. Your AnyCompany Financial Services, LLC credit card
 account 1111-XXXX-1111-XXXX has a minimum payment of $24.53 that is due by July
 31st."
```

Contenuto di `Sampletext2.txt`:

```
"Dear Max, based on your autopay settings for your account example1.org account, we
 will withdraw your payment on the due date from your bank account number XXXXXX1111
 with the routing number XXXXX0000. "
```

Contenuto di `Sampletext3.txt`:

```
"Jane, please submit any customer feedback from this weekend to AnySpa, 123 Main St,
 Anywhere and send comments to Alice at AnySpa@example.com."
```

Output:

```
{
    "JobId": "123456abcdeb0e11022f22a11EXAMPLE",
    "JobArn": "arn:aws:comprehend:us-west-2:111122223333:entities-detection-
job/123456abcdeb0e11022f22a11EXAMPLE",
    "JobStatus": "SUBMITTED"
}
```

Contenuto di `output.txt` con rientri di riga per una maggiore leggibilità:

```
{
"Entities": [
    {
    "BeginOffset": 6,
    "EndOffset": 15,
    "Score": 0.9994006636420306,
    "Text": "Zhang Wei",
    "Type": "PERSON"
    },
```

```
    {
    "BeginOffset": 22,
    "EndOffset": 26,
    "Score": 0.9976647915128143,
    "Text": "John",
    "Type": "PERSON"
    },
    {
    "BeginOffset": 33,
    "EndOffset": 67,
    "Score": 0.9984608700836206,
    "Text": "AnyCompany Financial Services, LLC",
    "Type": "ORGANIZATION"
    },
    {
    "BeginOffset": 88,
    "EndOffset": 107,
    "Score": 0.9868521019555556,
    "Text": "1111-XXXX-1111-XXXX",
    "Type": "OTHER"
    },
    {
    "BeginOffset": 133,
    "EndOffset": 139,
    "Score": 0.998242565709204,
    "Text": "$24.53",
    "Type": "QUANTITY"
    },
    {
    "BeginOffset": 155,
    "EndOffset": 164,
    "Score": 0.9993039263159287,
    "Text": "July 31st",
    "Type": "DATE"
    }
],
"File": "SampleText1.txt",
"Line": 0
}
{
"Entities": [
    {
    "BeginOffset": 5,
    "EndOffset": 8,
```

```
    "Score": 0.9866232147545232,
    "Text": "Max",
    "Type": "PERSON"
    },
    {
    "BeginOffset": 156,
    "EndOffset": 166,
    "Score": 0.9797723450933329,
    "Text": "XXXXXX1111",
    "Type": "OTHER"
    },
    {
    "BeginOffset": 191,
    "EndOffset": 200,
    "Score": 0.9247838572396843,
    "Text": "XXXXX0000",
    "Type": "OTHER"
    }
],
"File": "SampleText2.txt",
"Line": 0
}
{
 "Entities": [
    {
    "Score": 0.9990532994270325,
    "Type": "PERSON",
    "Text": "Jane",
    "BeginOffset": 0,
    "EndOffset": 4
    },
    {
    "Score": 0.9519651532173157,
    "Type": "DATE",
    "Text": "this weekend",
    "BeginOffset": 47,
    "EndOffset": 59
    },
    {
    "Score": 0.5566426515579224,
    "Type": "ORGANIZATION",
    "Text": "AnySpa",
    "BeginOffset": 63,
    "EndOffset": 69
```

```
    },
    {
    "Score": 0.8059805631637573,
    "Type": "LOCATION",
    "Text": "123 Main St, Anywhere",
    "BeginOffset": 71,
    "EndOffset": 92
    },
    {
    "Score": 0.998830258846283,
    "Type": "PERSON",
    "Text": "Alice",
    "BeginOffset": 114,
    "EndOffset": 119
    },
    {
    "Score": 0.997818112373352,
    "Type": "OTHER",
    "Text": "AnySpa@example.com",
    "BeginOffset": 123,
    "EndOffset": 138
    }
    ],
    "File": "SampleText3.txt",
    "Line": 0
}
```

Per ulteriori informazioni, consulta Analisi asincrona degli approfondimenti di Amazon Comprehend nella Guida per gli sviluppatori di Amazon Comprehend.

Esempio 2: come avviare un processo personalizzato di rilevamento delle entità

L'esempio `start-entities-detection-job` seguente avvia un processo di rilevamento delle entità personalizzato asincrono per tutti i file che si trovano all'indirizzo specificato dal tag `--input-data-config`. Il bucket S3 in questo esempio contiene `SampleFeedback1.txt`, `SampleFeedback2.txt` e `SampleFeedback3.txt`. Il modello di riconoscimento delle entità è stato addestrato sulla base dei feedback dell'assistenza clienti per riconoscere i nomi dei dispositivi. Al completamento del processo, la cartella `output` viene collocata nella posizione specificata dal tag `--output-data-config`. La cartella contiene `output.txt`, che elenca tutte le entità denominate rilevate in ogni file di testo e il punteggio di affidabilità del modello pre-addestrato per ogni previsione. L'output Json viene stampato su una riga per file, ma di seguito è formattato per ottimizzarne la leggibilità.

```
aws comprehend start-entities-detection-job \
    --job-name customentitiestest \
    --entity-recognizer-arn "arn:aws:comprehend:us-west-2:111122223333:entity-recognizer/entityrecognizer" \
    --language-code en \
    --input-data-config "S3Uri=s3://amzn-s3-demo-bucket/jobdata/" \
    --output-data-config "S3Uri=s3://amzn-s3-demo-destination-bucket/testfolder/" \
    --data-access-role-arn "arn:aws:iam::111122223333:role/service-role/AmazonComprehendServiceRole-IOrole"
```

Contenuto di `SampleFeedback1.txt`:

```
"I've been on the AnyPhone app have had issues for 24 hours when trying to pay bill. Cannot make payment. Sigh. | Oh man! Lets get that app up and running. DM me, and we can get to work!"
```

Contenuto di `SampleFeedback2.txt`:

```
"Hi, I have a discrepancy with my new bill. Could we get it sorted out? A rep added stuff I didnt sign up for when I did my AnyPhone 10 upgrade. | We can absolutely get this sorted!"
```

Contenuto di `SampleFeedback3.txt`:

```
"Is the by 1 get 1 free AnySmartPhone promo still going on? | Hi Christian! It ended yesterday, send us a DM if you have any questions and we can take a look at your options!"
```

Output:

```
{
    "JobId": "019ea9edac758806850fa8a79ff83021",
    "JobArn": "arn:aws:comprehend:us-west-2:111122223333:entities-detection-job/019ea9edac758806850fa8a79ff83021",
    "JobStatus": "SUBMITTED"
}
```

Contenuto di `output.txt` con rientri di riga per una maggiore leggibilità:

```
{
```

```
"Entities": [
    {
    "BeginOffset": 17,
    "EndOffset": 25,
    "Score": 0.9999728210205924,
    "Text": "AnyPhone",
    "Type": "DEVICE"
    }
],
"File": "SampleFeedback1.txt",
"Line": 0
}
{
"Entities": [
    {
    "BeginOffset": 123,
    "EndOffset": 133,
    "Score": 0.9999892116761524,
    "Text": "AnyPhone 10",
    "Type": "DEVICE"
    }
],
"File": "SampleFeedback2.txt",
"Line": 0
}
{
"Entities": [
    {
    "BeginOffset": 23,
    "EndOffset": 35,
    "Score": 0.9999971389852362,
    "Text": "AnySmartPhone",
    "Type": "DEVICE"
    }
],
"File": "SampleFeedback3.txt",
"Line": 0
}
```


Per ulteriori informazioni, consulta Riconoscimento delle entità personalizzate nella Guida per gli sviluppatori di Amazon Comprehend.

- Per i dettagli sull'API, consulta StartEntitiesDetectionJob AWS CLICommand Reference.

`start-events-detection-job`

Il seguente esempio di codice mostra come utilizzare`start-events-detection-job`.

AWS CLI

Come avviare un processo asincrono di rilevamento degli eventi

L'esempio `start-events-detection-job` seguente avvia un processo di rilevamento degli eventi asincrono per tutti i file che si trovano all'indirizzo specificato dal tag `--input-data-config`. I possibili tipi di eventi di destinazione includono `BANKRUPCTY`, `EMPLOYMENT`, `CORPORATE_ACQUISITION`, `INVESTMENT_GENERAL`, `CORPORATE_MERGER`, `IPO`, `RIGHTS_ISSUE`, `SECONDARY_OFFERING`, `SHELF_OFFERING`, `TENDER_OFFERING` e `STOCK_SPLIT`. Il bucket S3 in questo esempio contiene `SampleText1.txt`, `SampleText2.txt` e `SampleText3.txt`. Al completamento del processo, la cartella `output` viene collocata nella posizione specificata dal tag `--output-data-config`. La cartella contiene `SampleText1.txt.out`, `SampleText2.txt.out` e `SampleText3.txt.out`. L'output JSON viene stampato su una riga per file, ma di seguito è formattato per ottimizzarne la leggibilità.

```
aws comprehend start-events-detection-job \
    --job-name events-detection-1 \
    --input-data-config "S3Uri=s3://amzn-s3-demo-bucket/EventsData" \
    --output-data-config "S3Uri=s3://amzn-s3-demo-destination-bucket/testfolder/" \
    --data-access-role-arn arn:aws:iam::111122223333:role/service-role/
AmazonComprehendServiceRole-servicerole \
    --language-code en \
    --target-event-
types "BANKRUPTCY" "EMPLOYMENT" "CORPORATE_ACQUISITION" "CORPORATE_MERGER" "INVESTMENT_GENER
```

Contenuto di `SampleText1.txt`:

```
"Company AnyCompany grew by increasing sales and through acquisitions. After
 purchasing competing firms in 2020, AnyBusiness, a part of the AnyBusinessGroup,
 gave Jane Does firm a going rate of one cent a gallon or forty-two cents a barrel."
```

Contenuto di `SampleText2.txt`:

```
"In 2021, AnyCompany officially purchased AnyBusiness for 100 billion dollars,
 surprising and exciting the shareholders."
```

Contenuto di `SampleText3.txt`:

```
"In 2022, AnyCompany stock crashed 50. Eventually later that year they filed for
 bankruptcy."
```

Output:

```
{
    "JobId": "123456abcdeb0e11022f22a11EXAMPLE",
    "JobArn": "arn:aws:comprehend:us-west-2:111122223333:events-detection-
job/123456abcdeb0e11022f22a11EXAMPLE",
    "JobStatus": "SUBMITTED"
}
```

Contenuto di `SampleText1.txt.out` con rientri di riga per una maggiore leggibilità:

```
{
    "Entities": [
        {
        "Mentions": [
            {
            "BeginOffset": 8,
            "EndOffset": 18,
            "Score": 0.99977,
            "Text": "AnyCompany",
            "Type": "ORGANIZATION",
            "GroupScore": 1
            },
            {
            "BeginOffset": 112,
            "EndOffset": 123,
            "Score": 0.999747,
            "Text": "AnyBusiness",
            "Type": "ORGANIZATION",
            "GroupScore": 0.979826
            },
            {
            "BeginOffset": 171,
            "EndOffset": 175,
            "Score": 0.999615,
            "Text": "firm",
            "Type": "ORGANIZATION",
            "GroupScore": 0.871647
            }
```

```
        ]
        },
        {
        "Mentions": [
            {
            "BeginOffset": 97,
            "EndOffset": 102,
            "Score": 0.987687,
            "Text": "firms",
            "Type": "ORGANIZATION",
            "GroupScore": 1
            }
        ]
        },
        {
        "Mentions": [
            {
            "BeginOffset": 103,
            "EndOffset": 110,
            "Score": 0.999458,
            "Text": "in 2020",
            "Type": "DATE",
            "GroupScore": 1
            }
        ]
        },
        {
        "Mentions": [
            {
            "BeginOffset": 160,
            "EndOffset": 168,
            "Score": 0.999649,
            "Text": "John Doe",
            "Type": "PERSON",
            "GroupScore": 1
            }
        ]
        }
    ],
    "Events": [
        {
        "Type": "CORPORATE_ACQUISITION",
        "Arguments": [
            {
```

```
            "EntityIndex": 0,
            "Role": "INVESTOR",
            "Score": 0.99977
            }
        ],
        "Triggers": [
            {
            "BeginOffset": 56,
            "EndOffset": 68,
            "Score": 0.999967,
            "Text": "acquisitions",
            "Type": "CORPORATE_ACQUISITION",
            "GroupScore": 1
            }
        ]
        },
        {
        "Type": "CORPORATE_ACQUISITION",
        "Arguments": [
            {
            "EntityIndex": 1,
            "Role": "INVESTEE",
            "Score": 0.987687
            },
            {
            "EntityIndex": 2,
            "Role": "DATE",
            "Score": 0.999458
            },
            {
            "EntityIndex": 3,
            "Role": "INVESTOR",
            "Score": 0.999649
            }
        ],
        "Triggers": [
            {
            "BeginOffset": 76,
            "EndOffset": 86,
            "Score": 0.999973,
            "Text": "purchasing",
            "Type": "CORPORATE_ACQUISITION",
            "GroupScore": 1
            }
```

```
        ]
        }
    ],
    "File": "SampleText1.txt",
    "Line": 0
}
```

Contenuto di `SampleText2.txt.out`:

```
{
    "Entities": [
        {
        "Mentions": [
            {
            "BeginOffset": 0,
            "EndOffset": 7,
            "Score": 0.999473,
            "Text": "In 2021",
            "Type": "DATE",
            "GroupScore": 1
            }
        ]
        },
        {
        "Mentions": [
            {
            "BeginOffset": 9,
            "EndOffset": 19,
            "Score": 0.999636,
            "Text": "AnyCompany",
            "Type": "ORGANIZATION",
            "GroupScore": 1
            }
        ]
        },
        {
        "Mentions": [
            {
            "BeginOffset": 45,
            "EndOffset": 56,
            "Score": 0.999712,
            "Text": "AnyBusiness",
            "Type": "ORGANIZATION",
```

```
            "GroupScore": 1
            }
        ]
        },
        {
        "Mentions": [
            {
            "BeginOffset": 61,
            "EndOffset": 80,
            "Score": 0.998886,
            "Text": "100 billion dollars",
            "Type": "MONETARY_VALUE",
            "GroupScore": 1
            }
        ]
        }
    ],
    "Events": [
        {
        "Type": "CORPORATE_ACQUISITION",
        "Arguments": [
            {
            "EntityIndex": 3,
            "Role": "AMOUNT",
            "Score": 0.998886
            },
            {
            "EntityIndex": 2,
            "Role": "INVESTEE",
            "Score": 0.999712
            },
            {
            "EntityIndex": 0,
            "Role": "DATE",
            "Score": 0.999473
            },
            {
            "EntityIndex": 1,
            "Role": "INVESTOR",
            "Score": 0.999636
            }
        ],
        "Triggers": [
            {
```

```
            "BeginOffset": 31,
            "EndOffset": 40,
            "Score": 0.99995,
            "Text": "purchased",
            "Type": "CORPORATE_ACQUISITION",
            "GroupScore": 1
            }
        ]
        }
    ],
    "File": "SampleText2.txt",
    "Line": 0
}
```

Contenuto di `SampleText3.txt.out`:

```
{
    "Entities": [
        {
        "Mentions": [
            {
            "BeginOffset": 9,
            "EndOffset": 19,
            "Score": 0.999774,
            "Text": "AnyCompany",
            "Type": "ORGANIZATION",
            "GroupScore": 1
            },
            {
            "BeginOffset": 66,
            "EndOffset": 70,
            "Score": 0.995717,
            "Text": "they",
            "Type": "ORGANIZATION",
            "GroupScore": 0.997626
            }
        ]
        },
        {
        "Mentions": [
            {
            "BeginOffset": 50,
            "EndOffset": 65,
```

```
            "Score": 0.999656,
            "Text": "later that year",
            "Type": "DATE",
            "GroupScore": 1
            }
        ]
        }
    ],
    "Events": [
        {
        "Type": "BANKRUPTCY",
        "Arguments": [
            {
            "EntityIndex": 1,
            "Role": "DATE",
            "Score": 0.999656
            },
            {
            "EntityIndex": 0,
            "Role": "FILER",
            "Score": 0.995717
            }
        ],
        "Triggers": [
            {
            "BeginOffset": 81,
            "EndOffset": 91,
            "Score": 0.999936,
            "Text": "bankruptcy",
            "Type": "BANKRUPTCY",
            "GroupScore": 1
            }
        ]
        }
    ],
    "File": "SampleText3.txt",
    "Line": 0
}
```

Per ulteriori informazioni, consulta Analisi asincrona degli approfondimenti di Amazon Comprehend nella Guida per gli sviluppatori di Amazon Comprehend.

- Per i dettagli sull'API, consulta StartEventsDetectionJob AWS CLICommand Reference.

`start-flywheel-iteration`

Il seguente esempio di codice mostra come utilizzare`start-flywheel-iteration`.

AWS CLI

Come avviare un'iterazione del flywheel

L'esempio `start-flywheel-iteration` seguente avvia un'iterazione del flywheel. Questa operazione utilizza qualsiasi nuovo set di dati nel flywheel per addestrare una nuova versione del modello.

```
aws comprehend start-flywheel-iteration \
    --flywheel-arn arn:aws:comprehend:us-west-2:111122223333:flywheel/example-flywheel
```

Output:

```
{
    "FlywheelArn": "arn:aws:comprehend:us-west-2:111122223333:flywheel/example-flywheel",
    "FlywheelIterationId": "12345123TEXAMPLE"
}
```

Per ulteriori informazioni, consulta Panoramica del volano dei dati nella Guida per gli sviluppatori di Amazon Comprehend.

- Per i dettagli sull'API, consulta StartFlywheelIteration AWS CLICommand Reference.

`start-key-phrases-detection-job`

Il seguente esempio di codice mostra come utilizzare`start-key-phrases-detection-job`.

AWS CLI

Come avviare un processo di rilevamento delle frasi chiave

L'esempio `start-key-phrases-detection-job` seguente avvia un processo di rilevamento delle frasi chiave asincrono per tutti i file che si trovano all'indirizzo specificato dal tag `--input-data-config`. Il bucket S3 in questo esempio contiene `Sampletext1.txt`, `Sampletext2.txt` e `Sampletext3.txt`. Al completamento del processo, la cartella `output`

viene collocata nella posizione specificata dal tag `--output-data-config`. La cartella contiene il file `output.txt`, che a sua volta contiene tutte le frasi chiave denominate rilevate in ogni file di testo e il punteggio di affidabilità del modello pre-addestrato per ogni previsione. L'output Json viene stampato su una riga per file, ma di seguito è formattato per ottimizzarne la leggibilità.

```
aws comprehend start-key-phrases-detection-job \
    --job-name keyphrasesanalysistest1 \
    --language-code en \
    --input-data-config "S3Uri=s3://amzn-s3-demo-bucket/" \
    --output-data-config "S3Uri=s3://amzn-s3-demo-destination-bucket/testfolder/" \
    --data-access-role-arn "arn:aws:iam::111122223333:role/service-role/
AmazonComprehendServiceRole-example-role" \
    --language-code en
```

Contenuto di `Sampletext1.txt`:

```
"Hello Zhang Wei, I am John. Your AnyCompany Financial Services, LLC credit card
 account 1111-XXXX-1111-XXXX has a minimum payment of $24.53 that is due by July
 31st."
```

Contenuto di `Sampletext2.txt`:

```
"Dear Max, based on your autopay settings for your account Internet.org account, we
 will withdraw your payment on the due date from your bank account number XXXXXX1111
 with the routing number XXXXX0000. "
```

Contenuto di `Sampletext3.txt`:

```
"Jane, please submit any customer feedback from this weekend to Sunshine Spa, 123
 Main St, Anywhere and send comments to Alice at AnySpa@example.com."
```

Output:

```
{
    "JobId": "123456abcdeb0e11022f22a11EXAMPLE",
    "JobArn": "arn:aws:comprehend:us-west-2:111122223333:key-phrases-detection-
job/123456abcdeb0e11022f22a11EXAMPLE",
    "JobStatus": "SUBMITTED"
}
```

Contenuto di `output.txt` con rientri di riga per una maggiore leggibilità:


```
{
    "File": "SampleText1.txt",
    "KeyPhrases": [
        {
        "BeginOffset": 6,
        "EndOffset": 15,
        "Score": 0.9748965572679326,
        "Text": "Zhang Wei"
        },
        {
        "BeginOffset": 22,
        "EndOffset": 26,
        "Score": 0.9997344722354619,
        "Text": "John"
        },
        {
        "BeginOffset": 28,
        "EndOffset": 62,
        "Score": 0.9843791074032948,
        "Text": "Your AnyCompany Financial Services"
        },
        {
        "BeginOffset": 64,
        "EndOffset": 107,
        "Score": 0.8976122401721824,
        "Text": "LLC credit card account 1111-XXXX-1111-XXXX"
        },
        {
        "BeginOffset": 112,
        "EndOffset": 129,
        "Score": 0.9999612982629748,
        "Text": "a minimum payment"
        },
        {
        "BeginOffset": 133,
        "EndOffset": 139,
        "Score": 0.99975728947036,
        "Text": "$24.53"
        },
        {
        "BeginOffset": 155,
        "EndOffset": 164,
```

```
        "Score": 0.9940866241449973,
        "Text": "July 31st"
        }
    ],
    "Line": 0
    }
    {
    "File": "SampleText2.txt",
    "KeyPhrases": [
        {
        "BeginOffset": 0,
        "EndOffset": 8,
        "Score": 0.9974021100118472,
        "Text": "Dear Max"
        },
        {
        "BeginOffset": 19,
        "EndOffset": 40,
        "Score": 0.9961120519515884,
        "Text": "your autopay settings"
        },
        {
        "BeginOffset": 45,
        "EndOffset": 78,
        "Score": 0.9980620070116009,
        "Text": "your account Internet.org account"
        },
        {
        "BeginOffset": 97,
        "EndOffset": 109,
        "Score": 0.999919660140754,
        "Text": "your payment"
        },
        {
        "BeginOffset": 113,
        "EndOffset": 125,
        "Score": 0.9998370719754205,
        "Text": "the due date"
        },
        {
        "BeginOffset": 131,
        "EndOffset": 166,
        "Score": 0.9955068678502509,
        "Text": "your bank account number XXXXXX1111"
```

```
        },
        {
        "BeginOffset": 172,
        "EndOffset": 200,
        "Score": 0.8653433315829526,
        "Text": "the routing number XXXXX0000"
        }
    ],
    "Line": 0
    }
    {
    "File": "SampleText3.txt",
    "KeyPhrases": [
        {
        "BeginOffset": 0,
        "EndOffset": 4,
        "Score": 0.9142947833681668,
        "Text": "Jane"
        },
        {
        "BeginOffset": 20,
        "EndOffset": 41,
        "Score": 0.9984325676596763,
        "Text": "any customer feedback"
        },
        {
        "BeginOffset": 47,
        "EndOffset": 59,
        "Score": 0.9998782448150636,
        "Text": "this weekend"
        },
        {
        "BeginOffset": 63,
        "EndOffset": 75,
        "Score": 0.99866741830757,
        "Text": "Sunshine Spa"
        },
        {
        "BeginOffset": 77,
        "EndOffset": 88,
        "Score": 0.9695803485466054,
        "Text": "123 Main St"
        },
        {
```

```
        "BeginOffset": 108,
        "EndOffset": 116,
        "Score": 0.9997065928550928,
        "Text": "comments"
        },
        {
        "BeginOffset": 120,
        "EndOffset": 125,
        "Score": 0.9993466833825161,
        "Text": "Alice"
        },
        {
        "BeginOffset": 129,
        "EndOffset": 144,
        "Score": 0.9654563612885667,
        "Text": "AnySpa@example.com"
        }
    ],
    "Line": 0
}
```

Per ulteriori informazioni, consulta Analisi asincrona degli approfondimenti di Amazon Comprehend nella Guida per gli sviluppatori di Amazon Comprehend.

- Per i dettagli sull'API, consulta StartKeyPhrasesDetectionJob AWS CLICommand Reference.

`start-pii-entities-detection-job`

Il seguente esempio di codice mostra come utilizzare`start-pii-entities-detection-job`.

AWS CLI

Come avviare un processo asincrono di rilevamento delle PII

L'esempio `start-pii-entities-detection-job` seguente avvia un processo di rilevamento delle entità Informazioni di identificazione personale (PII) asincrono per tutti i file che si trovano all'indirizzo specificato dal tag `--input-data-config`. Il bucket S3 in questo esempio contiene `Sampletext1.txt`, `Sampletext2.txt` e `Sampletext3.txt`. Al completamento del processo, la cartella `output` viene collocata nella posizione specificata dal tag `--output-data-config`. La cartella contiene `SampleText1.txt.out`, `SampleText2.txt.out` e `SampleText3.txt.out`, che elencano le entità denominate all'interno di ogni file di testo.

L'output Json viene stampato su una riga per file, ma di seguito è formattato per ottimizzarne la leggibilità.

```
aws comprehend start-pii-entities-detection-job \
    --job-name entities_test \
    --language-code en \
    --input-data-config "S3Uri=s3://amzn-s3-demo-bucket/" \
    --output-data-config "S3Uri=s3://amzn-s3-demo-destination-bucket/testfolder/" \
    --data-access-role-arn arn:aws:iam::111122223333:role/service-role/
AmazonComprehendServiceRole-example-role \
    --language-code en \
    --mode ONLY_OFFSETS
```

Contenuto di `Sampletext1.txt`:

```
"Hello Zhang Wei, I am John. Your AnyCompany Financial Services, LLC credit card
 account 1111-XXXX-1111-XXXX has a minimum payment of $24.53 that is due by July
 31st."
```

Contenuto di `Sampletext2.txt`:

```
"Dear Max, based on your autopay settings for your account Internet.org account, we
 will withdraw your payment on the due date from your bank account number XXXXXX1111
 with the routing number XXXXX0000. "
```

Contenuto di `Sampletext3.txt`:

```
"Jane, please submit any customer feedback from this weekend to Sunshine Spa, 123
 Main St, Anywhere and send comments to Alice at AnySpa@example.com."
```

Output:

```
{
    "JobId": "123456abcdeb0e11022f22a11EXAMPLE",
    "JobArn": "arn:aws:comprehend:us-west-2:111122223333:pii-entities-detection-
job/123456abcdeb0e11022f22a11EXAMPLE",
    "JobStatus": "SUBMITTED"
}
```

Contenuto di `SampleText1.txt.out` con rientri di riga per una maggiore leggibilità:

```
{
    "Entities": [
        {
        "BeginOffset": 6,
        "EndOffset": 15,
        "Type": "NAME",
        "Score": 0.9998490510222595
        },
        {
        "BeginOffset": 22,
        "EndOffset": 26,
        "Type": "NAME",
        "Score": 0.9998937958019426
        },
        {
        "BeginOffset": 88,
        "EndOffset": 107,
        "Type": "CREDIT_DEBIT_NUMBER",
        "Score": 0.9554297245278491
        },
        {
        "BeginOffset": 155,
        "EndOffset": 164,
        "Type": "DATE_TIME",
        "Score": 0.9999720462925257
        }
    ],
    "File": "SampleText1.txt",
    "Line": 0
}
```


Contenuto di `SampleText2.txt.out` con rientri di riga per una maggiore leggibilità:


```
{
    "Entities": [
        {
        "BeginOffset": 5,
        "EndOffset": 8,
        "Type": "NAME",
        "Score": 0.9994390774924007
        },
        {
        "BeginOffset": 58,
```

```
        "EndOffset": 70,
        "Type": "URL",
        "Score": 0.9999958276922101
        },
        {
        "BeginOffset": 156,
        "EndOffset": 166,
        "Type": "BANK_ACCOUNT_NUMBER",
        "Score": 0.9999721058045592
        },
        {
        "BeginOffset": 191,
        "EndOffset": 200,
        "Type": "BANK_ROUTING",
        "Score": 0.9998968945989909
        }
    ],
    "File": "SampleText2.txt",
    "Line": 0
}
```

Contenuto di `SampleText3.txt.out` con rientri di riga per una maggiore leggibilità:

```
{
    "Entities": [
        {
        "BeginOffset": 0,
        "EndOffset": 4,
        "Type": "NAME",
        "Score": 0.999949934606805
        },
        {
        "BeginOffset": 77,
        "EndOffset": 88,
        "Type": "ADDRESS",
        "Score": 0.9999035300466904
        },
        {
        "BeginOffset": 120,
        "EndOffset": 125,
        "Type": "NAME",
        "Score": 0.9998203838716296
        },
```

```
        {
        "BeginOffset": 129,
        "EndOffset": 144,
        "Type": "EMAIL",
        "Score": 0.9998313473105228
        }
    ],
    "File": "SampleText3.txt",
    "Line": 0
}
```

Per ulteriori informazioni, consulta Analisi asincrona degli approfondimenti di Amazon Comprehend nella Guida per gli sviluppatori di Amazon Comprehend.

- Per i dettagli sull'API, consulta StartPiiEntitiesDetectionJob AWS CLICommand Reference.

`start-sentiment-detection-job`

Il seguente esempio di codice mostra come utilizzare`start-sentiment-detection-job`.

AWS CLI

Come avviare un processo asincrono di analisi del sentiment

L'esempio `start-sentiment-detection-job` seguente avvia un processo di rilevamento del sentiment asincrono per tutti i file che si trovano all'indirizzo specificato dal tag `--input-data-config`. Il bucket S3 in questo esempio contiene `SampleMovieReview1.txt`, `SampleMovieReview2.txt` e `SampleMovieReview3.txt`. Al completamento del processo, la cartella `output` viene collocata nella posizione specificata dal tag `--output-data-config`. La cartella contiene il file `output.txt`, che a sua volta contiene i sentiment prevalenti in ogni file di testo e il punteggio di affidabilità del modello pre-addestrato per ogni previsione. L'output Json viene stampato su una riga per file, ma di seguito è formattato per ottimizzarne la leggibilità.

```
aws comprehend start-sentiment-detection-job \
    --job-name example-sentiment-detection-job \
    --language-code en \
    --input-data-config "S3Uri=s3://amzn-s3-demo-bucket/MovieData" \
    --output-data-config "S3Uri=s3://amzn-s3-demo-destination-bucket/testfolder/" \
    --data-access-role-arn arn:aws:iam::111122223333:role/service-role/
AmazonComprehendServiceRole-example-role
```

Contenuto di `SampleMovieReview1.txt`:

```
"The film, AnyMovie2, is fairly predictable and just okay."
```

Contenuto di `SampleMovieReview2.txt`:

```
"AnyMovie2 is the essential sci-fi film that I grew up watching when I was a kid. I
 highly recommend this movie."
```

Contenuto di `SampleMovieReview3.txt`:

```
"Don't get fooled by the 'awards' for AnyMovie2. All parts of the film were poorly
 stolen from other modern directors."
```

Output:

```
{
    "JobId": "0b5001e25f62ebb40631a9a1a7fde7b3",
    "JobArn": "arn:aws:comprehend:us-west-2:111122223333:sentiment-detection-
job/0b5001e25f62ebb40631a9a1a7fde7b3",
    "JobStatus": "SUBMITTED"
}
```

Contenuto di `output.txt` con rientri di riga per una maggiore leggibilità:

```
{
    "File": "SampleMovieReview1.txt",
        "Line": 0,
        "Sentiment": "MIXED",
        "SentimentScore": {
            "Mixed": 0.6591159105300903,
            "Negative": 0.26492202281951904,
            "Neutral": 0.035430654883384705,
            "Positive": 0.04053137078881264
            }
        }
    {
    "File": "SampleMovieReview2.txt",
        "Line": 0,
        "Sentiment": "POSITIVE",
        "SentimentScore": {
            "Mixed": 0.00000871846623340 3888,
            "Negative": 0.00006134175055194646,
```

```
            "Neutral": 0.0002941041602753103,
            "Positive": 0.9996358156204224
            }
        }
    {
    "File": "SampleMovieReview3.txt",
        "Line": 0,
        "Sentiment": "NEGATIVE",
        "SentimentScore": {
            "Mixed": 0.004146667663007975,
            "Negative": 0.9645107984542847,
            "Neutral": 0.016559595242142677,
            "Positive": 0.014782938174903393
        }
    }
}
```

Per ulteriori informazioni, consulta Analisi asincrona degli approfondimenti di Amazon Comprehend nella Guida per gli sviluppatori di Amazon Comprehend.

- Per i dettagli sull'API, consulta StartSentimentDetectionJob AWS CLICommand Reference.

`start-targeted-sentiment-detection-job`

Il seguente esempio di codice mostra come utilizzare`start-targeted-sentiment-detection-job`.

AWS CLI

Come avviare un processo asincrono di analisi del sentiment mirato

L'esempio `start-targeted-sentiment-detection-job` seguente avvia un processo di rilevamento del sentiment mirato asincrono per tutti i file che si trovano all'indirizzo specificato dal tag `--input-data-config`. Il bucket S3 in questo esempio contiene `SampleMovieReview1.txt`, `SampleMovieReview2.txt` e `SampleMovieReview3.txt`. Al completamento del processo, `output.tar.gz` viene collocato nella posizione specificata dal tag `--output-data-configg`. `output.tar.gz` contiene i file `SampleMovieReview1.txt.out`, `SampleMovieReview2.txt.out` e `SampleMovieReview3.txt.out`, ciascuno dei quali contiene tutte le entità denominate e i sentiment associati per un singolo file di testo di input.

```
aws comprehend start-targeted-sentiment-detection-job \
    --job-name targeted_movie_review_analysis1 \
```

```
    --language-code en \
    --input-data-config "S3Uri=s3://amzn-s3-demo-bucket/MovieData" \
    --output-data-config "S3Uri=s3://amzn-s3-demo-destination-bucket/testfolder/" \
    --data-access-role-arn arn:aws:iam::111122223333:role/service-role/
AmazonComprehendServiceRole-example-role
```

Contenuto di `SampleMovieReview1.txt`:

```
"The film, AnyMovie, is fairly predictable and just okay."
```

Contenuto di `SampleMovieReview2.txt`:

```
"AnyMovie is the essential sci-fi film that I grew up watching when I was a kid. I
 highly recommend this movie."
```

Contenuto di `SampleMovieReview3.txt`:

```
"Don't get fooled by the 'awards' for AnyMovie. All parts of the film were poorly
 stolen from other modern directors."
```

Output:

```
{
    "JobId": "0b5001e25f62ebb40631a9a1a7fde7b3",
    "JobArn": "arn:aws:comprehend:us-west-2:111122223333:targeted-sentiment-
detection-job/0b5001e25f62ebb40631a9a1a7fde7b3",
    "JobStatus": "SUBMITTED"
}
```

Contenuto di `SampleMovieReview1.txt.out` con rientri di riga per una maggiore leggibilità:

```
{
    "Entities": [
        {
        "DescriptiveMentionIndex": [
            0
        ],
        "Mentions": [
            {
            "BeginOffset": 4,
            "EndOffset": 8,
```

```
            "Score": 0.994972,
            "GroupScore": 1,
            "Text": "film",
            "Type": "MOVIE",
            "MentionSentiment": {
                "Sentiment": "NEUTRAL",
                "SentimentScore": {
                "Mixed": 0,
                "Negative": 0,
                "Neutral": 1,
                "Positive": 0
                }
            }
            }
        ]
        },
        {
        "DescriptiveMentionIndex": [
            0
        ],
        "Mentions": [
            {
            "BeginOffset": 10,
            "EndOffset": 18,
            "Score": 0.631368,
            "GroupScore": 1,
            "Text": "AnyMovie",
            "Type": "ORGANIZATION",
            "MentionSentiment": {
                "Sentiment": "POSITIVE",
                "SentimentScore": {
                "Mixed": 0.001729,
                "Negative": 0.000001,
                "Neutral": 0.000318,
                "Positive": 0.997952
                }
            }
            }
        ]
        }
    ],
    "File": "SampleMovieReview1.txt",
    "Line": 0
```

```
}
```

Contenuto di `SampleMovieReview2.txt.out` con rientri di riga per una maggiore leggibilità:

```
{
    "Entities": [
        {
        "DescriptiveMentionIndex": [
            0
        ],
        "Mentions": [
            {
            "BeginOffset": 0,
            "EndOffset": 8,
            "Score": 0.854024,
            "GroupScore": 1,
            "Text": "AnyMovie",
            "Type": "MOVIE",
            "MentionSentiment": {
                "Sentiment": "POSITIVE",
                "SentimentScore": {
                "Mixed": 0,
                "Negative": 0,
                "Neutral": 0.000007,
                "Positive": 0.999993
                }
            }
            },
            {
            "BeginOffset": 104,
            "EndOffset": 109,
            "Score": 0.999129,
            "GroupScore": 0.502937,
            "Text": "movie",
            "Type": "MOVIE",
            "MentionSentiment": {
                "Sentiment": "POSITIVE",
                "SentimentScore": {
                "Mixed": 0,
                "Negative": 0,
                "Neutral": 0,
                "Positive": 1
                }
```

```
            }
            },
            {
            "BeginOffset": 33,
            "EndOffset": 37,
            "Score": 0.999823,
            "GroupScore": 0.999252,
            "Text": "film",
            "Type": "MOVIE",
            "MentionSentiment": {
                "Sentiment": "POSITIVE",
                "SentimentScore": {
                "Mixed": 0,
                "Negative": 0,
                "Neutral": 0.000001,
                "Positive": 0.999999
                }
            }
            }
        ]
        },
        {
        "DescriptiveMentionIndex": [
            0,
            1,
            2
        ],
        "Mentions": [
            {
            "BeginOffset": 43,
            "EndOffset": 44,
            "Score": 0.999997,
            "GroupScore": 1,
            "Text": "I",
            "Type": "PERSON",
            "MentionSentiment": {
                "Sentiment": "NEUTRAL",
                "SentimentScore": {
                "Mixed": 0,
                "Negative": 0,
                "Neutral": 1,
                "Positive": 0
                }
            }
```

```
            },
            {
            "BeginOffset": 80,
            "EndOffset": 81,
            "Score": 0.999996,
            "GroupScore": 0.52523,
            "Text": "I",
            "Type": "PERSON",
            "MentionSentiment": {
                "Sentiment": "NEUTRAL",
                "SentimentScore": {
                "Mixed": 0,
                "Negative": 0,
                "Neutral": 1,
                "Positive": 0
                }
            }
            },
            {
            "BeginOffset": 67,
            "EndOffset": 68,
            "Score": 0.999994,
            "GroupScore": 0.999499,
            "Text": "I",
            "Type": "PERSON",
            "MentionSentiment": {
                "Sentiment": "NEUTRAL",
                "SentimentScore": {
                "Mixed": 0,
                "Negative": 0,
                "Neutral": 1,
                "Positive": 0
                }
            }
            }
        ]
        },
        {
        "DescriptiveMentionIndex": [
            0
        ],
        "Mentions": [
            {
            "BeginOffset": 75,
```

```
            "EndOffset": 78,
            "Score": 0.999978,
            "GroupScore": 1,
            "Text": "kid",
            "Type": "PERSON",
            "MentionSentiment": {
                "Sentiment": "NEUTRAL",
                "SentimentScore": {
                "Mixed": 0,
                "Negative": 0,
                "Neutral": 1,
                "Positive": 0
                }
            }
            }
        ]
        }
    ],
    "File": "SampleMovieReview2.txt",
    "Line": 0
}
```

Contenuto di `SampleMovieReview3.txt.out` con rientri di riga per una maggiore leggibilità:

```
{
    "Entities": [
        {
        "DescriptiveMentionIndex": [
            1
        ],
        "Mentions": [
            {
            "BeginOffset": 64,
            "EndOffset": 68,
            "Score": 0.992953,
            "GroupScore": 0.999814,
            "Text": "film",
            "Type": "MOVIE",
            "MentionSentiment": {
                "Sentiment": "NEUTRAL",
                "SentimentScore": {
                "Mixed": 0.000004,
                "Negative": 0.010425,
```

```
                "Neutral": 0.989543,
                "Positive": 0.000027
                }
            }
            },
            {
            "BeginOffset": 37,
            "EndOffset": 45,
            "Score": 0.999782,
            "GroupScore": 1,
            "Text": "AnyMovie",
            "Type": "ORGANIZATION",
            "MentionSentiment": {
                "Sentiment": "POSITIVE",
                "SentimentScore": {
                "Mixed": 0.000095,
                "Negative": 0.039847,
                "Neutral": 0.000673,
                "Positive": 0.959384
                }
            }
            }
        ]
        },
        {
        "DescriptiveMentionIndex": [
            0
        ],
        "Mentions": [
            {
            "BeginOffset": 47,
            "EndOffset": 50,
            "Score": 0.999991,
            "GroupScore": 1,
            "Text": "All",
            "Type": "QUANTITY",
            "MentionSentiment": {
                "Sentiment": "NEUTRAL",
                "SentimentScore": {
                "Mixed": 0.000001,
                "Negative": 0.000001,
                "Neutral": 0.999998,
                "Positive": 0
                }
```

```
            }
            }
        ]
        },
        {
        "DescriptiveMentionIndex": [
            0
        ],
        "Mentions": [
            {
            "BeginOffset": 106,
            "EndOffset": 115,
            "Score": 0.542083,
            "GroupScore": 1,
            "Text": "directors",
            "Type": "PERSON",
            "MentionSentiment": {
                "Sentiment": "NEUTRAL",
                "SentimentScore": {
                "Mixed": 0,
                "Negative": 0,
                "Neutral": 1,
                "Positive": 0
                }
            }
            }
        ]
        }
    ],
    "File": "SampleMovieReview3.txt",
    "Line": 0
}
```

Per ulteriori informazioni, consulta Analisi asincrona degli approfondimenti di Amazon Comprehend nella Guida per gli sviluppatori di Amazon Comprehend.

- Per i dettagli sull'API, consulta StartTargetedSentimentDetectionJob AWS CLICommand Reference.

`start-topics-detection-job`

Il seguente esempio di codice mostra come utilizzare`start-topics-detection-job`.

AWS CLI

Come avviare un processo di analisi del rilevamento degli argomenti

L'esempio `start-topics-detection-job` seguente avvia un processo asincrono di rilevamento degli argomenti per tutti i file che si trovano all'indirizzo specificato dal tag `--input-data-config`. Al completamento del processo, la cartella `output` viene collocata nella posizione specificata dal tag `--ouput-data-config`. `output` contiene topic-terms.csv e doc-topics.csv. Il primo file di output (topic-terms.csv) è un elenco di argomenti nella raccolta. Per ogni argomento, per impostazione predefinita l'elenco include i termini principali per argomento in base al loro peso. Il secondo file (`doc-topics.csv`) elenca i documenti associati a un argomento e la proporzione del documento interessato dall'argomento.

```
aws comprehend start-topics-detection-job \
    --job-name example_topics_detection_job \
    --language-code en \
    --input-data-config "S3Uri=s3://amzn-s3-demo-bucket/" \
    --output-data-config "S3Uri=s3://amzn-s3-demo-destination-bucket/testfolder/" \
    --data-access-role-arn arn:aws:iam::111122223333:role/service-role/
AmazonComprehendServiceRole-example-role \
    --language-code en
```

Output:

```
{
    "JobId": "123456abcdeb0e11022f22a11EXAMPLE",
    "JobArn": "arn:aws:comprehend:us-west-2:111122223333:key-phrases-detection-
job/123456abcdeb0e11022f22a11EXAMPLE",
    "JobStatus": "SUBMITTED"
}
```

Per ulteriori informazioni, consulta Modellazione degli argomenti nella Guida per gli sviluppatori di Amazon Comprehend.

- Per i dettagli sull'API, consulta StartTopicsDetectionJob AWS CLICommand Reference.

`stop-dominant-language-detection-job`

Il seguente esempio di codice mostra come utilizzare`stop-dominant-language-detection-job`.

AWS CLI

Come arrestare un processo asincrono di rilevamento della lingua dominante

L'esempio `stop-dominant-language-detection-job` seguente arresta un processo di rilevamento asincrono della lingua dominante in corso. Se lo stato corrente del processo è `IN_PROGRESS`, il processo viene contrassegnato per la terminazione e viene impostato lo `STOP_REQUESTED`. Se il processo viene completato prima di poter essere arrestato, viene messo in stato `COMPLETED`.

```
aws comprehend stop-dominant-language-detection-job \
    --job-id 123456abcdeb0e11022f22a11EXAMPLE
```

Output:

```
{
    "JobId": "123456abcdeb0e11022f22a11EXAMPLE,
    "JobStatus": "STOP_REQUESTED"
}
```

Per ulteriori informazioni, consulta Analisi asincrona degli approfondimenti di Amazon Comprehend nella Guida per gli sviluppatori di Amazon Comprehend.

- Per i dettagli sull'API, consulta StopDominantLanguageDetectionJob AWS CLICommand Reference.

`stop-entities-detection-job`

Il seguente esempio di codice mostra come utilizzare`stop-entities-detection-job`.

AWS CLI

Come arrestare un processo asincrono di rilevamento delle entità

L'esempio `stop-entities-detection-job` seguente arresta un processo di rilevamento asincrono delle entità in corso. Se lo stato corrente del processo è `IN_PROGRESS`, il processo viene contrassegnato per la terminazione e viene impostato lo `STOP_REQUESTED`. Se il processo viene completato prima di poter essere arrestato, viene messo in stato `COMPLETED`.

```
aws comprehend stop-entities-detection-job \
```

```
    --job-id 123456abcdeb0e11022f22a11EXAMPLE
```

Output:

```
{
    "JobId": "123456abcdeb0e11022f22a11EXAMPLE,
    "JobStatus": "STOP_REQUESTED"
}
```

Per ulteriori informazioni, consulta Analisi asincrona degli approfondimenti di Amazon Comprehend nella Guida per gli sviluppatori di Amazon Comprehend.

- Per i dettagli sull'API, consulta StopEntitiesDetectionJob AWS CLICommand Reference.

`stop-events-detection-job`

Il seguente esempio di codice mostra come utilizzare`stop-events-detection-job`.

AWS CLI

Come arrestare un processo asincrono di rilevamento degli eventi

L'esempio `stop-events-detection-job` seguente arresta un processo di rilevamento asincrono degli eventi in corso. Se lo stato corrente del processo è `IN_PROGRESS`, il processo viene contrassegnato per la terminazione e viene impostato lo `STOP_REQUESTED`. Se il processo viene completato prima di poter essere arrestato, viene messo in stato `COMPLETED`.

```
aws comprehend stop-events-detection-job \
    --job-id 123456abcdeb0e11022f22a11EXAMPLE
```

Output:

```
{
    "JobId": "123456abcdeb0e11022f22a11EXAMPLE,
    "JobStatus": "STOP_REQUESTED"
}
```

Per ulteriori informazioni, consulta Analisi asincrona degli approfondimenti di Amazon Comprehend nella Guida per gli sviluppatori di Amazon Comprehend.

- Per i dettagli sull'API, consulta StopEventsDetectionJob AWS CLICommand Reference.

`stop-key-phrases-detection-job`

Il seguente esempio di codice mostra come utilizzare`stop-key-phrases-detection-job`.

AWS CLI

Come arrestare un processo asincrono di rilevamento delle frasi chiave

L'esempio `stop-key-phrases-detection-job` seguente arresta un processo di rilevamento asincrono delle frasi chiave in corso. Se lo stato corrente del processo è `IN_PROGRESS`, il processo viene contrassegnato per la terminazione e viene impostato lo `STOP_REQUESTED`. Se il processo viene completato prima di poter essere arrestato, viene messo in stato `COMPLETED`.

```
aws comprehend stop-key-phrases-detection-job \
    --job-id 123456abcdeb0e11022f22a11EXAMPLE
```

Output:

```
{
    "JobId": "123456abcdeb0e11022f22a11EXAMPLE,
    "JobStatus": "STOP_REQUESTED"
}
```

Per ulteriori informazioni, consulta Analisi asincrona degli approfondimenti di Amazon Comprehend nella Guida per gli sviluppatori di Amazon Comprehend.

- Per i dettagli sull'API, consulta StopKeyPhrasesDetectionJob AWS CLICommand Reference.

`stop-pii-entities-detection-job`

Il seguente esempio di codice mostra come utilizzare`stop-pii-entities-detection-job`.

AWS CLI

Come arrestare un processo asincrono di rilevamento delle entità pii

L'esempio `stop-pii-entities-detection-job` seguente arresta un processo di rilevamento asincrono delle entità pii in corso. Se lo stato corrente del processo è `IN_PROGRESS`, il processo viene contrassegnato per la terminazione e viene impostato lo `STOP_REQUESTED`. Se il processo viene completato prima di poter essere arrestato, viene messo in stato `COMPLETED`.

```
aws comprehend stop-pii-entities-detection-job \
```

```
    --job-id 123456abcdeb0e11022f22a11EXAMPLE
```

Output:

```
{
    "JobId": "123456abcdeb0e11022f22a11EXAMPLE,
    "JobStatus": "STOP_REQUESTED"
}
```

Per ulteriori informazioni, consulta Analisi asincrona degli approfondimenti di Amazon Comprehend nella Guida per gli sviluppatori di Amazon Comprehend.

- Per i dettagli sull'API, consulta StopPiiEntitiesDetectionJob AWS CLICommand Reference.

`stop-sentiment-detection-job`

Il seguente esempio di codice mostra come utilizzare`stop-sentiment-detection-job`.

AWS CLI

Come arrestare un processo asincrono di rilevamento del sentiment

L'esempio `stop-sentiment-detection-job` seguente arresta un processo di rilevamento asincrono del sentiment in corso. Se lo stato corrente del processo è `IN_PROGRESS`, il processo viene contrassegnato per la terminazione e viene impostato lo `STOP_REQUESTED`. Se il processo viene completato prima di poter essere arrestato, viene messo in stato `COMPLETED`.

```
aws comprehend stop-sentiment-detection-job \
    --job-id 123456abcdeb0e11022f22a11EXAMPLE
```

Output:

```
{
    "JobId": "123456abcdeb0e11022f22a11EXAMPLE,
    "JobStatus": "STOP_REQUESTED"
}
```

Per ulteriori informazioni, consulta Analisi asincrona degli approfondimenti di Amazon Comprehend nella Guida per gli sviluppatori di Amazon Comprehend.

- Per i dettagli sull'API, consulta StopSentimentDetectionJob AWS CLICommand Reference.

`stop-targeted-sentiment-detection-job`

Il seguente esempio di codice mostra come utilizzare`stop-targeted-sentiment-detection-job`.

AWS CLI

Come arrestare un processo asincrono di rilevamento del sentiment mirato

L'esempio `stop-targeted-sentiment-detection-job` seguente arresta un processo di rilevamento asincrono del sentiment mirato in corso. Se lo stato corrente del processo è `IN_PROGRESS`, il processo viene contrassegnato per la terminazione e viene impostato lo `STOP_REQUESTED`. Se il processo viene completato prima di poter essere arrestato, viene messo in stato `COMPLETED`.

```
aws comprehend stop-targeted-sentiment-detection-job \
    --job-id 123456abcdeb0e11022f22a11EXAMPLE
```

Output:

```
{
    "JobId": "123456abcdeb0e11022f22a11EXAMPLE,
    "JobStatus": "STOP_REQUESTED"
}
```

Per ulteriori informazioni, consulta Analisi asincrona degli approfondimenti di Amazon Comprehend nella Guida per gli sviluppatori di Amazon Comprehend.

- Per i dettagli sull'API, consulta StopTargetedSentimentDetectionJob AWS CLICommand Reference.

`stop-training-document-classifier`

Il seguente esempio di codice mostra come utilizzare`stop-training-document-classifier`.

AWS CLI

Come arrestare l'addestramento di un modello di classificazione dei documenti

L'esempio `stop-training-document-classifier` seguente arresta l'addestramento di un modello di classificazione di documenti ancora in corso.

```
aws comprehend stop-training-document-classifier
    --document-classifier-arn arn:aws:comprehend:us-west-2:111122223333:document-classifier/example-classifier
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Creazione e gestione di modelli personalizzati nella Guida per gli sviluppatori di Amazon Comprehend.

- Per i dettagli sull'API, consulta StopTrainingDocumentClassifier AWS CLICommand Reference.

`stop-training-entity-recognizer`

Il seguente esempio di codice mostra come utilizzare`stop-training-entity-recognizer`.

AWS CLI

Come arrestare l'addestramento di un modello di riconoscimento delle entità

L'esempio `stop-training-entity-recognizer` seguente arresta l'addestramento di un modello di riconoscimento delle entità ancora in corso.

```
aws comprehend stop-training-entity-recognizer
    --entity-recognizer-arn "arn:aws:comprehend:us-west-2:111122223333:entity-recognizer/examplerecognizer1"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Creazione e gestione di modelli personalizzati nella Guida per gli sviluppatori di Amazon Comprehend.

- Per i dettagli sull'API, consulta StopTrainingEntityRecognizer AWS CLICommand Reference.

`tag-resource`

Il seguente esempio di codice mostra come utilizzare`tag-resource`.

AWS CLI

Esempio 1: come aggiungere un tag a una risorsa

L'esempio `tag-resource` seguente aggiunge un singolo tag a una risorsa Amazon Comprehend.

```
aws comprehend tag-resource \
    --resource-arn arn:aws:comprehend:us-west-2:111122223333:document-classifier/
example-classifier/version/1 \
    --tags Key=Location,Value=Seattle
```

Questo comando non ha output.

Per ulteriori informazioni, consulta Tagging delle risorse nella Guida per gli sviluppatori di Amazon Comprehend.

Esempio 2: come aggiungere più tag a una risorsa

L'esempio `tag-resource` seguente aggiunge più tag a una risorsa Amazon Comprehend.

```
aws comprehend tag-resource \
    --resource-arn "arn:aws:comprehend:us-west-2:111122223333:document-classifier/
example-classifier/version/1" \
    --tags Key=location,Value=Seattle Key=Department,Value=Finance
```

Questo comando non ha output.

Per ulteriori informazioni, consulta Tagging delle risorse nella Guida per gli sviluppatori di Amazon Comprehend.

- Per i dettagli sull'API, consulta TagResource AWS CLICommand Reference.

`untag-resource`

Il seguente esempio di codice mostra come utilizzare`untag-resource`.

AWS CLI

Esempio 1: come rimuovere un singolo tag da una risorsa

L'esempio `untag-resource` seguente rimuove un singolo tag da una risorsa Amazon Comprehend.

```
aws comprehend untag-resource \
```

```
    --resource-arn arn:aws:comprehend:us-west-2:111122223333:document-classifier/
example-classifier/version/1
    --tag-keys Location
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Tagging delle risorse nella Guida per gli sviluppatori di Amazon Comprehend.

Esempio 2: come rimuovere più tag da una risorsa

L'esempio `untag-resource` seguente rimuove più tag da una risorsa Amazon Comprehend.

```
aws comprehend untag-resource \
    --resource-arn arn:aws:comprehend:us-west-2:111122223333:document-classifier/
example-classifier/version/1
    --tag-keys Location Department
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Tagging delle risorse nella Guida per gli sviluppatori di Amazon Comprehend.

- Per i dettagli sull'API, consulta UntagResource AWS CLICommand Reference.

`update-endpoint`

Il seguente esempio di codice mostra come utilizzare`update-endpoint`.

AWS CLI

Esempio 1: come aggiornare le unità di inferenza di un endpoint

L'esempio `update-endpoint` seguente aggiorna le informazioni su un endpoint. In questo esempio, viene aumentato il numero di unità di inferenza.

```
aws comprehend update-endpoint \
    --endpoint-arn arn:aws:comprehend:us-west-2:111122223333:document-classifier-
endpoint/example-classifier-endpoint
    --desired-inference-units 2
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Gestione degli endpoint di Amazon Comprehend nella Guida per gli sviluppatori di Amazon Comprehend.

Esempio 2: come aggiornare il modello attivo di un endpoint

L'esempio `update-endpoint` seguente aggiorna le informazioni su un endpoint. In questo esempio, il modello attivo viene modificato.

```
aws comprehend update-endpoint \
    --endpoint-arn arn:aws:comprehend:us-west-2:111122223333:document-classifier-endpoint/example-classifier-endpoint
    --active-model-arn arn:aws:comprehend:us-west-2:111122223333:document-classifier/example-classifier-new
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Gestione degli endpoint di Amazon Comprehend nella Guida per gli sviluppatori di Amazon Comprehend.

- Per i dettagli sull'API, consulta UpdateEndpoint AWS CLICommand Reference.

`update-flywheel`

Il seguente esempio di codice mostra come utilizzare`update-flywheel`.

AWS CLI

Come aggiornare la configurazione di un flywheel

L'esempio `update-flywheel` seguente aggiorna una configurazione del flywheel. In questo esempio, il modello attivo per il flywheel viene aggiornato.

```
aws comprehend update-flywheel \
    --flywheel-arn arn:aws:comprehend:us-west-2:111122223333:flywheel/example-flywheel-1 \
    --active-model-arn arn:aws:comprehend:us-west-2:111122223333:document-classifier/example-classifier/version/new-example-classifier-model
```

Output:

```
{
    "FlywheelProperties": {
        "FlywheelArn": "arn:aws:comprehend:us-west-2:111122223333:flywheel/flywheel-
entity",
        "ActiveModelArn": "arn:aws:comprehend:us-west-2:111122223333:document-
classifier/example-classifier/version/new-example-classifier-model",
        "DataAccessRoleArn": "arn:aws:iam::111122223333:role/service-role/
AmazonComprehendServiceRole-example-role",
        "TaskConfig": {
            "LanguageCode": "en",
            "DocumentClassificationConfig": {
                "Mode": "MULTI_CLASS"
            }
        },
        "DataLakeS3Uri": "s3://amzn-s3-demo-bucket/flywheel-entity/
schemaVersion=1/20230616T200543Z/",
        "DataSecurityConfig": {},
        "Status": "ACTIVE",
        "ModelType": "DOCUMENT_CLASSIFIER",
        "CreationTime": "2023-06-16T20:05:43.242000+00:00",
        "LastModifiedTime": "2023-06-19T04:00:43.027000+00:00",
        "LatestFlywheelIteration": "20230619T040032Z"
    }
}
```

Per ulteriori informazioni, consulta Panoramica del volano dei dati nella Guida per gli sviluppatori di Amazon Comprehend.

- Per i dettagli sull'API, consulta UpdateFlywheel AWS CLICommand Reference.

Esempi di utilizzo di Amazon Comprehend Medical AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando Amazon Comprehend Medical. AWS Command Line Interface

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`describe-entities-detection-v2-job`

Il seguente esempio di codice mostra come usare`describe-entities-detection-v2-job`.

AWS CLI

Come descrivere un processo di rilevamento delle entità

L'esempio `describe-entities-detection-v2-job` seguente mostra le proprietà associate a un processo di rilevamento delle entità asincrono.

```
aws comprehendmedical describe-entities-detection-v2-job \
    --job-id "ab9887877365fe70299089371c043b96"
```

Output:

```
{
    "ComprehendMedicalAsyncJobProperties": {
        "JobId": "ab9887877365fe70299089371c043b96",
        "JobStatus": "COMPLETED",
        "SubmitTime": "2020-03-18T21:20:15.614000+00:00",
        "EndTime": "2020-03-18T21:27:07.350000+00:00",
        "ExpirationTime": "2020-07-16T21:20:15+00:00",
        "InputDataConfig": {
            "S3Bucket": "comp-med-input",
            "S3Key": ""
        },
        "OutputDataConfig": {
            "S3Bucket": "comp-med-output",
            "S3Key": "867139942017-EntitiesDetection-
ab9887877365fe70299089371c043b96/"
        },
        "LanguageCode": "en",
        "DataAccessRoleArn": "arn:aws:iam::867139942017:role/
ComprehendMedicalBatchProcessingRole",
        "ModelVersion": "DetectEntitiesModelV20190930"
```

```
    }
}
```

Per ulteriori informazioni, consulta Batch APIs nella Amazon Comprehend Medical Developer Guide.

- Per i dettagli sull'API, consulta DescribeEntitiesDetectionV2Job in AWS CLI Command Reference.

`describe-icd10-cm-inference-job`

Il seguente esempio di codice mostra come utilizzare. `describe-icd10-cm-inference-job`

AWS CLI

Come descrivere un processo di inferenza di ICD-10-CM

L'esempio `describe-icd10-cm-inference-job` seguente descrive le proprietà del processo di inferenza richiesto con il job-id specificato.

```
aws comprehendmedical describe-icd10-cm-inference-job \
    --job-id "5780034166536cdb52ffa3295a1b00a7"
```

Output:

```
{
    "ComprehendMedicalAsyncJobProperties": {
        "JobId": "5780034166536cdb52ffa3295a1b00a7",
        "JobStatus": "COMPLETED",
        "SubmitTime": "2020-05-18T21:20:15.614000+00:00",
        "EndTime": "2020-05-18T21:27:07.350000+00:00",
        "ExpirationTime": "2020-09-16T21:20:15+00:00",
        "InputDataConfig": {
            "S3Bucket": "comp-med-input",
            "S3Key": "AKIAIOSFODNN7EXAMPLE"
        },
        "OutputDataConfig": {
            "S3Bucket": "comp-med-output",
            "S3Key": "AKIAIOSFODNN7EXAMPLE"
        },
        "LanguageCode": "en",
```

```
        "DataAccessRoleArn": "arn:aws:iam::867139942017:role/
ComprehendMedicalBatchProcessingRole",
        "ModelVersion":  "0.1.0"
    }
}
```

Per ulteriori informazioni, consulta Analisi in batch del collegamento delle rappresentazioni formali nella Guida per gli sviluppatori di Amazon Comprehend Medical.

- Per i dettagli sull'API, vedi DescribeIcd10 CmInferenceJob in AWS CLI Command Reference.

`describe-phi-detection-job`

Il seguente esempio di codice mostra come utilizzare`describe-phi-detection-job`.

AWS CLI

Come descrivere un processo di rilevamento di PHI

L'esempio `describe-phi-detection-job` seguente mostra le proprietà associate a un processo di rilevamento di informazioni sanitarie protette (PHI) asincrono.

```
aws comprehendmedical describe-phi-detection-job \
    --job-id "4750034166536cdb52ffa3295a1b00a3"
```

Output:

```
{
    "ComprehendMedicalAsyncJobProperties": {
        "JobId": "4750034166536cdb52ffa3295a1b00a3",
        "JobStatus": "COMPLETED",
        "SubmitTime": "2020-03-19T20:38:37.594000+00:00",
        "EndTime": "2020-03-19T20:45:07.894000+00:00",
        "ExpirationTime": "2020-07-17T20:38:37+00:00",
        "InputDataConfig": {
            "S3Bucket": "comp-med-input",
            "S3Key": ""
        },
        "OutputDataConfig": {
            "S3Bucket": "comp-med-output",
            "S3Key": "867139942017-PHIDetection-4750034166536cdb52ffa3295a1b00a3/"
```

```
        },
        "LanguageCode": "en",
        "DataAccessRoleArn": "arn:aws:iam::867139942017:role/
ComprehendMedicalBatchProcessingRole",
        "ModelVersion": "PHIModelV20190903"
    }
}
```

Per ulteriori informazioni, consulta Batch APIs nella Amazon Comprehend Medical Developer Guide.

- Per i dettagli sull'API, consulta DescribePhiDetectionJob AWS CLICommand Reference.

`describe-rx-norm-inference-job`

Il seguente esempio di codice mostra come utilizzare`describe-rx-norm-inference-job`.

AWS CLI

Descrivere un lavoro di RxNorm inferenza

L'esempio `describe-rx-norm-inference-job` seguente descrive le proprietà del processo di inferenza richiesto con il job-id specificato.

```
aws comprehendmedical describe-rx-norm-inference-job \
    --job-id "eg8199877365fc70299089371c043b96"
```

Output:

```
{
    "ComprehendMedicalAsyncJobProperties": {
        "JobId": "g8199877365fc70299089371c043b96",
        "JobStatus": "COMPLETED",
        "SubmitTime": "2020-05-18T21:20:15.614000+00:00",
        "EndTime": "2020-05-18T21:27:07.350000+00:00",
        "ExpirationTime": "2020-09-16T21:20:15+00:00",
        "InputDataConfig": {
            "S3Bucket": "comp-med-input",
            "S3Key": "AKIAIOSFODNN7EXAMPLE"
        },
        "OutputDataConfig": {
```

```
            "S3Bucket": "comp-med-output",
            "S3Key": "AKIAIOSFODNN7EXAMPLE"
        },
        "LanguageCode": "en",
        "DataAccessRoleArn": "arn:aws:iam::867139942017:role/
ComprehendMedicalBatchProcessingRole",
        "ModelVersion": "0.0.0"
    }
}
```

Per ulteriori informazioni, consulta Analisi in batch del collegamento delle rappresentazioni formali nella Guida per gli sviluppatori di Amazon Comprehend Medical.

- Per i dettagli sull'API, consulta DescribeRxNormInferenceJob AWS CLICommand Reference.

`describe-snomedct-inference-job`

Il seguente esempio di codice mostra come utilizzare`describe-snomedct-inference-job`.

AWS CLI

Come descrivere un processo di inferenza di SNOMED CT

L'esempio `describe-snomedct-inference-job` seguente descrive le proprietà del processo di inferenza richiesto con il job-id specificato.

```
aws comprehendmedical describe-snomedct-inference-job \
    --job-id "2630034166536cdb52ffa3295a1b00a7"
```

Output:

```
{
    "ComprehendMedicalAsyncJobProperties": {
        "JobId": "2630034166536cdb52ffa3295a1b00a7",
        "JobStatus": "COMPLETED",
        "SubmitTime": "2021-12-18T21:20:15.614000+00:00",
        "EndTime": "2021-12-18T21:27:07.350000+00:00",
        "ExpirationTime": "2022-05-16T21:20:15+00:00",
        "InputDataConfig": {
            "S3Bucket": "comp-med-input",
            "S3Key": "AKIAIOSFODNN7EXAMPLE"
        },
```

```
        "OutputDataConfig": {
            "S3Bucket": "comp-med-output",
            "S3Key": "AKIAIOSFODNN7EXAMPLE"
        },
        "LanguageCode": "en",
        "DataAccessRoleArn": "arn:aws:iam::867139942017:role/
ComprehendMedicalBatchProcessingRole",
        "ModelVersion":  "0.1.0"
    }
}
```

Per ulteriori informazioni, consulta Analisi in batch del collegamento delle rappresentazioni formali nella Guida per gli sviluppatori di Amazon Comprehend Medical.

- Per i dettagli sull'API, consulta DescribeSnomedctInferenceJob AWS CLICommand Reference.

`detect-entities-v2`

Il seguente esempio di codice mostra come utilizzare`detect-entities-v2`.

AWS CLI

Esempio 1: come rilevare le entità direttamente dal testo

L'esempio `detect-entities-v2` seguente mostra le entità rilevate e le etichetta in base al tipo, direttamente dal testo di input.

```
aws comprehendmedical detect-entities-v2 \
    --text "Sleeping trouble on present dosage of Clonidine. Severe rash on face and
 leg, slightly itchy."
```

Output:

```
{
    "Id": 0,
    "BeginOffset": 38,
    "EndOffset": 47,
    "Score": 0.9942955374717712,
    "Text": "Clonidine",
    "Category": "MEDICATION",
    "Type": "GENERIC_NAME",
    "Traits": []
```

```
}
```

Per ulteriori informazioni, consulta Rilevare le entità (versione 2) nella Guida per gli sviluppatori di Amazon Comprehend Medical.

Esempio 2: come rilevare entità da un percorso di file

L'esempio `detect-entities-v2` seguente mostra le entità rilevate e le etichetta in base al tipo da un percorso di file.

```
aws comprehendmedical detect-entities-v2 \
    --text file://medical_entities.txt
```

Contenuto di `medical_entities.txt`:

```
{
    "Sleeping trouble on present dosage of Clonidine. Severe rash on face and leg,
 slightly itchy."
}
```

Output:

```
{
    "Id": 0,
    "BeginOffset": 38,
    "EndOffset": 47,
    "Score": 0.9942955374717712,
    "Text": "Clonidine",
    "Category": "MEDICATION",
    "Type": "GENERIC_NAME",
    "Traits": []
}
```

Per ulteriori informazioni, consulta Rilevare le entità (versione 2) nella Guida per gli sviluppatori di Amazon Comprehend Medical.

- Per i dettagli sull'API, consulta DetectEntitiesV2 in AWS CLI Command Reference.

`detect-phi`

Il seguente esempio di codice mostra come utilizzare`detect-phi`.

AWS CLI

Esempio 1: come rilevare informazioni sanitarie protette (PHI) direttamente dal testo

L'esempio `detect-phi` seguente mostra le entità di informazioni sanitarie protette (PHI) rilevate direttamente dal testo di input.

```
aws comprehendmedical detect-phi \
    --text "Patient Carlos Salazar presented with rash on his upper extremities and dry cough. He lives at 100 Main Street, Anytown, USA where he works from his home as a carpenter."
```

Output:

```
{
    "Entities": [
        {
            "Id": 0,
            "BeginOffset": 8,
            "EndOffset": 21,
            "Score": 0.9914507269859314,
            "Text": "Carlos Salazar",
            "Category": "PROTECTED_HEALTH_INFORMATION",
            "Type": "NAME",
            "Traits": []
        },
        {
            "Id": 1,
            "BeginOffset": 94,
            "EndOffset": 109,
            "Score": 0.871849775314331,
            "Text": "100 Main Street, Anytown, USA",
            "Category": "PROTECTED_HEALTH_INFORMATION",
            "Type": "ADDRESS",
            "Traits": []
        },
        {
            "Id": 2,
            "BeginOffset": 145,
            "EndOffset": 154,
            "Score": 0.8302185535430908,
            "Text": "carpenter",
            "Category": "PROTECTED_HEALTH_INFORMATION",
```

```
            "Type": "PROFESSION",
            "Traits": []
        }
    ],
    "ModelVersion": "0.0.0"
}
```

Per ulteriori informazioni, consulta Rilevare le informazioni sanitarie protette nella Guida per gli sviluppatori di Amazon Comprehend Medical.

Esempio 2: come rilevare informazioni sanitarie protette (PHI) direttamente da un percorso di file

L'esempio `detect-phi` seguente mostra le entità di informazioni sanitarie protette (PHI) rilevate da un percorso di file.

```
aws comprehendmedical detect-phi \
    --text file://phi.txt
```

Contenuto di `phi.txt`:

```
"Patient Carlos Salazar presented with a rash on his upper extremities and a dry
 cough. He lives at 100 Main Street, Anytown, USA, where he works from his home as a
 carpenter."
```

Output:

```
{
    "Entities": [
        {
            "Id": 0,
            "BeginOffset": 8,
            "EndOffset": 21,
            "Score": 0.9914507269859314,
            "Text": "Carlos Salazar",
            "Category": "PROTECTED_HEALTH_INFORMATION",
            "Type": "NAME",
            "Traits": []
        },
        {
            "Id": 1,
            "BeginOffset": 94,
```

```
            "EndOffset": 109,
            "Score": 0.871849775314331,
            "Text": "100 Main Street, Anytown, USA",
            "Category": "PROTECTED_HEALTH_INFORMATION",
            "Type": "ADDRESS",
            "Traits": []
        },
        {
            "Id": 2,
            "BeginOffset": 145,
            "EndOffset": 154,
            "Score": 0.8302185535430908,
            "Text": "carpenter",
            "Category": "PROTECTED_HEALTH_INFORMATION",
            "Type": "PROFESSION",
            "Traits": []
        }
    ],
    "ModelVersion": "0.0.0"
}
```

Per ulteriori informazioni, consulta Rilevare le informazioni sanitarie protette nella Guida per gli sviluppatori di Amazon Comprehend Medical.

- Per i dettagli sull'API, consulta DetectPhi AWS CLICommand Reference.

`infer-icd10-cm`

Il seguente esempio di codice mostra come utilizzare`infer-icd10-cm`.

AWS CLI

Esempio 1: come rilevare entità relative alle condizioni mediche e collegarle alla rappresentazione formale di ICD-10-CM direttamente dal testo

L'esempio `infer-icd10-cm` seguente etichetta le entità di condizioni mediche rilevate e le collega ai codici presenti nella versione 2019 della Classificazione internazionale delle malattie - modificazione clinica (ICD-10-CM).

```
aws comprehendmedical infer-icd10-cm \
    --text "The patient complains of abdominal pain, has a long-standing history of
 diabetes treated with Micronase daily."
```

Output:


```
{
    "Entities": [
        {
            "Id": 0,
            "Text": "abdominal pain",
            "Category": "MEDICAL_CONDITION",
            "Type": "DX_NAME",
            "Score": 0.9475538730621338,
            "BeginOffset": 28,
            "EndOffset": 42,
            "Attributes": [],
            "Traits": [
                {
                    "Name": "SYMPTOM",
                    "Score": 0.6724207401275635
                }
            ],
            "ICD10CMConcepts": [
                {
                    "Description": "Unspecified abdominal pain",
                    "Code": "R10.9",
                    "Score": 0.6904221177101135
                },
                {
                    "Description": "Epigastric pain",
                    "Code": "R10.13",
                    "Score": 0.1364113688468933
                },
                {
                    "Description": "Generalized abdominal pain",
                    "Code": "R10.84",
                    "Score": 0.12508003413677216
                },
                {
                    "Description": "Left lower quadrant pain",
                    "Code": "R10.32",
                    "Score": 0.10063883662223816
                },
                {
                    "Description": "Lower abdominal pain, unspecified",
                    "Code": "R10.30",
                    "Score": 0.09933677315711975
```

```
                }
            ]
        },
        {
            "Id": 1,
            "Text": "diabetes",
            "Category": "MEDICAL_CONDITION",
            "Type": "DX_NAME",
            "Score": 0.9899052977561951,
            "BeginOffset": 75,
            "EndOffset": 83,
            "Attributes": [],
            "Traits": [
                {
                    "Name": "DIAGNOSIS",
                    "Score": 0.9258432388305664
                }
            ],
            "ICD10CMConcepts": [
                {
                    "Description": "Type 2 diabetes mellitus without complications",
                    "Code": "E11.9",
                    "Score": 0.7158446311950684
                },
                {
                    "Description": "Family history of diabetes mellitus",
                    "Code": "Z83.3",
                    "Score": 0.5704703330993652
                },
                {
                    "Description": "Family history of other endocrine, nutritional
and metabolic diseases",
                    "Code": "Z83.49",
                    "Score": 0.19856023788452148
                },
                {
                    "Description": "Type 1 diabetes mellitus with ketoacidosis
without coma",
                    "Code": "E10.10",
                    "Score": 0.13285516202449799
                },
                {
                    "Description": "Type 2 diabetes mellitus with hyperglycemia",
                    "Code": "E11.65",
```

```
                    "Score": 0.0993388369679451
                }
            ]
        }
    ],
    "ModelVersion": "0.1.0"
}
```

Per ulteriori informazioni, consulta Infer ICD1 0-CM nella Amazon Comprehend Medical Developer Guide.

Esempio 2: come rilevare entità di condizioni mediche e collegarle alla rappresentazione formale di ICD-10-CM da un percorso di file

L'esempio `infer-icd-10-cm` seguente etichetta le entità di condizioni mediche rilevate e le collega ai codici presenti nella versione 2019 della Classificazione internazionale delle malattie - modificazione clinica (ICD-10-CM).

```
aws comprehendmedical infer-icd10-cm \
    --text file://icd10cm.txt
```

Contenuto di `icd10cm.txt`:

```
{
    "The patient complains of abdominal pain, has a long-standing history of
 diabetes treated with Micronase daily."
}
```

Output:

```
{
    "Entities": [
        {
            "Id": 0,
            "Text": "abdominal pain",
            "Category": "MEDICAL_CONDITION",
            "Type": "DX_NAME",
            "Score": 0.9475538730621338,
            "BeginOffset": 28,
            "EndOffset": 42,
            "Attributes": [],
            "Traits": [
```

```
                {
                    "Name": "SYMPTOM",
                    "Score": 0.6724207401275635
                }
            ],
            "ICD10CMConcepts": [
                {
                    "Description": "Unspecified abdominal pain",
                    "Code": "R10.9",
                    "Score": 0.6904221177101135
                },
                {
                    "Description": "Epigastric pain",
                    "Code": "R10.13",
                    "Score": 0.1364113688468933
                },
                {
                    "Description": "Generalized abdominal pain",
                    "Code": "R10.84",
                    "Score": 0.12508003413677216
                },
                {
                    "Description": "Left lower quadrant pain",
                    "Code": "R10.32",
                    "Score": 0.10063883662223816
                },
                {
                    "Description": "Lower abdominal pain, unspecified",
                    "Code": "R10.30",
                    "Score": 0.09933677315711975
                }
            ]
        },
        {
            "Id": 1,
            "Text": "diabetes",
            "Category": "MEDICAL_CONDITION",
            "Type": "DX_NAME",
            "Score": 0.9899052977561951,
            "BeginOffset": 75,
            "EndOffset": 83,
            "Attributes": [],
            "Traits": [
                {
```

```
                    "Name": "DIAGNOSIS",
                    "Score": 0.9258432388305664
                }
            ],
            "ICD10CMConcepts": [
                {
                    "Description": "Type 2 diabetes mellitus without complications",
                    "Code": "E11.9",
                    "Score": 0.7158446311950684
                },
                {
                    "Description": "Family history of diabetes mellitus",
                    "Code": "Z83.3",
                    "Score": 0.5704703330993652
                },
                {
                    "Description": "Family history of other endocrine, nutritional
 and metabolic diseases",
                    "Code": "Z83.49",
                    "Score": 0.19856023788452148
                },
                {
                    "Description": "Type 1 diabetes mellitus with ketoacidosis
 without coma",
                    "Code": "E10.10",
                    "Score": 0.13285516202449799
                },
                {
                    "Description": "Type 2 diabetes mellitus with hyperglycemia",
                    "Code": "E11.65",
                    "Score": 0.0993388369679451
                }
            ]
        }
    ],
    "ModelVersion": "0.1.0"
}
```

Per ulteriori informazioni, consulta ICD1Infer-0-CM nella Amazon Comprehend Medical Developer Guide.

- Per i dettagli sull'API, consulta InferIcd10Cm in Command Reference.AWS CLI

`infer-rx-norm`

Il seguente esempio di codice mostra come utilizzare`infer-rx-norm`.

AWS CLI

Esempio 1: per rilevare le entità del farmaco e collegarle RxNorm direttamente dal testo

L'`infer-rx-norm`esempio seguente mostra ed etichetta le entità terapeutiche rilevate e le collega agli identificatori concettuali (RxCUI) del database della National Library of Medicine. RxNorm

```
aws comprehendmedical infer-rx-norm \
    --text "Patient reports taking Levothyroxine 125 micrograms p.o. once daily, but
 denies taking Synthroid."
```

Output:

```
{
    "Entities": [
        {
            "Id": 0,
            "Text": "Levothyroxine",
            "Category": "MEDICATION",
            "Type": "GENERIC_NAME",
            "Score": 0.9996285438537598,
            "BeginOffset": 23,
            "EndOffset": 36,
            "Attributes": [
                {
                    "Type": "DOSAGE",
                    "Score": 0.9892290830612183,
                    "RelationshipScore": 0.9997978806495667,
                    "Id": 1,
                    "BeginOffset": 37,
                    "EndOffset": 51,
                    "Text": "125 micrograms",
                    "Traits": []
                },
                {
                    "Type": "ROUTE_OR_MODE",
                    "Score": 0.9988924860954285,
                    "RelationshipScore": 0.998291552066803,
```

```
                    "Id": 2,
                    "BeginOffset": 52,
                    "EndOffset": 56,
                    "Text": "p.o.",
                    "Traits": []
                },
                {
                    "Type": "FREQUENCY",
                    "Score": 0.9953463673591614,
                    "RelationshipScore": 0.9999889135360718,
                    "Id": 3,
                    "BeginOffset": 57,
                    "EndOffset": 67,
                    "Text": "once daily",
                    "Traits": []
                }
            ],
            "Traits": [],
            "RxNormConcepts": [
                {
                    "Description": "Levothyroxine Sodium 0.125 MG Oral Tablet",
                    "Code": "966224",
                    "Score": 0.9912070631980896
                },
                {
                    "Description": "Levothyroxine Sodium 0.125 MG Oral Capsule",
                    "Code": "966405",
                    "Score": 0.8698278665542603
                },
                {
                    "Description": "Levothyroxine Sodium 0.125 MG Oral Tablet
[Synthroid]",
                    "Code": "966191",
                    "Score": 0.7448257803916931
                },
                {
                    "Description": "levothyroxine",
                    "Code": "10582",
                    "Score": 0.7050482630729675
                },
                {
                    "Description": "Levothyroxine Sodium 0.125 MG Oral Tablet
[Levoxyl]",
                    "Code": "966190",
```

```
                    "Score": 0.6921631693840027
                }
            ]
        },
        {
            "Id": 4,
            "Text": "Synthroid",
            "Category": "MEDICATION",
            "Type": "BRAND_NAME",
            "Score": 0.9946461319923401,
            "BeginOffset": 86,
            "EndOffset": 95,
            "Attributes": [],
            "Traits": [
                {
                    "Name": "NEGATION",
                    "Score": 0.5167351961135864
                }
            ],
            "RxNormConcepts": [
                {
                    "Description": "Synthroid",
                    "Code": "224920",
                    "Score": 0.9462039470672607
                },
                {
                    "Description": "Levothyroxine Sodium 0.088 MG Oral Tablet
[Synthroid]",
                    "Code": "966282",
                    "Score": 0.8309829235076904
                },
                {
                    "Description": "Levothyroxine Sodium 0.125 MG Oral Tablet
[Synthroid]",
                    "Code": "966191",
                    "Score": 0.4945160448551178
                },
                {
                    "Description": "Levothyroxine Sodium 0.05 MG Oral Tablet
[Synthroid]",
                    "Code": "966247",
                    "Score": 0.3674522042274475
                },
                {
```

```
                    "Description": "Levothyroxine Sodium 0.025 MG Oral Tablet
 [Synthroid]",
                    "Code": "966158",
                    "Score": 0.2588822841644287
                }
            ]
        }
    ],
    "ModelVersion": "0.0.0"
}
```

Per ulteriori informazioni, consulta Infer RxNorm nella Amazon Comprehend Medical Developer Guide.

Esempio 2: per rilevare le entità del farmaco e collegarle RxNorm da un percorso di file.

L'`infer-rx-norm`esempio seguente mostra ed etichetta le entità terapeutiche rilevate e le collega agli identificatori concettuali (RxCUI) del database della National Library of Medicine. RxNorm

```
aws comprehendmedical infer-rx-norm \
    --text file://rxnorm.txt
```

Contenuto di `rxnorm.txt`:

```
{
    "Patient reports taking Levothyroxine 125 micrograms p.o. once daily, but denies
 taking Synthroid."
}
```

Output:

```
{
    "Entities": [
        {
            "Id": 0,
            "Text": "Levothyroxine",
            "Category": "MEDICATION",
            "Type": "GENERIC_NAME",
            "Score": 0.9996285438537598,
            "BeginOffset": 23,
            "EndOffset": 36,
```

```
            "Attributes": [
                {
                    "Type": "DOSAGE",
                    "Score": 0.9892290830612183,
                    "RelationshipScore": 0.9997978806495667,
                    "Id": 1,
                    "BeginOffset": 37,
                    "EndOffset": 51,
                    "Text": "125 micrograms",
                    "Traits": []
                },
                {
                    "Type": "ROUTE_OR_MODE",
                    "Score": 0.9988924860954285,
                    "RelationshipScore": 0.998291552066803,
                    "Id": 2,
                    "BeginOffset": 52,
                    "EndOffset": 56,
                    "Text": "p.o.",
                    "Traits": []
                },
                {
                    "Type": "FREQUENCY",
                    "Score": 0.9953463673591614,
                    "RelationshipScore": 0.9999889135360718,
                    "Id": 3,
                    "BeginOffset": 57,
                    "EndOffset": 67,
                    "Text": "once daily",
                    "Traits": []
                }
            ],
            "Traits": [],
            "RxNormConcepts": [
                {
                    "Description": "Levothyroxine Sodium 0.125 MG Oral Tablet",
                    "Code": "966224",
                    "Score": 0.9912070631980896
                },
                {
                    "Description": "Levothyroxine Sodium 0.125 MG Oral Capsule",
                    "Code": "966405",
                    "Score": 0.8698278665542603
                },
```

```
                {
                    "Description": "Levothyroxine Sodium 0.125 MG Oral Tablet
[Synthroid]",
                    "Code": "966191",
                    "Score": 0.7448257803916931
                },
                {
                    "Description": "levothyroxine",
                    "Code": "10582",
                    "Score": 0.7050482630729675
                },
                {
                    "Description": "Levothyroxine Sodium 0.125 MG Oral Tablet
[Levoxyl]",
                    "Code": "966190",
                    "Score": 0.6921631693840027
                }
            ]
        },
        {
            "Id": 4,
            "Text": "Synthroid",
            "Category": "MEDICATION",
            "Type": "BRAND_NAME",
            "Score": 0.9946461319923401,
            "BeginOffset": 86,
            "EndOffset": 95,
            "Attributes": [],
            "Traits": [
                {
                    "Name": "NEGATION",
                    "Score": 0.5167351961135864
                }
            ],
            "RxNormConcepts": [
                {
                    "Description": "Synthroid",
                    "Code": "224920",
                    "Score": 0.9462039470672607
                },
                {
                    "Description": "Levothyroxine Sodium 0.088 MG Oral Tablet
[Synthroid]",
                    "Code": "966282",
```

```
                    "Score": 0.8309829235076904
                },
                {
                    "Description": "Levothyroxine Sodium 0.125 MG Oral Tablet
 [Synthroid]",
                    "Code": "966191",
                    "Score": 0.4945160448551178
                },
                {
                    "Description": "Levothyroxine Sodium 0.05 MG Oral Tablet
 [Synthroid]",
                    "Code": "966247",
                    "Score": 0.3674522042274475
                },
                {
                    "Description": "Levothyroxine Sodium 0.025 MG Oral Tablet
 [Synthroid]",
                    "Code": "966158",
                    "Score": 0.2588822841644287
                }
            ]
        }
    ],
    "ModelVersion": "0.0.0"
}
```

Per ulteriori informazioni, consulta Infer RxNorm nella Amazon Comprehend Medical Developer Guide.

- Per i dettagli sull'API, consulta AWS CLI Command InferRxNormReference.

`infer-snomedct`

Il seguente esempio di codice mostra come utilizzare`infer-snomedct`.

AWS CLI

Esempio: come rilevare entità e collegarle alla rappresentazione formale di SNOMED CT direttamente dal testo

L'esempio `infer-snomedct` seguente rileva entità mediche e le collega ai concetti della versione 2021-03 della Systematized Nomenclature of Medicine - Clinical Terms (SNOMED CT).

```
aws comprehendmedical infer-snomedct \
    --text "The patient complains of abdominal pain, has a long-standing history of
 diabetes treated with Micronase daily."
```

Output:


```
{
    "Entities": [
        {
            "Id": 3,
            "BeginOffset": 26,
            "EndOffset": 40,
            "Score": 0.9598260521888733,
            "Text": "abdominal pain",
            "Category": "MEDICAL_CONDITION",
            "Type": "DX_NAME",
            "Traits": [
                {
                    "Name": "SYMPTOM",
                    "Score": 0.6819021701812744
                }
            ]
        },
        {
            "Id": 4,
            "BeginOffset": 73,
            "EndOffset": 81,
            "Score": 0.9905840158462524,
            "Text": "diabetes",
            "Category": "MEDICAL_CONDITION",
            "Type": "DX_NAME",
            "Traits": [
                {
                    "Name": "DIAGNOSIS",
                    "Score": 0.9255214333534241
                }
            ]
        },
        {
            "Id": 1,
            "BeginOffset": 95,
            "EndOffset": 104,
            "Score": 0.6371926665306091,
```

```
            "Text": "Micronase",
            "Category": "MEDICATION",
            "Type": "BRAND_NAME",
            "Traits": [],
            "Attributes": [
                {
                    "Type": "FREQUENCY",
                    "Score": 0.9761165380477905,
                    "RelationshipScore": 0.9984188079833984,
                    "RelationshipType": "FREQUENCY",
                    "Id": 2,
                    "BeginOffset": 105,
                    "EndOffset": 110,
                    "Text": "daily",
                    "Category": "MEDICATION",
                    "Traits": []
                }
            ]
        }
    ],
    "UnmappedAttributes": [],
    "ModelVersion": "1.0.0"
}
```

Per ulteriori informazioni, consulta InferSNOMEDCT nella Guida per gli sviluppatori di Amazon Comprehend Medical.

- Per i dettagli sull'API, consulta InferSnomedct AWS CLICommand Reference.

`list-entities-detection-v2-jobs`

Il seguente esempio di codice mostra come utilizzare`list-entities-detection-v2-jobs`.

AWS CLI

Come elencare i processi di rilevamento delle entità

L'esempio `list-entities-detection-v2-jobs` seguente elenca i processi di rilevamento asincroni correnti.

```
aws comprehendmedical list-entities-detection-v2-jobs
```

Output:

```
{
    "ComprehendMedicalAsyncJobPropertiesList": [
        {
            "JobId": "ab9887877365fe70299089371c043b96",
            "JobStatus": "COMPLETED",
            "SubmitTime": "2020-03-19T20:38:37.594000+00:00",
            "EndTime": "2020-03-19T20:45:07.894000+00:00",
            "ExpirationTime": "2020-07-17T20:38:37+00:00",
            "InputDataConfig": {
                "S3Bucket": "comp-med-input",
                "S3Key": ""
            },
            "OutputDataConfig": {
                "S3Bucket": "comp-med-output",
                "S3Key": "867139942017-EntitiesDetection-
ab9887877365fe70299089371c043b96/"
            },
            "LanguageCode": "en",
            "DataAccessRoleArn": "arn:aws:iam::867139942017:role/
ComprehendMedicalBatchProcessingRole",
            "ModelVersion": "DetectEntitiesModelV20190930"
        }
    ]
}
```

Per ulteriori informazioni, consulta Batch APIs nella Amazon Comprehend Medical Developer Guide.

- Per i dettagli sull'API, consulta ListEntitiesDetectionV2Jobs in AWS CLI Command Reference.

`list-icd10-cm-inference-jobs`

Il seguente esempio di codice mostra come utilizzare. `list-icd10-cm-inference-jobs`

AWS CLI

Come elencare tutti i processi di inferenza di ICD-10-CM correnti

L'esempio seguente mostra come l'operazione `list-icd10-cm-inference-jobs` restituisce un elenco dei processi di inferenza in batch di ICD-10-CM asincroni attualmente in corso.

```
aws comprehendmedical list-icd10-cm-inference-jobs
```

Output:

```
{
    "ComprehendMedicalAsyncJobPropertiesList": [
        {
            "JobId": "5780034166536cdb52ffa3295a1b00a7",
            "JobStatus": "COMPLETED",
            "SubmitTime": "2020-05-19T20:38:37.594000+00:00",
            "EndTime": "2020-05-19T20:45:07.894000+00:00",
            "ExpirationTime": "2020-09-17T20:38:37+00:00",
            "InputDataConfig": {
                "S3Bucket": "comp-med-input",
                "S3Key": "AKIAIOSFODNN7EXAMPLE"
            },
            "OutputDataConfig": {
                "S3Bucket": "comp-med-output",
                "S3Key": "AKIAIOSFODNN7EXAMPLE"
            },
            "LanguageCode": "en",
            "DataAccessRoleArn": "arn:aws:iam::867139942017:role/
ComprehendMedicalBatchProcessingRole",
            "ModelVersion":  "0.1.0"
        }
    ]
}
```

Per ulteriori informazioni, consulta Analisi in batch del collegamento delle rappresentazioni formali nella Guida per gli sviluppatori di Amazon Comprehend Medical.

- Per i dettagli sull'API, vedi ListIcd10 CmInferenceJobs in AWS CLI Command Reference.

`list-phi-detection-jobs`

Il seguente esempio di codice mostra come utilizzare`list-phi-detection-jobs`.

AWS CLI

Come elencare i processi di rilevamento di informazioni sanitarie protette (PHI)

L'esempio `list-phi-detection-jobs` seguente elenca i processi di rilevamento di informazioni sanitarie protette (PHI) attuali.

```
aws comprehendmedical list-phi-detection-jobs
```

Output:

```
{
    "ComprehendMedicalAsyncJobPropertiesList": [
        {
            "JobId": "4750034166536cdb52ffa3295a1b00a3",
            "JobStatus": "COMPLETED",
            "SubmitTime": "2020-03-19T20:38:37.594000+00:00",
            "EndTime": "2020-03-19T20:45:07.894000+00:00",
            "ExpirationTime": "2020-07-17T20:38:37+00:00",
            "InputDataConfig": {
                "S3Bucket": "comp-med-input",
                "S3Key": ""
            },
            "OutputDataConfig": {
                "S3Bucket": "comp-med-output",
                "S3Key": "867139942017-
PHIDetection-4750034166536cdb52ffa3295a1b00a3/"
            },
            "LanguageCode": "en",
            "DataAccessRoleArn": "arn:aws:iam::867139942017:role/
ComprehendMedicalBatchProcessingRole",
            "ModelVersion": "PHIModelV20190903"
        }
    ]
}
```

Per ulteriori informazioni, consulta Batch APIs nella Amazon Comprehend Medical Developer Guide.

- Per i dettagli sull'API, consulta ListPhiDetectionJobs AWS CLICommand Reference.

`list-rx-norm-inference-jobs`

Il seguente esempio di codice mostra come utilizzare`list-rx-norm-inference-jobs`.

AWS CLI

Come elencare tutti i processi di inferenza di Rx-Norm correnti

L'esempio seguente mostra come `list-rx-norm-inference-jobs` restituisce un elenco dei processi di inferenza in batch di Rx-Norm asincroni attualmente in corso.

```
aws comprehendmedical list-rx-norm-inference-jobs
```

Output:

```
{
    "ComprehendMedicalAsyncJobPropertiesList": [
        {
            "JobId": "4980034166536cfb52gga3295a1b00a3",
            "JobStatus": "COMPLETED",
            "SubmitTime": "2020-05-19T20:38:37.594000+00:00",
            "EndTime": "2020-05-19T20:45:07.894000+00:00",
            "ExpirationTime": "2020-09-17T20:38:37+00:00",
            "InputDataConfig": {
                "S3Bucket": "comp-med-input",
                "S3Key": "AKIAIOSFODNN7EXAMPLE"
            },
            "OutputDataConfig": {
                "S3Bucket": "comp-med-output",
                "S3Key": "AKIAIOSFODNN7EXAMPLE"
            },
            "LanguageCode": "en",
            "DataAccessRoleArn": "arn:aws:iam::867139942017:role/
ComprehendMedicalBatchProcessingRole",
            "ModelVersion": "0.0.0"
        }
    ]
}
```

Per ulteriori informazioni, consulta Analisi in batch del collegamento delle rappresentazioni formali nella Guida per gli sviluppatori di Amazon Comprehend Medical.

- Per i dettagli sull'API, consulta ListRxNormInferenceJobs AWS CLICommand Reference.

`list-snomedct-inference-jobs`

Il seguente esempio di codice mostra come utilizzare`list-snomedct-inference-jobs`.

AWS CLI

Come elencare tutti i processi di inferenza di SNOMED CT

L'esempio seguente mostra come l'operazione `list-snomedct-inference-jobs` restituisce un elenco dei processi di inferenza in batch di SNOMED CT asincroni attualmente in corso.

```
aws comprehendmedical list-snomedct-inference-jobs
```

Output:

```
{
    "ComprehendMedicalAsyncJobPropertiesList": [
        {
            "JobId": "5780034166536cdb52ffa3295a1b00a7",
            "JobStatus": "COMPLETED",
            "SubmitTime": "2020-05-19T20:38:37.594000+00:00",
            "EndTime": "2020-05-19T20:45:07.894000+00:00",
            "ExpirationTime": "2020-09-17T20:38:37+00:00",
            "InputDataConfig": {
                "S3Bucket": "comp-med-input",
                "S3Key": "AKIAIOSFODNN7EXAMPLE"
            },
            "OutputDataConfig": {
                "S3Bucket": "comp-med-output",
                "S3Key": "AKIAIOSFODNN7EXAMPLE"
            },
            "LanguageCode": "en",
            "DataAccessRoleArn": "arn:aws:iam::867139942017:role/
ComprehendMedicalBatchProcessingRole",
            "ModelVersion":  "0.1.0"
        }
    ]
}
```

Per ulteriori informazioni, consulta Analisi in batch del collegamento delle rappresentazioni formali nella Guida per gli sviluppatori di Amazon Comprehend Medical.

- Per i dettagli sull'API, consulta ListSnomedctInferenceJobs AWS CLICommand Reference.

`start-entities-detection-v2-job`

Il seguente esempio di codice mostra come utilizzare`start-entities-detection-v2-job`.

AWS CLI

Come avviare un processo di rilevamento delle entità

L'esempio `start-entities-detection-v2-job` seguente avvia un processo di rilevamento delle entità asincrono.

```
aws comprehendmedical start-entities-detection-v2-job \
    --input-data-config "S3Bucket=comp-med-input" \
    --output-data-config "S3Bucket=comp-med-output" \
    --data-access-role-arn arn:aws:iam::867139942017:role/
ComprehendMedicalBatchProcessingRole \
    --language-code en
```

Output:

```
{
    "JobId": "ab9887877365fe70299089371c043b96"
}
```

Per ulteriori informazioni, consulta Batch APIs nella Amazon Comprehend Medical Developer Guide.

- Per i dettagli sull'API, consulta StartEntitiesDetectionV2Job in AWS CLI Command Reference.

`start-icd10-cm-inference-job`

Il seguente esempio di codice mostra come utilizzare. `start-icd10-cm-inference-job`

AWS CLI

Come avviare un processo di inferenza di ICD-10-CM

L'esempio `start-icd10-cm-inference-job` seguente avvia un processo di analisi in batch di inferenza di ICD-10-CM.

```
aws comprehendmedical start-icd10-cm-inference-job \
    --input-data-config "S3Bucket=comp-med-input" \
    --output-data-config "S3Bucket=comp-med-output" \
    --data-access-role-arn arn:aws:iam::867139942017:role/
ComprehendMedicalBatchProcessingRole \
```

```
    --language-code en
```

Output:

```
{
    "JobId": "ef7289877365fc70299089371c043b96"
}
```

Per ulteriori informazioni, consulta Analisi in batch del collegamento delle rappresentazioni formali nella Guida per gli sviluppatori di Amazon Comprehend Medical.

- Per i dettagli sull'API, vedi StartIcd10 CmInferenceJob in AWS CLI Command Reference.

`start-phi-detection-job`

Il seguente esempio di codice mostra come utilizzare`start-phi-detection-job`.

AWS CLI

Come avviare un processo di rilevamento di PHI

L'esempio `start-phi-detection-job` seguente avvia un processo di rilevamento delle entità di PHI asincrono.

```
aws comprehendmedical start-phi-detection-job \
    --input-data-config "S3Bucket=comp-med-input" \
    --output-data-config "S3Bucket=comp-med-output" \
    --data-access-role-arn arn:aws:iam::867139942017:role/
ComprehendMedicalBatchProcessingRole \
    --language-code en
```

Output:

```
{
    "JobId": "ab9887877365fe70299089371c043b96"
}
```

Per ulteriori informazioni, consulta Batch APIs nella Amazon Comprehend Medical Developer Guide.

- Per i dettagli sull'API, consulta StartPhiDetectionJob AWS CLICommand Reference.

`start-rx-norm-inference-job`

Il seguente esempio di codice mostra come utilizzare`start-rx-norm-inference-job`.

AWS CLI

Per avviare un processo di RxNorm inferenza

L'`start-rx-norm-inference-job`esempio seguente avvia un processo di analisi in batch di RxNorm inferenza.

```
aws comprehendmedical start-rx-norm-inference-job \
    --input-data-config "S3Bucket=comp-med-input" \
    --output-data-config "S3Bucket=comp-med-output" \
    --data-access-role-arn arn:aws:iam::867139942017:role/
ComprehendMedicalBatchProcessingRole \
    --language-code en
```

Output:

```
{
    "JobId": "eg8199877365fc70299089371c043b96"
}
```

Per ulteriori informazioni, consulta Analisi in batch del collegamento delle rappresentazioni formali nella Guida per gli sviluppatori di Amazon Comprehend Medical.

- Per i dettagli sull'API, vedere StartRxNormInferenceJobin AWS CLI Command Reference.

`start-snomedct-inference-job`

Il seguente esempio di codice mostra come utilizzare`start-snomedct-inference-job`.

AWS CLI

Come avviare un processo di inferenza di SNOMED CT

L'esempio `start-snomedct-inference-job` seguente avvia un processo di analisi in batch di inferenza di SNOMED CT.

```
aws comprehendmedical start-snomedct-inference-job \
    --input-data-config "S3Bucket=comp-med-input" \
    --output-data-config "S3Bucket=comp-med-output" \
    --data-access-role-arn arn:aws:iam::867139942017:role/
ComprehendMedicalBatchProcessingRole \
    --language-code en
```

Output:

```
{
    "JobId": "dg7289877365fc70299089371c043b96"
}
```

Per ulteriori informazioni, consulta Analisi in batch del collegamento delle rappresentazioni formali nella Guida per gli sviluppatori di Amazon Comprehend Medical.

- Per i dettagli sull'API, consulta StartSnomedctInferenceJob AWS CLICommand Reference.

`stop-entities-detection-v2-job`

Il seguente esempio di codice mostra come utilizzare`stop-entities-detection-v2-job`.

AWS CLI

Come interrompere un processo di rilevamento delle entità

L'esempio `stop-entities-detection-v2-job` seguente interrompe un processo di rilevamento delle entità asincrono.

```
aws comprehendmedical stop-entities-detection-v2-job \
    --job-id "ab9887877365fe70299089371c043b96"
```

Output:

```
{
    "JobId": "ab9887877365fe70299089371c043b96"
}
```

Per ulteriori informazioni, consulta Batch APIs nella Amazon Comprehend Medical Developer Guide.

- Per i dettagli sull'API, consulta StopEntitiesDetectionV2Job in AWS CLI Command Reference.

`stop-icd10-cm-inference-job`

Il seguente esempio di codice mostra come utilizzare. `stop-icd10-cm-inference-job`

AWS CLI

Come interrompere un processo di inferenza di ICD-10-CM

L'esempio `stop-icd10-cm-inference-job` seguente interrompe un processo di analisi in batch di inferenza di ICD-10-CM.

```
aws comprehendmedical stop-icd10-cm-inference-job \
    --job-id "4750034166536cdb52ffa3295a1b00a3"
```

Output:

```
{
    "JobId": "ef7289877365fc70299089371c043b96",
}
```

Per ulteriori informazioni, consulta Analisi in batch del collegamento delle rappresentazioni formali nella Guida per gli sviluppatori di Amazon Comprehend Medical.

- Per i dettagli sull'API, vedi StopIcd10 CmInferenceJob in AWS CLI Command Reference.

`stop-phi-detection-job`

Il seguente esempio di codice mostra come utilizzare`stop-phi-detection-job`.

AWS CLI

Come interrompere un processo di rilevamento di informazioni sanitarie protette (PHI)

L'esempio `stop-phi-detection-job` seguente interrompe un processo di rilevamento di informazioni sanitarie protette (PHI) asincrono.

```
aws comprehendmedical stop-phi-detection-job \
```

```
    --job-id "4750034166536cdb52ffa3295a1b00a3"
```

Output:

```
{
    "JobId": "ab9887877365fe70299089371c043b96"
}
```

Per ulteriori informazioni, consulta Batch APIs nella Amazon Comprehend Medical Developer Guide.

- Per i dettagli sull'API, consulta StopPhiDetectionJob AWS CLICommand Reference.

`stop-rx-norm-inference-job`

Il seguente esempio di codice mostra come utilizzare`stop-rx-norm-inference-job`.

AWS CLI

Per interrompere un processo di RxNorm inferenza

L'esempio `stop-rx-norm-inference-job` seguente interrompe un processo di analisi in batch di inferenza di ICD-10-CM.

```
aws comprehendmedical stop-rx-norm-inference-job \
    --job-id "eg8199877365fc70299089371c043b96"
```

Output:

```
{
    "JobId": "eg8199877365fc70299089371c043b96",
}
```

Per ulteriori informazioni, consulta Analisi in batch del collegamento delle rappresentazioni formali nella Guida per gli sviluppatori di Amazon Comprehend Medical.

- Per i dettagli sull'API, consulta StopRxNormInferenceJob AWS CLICommand Reference.

`stop-snomedct-inference-job`

Il seguente esempio di codice mostra come utilizzare`stop-snomedct-inference-job`.

AWS CLI

Come interrompere un processo di inferenza di SNOMED CT

L'esempio `stop-snomedct-inference-job` seguente interrompe un processo di analisi in batch di inferenza di SNOMED CT.

```
aws comprehendmedical stop-snomedct-inference-job \
    --job-id "8750034166436cdb52ffa3295a1b00a1"
```

Output:

```
{
    "JobId": "8750034166436cdb52ffa3295a1b00a1",
}
```

Per ulteriori informazioni, consulta Analisi in batch del collegamento delle rappresentazioni formali nella Guida per gli sviluppatori di Amazon Comprehend Medical.

- Per i dettagli sull'API, consulta StopSnomedctInferenceJob AWS CLICommand Reference.

AWS Config esempi utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with AWS Config.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`delete-config-rule`

Il seguente esempio di codice mostra come utilizzare`delete-config-rule`.

AWS CLI

Per eliminare una regola di AWS Config

Il comando seguente elimina una regola AWS Config denominata: `MyConfigRule`

```
aws configservice delete-config-rule --config-rule-name MyConfigRule
```

- Per i dettagli sull'API, vedere DeleteConfigRulein AWS CLI Command Reference.

`delete-delivery-channel`

Il seguente esempio di codice mostra come utilizzare`delete-delivery-channel`.

AWS CLI

Come eliminare un canale di distribuzione

Il comando seguente elimina il canale di distribuzione predefinito:

```
aws configservice delete-delivery-channel --delivery-channel-name default
```

- Per i dettagli sull'API, consulta DeleteDeliveryChannel AWS CLICommand Reference.

`delete-evaluation-results`

Il seguente esempio di codice mostra come utilizzare`delete-evaluation-results`.

AWS CLI

Come eliminare manualmente i risultati della valutazione

Il comando seguente elimina i risultati di valutazione correnti per la regola AWS gestita s3-: bucket-versioning-enabled

```
aws configservice delete-evaluation-results --config-rule-name s3-bucket-versioning-
enabled
```

- Per i dettagli sull'API, vedere DeleteEvaluationResultsin AWS CLI Command Reference.

`deliver-config-snapshot`

Il seguente esempio di codice mostra come utilizzare`deliver-config-snapshot`.

AWS CLI

Come distribuire uno snapshot di configurazione

Il comando seguente invia uno snapshot di configurazione per il bucket Amazon S3 appartenente al canale di distribuzione predefinito:

```
aws configservice deliver-config-snapshot --delivery-channel-name default
```

Output:

```
{
    "configSnapshotId": "d0333b00-a683-44af-921e-examplefb794"
}
```

- Per i dettagli sull'API, consulta DeliverConfigSnapshot AWS CLICommand Reference.

`describe-compliance-by-config-rule`

Il seguente esempio di codice mostra come utilizzare`describe-compliance-by-config-rule`.

AWS CLI

Per ottenere informazioni sulla conformità per le regole di AWS Config

Il comando seguente restituisce informazioni sulla conformità per ogni regola di AWS Config violata da una o più risorse: AWS

```
aws configservice describe-compliance-by-config-rule --compliance-
types NON_COMPLIANT
```

Nell'output, il valore di ogni attributo `CappedCount` indica il numero di risorse non conformi alla regola correlata. Ad esempio, l'output seguente indica che 3 risorse non sono conformi alla regola denominata `InstanceTypesAreT2micro`.

Output:

```
{
    "ComplianceByConfigRules": [
        {
            "Compliance": {
                "ComplianceContributorCount": {
                    "CappedCount": 3,
                    "CapExceeded": false
                },
                "ComplianceType": "NON_COMPLIANT"
            },
            "ConfigRuleName": "InstanceTypesAreT2micro"
        },
        {
            "Compliance": {
                "ComplianceContributorCount": {
                    "CappedCount": 10,
                    "CapExceeded": false
                },
                "ComplianceType": "NON_COMPLIANT"
            },
            "ConfigRuleName": "RequiredTagsForVolumes"
        }
    ]
}
```

- Per i dettagli sull'API, consulta AWS CLI Command DescribeComplianceByConfigRuleReference.

`describe-compliance-by-resource`

Il seguente esempio di codice mostra come utilizzare`describe-compliance-by-resource`.

AWS CLI

Per ottenere informazioni sulla conformità delle tue AWS risorse

Il comando seguente restituisce informazioni sulla conformità per ogni EC2 istanza registrata da AWS Config e che viola una o più regole:

```
aws configservice describe-compliance-by-resource --resource-type AWS::EC2::Instance
 --compliance-types NON_COMPLIANT
```

Nell'output, il valore di ogni attributo `CappedCount` indica il numero di regole violate dalla risorsa. Ad esempio, l'output seguente indica che l'istanza `i-1a2b3c4d` viola 2 regole.

Output:

```
{
    "ComplianceByResources": [
        {
            "ResourceType": "AWS::EC2::Instance",
            "ResourceId": "i-1a2b3c4d",
            "Compliance": {
                "ComplianceContributorCount": {
                    "CappedCount": 2,
                    "CapExceeded": false
                },
                "ComplianceType": "NON_COMPLIANT"
            }
        },
        {
            "ResourceType": "AWS::EC2::Instance",
            "ResourceId": "i-2a2b3c4d ",
            "Compliance": {
                "ComplianceContributorCount": {
                    "CappedCount": 3,
                    "CapExceeded": false
                },
                "ComplianceType": "NON_COMPLIANT"
            }
        }
    ]
}
```

- Per i dettagli sull'API, consulta AWS CLI Command DescribeComplianceByResourceReference.

`describe-config-rule-evaluation-status`

Il seguente esempio di codice mostra come utilizzare`describe-config-rule-evaluation-status`.

AWS CLI

Per ottenere informazioni sullo stato di una regola di AWS Config

Il comando seguente restituisce le informazioni sullo stato di una regola AWS Config denominata: `MyConfigRule`

```
aws configservice describe-config-rule-evaluation-status --config-rule-
names MyConfigRule
```

Output:

```
{
    "ConfigRulesEvaluationStatus": [
        {
            "ConfigRuleArn": "arn:aws:config:us-east-1:123456789012:config-rule/
config-rule-abcdef",
            "FirstActivatedTime": 1450311703.844,
            "ConfigRuleId": "config-rule-abcdef",
            "LastSuccessfulInvocationTime": 1450314643.156,
            "ConfigRuleName": "MyConfigRule"
        }
    ]
}
```

- Per i dettagli sull'API, vedere DescribeConfigRuleEvaluationStatusin AWS CLI Command Reference.

`describe-config-rules`

Il seguente esempio di codice mostra come utilizzare`describe-config-rules`.

AWS CLI

Per ottenere i dettagli di una regola di AWS Config

Il comando seguente restituisce i dettagli per una regola AWS Config denominata: `InstanceTypesAreT2micro`

```
aws configservice describe-config-rules --config-rule-names InstanceTypesAreT2micro
```

Output:

```
{
    "ConfigRules": [
        {
            "ConfigRuleState": "ACTIVE",
            "Description": "Evaluates whether EC2 instances are the t2.micro type.",
            "ConfigRuleName": "InstanceTypesAreT2micro",
            "ConfigRuleArn": "arn:aws:config:us-east-1:123456789012:config-rule/
config-rule-abcdef",
            "Source": {
                "Owner": "CUSTOM_LAMBDA",
                "SourceIdentifier": "arn:aws:lambda:us-
east-1:123456789012:function:InstanceTypeCheck",
                "SourceDetails": [
                    {
                        "EventSource": "aws.config",
                        "MessageType": "ConfigurationItemChangeNotification"
                    }
                ]
            },
            "InputParameters": "{\"desiredInstanceType\":\"t2.micro\"}",
            "Scope": {
                "ComplianceResourceTypes": [
                    "AWS::EC2::Instance"
                ]
            },
            "ConfigRuleId": "config-rule-abcdef"
        }
    ]
}
```

- Per i dettagli sull'API, vedere DescribeConfigRulesin AWS CLI Command Reference.

`describe-configuration-recorder-status`

Il seguente esempio di codice mostra come utilizzare`describe-configuration-recorder-status`.

AWS CLI

Come ottenere informazioni sullo stato del registratore di configurazione

Il comando seguente restituisce lo stato del registratore di configurazione predefinito.

```
aws configservice describe-configuration-recorder-status
```

Output:

```
{
    "ConfigurationRecordersStatus": [
        {
            "name": "default",
            "lastStatus": "SUCCESS",
            "recording": true,
            "lastStatusChangeTime": 1452193834.344,
            "lastStartTime": 1441039997.819,
            "lastStopTime": 1441039992.835
        }
    ]
}
```

- Per i dettagli sull'API, consulta DescribeConfigurationRecorderStatus AWS CLICommand Reference.

`describe-configuration-recorders`

Il seguente esempio di codice mostra come utilizzare`describe-configuration-recorders`.

AWS CLI

Come ottenere dettagli sul registratore di configurazione

Il comando seguente restituisce i dettagli del registratore di configurazione predefinito.

```
aws configservice describe-configuration-recorders
```

Output:

```
{
    "ConfigurationRecorders": [
        {
            "recordingGroup": {
                "allSupported": true,
                "resourceTypes": [],
```

```
                "includeGlobalResourceTypes": true
            },
            "roleARN": "arn:aws:iam::123456789012:role/config-ConfigRole-
A1B2C3D4E5F6",
            "name": "default"
        }
    ]
}
```

- Per i dettagli sull'API, consulta DescribeConfigurationRecorders AWS CLICommand Reference.

`describe-delivery-channel-status`

Il seguente esempio di codice mostra come utilizzare`describe-delivery-channel-status`.

AWS CLI

Come ottenere informazioni sullo stato del canale di distribuzione

Il comando seguente restituisce lo stato del canale di distribuzione:

```
aws configservice describe-delivery-channel-status
```

Output:

```
{
    "DeliveryChannelsStatus": [
        {
            "configStreamDeliveryInfo": {
                "lastStatusChangeTime": 1452193834.381,
                "lastStatus": "SUCCESS"
            },
            "configHistoryDeliveryInfo": {
                "lastSuccessfulTime": 1450317838.412,
                "lastStatus": "SUCCESS",
                "lastAttemptTime": 1450317838.412
            },
            "configSnapshotDeliveryInfo": {
                "lastSuccessfulTime": 1452185597.094,
                "lastStatus": "SUCCESS",
                "lastAttemptTime": 1452185597.094
            },
```

```
            "name": "default"
        }
    ]
}
```

- Per i dettagli sull'API, consulta DescribeDeliveryChannelStatus AWS CLICommand Reference.

`describe-delivery-channels`

Il seguente esempio di codice mostra come utilizzare`describe-delivery-channels`.

AWS CLI

Come ottenere dettagli sul canale di distribuzione

Il comando seguente restituisce i dettagli del canale di distribuzione.

```
aws configservice describe-delivery-channels
```

Output:

```
{
    "DeliveryChannels": [
        {
            "snsTopicARN": "arn:aws:sns:us-east-1:123456789012:config-topic",
            "name": "default",
            "s3BucketName": "config-bucket-123456789012"
        }
    ]
}
```

- Per i dettagli sull'API, consulta DescribeDeliveryChannels AWS CLICommand Reference.

`get-compliance-details-by-config-rule`

Il seguente esempio di codice mostra come utilizzare`get-compliance-details-by-config-rule`.

AWS CLI

Per ottenere i risultati della valutazione per una regola di AWS Config

Il comando seguente restituisce i risultati della valutazione per tutte le risorse che non sono conformi a una regola AWS Config denominata: `InstanceTypesAreT2micro`

```
aws configservice get-compliance-details-by-config-rule --config-rule-
name InstanceTypesAreT2micro --compliance-types NON_COMPLIANT
```

Output:

```
{
    "EvaluationResults": [
        {
            "EvaluationResultIdentifier": {
                "OrderingTimestamp": 1450314635.065,
                "EvaluationResultQualifier": {
                    "ResourceType": "AWS::EC2::Instance",
                    "ResourceId": "i-1a2b3c4d",
                    "ConfigRuleName": "InstanceTypesAreT2micro"
                }
            },
            "ResultRecordedTime": 1450314645.261,
            "ConfigRuleInvokedTime": 1450314642.948,
            "ComplianceType": "NON_COMPLIANT"
        },
        {
            "EvaluationResultIdentifier": {
                "OrderingTimestamp": 1450314635.065,
                "EvaluationResultQualifier": {
                    "ResourceType": "AWS::EC2::Instance",
                    "ResourceId": "i-2a2b3c4d",
                    "ConfigRuleName": "InstanceTypesAreT2micro"
                }
            },
            "ResultRecordedTime": 1450314645.18,
            "ConfigRuleInvokedTime": 1450314642.902,
            "ComplianceType": "NON_COMPLIANT"
        },
        {
            "EvaluationResultIdentifier": {
                "OrderingTimestamp": 1450314635.065,
                "EvaluationResultQualifier": {
                    "ResourceType": "AWS::EC2::Instance",
                    "ResourceId": "i-3a2b3c4d",
                    "ConfigRuleName": "InstanceTypesAreT2micro"
```

```
                }
            },
            "ResultRecordedTime": 1450314643.346,
            "ConfigRuleInvokedTime": 1450314643.124,
            "ComplianceType": "NON_COMPLIANT"
        }
    ]
}
```

- Per i dettagli sull'API, consulta GetComplianceDetailsByConfigRule AWS CLICommand Reference.

`get-compliance-details-by-resource`

Il seguente esempio di codice mostra come utilizzare`get-compliance-details-by-resource`.

AWS CLI

Per ottenere i risultati della valutazione di una AWS risorsa

Il comando seguente restituisce i risultati della valutazione per ogni regola a cui l' EC2 istanza `i-1a2b3c4d` non è conforme:

```
aws configservice get-compliance-details-by-resource --resource-
type AWS::EC2::Instance --resource-id i-1a2b3c4d --compliance-types NON_COMPLIANT
```

Output:

```
{
    "EvaluationResults": [
        {
            "EvaluationResultIdentifier": {
                "OrderingTimestamp": 1450314635.065,
                "EvaluationResultQualifier": {
                    "ResourceType": "AWS::EC2::Instance",
                    "ResourceId": "i-1a2b3c4d",
                    "ConfigRuleName": "InstanceTypesAreT2micro"
                }
            },
            "ResultRecordedTime": 1450314643.288,
            "ConfigRuleInvokedTime": 1450314643.034,
            "ComplianceType": "NON_COMPLIANT"
```

```
        },
        {
            "EvaluationResultIdentifier": {
                "OrderingTimestamp": 1450314635.065,
                "EvaluationResultQualifier": {
                    "ResourceType": "AWS::EC2::Instance",
                    "ResourceId": "i-1a2b3c4d",
                    "ConfigRuleName": "RequiredTagForEC2Instances"
                }
            },
            "ResultRecordedTime": 1450314645.261,
            "ConfigRuleInvokedTime": 1450314642.948,
            "ComplianceType": "NON_COMPLIANT"
        }
    ]
}
```

- Per i dettagli sull'API, consulta GetComplianceDetailsByResource AWS CLICommand Reference.

`get-compliance-summary-by-config-rule`

Il seguente esempio di codice mostra come utilizzare`get-compliance-summary-by-config-rule`.

AWS CLI

Per ottenere il riepilogo della conformità per le tue regole di AWS Config

Il comando seguente restituisce il numero di regole conformi e il numero di regole non conformi.

```
aws configservice get-compliance-summary-by-config-rule
```

Nell'output, il valore di ogni attributo `CappedCount` indica il numero di regole conformi o non conformi.

Output:

```
{
    "ComplianceSummary": {
        "NonCompliantResourceCount": {
            "CappedCount": 3,
```

```
            "CapExceeded": false
        },
        "ComplianceSummaryTimestamp": 1452204131.493,
        "CompliantResourceCount": {
            "CappedCount": 2,
            "CapExceeded": false
        }
    }
}
```

- Per i dettagli sull'API, consulta GetComplianceSummaryByConfigRule AWS CLICommand Reference.

`get-compliance-summary-by-resource-type`

Il seguente esempio di codice mostra come utilizzare`get-compliance-summary-by-resource-type`.

AWS CLI

Come ottenere il riepilogo della conformità per tutti i tipi di risorsa

Il comando seguente restituisce il numero di AWS risorse non conformi e il numero di risorse conformi:

```
aws configservice get-compliance-summary-by-resource-type
```

Nell'output, il valore di ogni attributo `CappedCount` indica il numero di risorse conformi o non conformi.

Output:

```
{
    "ComplianceSummariesByResourceType": [
        {
            "ComplianceSummary": {
                "NonCompliantResourceCount": {
                    "CappedCount": 16,
                    "CapExceeded": false
                },
                "ComplianceSummaryTimestamp": 1453237464.543,
                "CompliantResourceCount": {
```

```
                    "CappedCount": 10,
                    "CapExceeded": false
                }
            }
        }
    ]
}
```

Come ottenere il riepilogo della conformità per un tipo di risorsa specifico

Il comando seguente restituisce il numero di EC2 istanze non conformi e il numero di istanze conformi:

```
aws configservice get-compliance-summary-by-resource-type --resource-
types AWS::EC2::Instance
```

Nell'output, il valore di ogni attributo `CappedCount` indica il numero di risorse conformi o non conformi.

Output:

```
{
    "ComplianceSummariesByResourceType": [
        {
            "ResourceType": "AWS::EC2::Instance",
            "ComplianceSummary": {
                "NonCompliantResourceCount": {
                    "CappedCount": 3,
                    "CapExceeded": false
                },
                "ComplianceSummaryTimestamp": 1452204923.518,
                "CompliantResourceCount": {
                    "CappedCount": 7,
                    "CapExceeded": false
                }
            }
        }
    ]
}
```

- Per i dettagli sull'API, consulta Command Reference. GetComplianceSummaryByResourceTypeAWS CLI

`get-resource-config-history`

Il seguente esempio di codice mostra come utilizzare`get-resource-config-history`.

AWS CLI

Per ottenere la cronologia di configurazione di una AWS risorsa

Il comando seguente restituisce un elenco di elementi di configurazione per un' EC2 istanza con un ID di`i-1a2b3c4d`:

```
aws configservice get-resource-config-history --resource-type AWS::EC2::Instance --resource-id i-1a2b3c4d
```

- Per i dettagli sull'API, consulta GetResourceConfigHistory AWS CLICommand Reference.

`get-status`

Il seguente esempio di codice mostra come utilizzare`get-status`.

AWS CLI

Per ottenere lo stato di AWS Config

Il comando seguente restituisce lo stato del canale di distribuzione e del registratore di configurazione:

```
aws configservice get-status
```

Output:

```
Configuration Recorders:

name: default
recorder: ON
last status: SUCCESS

Delivery Channels:

name: default
last stream delivery status: SUCCESS
```

```
last history delivery status: SUCCESS
last snapshot delivery status: SUCCESS
```

- Per i dettagli sull'API, consulta GetStatus AWS CLICommand Reference.

`list-discovered-resources`

Il seguente esempio di codice mostra come utilizzare`list-discovered-resources`.

AWS CLI

Per elencare le risorse scoperte da AWS Config

Il comando seguente elenca le EC2 istanze rilevate da AWS Config:

```
aws configservice list-discovered-resources --resource-type AWS::EC2::Instance
```

Output:

```
{
    "resourceIdentifiers": [
        {
            "resourceType": "AWS::EC2::Instance",
            "resourceId": "i-1a2b3c4d"
        },
        {
            "resourceType": "AWS::EC2::Instance",
            "resourceId": "i-2a2b3c4d"
        },
        {
            "resourceType": "AWS::EC2::Instance",
            "resourceId": "i-3a2b3c4d"
        }
    ]
}
```

- Per i dettagli sull'API, consulta AWS CLI Command ListDiscoveredResourcesReference.

`put-config-rule`

Il seguente esempio di codice mostra come utilizzare`put-config-rule`.

AWS CLI

Per aggiungere una regola AWS Config gestita

Il comando seguente fornisce codice JSON per aggiungere una regola Config AWS gestita:

```
aws configservice put-config-rule --config-rule file://
RequiredTagsForEC2Instances.json
```

`RequiredTagsForEC2Instances.json` è un file JSON contenente la configurazione delle regole.

```
{
  "ConfigRuleName": "RequiredTagsForEC2Instances",
  "Description": "Checks whether the CostCenter and Owner tags are applied to EC2
 instances.",
  "Scope": {
    "ComplianceResourceTypes": [
      "AWS::EC2::Instance"
    ]
  },
  "Source": {
    "Owner": "AWS",
    "SourceIdentifier": "REQUIRED_TAGS"
  },
  "InputParameters": "{\"tag1Key\":\"CostCenter\",\"tag2Key\":\"Owner\"}"
}
```

Per l'`ComplianceResourceTypes`attributo, questo codice JSON limita l'ambito alle risorse del `AWS::EC2::Instance` tipo, quindi AWS Config valuterà EC2 solo le istanze rispetto alla regola. Poiché si tratta di una regola gestita, l'attributo `Owner` è impostato su `AWS` e l'attributo `SourceIdentifier` è impostato sull'identificatore della regola, ovvero `REQUIRED_TAGS`. Per l'attributo `InputParameters`, vengono specificate le chiavi di tag (`CostCenter` e `Owner`) richieste dalla regola.

Se il comando ha esito positivo, AWS Config non restituisce alcun output. Per verificare la configurazione della regola, esegui il describe-config-rules comando e specifica il nome della regola.

Come aggiungere una regola di configurazione gestita dal cliente

Il comando seguente fornisce il codice JSON per aggiungere una regola di configurazione gestita dal cliente.

```
aws configservice put-config-rule --config-rule file://InstanceTypesAreT2micro.json
```

`InstanceTypesAreT2micro.json` è un file JSON contenente la configurazione delle regole.

```
{
  "ConfigRuleName": "InstanceTypesAreT2micro",
  "Description": "Evaluates whether EC2 instances are the t2.micro type.",
  "Scope": {
    "ComplianceResourceTypes": [
      "AWS::EC2::Instance"
    ]
  },
  "Source": {
    "Owner": "CUSTOM_LAMBDA",
    "SourceIdentifier": "arn:aws:lambda:us-
east-1:123456789012:function:InstanceTypeCheck",
    "SourceDetails": [
      {
        "EventSource": "aws.config",
        "MessageType": "ConfigurationItemChangeNotification"
      }
    ]
  },
  "InputParameters": "{\"desiredInstanceType\":\"t2.micro\"}"
}
```

Per l'`ComplianceResourceTypes`attributo, questo codice JSON limita l'ambito alle risorse del `AWS::EC2::Instance` tipo, quindi AWS Config valuterà EC2 solo le istanze rispetto alla regola. Poiché questa regola è una regola gestita dal cliente, l'`Owner`attributo è impostato su e l'`SourceIdentifier`attributo è impostato sull'ARN della funzione Lambda AWS . `CUSTOM_LAMBDA` L'oggetto `SourceDetails` è obbligatorio. I parametri specificati per l'`InputParameters`attributo vengono passati alla funzione AWS Lambda quando AWS Config la richiama per valutare le risorse rispetto alla regola.

Se il comando ha esito positivo, AWS Config non restituisce alcun output. Per verificare la configurazione della regola, esegui il describe-config-rules comando e specifica il nome della regola.

- Per i dettagli sull'API, consulta PutConfigRule AWS CLICommand Reference.

put-configuration-recorder

Il seguente esempio di codice mostra come utilizzare`put-configuration-recorder`.

AWS CLI

Esempio 1: come registrare tutte le risorse supportate

Il comando seguente crea un registratore di configurazione che tiene traccia delle modifiche a tutti i tipi di risorse supportati, inclusi i tipi di risorse globali:

```
aws configservice put-configuration-recorder \
    --configuration-recorder name=default,roleARN=arn:aws:iam::123456789012:role/config-role \
    --recording-group allSupported=true,includeGlobalResourceTypes=true
```

Se il comando ha esito positivo, AWS Config non restituisce alcun output. Per verificare le impostazioni del registratore di configurazione, esegui il comando. describe-configuration-recorders

Esempio 2: come registrare tipi specifici di risorse

Il comando seguente crea un registratore di configurazione che tiene traccia delle modifiche solo ai tipi di risorse specificati nel file JSON per l'opzione --recording-group:

```
aws configservice put-configuration-recorder \
    --configuration-recorder name=default,roleARN=arn:aws:iam::123456789012:role/config-role \
    --recording-group file://recordingGroup.json
```

RecordingGroup.json è un file JSON che specifica i tipi di risorse che Config registrerà: AWS

```
{
    "allSupported": false,
    "includeGlobalResourceTypes": false,
    "resourceTypes": [
        "AWS::EC2::EIP",
        "AWS::EC2::Instance",
        "AWS::EC2::NetworkAcl",
        "AWS::EC2::SecurityGroup",
        "AWS::CloudTrail::Trail",
```

```
        "AWS::EC2::Volume",
        "AWS::EC2::VPC",
        "AWS::IAM::User",
        "AWS::IAM::Policy"
    ]
}
```

Prima di poter specificare i tipi di risorse per la chiave ResourceTypes, è necessario impostare le opzioni AllSupported e Types su false o includeGlobalResource ometterle.

Se il comando ha esito positivo, AWS Config non restituisce alcun output. Per verificare le impostazioni del registratore di configurazione, esegui il comando. describe-configuration-recorders

Esempio 3: come selezionare tutte le risorse supportate, esclusi tipi specifici di risorse

Il comando seguente crea un registratore di configurazione che tiene traccia delle modifiche a tutti i tipi di risorse supportati in questo momento e in futuro escludendo quelli specificati nel file JSON per l'opzione --recording-group:

```
aws configservice put-configuration-recorder \
    --configuration-recorder name=default,roleARN=arn:aws:iam::123456789012:role/
config-role \
    --recording-group file://recordingGroup.json
```

RecordingGroup.json è un file JSON che specifica i tipi di risorse che Config registrerà: AWS

```
{
    "allSupported": false,
    "exclusionByResourceTypes": {
        "resourceTypes": [
        "AWS::Redshift::ClusterSnapshot",
        "AWS::RDS::DBClusterSnapshot",
        "AWS::CloudFront::StreamingDistribution"
    ]
    },
        "includeGlobalResourceTypes": false,
        "recordingStrategy": {
        "useOnly": "EXCLUSION_BY_RESOURCE_TYPES"
    },
}
```

Prima di poter specificare i tipi di risorse da escludere dalla registrazione: 1) È necessario impostare le opzioni allSupported e Types su false o ometterle e 2) è necessario impostare il includeGlobalResource campo useOnly di su EXCLUSION_BY_RESOURCE_TYPES. RecordingStrategy

Se il comando ha esito positivo, AWS Config non restituisce alcun output. Per verificare le impostazioni del registratore di configurazione, esegui il comando. describe-configuration-recorders

- Per i dettagli sull'API, consulta PutConfigurationRecorder AWS CLICommand Reference.

`put-delivery-channel`

Il seguente esempio di codice mostra come utilizzare`put-delivery-channel`.

AWS CLI

Come creare un canale di distribuzione

Il comando seguente fornisce le impostazioni per il canale di distribuzione come codice JSON.

```
aws configservice put-delivery-channel --delivery-channel file://
deliveryChannel.json
```

Il file `deliveryChannel.json` specifica gli attributi del canale di distribuzione.

```
{
    "name": "default",
    "s3BucketName": "config-bucket-123456789012",
    "snsTopicARN": "arn:aws:sns:us-east-1:123456789012:config-topic",
    "configSnapshotDeliveryProperties": {
        "deliveryFrequency": "Twelve_Hours"
    }
}
```

L'esempio seguente imposta i seguenti attributi:

`name`: il nome del canale di distribuzione. Per impostazione predefinita, AWS Config assegna il nome `default` a un nuovo canale di consegna. Non è possibile aggiornare il nome del canale di consegna con il comando. `put-delivery-channel` Per i passaggi per modificare il nome, consulta Ridenominazione del canale di distribuzione (`s3BucketName`). - Il nome del bucket

Amazon S3 a cui AWS Config fornisce istantanee di configurazione e file di cronologia della configurazione. Se si specifica un bucket che appartiene a un altro AWS account, tale bucket deve disporre di politiche che concedano le autorizzazioni di accesso a Config. AWS Per ulteriori informazioni, consulta Autorizzazioni per il bucket Amazon S3.

`snsTopicARN`- L'Amazon Resource Name (ARN) dell'argomento Amazon SNS a cui AWS Config invia notifiche sulle modifiche alla configurazione. Se scegli un argomento da un altro account, l'argomento deve avere politiche che concedano le autorizzazioni di accesso a Config. AWS Per ulteriori informazioni, consulta Autorizzazioni per l'argomento Amazon SNS.

`configSnapshotDeliveryProperties`- Contiene l'`deliveryFrequency`attributo, che imposta la frequenza con cui AWS Config fornisce istantanee di configurazione e la frequenza con cui richiama le valutazioni per le regole Config periodiche.

Se il comando ha esito positivo, AWS Config non restituisce alcun output. Per verificare le impostazioni del tuo canale di consegna, esegui il describe-delivery-channels comando.

- Per i dettagli sull'API, consulta PutDeliveryChannel AWS CLICommand Reference.

`start-config-rules-evaluation`

Il seguente esempio di codice mostra come utilizzare`start-config-rules-evaluation`.

AWS CLI

Per eseguire una valutazione su richiesta per le regole AWS Config

Il comando seguente avvia una valutazione per due regole AWS gestite:

```
aws configservice start-config-rules-evaluation --config-rule-names s3-bucket-versioning-enabled cloudtrail-enabled
```

- Per i dettagli sull'API, vedere StartConfigRulesEvaluationin AWS CLI Command Reference.

`start-configuration-recorder`

Il seguente esempio di codice mostra come utilizzare`start-configuration-recorder`.

AWS CLI

Come avviare il registratore di configurazione

Il comando seguente avvia il registratore di configurazione predefinito:

```
aws configservice start-configuration-recorder --configuration-recorder-name default
```

Se il comando ha esito positivo, AWS Config non restituisce alcun output. Per verificare che AWS Config stia registrando le tue risorse, esegui il comando get-status.

- Per i dettagli sull'API, consulta Command StartConfigurationRecorderReference AWS CLI .

`stop-configuration-recorder`

Il seguente esempio di codice mostra come utilizzare`stop-configuration-recorder`.

AWS CLI

Come arrestare il registratore di configurazione

Il comando seguente arresta il registratore di configurazione predefinito:

```
aws configservice stop-configuration-recorder --configuration-recorder-name default
```

Se il comando ha esito positivo, AWS Config non restituisce alcun output. Per verificare che AWS Config non stia registrando le tue risorse, esegui il comando get-status.

- Per i dettagli sull'API, consulta Command StopConfigurationRecorderReference AWS CLI .

`subscribe`

Il seguente esempio di codice mostra come utilizzare`subscribe`.

AWS CLI

Per abbonarsi a AWS Config

Il comando seguente crea il canale di distribuzione e il registratore di configurazione predefiniti. Il comando specifica anche il bucket Amazon S3 e l'argomento Amazon SNS a AWS cui Config fornirà le informazioni di configurazione:

```
aws configservice subscribe --s3-bucket config-bucket-123456789012
 --sns-topic arn:aws:sns:us-east-1:123456789012:config-topic --iam-
role arn:aws:iam::123456789012:role/ConfigRole-A1B2C3D4E5F6
```

Output:

```
Using existing S3 bucket: config-bucket-123456789012
Using existing SNS topic: arn:aws:sns:us-east-1:123456789012:config-topic
Subscribe succeeded:

Configuration Recorders: [
    {
        "recordingGroup": {
            "allSupported": true,
            "resourceTypes": [],
            "includeGlobalResourceTypes": false
        },
        "roleARN": "arn:aws:iam::123456789012:role/ConfigRole-A1B2C3D4E5F6",
        "name": "default"
    }
]

Delivery Channels: [
    {
        "snsTopicARN": "arn:aws:sns:us-east-1:123456789012:config-topic",
        "name": "default",
        "s3BucketName": "config-bucket-123456789012"
    }
]
```

- Per informazioni dettagliate sull'API, consulta Subscribe nella Documentazione di riferimento dei comandi della AWS CLI .

Esempi di utilizzo di Amazon Connect AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando Amazon Connect. AWS Command Line Interface

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`create-user`

Il seguente esempio di codice mostra come usare`create-user`.

AWS CLI

Come creare un utente

L'esempio `create-user` seguente aggiunge un utente con gli attributi specificati all'istanza Amazon Connect specificata.

```
aws connect create-user \
    --username Mary \
    --password Pass@Word1 \
    --identity-info FirstName=Mary,LastName=Major \
    --phone-
config PhoneType=DESK_PHONE,AutoAccept=true,AfterContactWorkTimeLimit=60,DeskPhoneNumber=
+15555551212 \
    --security-profile-id 12345678-1111-2222-aaaa-a1b2c3d4f5g7 \
    --routing-profile-id 87654321-9999-3434-abcd-x1y2z3a1b2c3 \
    --instance-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Output:

```
{
    "UserId": "87654321-2222-1234-1234-111234567891",
    "UserArn": "arn:aws:connect:us-west-2:123456789012:instance/a1b2c3d4-5678-90ab-
cdef-EXAMPLE11111/agent/87654321-2222-1234-1234-111234567891"
}
```

Per ulteriori informazioni, consulta Aggiungere utenti nella Guida per l'amministratore di Amazon Connect.

- Per i dettagli sull'API, consulta CreateUser AWS CLICommand Reference.

`delete-user`

Il seguente esempio di codice mostra come utilizzare`delete-user`.

AWS CLI

Come eliminare un utente

L'esempio `delete-user` seguente elimina l'utente specificato dall'istanza Amazon Connect specificata.

```
aws connect delete-user \
    --instance-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 \
    --user-id 87654321-2222-1234-1234-111234567891
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Gestione degli utenti nella Guida per l'amministratore di Amazon Connect.

- Per i dettagli sull'API, consulta DeleteUser AWS CLICommand Reference.

`describe-user-hierarchy-group`

Il seguente esempio di codice mostra come utilizzare`describe-user-hierarchy-group`.

AWS CLI

Come visualizzare i dettagli di un gruppo gerarchico

L'esempio `describe-user-hierarchy-group` seguente mostra i dettagli del gruppo di gerarchia Amazon Connect specificato.

```
aws connect describe-user-hierarchy-group \
    --hierarchy-group-id 12345678-1111-2222-800e-aaabbb555gg \
    --instance-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Output:

```
{
     "HierarchyGroup": {
         "Id": "12345678-1111-2222-800e-a2b3c4d5f6g7",
         "Arn": "arn:aws:connect:us-west-2:123456789012:instance/a1b2c3d4-5678-90ab-
cdef-EXAMPLE11111/agent-group/12345678-1111-2222-800e-a2b3c4d5f6g7",
         "Name": "Example Corporation",
         "LevelId": "1",
```

```
        "HierarchyPath": {
            "LevelOne": {
                "Id": "abcdefgh-3333-4444-8af3-201123456789",
                "Arn": "arn:aws:connect:us-west-2:123456789012:instance/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111/agent-group/
abcdefgh-3333-4444-8af3-201123456789",
                "Name": "Example Corporation"
            }
        }
    }
}
```

Per ulteriori informazioni, consulta Configurare le gerarchie di agenti nella Guida per l'amministratore di Amazon Connect.

- Per i dettagli sull'API, consulta DescribeUserHierarchyGroup AWS CLICommand Reference.

`describe-user-hierarchy-structure`

Il seguente esempio di codice mostra come utilizzare`describe-user-hierarchy-structure`.

AWS CLI

Come visualizzare i dettagli di una struttura gerarchica

L'esempio `describe-user-hierarchy-structure` seguente mostra i dettagli della struttura gerarchica per l'istanza Amazon Connect specificata.

```
aws connect describe-user-hierarchy-group \
    --instance-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Output:

```
{
    "HierarchyStructure": {
        "LevelOne": {
            "Id": "12345678-1111-2222-800e-aaabbb555gg",
            "Arn": "arn:aws:connect:us-west-2:123456789012:instance/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111/agent-group-level/1",
            "Name": "Corporation"
        },
        "LevelTwo": {
```

```
            "Id": "87654321-2222-3333-ac99-123456789102",
            "Arn": "arn:aws:connect:us-west-2:123456789012:instance/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111/agent-group-level/2",
            "Name": "Services Division"
        },
        "LevelThree": {
            "Id": "abcdefgh-3333-4444-8af3-201123456789",
            "Arn": "arn:aws:connect:us-west-2:123456789012:instance/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111/agent-group-level/3",
            "Name": "EU Site"
        }
    }
}
```

Per ulteriori informazioni, consulta Configurare le gerarchie di agenti nella Guida per l'amministratore di Amazon Connect.

- Per i dettagli sull'API, consulta DescribeUserHierarchyStructure AWS CLICommand Reference.

`describe-user`

Il seguente esempio di codice mostra come utilizzare`describe-user`.

AWS CLI

Come visualizzare i dettagli di un utente

L'esempio `describe-user` seguente mostra i dettagli dell'utente Amazon Connect specificato.

```
aws connect describe-user \
    --user-id 0c245dc0-0cf5-4e37-800e-2a7481cc8a60
    --instance-id 40c83b68-ea62-414c-97bb-d018e39e158e
```

Output:

```
{
    "User": {
        "Id": "0c245dc0-0cf5-4e37-800e-2a7481cc8a60",
        "Arn": "arn:aws:connect:us-west-2:123456789012:instance/40c83b68-
ea62-414c-97bb-d018e39e158e/agent/0c245dc0-0cf5-4e37-800e-2a7481cc8a60",
        "Username": "Jane",
        "IdentityInfo": {
            "FirstName": "Jane",
```

```
            "LastName": "Doe",
            "Email": "example.com"
        },
        "PhoneConfig": {
            "PhoneType": "SOFT_PHONE",
            "AutoAccept": false,
            "AfterContactWorkTimeLimit": 0,
            "DeskPhoneNumber": ""
        },
        "DirectoryUserId": "8b444cf6-b368-4f29-ba18-07af27405658",
        "SecurityProfileIds": [
            "b6f85a42-1dc5-443b-b621-de0abf70c9cf"
        ],
        "RoutingProfileId": "0be36ee9-2b5f-4ef4-bcf7-87738e5be0e5",
        "Tags": {}
    }
}
```

Per ulteriori informazioni, consulta Gestione degli utenti nella Guida per l'amministratore di Amazon Connect.

- Per i dettagli sull'API, consulta DescribeUser AWS CLICommand Reference.

`get-contact-attributes`

Il seguente esempio di codice mostra come utilizzare`get-contact-attributes`.

AWS CLI

Come recuperare gli attributi di un contatto

L'esempio `get-contact-attributes` seguente recupera gli attributi che sono stati impostati per il contatto Amazon Connect specificato.

```
aws connect get-contact-attributes \
    --instance-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 \
    --initial-contact-id 12345678-1111-2222-800e-a2b3c4d5f6g7
```

Output:

```
{
    "Attributes": {
```

```
        "greetingPlayed": "true"
    }
}
```

Per ulteriori informazioni, consulta la pagina Uso degli attributi di contatto di Amazon Connect nella Guida per l'amministratore di Amazon Connect.

- Per i dettagli sull'API, consulta GetContactAttributes AWS CLICommand Reference.

`list-contact-flows`

Il seguente esempio di codice mostra come utilizzare`list-contact-flows`.

AWS CLI

Come elencare i flussi di contatti in un'istanza

L'esempio `list-contact-flows` seguente mostra i flussi di contatto nell'istanza Amazon Connect specificata.

```
aws connect list-contact-flows \
    --instance-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Output:

```
{
     "ContactFlowSummaryList": [
         {
             "Id": "12345678-1111-2222-800e-a2b3c4d5f6g7",
             "Arn": "arn:aws:connect:us-west-2:123456789012:instance/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111/contact-flow/12345678-1111-2222-800e-
a2b3c4d5f6g7",
             "Name": "Default queue transfer",
             "ContactFlowType": "QUEUE_TRANSFER"
         },
         {
             "Id": "87654321-2222-3333-ac99-123456789102",
             "Arn": "arn:aws:connect:us-west-2:123456789012:instance/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111/contact-flow/87654321-2222-3333-
ac99-123456789102",
             "Name": "Default agent hold",
             "ContactFlowType": "AGENT_HOLD"
         },
```

```
        {
            "Id": "abcdefgh-3333-4444-8af3-201123456789",
                "Arn": "arn:aws:connect:us-west-2:123456789012:instance/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111/contact-flow/
abcdefgh-3333-4444-8af3-201123456789",
            "Name": "Default customer hold",
            "ContactFlowType": "CUSTOMER_HOLD"
        },
    ]
}
```

Per ulteriori informazioni, consulta Creare flussi di contatti Amazon Connect nella Guida per l'amministratore di Amazon Connect.

- Per i dettagli sull'API, consulta ListContactFlows AWS CLICommand Reference.

`list-hours-of-operations`

Il seguente esempio di codice mostra come utilizzare`list-hours-of-operations`.

AWS CLI

Come elencare gli orari di funzionamento di un'istanza

L'esempio `list-hours-of-operations` seguente elenca gli orari di operatività per l'istanza Amazon Connect specificata.

```
aws connect list-hours-of-operations \
    --instance-id 40c83b68-ea62-414c-97bb-d018e39e158e
```

Output:

```
{
    "HoursOfOperationSummaryList": [
        {
            "Id": "d69f1f84-7457-4924-8fbe-e64875546259",
            "Arn": "arn:aws:connect:us-west-2:123456789012:instance/40c83b68-
ea62-414c-97bb-d018e39e158e/operating-hours/d69f1f84-7457-4924-8fbe-e64875546259",
            "Name": "Basic Hours"
        }
    ]
}
```

Per ulteriori informazioni, consulta Impostazione degli orari di attività di una coda nella Guida per l'amministratore di Amazon Connect.

- Per i dettagli sull'API, consulta ListHoursOfOperations AWS CLICommand Reference.

`list-phone-numbers`

Il seguente esempio di codice mostra come utilizzare`list-phone-numbers`.

AWS CLI

Come elencare i numeri di telefono di un'istanza

L'esempio `list-phone-numbers` seguente elenca i numeri di telefono nell'istanza Amazon Connect specificata.

```
aws connect list-phone-numbers \
    --instance-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Output:

```
{
    "PhoneNumberSummaryList": [
        {
            "Id": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
            "Arn": "arn:aws:connect:us-west-2:123456789012:instance/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111/phone-number/xyz80zxy-xyz1-80zx-
zx80-11111EXAMPLE",
            "PhoneNumber": "+17065551212",
            "PhoneNumberType": "DID",
            "PhoneNumberCountryCode": "US"
        },
        {
            "Id": "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
            "Arn": "arn:aws:connect:us-west-2:123456789012:instance/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111/phone-number/ccc0ccc-xyz1-80zx-
zx80-22222EXAMPLE",
            "PhoneNumber": "+18555551212",
            "PhoneNumberType": "TOLL_FREE",
            "PhoneNumberCountryCode": "US"
        }
    ]
```

```
}
```

Per ulteriori informazioni, consulta Configurare i numeri di telefono per il contact center nella Guida per l'amministratore di Amazon Connect.

- Per i dettagli sull'API, consulta ListPhoneNumbers AWS CLICommand Reference.

`list-queues`

Il seguente esempio di codice mostra come utilizzare`list-queues`.

AWS CLI

Come elencare le code in un'istanza

L'esempio `list-queues` seguente elenca le code nell'istanza Amazon Connect specificata.

```
aws connect list-queues \
    --instance-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Output:

```
{
    "QueueSummaryList": [
        {
            "Id": "12345678-1111-2222-800e-a2b3c4d5f6g7",
            "Arn": "arn:aws:connect:us-west-2:123456789012:instance/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111/queue/agent/12345678-1111-2222-800e-
a2b3c4d5f6g7",
            "QueueType": "AGENT"
        },
        {
            "Id": "87654321-2222-3333-ac99-123456789102",
            "Arn": "arn:aws:connect:us-west-2:123456789012:instance/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111/queue/agent/87654321-2222-3333-
ac99-123456789102",
            "QueueType": "AGENT"
        },
        {
            "Id": "abcdefgh-3333-4444-8af3-201123456789",
            "Arn": "arn:aws:connect:us-west-2:123456789012:instance/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111/queue/agent/
abcdefgh-3333-4444-8af3-201123456789",
```

```
            "QueueType": "AGENT"
        },
        {
            "Id": "hgfedcba-4444-5555-a31f-123456789102",
            "Arn": "arn:aws:connect:us-west-2:123456789012:instance/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111/queue/hgfedcba-4444-5555-a31f-123456789102",
            "Name": "BasicQueue",
            "QueueType": "STANDARD"
        },
    ]
}
```

Per ulteriori informazioni, consulta Creare una coda nella Guida per l'amministratore di Amazon Connect.

- Per i dettagli sull'API, consulta ListQueues AWS CLICommand Reference.

`list-routing-profiles`

Il seguente esempio di codice mostra come utilizzare`list-routing-profiles`.

AWS CLI

Come elencare i profili di routing in un'istanza

L'esempio `list-routing-profiles` seguente elenca i profili di routing nell'istanza Amazon Connect specificata.

```
aws connect list-routing-profiles \
    --instance-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Output:

```
{
    "RoutingProfileSummaryList": [
        {
            "Id": "12345678-1111-2222-800e-a2b3c4d5f6g7",
            "Arn": "arn:aws:connect:us-west-2:123456789012:instance/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111/routing-profile/12345678-1111-2222-800e-
a2b3c4d5f6g7",
            "Name": "Basic Routing Profile"
        },
```

```
    ]
}
```

Per ulteriori informazioni, consulta Creazione di un profilo di instradamento nella Guida per l'amministratore di Amazon Connect.

- Per i dettagli sull'API, consulta ListRoutingProfiles AWS CLICommand Reference.

`list-security-profiles`

Il seguente esempio di codice mostra come utilizzare`list-security-profiles`.

AWS CLI

Come elencare i profili di sicurezza in un'istanza

L'esempio `list-security-profiles` seguente elenca i profili di sicurezza nell'istanza Amazon Connect specificata.

```
aws connect list-security-profiles \
    --instance-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Output:

```
{
    "SecurityProfileSummaryList": [
        {
            "Id": "12345678-1111-2222-800e-a2b3c4d5f6g7",
            "Arn": "arn:aws:connect:us-west-2:123456789012:instance/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111/security-profile/12345678-1111-2222-800e-
a2b3c4d5f6g7",
            "Name": "CallCenterManager"
        },
        {
            "Id": "87654321-2222-3333-ac99-123456789102",
            "Arn": "arn:aws:connect:us-west-2:123456789012:instance/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111/security-profile/87654321-2222-3333-
ac99-123456789102",
            "Name": "QualityAnalyst"
        },
        {
            "Id": "abcdefgh-3333-4444-8af3-201123456789",
```

```
            "Arn": "arn:aws:connect:us-west-2:123456789012:instance/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111/security-profile/
abcdefgh-3333-4444-8af3-201123456789",
            "Name": "Agent"
        },
        {
            "Id": "12345678-1111-2222-800e-x2y3c4d5fzzzz",
            "Arn": "arn:aws:connect:us-west-2:123456789012:instance/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111/security-profile/12345678-1111-2222-800e-
x2y3c4d5fzzzz",
            "Name": "Admin"
        }
    ]
}
```

Per ulteriori informazioni, consulta Assegnare le autorizzazioni: profili di sicurezza nella Guida per l'amministratore di Amazon Connect.

- Per i dettagli sull'API, consulta ListSecurityProfiles AWS CLICommand Reference.

`list-user-hierarchy-groups`

Il seguente esempio di codice mostra come utilizzare`list-user-hierarchy-groups`.

AWS CLI

Come elencare i gruppi gerarchici di utenti in un'istanza

L'esempio `list-user-hierarchy-groups` seguente elenca i gruppi di gerarchie di utenti nell'istanza Amazon Connect specificata.

```
aws connect list-user-hierarchy-groups \
    --instance-id 40c83b68-ea62-414c-97bb-d018e39e158e
```

Output:

```
{
    "UserHierarchyGroupSummaryList": [
        {
            "Id": "0e2f6d1d-b3ca-494b-8dbc-ba81d9f8182a",
            "Arn": "arn:aws:connect:us-west-2:123456789012:instance/40c83b68-
ea62-414c-97bb-d018e39e158e/agent-group/0e2f6d1d-b3ca-494b-8dbc-ba81d9f8182a",
```

```
            "Name": "Example Corporation"
        },
    ]
}
```

Per ulteriori informazioni, consulta Configurare le gerarchie di agenti nella Guida per l'amministratore di Amazon Connect.

- Per i dettagli sull'API, consulta ListUserHierarchyGroups AWS CLICommand Reference.

`list-users`

Il seguente esempio di codice mostra come utilizzare`list-users`.

AWS CLI

Come elencare i gruppi gerarchici di utenti in un'istanza

L'esempio `list-users` seguente elenca gli utenti nell'istanza Amazon Connect specificata.

```
aws connect list-users \
    --instance-id 40c83b68-ea62-414c-97bb-d018e39e158e
```

Output:

```
{
    "UserSummaryList": [
        {
            "Id": "0c245dc0-0cf5-4e37-800e-2a7481cc8a60",
            "Arn": "arn:aws:connect:us-west-2:123456789012:instance/40c83b68-
ea62-414c-97bb-d018e39e158e/agent/0c245dc0-0cf5-4e37-800e-2a7481cc8a60",
            "Username": "Jane"
        },
        {
            "Id": "46f0c67c-3fc7-4806-ac99-403798788c14",
            "Arn": "arn:aws:connect:us-west-2:123456789012:instance/40c83b68-
ea62-414c-97bb-d018e39e158e/agent/46f0c67c-3fc7-4806-ac99-403798788c14",
            "Username": "Paulo"
        },
        {
            "Id": "55a83578-95e1-4710-8af3-2b7afe310e48",
            "Arn": "arn:aws:connect:us-west-2:123456789012:instance/40c83b68-
ea62-414c-97bb-d018e39e158e/agent/55a83578-95e1-4710-8af3-2b7afe310e48",
```

```
            "Username": "JohnD"
        },
        {
            "Id": "703e27b5-c9f0-4f1f-a239-64ccbb160125",
            "Arn": "arn:aws:connect:us-west-2:123456789012:instance/40c83b68-
ea62-414c-97bb-d018e39e158e/agent/703e27b5-c9f0-4f1f-a239-64ccbb160125",
            "Username": "JohnS"
        }
    ]
}
```

Per ulteriori informazioni, consulta Aggiungere utenti nella Guida per l'amministratore di Amazon Connect.

- Per i dettagli sull'API, consulta ListUsers AWS CLICommand Reference.

`update-contact-attributes`

Il seguente esempio di codice mostra come utilizzare`update-contact-attributes`.

AWS CLI

Come aggiornare l'attributo di un contatto

L'esempio `update-contact-attributes` seguente aggiorna l'attributo `greetingPlayed` per l'utente Amazon Connect specificato.

```
aws connect update-contact-attributes \
    --initial-contact-id 11111111-2222-3333-4444-12345678910 \
    --instance-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 \
    --attributes greetingPlayed=false
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta la pagina Uso degli attributi di contatto di Amazon Connect nella Guida per l'amministratore di Amazon Connect.

- Per i dettagli sull'API, consulta UpdateContactAttributes AWS CLICommand Reference.

`update-user-hierarchy`

Il seguente esempio di codice mostra come utilizzare`update-user-hierarchy`.

AWS CLI

Come aggiornare la gerarchia degli utenti

L'esempio `update-user-hierarchy` seguente aggiorna la gerarchia degli agenti per l'utente Amazon Connect specificato.

```
aws connect update-user-hierarchy \
    --hierarchy-group-id 12345678-a1b2-c3d4-e5f6-123456789abc \
    --user-id 87654321-2222-1234-1234-111234567891 \
    --instance-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Configurare le impostazioni dell'agente nella Guida per l'amministratore di Amazon Connect.

- Per i dettagli sull'API, consulta UpdateUserHierarchy AWS CLICommand Reference.

`update-user-identity-info`

Il seguente esempio di codice mostra come utilizzare`update-user-identity-info`.

AWS CLI

Come aggiornare le informazioni sull'identità di un utente

L'esempio `update-user-identity-info` seguente aggiorna le informazioni sull'identità per l'utente Amazon Connect specificato.

```
aws connect update-user-identity-info \
    --identity-info FirstName=Mary,LastName=Major,Email=marym@example.com \
    --user-id 87654321-2222-1234-1234-111234567891 \
    --instance-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Configurare le impostazioni dell'agente nella Guida per l'amministratore di Amazon Connect.

- Per i dettagli sull'API, consulta UpdateUserIdentityInfo AWS CLICommand Reference.

update-user-phone-config

Il seguente esempio di codice mostra come utilizzareupdate-user-phone-config.

AWS CLI

Come aggiornare la configurazione del telefono di un utente

L'esempio update-user-phone-config seguente aggiorna la configurazione del telefono per l'utente specificato.

```
aws connect update-user-phone-config \
    --phone-
config PhoneType=SOFT_PHONE,AutoAccept=false,AfterContactWorkTimeLimit=60,DeskPhoneNumber=
+18005551212 \
    --user-id 12345678-4444-3333-2222-111122223333 \
    --instance-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Configurare le impostazioni dell'agente nella Guida per l'amministratore di Amazon Connect.

- Per i dettagli sull'API, consulta UpdateUserPhoneConfig AWS CLICommand Reference.

update-user-routing-profile

Il seguente esempio di codice mostra come utilizzareupdate-user-routing-profile.

AWS CLI

Come aggiornare il profilo di routing di un utente

L'esempio update-user-routing-profile seguente aggiorna il profilo di routing per l'utente Amazon Connect specificato.

```
aws connect update-user-routing-profile \
    --routing-profile-id 12345678-1111-3333-2222-4444EXAMPLE \
    --user-id 87654321-2222-1234-1234-111234567891 \
    --instance-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Configurare le impostazioni dell'agente nella Guida per l'amministratore di Amazon Connect.

- Per i dettagli sull'API, consulta UpdateUserRoutingProfile AWS CLICommand Reference.

`update-user-security-profiles`

Il seguente esempio di codice mostra come utilizzare`update-user-security-profiles`.

AWS CLI

Come aggiornare i profili di sicurezza di un utente

L'esempio `update-user-security-profiles` seguente aggiorna il profilo di sicurezza per l'utente Amazon Connect specificato.

```
aws connect update-user-security-profiles \
    --security-profile-ids 12345678-1234-1234-1234-1234567892111 \
    --user-id 87654321-2222-1234-1234-111234567891 \
    --instance-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Assegnare le autorizzazioni: profili di sicurezza nella Guida per l'amministratore di Amazon Connect.

- Per i dettagli sull'API, consulta UpdateUserSecurityProfiles AWS CLICommand Reference.

AWS Cost and Usage Report esempi utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with AWS Cost and Usage Report.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`delete-report-definition`

Il seguente esempio di codice mostra come utilizzare`delete-report-definition`.

AWS CLI

Per eliminare un rapporto AWS sui costi e sull'utilizzo

Questo esempio elimina un rapporto AWS sui costi e sull'utilizzo.

Comando:

```
aws cur --region us-east-1 delete-report-definition --report-name "ExampleReport"
```

- Per i dettagli sull'API, vedere DeleteReportDefinitionin AWS CLI Command Reference.

`describe-report-definitions`

Il seguente esempio di codice mostra come utilizzare`describe-report-definitions`.

AWS CLI

Per recuperare un elenco di report AWS sui costi e sull'utilizzo

Questo esempio descrive un elenco di report AWS sui costi e sull'utilizzo di proprietà di un account.

Comando:

```
aws cur --region us-east-1 describe-report-definitions --max-items 5
```

Output:

```
  {
"ReportDefinitions": [
```

```
  {
      "ReportName": "ExampleReport",
      "Compression": "ZIP",
      "S3Region": "us-east-1",
      "Format": "textORcsv",
      "S3Prefix": "exampleprefix",
      "S3Bucket": "example-s3-bucket",
      "TimeUnit": "DAILY",
      "AdditionalArtifacts": [
          "REDSHIFT",
          "QUICKSIGHT"
      ],
      "AdditionalSchemaElements": [
          "RESOURCES"
      ]
  }
]
  }
```

- Per i dettagli sull'API, consulta DescribeReportDefinitions AWS CLICommand Reference.

`put-report-definition`

Il seguente esempio di codice mostra come utilizzare`put-report-definition`.

AWS CLI

Per creare un rapporto AWS sui costi e sull'utilizzo

L'`put-report-definition`esempio seguente crea un rapporto giornaliero AWS sui costi e sull'utilizzo che puoi caricare su Amazon Redshift o Amazon. QuickSight

```
aws cur put-report-definition --report-definition file://report-definition.json
```

Contenuto di `report-definition.json`:

```
{
    "ReportName": "ExampleReport",
    "TimeUnit": "DAILY",
    "Format": "textORcsv",
    "Compression": "ZIP",
    "AdditionalSchemaElements": [
```

```
        "RESOURCES"
    ],
    "S3Bucket": "example-s3-bucket",
    "S3Prefix": "exampleprefix",
    "S3Region": "us-east-1",
    "AdditionalArtifacts": [
        "REDSHIFT",
        "QUICKSIGHT"
    ]
}
```

- Per i dettagli sull'API, consulta PutReportDefinition AWS CLICommand Reference.

Esempi di utilizzo del servizio Cost Explorer AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando il servizio AWS Command Line Interface with Cost Explorer.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`get-cost-and-usage`

Il seguente esempio di codice mostra come utilizzare`get-cost-and-usage`.

AWS CLI

Come recuperare l'utilizzo S3 di un account per il mese di settembre 2017

L'esempio `get-cost-and-usage` seguente recupera l'utilizzo di S3 di un account per il mese di settembre 2017.

```
aws ce get-cost-and-usage \
    --time-period Start=2017-09-01,End=2017-10-01 \
    --granularity MONTHLY \
    --metrics "BlendedCost" "UnblendedCost" "UsageQuantity" \
    --group-by Type=DIMENSION,Key=SERVICE Type=TAG,Key=Environment \
    --filter file://filters.json
```

Contenuto di `filters.json`:

```
{
    "Dimensions": {
        "Key": "SERVICE",
        "Values": [
            "Amazon Simple Storage Service"
        ]
    }
}
```

Output:

```
{
    "GroupDefinitions": [
        {
            "Type": "DIMENSION",
            "Key": "SERVICE"
        },
        {
            "Type": "TAG",
            "Key": "Environment"
        }
    ],
    "ResultsByTime": [
        {
            "Estimated": false,
            "TimePeriod": {
                "Start": "2017-09-01",
                "End": "2017-10-01"
            },
            "Total": {},
            "Groups": [
                {
                    "Keys": [
```

```
                        "Amazon Simple Storage Service",
                        "Environment$"
                    ],
                    "Metrics": {
                        "BlendedCost": {
                            "Amount": "40.3527508453",
                            "Unit": "USD"
                        },
                        "UnblendedCost": {
                            "Amount": "40.3543773134",
                            "Unit": "USD"
                        },
                        "UsageQuantity": {
                            "Amount": "9312771.098461578",
                            "Unit": "N/A"
                        }
                    }
                },
                {
                    "Keys": [
                        "Amazon Simple Storage Service",
                        "Environment$Dev"
                    ],
                    "Metrics": {
                        "BlendedCost": {
                            "Amount": "0.2682364644",
                            "Unit": "USD"
                        },
                        "UnblendedCost": {
                            "Amount": "0.2682364644",
                            "Unit": "USD"
                        },
                        "UsageQuantity": {
                            "Amount": "22403.4395271182",
                            "Unit": "N/A"
                        }
                    }
                }
            ]
        }
    ]
}
```

- Per i dettagli sull'API, consulta GetCostAndUsage AWS CLICommand Reference.

`get-dimension-values`

Il seguente esempio di codice mostra come utilizzare`get-dimension-values`.

AWS CLI

Come recuperare i tag della dimensione SERVICE, con il valore "Elastic"

Questo esempio recupera i tag per la dimensione SERVICE, con il valore "Elastic" dal 1° gennaio 2017 al 18 maggio 2017.

Comando:

```
aws ce get-dimension-values --search-string Elastic --time-
period Start=2017-01-01,End=2017-05-18 --dimension SERVICE
```

Output:

```
{
   "TotalSize": 6,
   "DimensionValues": [
       {
           "Attributes": {},
           "Value": "Amazon ElastiCache"
       },
       {
           "Attributes": {},
           "Value": "EC2 - Other"
       },
       {
           "Attributes": {},
           "Value": "Amazon Elastic Compute Cloud - Compute"
       },
       {
           "Attributes": {},
           "Value": "Amazon Elastic Load Balancing"
       },
       {
           "Attributes": {},
           "Value": "Amazon Elastic MapReduce"
       },
       {
           "Attributes": {},
```

```
            "Value": "Amazon Elasticsearch Service"
        }
    ],
    "ReturnSize": 6
}
```

- Per i dettagli sull'API, consulta GetDimensionValues AWS CLICommand Reference.

`get-reservation-coverage`

Il seguente esempio di codice mostra come utilizzare`get-reservation-coverage`.

AWS CLI

Per recuperare la copertura delle prenotazioni per le istanze EC2 t2.nano nella regione us-east-1

Questo esempio recupera la copertura delle prenotazioni per le istanze EC2 t2.nano nella regione us-east-1 per luglio-settembre 2017.

Comando:

```
aws ce get-reservation-coverage --time-period Start=2017-07-01,End=2017-10-01 --
group-by Type=Dimension,Key=REGION --filter file://filters.json
```

filters.json:

```
{
    "And": [
      {
        "Dimensions": {
          "Key": "INSTANCE_TYPE",
          "Values": [
            "t2.nano"
          ]
        },
        "Dimensions": {
          "Key": "REGION",
          "Values": [
            "us-east-1"
          ]
        }
      }
```

```
    ]
 }
```

Output:

```
{
   "TotalSize": 6,
   "DimensionValues": [
       {
           "Attributes": {},
           "Value": "Amazon ElastiCache"
       },
       {
           "Attributes": {},
           "Value": "EC2 - Other"
       },
       {
           "Attributes": {},
           "Value": "Amazon Elastic Compute Cloud - Compute"
       },
       {
           "Attributes": {},
           "Value": "Amazon Elastic Load Balancing"
       },
       {
           "Attributes": {},
           "Value": "Amazon Elastic MapReduce"
       },
       {
           "Attributes": {},
           "Value": "Amazon Elasticsearch Service"
       }
   ],
   "ReturnSize": 6
}
```

- Per i dettagli sull'API, consulta Command Reference. GetReservationCoverageAWS CLI

`get-reservation-purchase-recommendation`

Il seguente esempio di codice mostra come utilizzare`get-reservation-purchase-recommendation`.

AWS CLI

Per recuperare i consigli di prenotazione per Partial Upfront EC2 RIs con un periodo di tre anni

L'`get-reservation-purchase-recommendation`esempio seguente recupera i consigli per le EC2 istanze Partial Upfront con una durata di tre anni, in base agli ultimi 60 giorni di utilizzo. EC2

```
aws ce get-reservation-purchase-recommendation \
    --service "Amazon Redshift" \
    --lookback-period-in-days SIXTY_DAYS \
    --term-in-years THREE_YEARS \
    --payment-option PARTIAL_UPFRONT
```

Output:

```
{
    "Recommendations": [],
    "Metadata": {
        "GenerationTimestamp": "2018-08-08T15:20:57Z",
        "RecommendationId": "00d59dde-a1ad-473f-8ff2-iexample3330b"
    }
}
```

- Per i dettagli sull'API, consulta GetReservationPurchaseRecommendationCommand Reference.AWS CLI

`get-reservation-utilization`

Il seguente esempio di codice mostra come utilizzare`get-reservation-utilization`.

AWS CLI

Come concedere l'autorizzazione a recuperare l'utilizzo della prenotazione per l'account

L'esempio `get-reservation-utilization` seguente recupera l'utilizzo delle istanze riservate per tutti i tipi di istanza t2.nano dal giorno 01-03-2018 al giorno 01-08-2018 per l'account.

```
aws ce get-reservation-utilization \
    --time-period Start=2018-03-01,End=2018-08-01 \
    --filter file://filters.json
```

Contenuto di `filters.json`:

```
{
    "Dimensions": {
        "Key": "INSTANCE_TYPE",
        "Values": [
            "t2.nano"
        ]
    }
}
```

Output:

```
{
    "Total": {
        "TotalAmortizedFee": "0",
        "UtilizationPercentage": "0",
        "PurchasedHours": "0",
        "NetRISavings": "0",
        "TotalActualHours": "0",
        "AmortizedRecurringFee": "0",
        "UnusedHours": "0",
        "TotalPotentialRISavings": "0",
        "OnDemandCostOfRIHoursUsed": "0",
        "AmortizedUpfrontFee": "0"
    },
    "UtilizationsByTime": []
}
```

- Per i dettagli sull'API, consulta GetReservationUtilization AWS CLICommand Reference.

`get-tags`

Il seguente esempio di codice mostra come utilizzare`get-tags`.

AWS CLI

Come recuperare chiavi e valori per un tag di allocazione dei costi

Questo esempio recupera tutti i tag di allocazione dei costi con la chiave "Project" e un valore che contiene "secretProject".

Comando:

```
aws ce get-tags --search-string secretProject --time-
period Start=2017-01-01,End=2017-05-18 --tag-key Project
```

Output:

```
{
  "ReturnSize": 2,
  "Tags": [
    "secretProject1",
    "secretProject2"
  ],
  "TotalSize": 2
}
```

- Per i dettagli sull'API, consulta GetTags AWS CLICommand Reference.

Esempi di utilizzo di Firehose AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with Firehose.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`list-delivery-streams`

Il seguente esempio di codice mostra come utilizzare`list-delivery-streams`.

AWS CLI

Come elencare i flussi di distribuzione disponibili

L'`list-delivery-streams`esempio seguente elenca i flussi di consegna disponibili nel tuo AWS account.

```
aws firehose list-delivery-streams
```

Output:

```
{
    "DeliveryStreamNames": [
        "my-stream"
    ],
    "HasMoreDeliveryStreams": false
}
```

Per ulteriori informazioni, consulta Creazione di un flusso di distribuzione Amazon Kinesis Data Firehose nella Guida per sviluppatori Amazon Kinesis Data Firehose.

- Per i dettagli sull'API, consulta ListDeliveryStreams AWS CLICommand Reference.

`put-record-batch`

Il seguente esempio di codice mostra come utilizzare`put-record-batch`.

AWS CLI

Come scrivere più record in un flusso

L'esempio `put-record-batch` seguente scrive tre record in un flusso. I dati sono codificati nel formato Base64.

```
aws firehose put-record-batch \
    --delivery-stream-name my-stream \
    --records file://records.json
```

Contenuto di `myfile.json`:

```
[
    {"Data": "Rmlyc3QgdGhpbmc="},
    {"Data": "U2Vjb25kIHRoaW5n"},
    {"Data": "VGhpcmQgdGhpbmc="}
]
```

Output:

```
{
    "FailedPutCount": 0,
    "Encrypted": false,
    "RequestResponses": [
        {
            "RecordId": "9D2OJ6t2EqCTZTXwGzeSv/EVHxRoRCw89xd+o3+sXg8DhYOaWKPSmZy/
CGlRVEys1u1xbeKh6VofEYKkoeiDrcjrxhQp9iF7sUW7pujiMEQ5LzlrzCkGosxQn
+3boDnURDEaD42V7GiixpOyLJkYZcae1i7HzlCEoy9LJhMr8EjDSi4Om/9Vc2uhwwuAtGE0XKpxJ2WD7ZRWtAnYlKAnv
        },
        {
            "RecordId": "jFirejqxCLlK5xjH/UNmlMVcjktEN76I7916X9PaZ
+PVaOSXDfU1WGOqEZhxq2js7xcZ552eoeDxsuTU1MSq9nZTbVfb6cQTIXnm/GsuF37Uhg67GKmR5z90l6XKJ
+/+pDloFv7Hh9a3oUS6wYm3DcNRLTHHAimANp1PhkQvWpvLRfzbuCUkBphR2QVzhP9OiHLbzGwy8/
DfH8sqWEUYASNJKS8GXP5s"
        },
        {
            "RecordId":
 "oy0amQ40o5Y2YV4vxzufdcMOOw6n3EPr3tpPJGoYVNKH4APPVqNcbUgefo1stEFRg4hTLrf2k6eliHu/9+YJ5R3iie
DTBt3qBlmTj7Xq8SKVbO1S7YvMTpWkMKA86f8JfmT8BMKoMb4XZS/sOkQLe+qh0sYKXWl"
        }
    ]
}
```

Per ulteriori informazioni, consulta Invio di dati a un flusso di distribuzione Amazon Kinesis Data Firehose nella Guida per sviluppatori di Amazon Kinesis Data Firehose.

- Per i dettagli sull'API, consulta PutRecordBatch AWS CLICommand Reference.

`put-record`

Il seguente esempio di codice mostra come utilizzare`put-record`.

AWS CLI

Come scrivere un record in un flusso

L'esempio `put-record` seguente scrive dati in un flusso. I dati sono codificati nel formato Base64.

```
aws firehose put-record \
    --delivery-stream-name my-stream \
```

```
    --record '{"Data":"SGVsbG8gd29ybGQ="}'
```

Output:

```
{
    "RecordId": "RjB5K/nnoGFHqwTsZlNd/
TTqvjE8V5dsyXZTQn2JXrdpMTOwssyEb6nfC8fwf1whhwnItt4mvrn+gsqeK5jB7QjuLg283+Ps4Sz/
j1Xujv31iDhnPdaLw4BOyM9Amv7PcCuB2079RuM0NhoakbyUymlwY8yt20G8X2420wu1jlFafhci4erAt7QhDEvpwuK8
    "Encrypted": false
}
```

Per ulteriori informazioni, consulta Invio di dati a un flusso di distribuzione Amazon Kinesis Data Firehose nella Guida per sviluppatori di Amazon Kinesis Data Firehose.

- Per i dettagli sull'API, consulta PutRecord AWS CLICommand Reference.

Esempi di utilizzo di Amazon Data Lifecycle Manager AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando Amazon Data Lifecycle Manager. AWS Command Line Interface

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`create-default-role`

Il seguente esempio di codice mostra come usare. `create-default-role`

AWS CLI

Come creare il ruolo IAM richiesto per Amazon DLM

L'`dlm create-default-role`esempio seguente crea il ruolo AWS DataLifecycleManagerDefaultRole predefinito per la gestione delle istantanee.

```
aws dlm create-default-role \
    --resource-type snapshot
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Ruoli di servizio predefiniti per Amazon Data Lifecycle Manager nella Guida per l'utente di Amazon Elastic Compute Cloud.

- Per i dettagli sull'API, vedere CreateDefaultRolein AWS CLI Command Reference.

`create-lifecycle-policy`

Il seguente esempio di codice mostra come utilizzare`create-lifecycle-policy`.

AWS CLI

Come creare una policy del ciclo di vita

L'esempio `create-lifecycle-policy` seguente crea una policy sul ciclo di vita che genera uno snapshot giornaliero dei volumi all'ora specificata. I tag specificati vengono aggiunti alle istantanee e i tag vengono inoltre copiati dal volume e aggiunti agli snapshot. Se la creazione di un nuovo snapshot supera il numero massimo specificato, lo snapshot meno recente viene eliminato.

```
aws dlm create-lifecycle-policy \
    --description "My first policy" \
    --state ENABLED \
    --execution-role-arn arn:aws:iam::12345678910:role/
AWSDataLifecycleManagerDefaultRole \
    --policy-details file://policyDetails.json
```

Contenuto di `policyDetails.json`:

```
{
    "ResourceTypes": [
        "VOLUME"
    ],
    "TargetTags": [
        {
```

```
            "Key": "costCenter",
            "Value": "115"
        }
    ],
    "Schedules":[
        {
            "Name": "DailySnapshots",
            "CopyTags": true,
            "TagsToAdd": [
                {
                    "Key": "type",
                     "Value": "myDailySnapshot"
                }
            ],
            "CreateRule": {
                "Interval": 24,
                "IntervalUnit": "HOURS",
                "Times": [
                    "03:00"
                ]
            },
            "RetainRule": {
                "Count":5
            }
        }
    ]
}
```

Output:

```
{
    "PolicyId": "policy-0123456789abcdef0"
}
```

- Per i dettagli sull'API, consulta CreateLifecyclePolicy AWS CLICommand Reference.

`delete-lifecycle-policy`

Il seguente esempio di codice mostra come utilizzare`delete-lifecycle-policy`.

AWS CLI

Come eliminare una policy per il ciclo di vita dei dati

L'esempio seguente elimina la policy sul ciclo di vita specificata:

```
aws dlm delete-lifecycle-policy --policy-id policy-0123456789abcdef0
```

- Per i dettagli sull'API, consulta DeleteLifecyclePolicy AWS CLICommand Reference.

`get-lifecycle-policies`

Il seguente esempio di codice mostra come utilizzare`get-lifecycle-policies`.

AWS CLI

Come ottenere un riepilogo delle policy sul ciclo di vita

L'esempio `get-lifecycle-policies` seguente elenca tutte le policy sul ciclo di vita.

```
aws dlm get-lifecycle-policies
```

Output:

```
{
    "Policies": [
        {
            "PolicyId": "policy-0123456789abcdef0",
            "Description": "My first policy",
            "State": "ENABLED"
        }
    ]
}
```

- Per i dettagli sull'API, consulta GetLifecyclePolicies AWS CLICommand Reference.

`get-lifecycle-policy`

Il seguente esempio di codice mostra come utilizzare`get-lifecycle-policy`.

AWS CLI

Come eliminare una policy sul ciclo di vita

L'esempio `get-lifecycle-policy` seguente mostra i dettagli della policy sul ciclo di vita specificata.

```
aws dlm get-lifecycle-policy \
    --policy-id policy-0123456789abcdef0
```

Output:

```
{
    "Policy": {
        "PolicyId": "policy-0123456789abcdef0",
        "Description": "My policy",
        "State": "ENABLED",
        "ExecutionRoleArn": "arn:aws:iam::123456789012:role/
AWSDataLifecycleManagerDefaultRole",
        "DateCreated": "2019-08-08T17:45:42Z",
        "DateModified": "2019-08-08T17:45:42Z",
        "PolicyDetails": {
            "PolicyType": "EBS_SNAPSHOT_MANAGEMENT",
            "ResourceTypes": [
                "VOLUME"
            ],
            "TargetTags": [
              {
                  "Key": "costCenter",
                  "Value": "115"
              }
            ],
            "Schedules": [
              {
                  "Name": "DailySnapshots",
                  "CopyTags": true,
                  "TagsToAdd": [
                    {
                        "Key": "type",
                        "Value": "myDailySnapshot"
                    }
                  ],
                  "CreateRule": {
                    "Interval": 24,
                    "IntervalUnit": "HOURS",
                    "Times": [
                        "03:00"
```

```
                    ]
                },
                "RetainRule": {
                    "Count": 5
                }
            }
        ]
    }
}
```

- Per i dettagli sull'API, consulta GetLifecyclePolicy AWS CLICommand Reference.

`update-lifecycle-policy`

Il seguente esempio di codice mostra come utilizzare`update-lifecycle-policy`.

AWS CLI

Esempio 1: come abilitare una policy del ciclo di vita

L'esempio `update-lifecycle-policy` seguente abilita la policy sul ciclo di vita specificata.

```
aws dlm update-lifecycle-policy \
    --policy-id policy-0123456789abcdef0 \
    --state ENABLED
```

Esempio 2: come disabilitare una policy del ciclo di vita

L'esempio `update-lifecycle-policy` seguente disabilita la policy sul ciclo di vita specificata.

```
aws dlm update-lifecycle-policy \
    --policy-id policy-0123456789abcdef0 \
    --state DISABLED
```

Esempio 3: come aggiornare i dettagli della policy sul ciclo di vita

L'esempio `update-lifecycle-policy` seguente aggiorna i tag di destinazione per la policy del ciclo di vita specificata.

```
aws dlm update-lifecycle-policy \
    --policy-id policy-0123456789abcdef0
```

```
    --policy-details file://policyDetails.json
```

Contenuto di `policyDetails.json`. Gli altri dettagli non citati in questo file non vengono modificati dal comando.

```
{
    "TargetTags": [
        {
            "Key": "costCenter",
            "Value": "120"
        },
        {
            "Key": "project",
            "Value": "lima"
        }
    ]
}
```

- Per i dettagli sull'API, consulta UpdateLifecyclePolicy AWS CLICommand Reference.

AWS Data Pipeline esempi utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with AWS Data Pipeline.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`activate-pipeline`

Il seguente esempio di codice mostra come utilizzare`activate-pipeline`.

AWS CLI

Come attivare una pipeline

Questo esempio attiva la pipeline specificata:

```
aws datapipeline activate-pipeline --pipeline-id df-00627471SOVYZEXAMPLE
```

Per attivare la pipeline a una data e a un'ora specifiche, utilizza il comando seguente:

```
aws datapipeline activate-pipeline --pipeline-id df-00627471SOVYZEXAMPLE --start-
timestamp 2015-04-07T00:00:00Z
```

- Per i dettagli sull'API, consulta ActivatePipeline AWS CLICommand Reference.

`add-tags`

Il seguente esempio di codice mostra come utilizzare`add-tags`.

AWS CLI

Come aggiungere un tag a una pipeline

Questo esempio aggiunge il tag specificato alla pipeline specificata:

```
aws datapipeline add-tags --pipeline-id df-00627471SOVYZEXAMPLE --
tags key=environment,value=production key=owner,value=sales
```

Per visualizzare i tag, utilizza il comando describe-pipelines. Ad esempio, i tag aggiunti al comando example vengono visualizzati come segue nell'output di describe-pipelines:

```
{
    ...
        "tags": [
            {
                "value": "production",
                "key": "environment"
            },
            {
                "value": "sales",
```

```
                "key": "owner"
            }
        ]
    ...
}
```

- Per i dettagli sull'API, consulta AddTags AWS CLICommand Reference.

`create-pipeline`

Il seguente esempio di codice mostra come utilizzare`create-pipeline`.

AWS CLI

Come creare una pipeline

Questo esempio crea una pipeline:

```
aws datapipeline create-pipeline --name my-pipeline --unique-id my-pipeline-token
```

Di seguito è riportato un output di esempio:

```
{
    "pipelineId": "df-00627471SOVYZEXAMPLE"
}
```

- Per i dettagli sull'API, consulta CreatePipeline AWS CLICommand Reference.

`deactivate-pipeline`

Il seguente esempio di codice mostra come utilizzare`deactivate-pipeline`.

AWS CLI

Come disattivare una pipeline

Questo esempio disattiva la pipeline specificata:

```
aws datapipeline deactivate-pipeline --pipeline-id df-00627471SOVYZEXAMPLE
```

Per disattivare la pipeline solo dopo che tutte le attività in esecuzione sono terminate, utilizza il comando seguente:

```
aws datapipeline deactivate-pipeline --pipeline-id df-00627471SOVYZEXAMPLE --no-cancel-active
```

- Per i dettagli sull'API, consulta DeactivatePipeline AWS CLICommand Reference.

`delete-pipeline`

Il seguente esempio di codice mostra come utilizzare`delete-pipeline`.

AWS CLI

Come eliminare una pipeline

Questo esempio elimina la pipeline specificata:

```
aws datapipeline delete-pipeline --pipeline-id df-00627471SOVYZEXAMPLE
```

- Per i dettagli sull'API, consulta DeletePipeline AWS CLICommand Reference.

`describe-pipelines`

Il seguente esempio di codice mostra come utilizzare`describe-pipelines`.

AWS CLI

Come descrivere le pipeline

Questo esempio descrive come eliminare la pipeline specificata:

```
aws datapipeline describe-pipelines --pipeline-ids df-00627471SOVYZEXAMPLE
```

Di seguito è riportato un output di esempio:

```
{
  "pipelineDescriptionList": [
      {
          "fields": [
              {
```

```
                "stringValue": "PENDING",
                "key": "@pipelineState"
            },
            {
                "stringValue": "my-pipeline",
                "key": "name"
            },
            {
                "stringValue": "2015-04-07T16:05:58",
                "key": "@creationTime"
            },
            {
                "stringValue": "df-00627471SOVYZEXAMPLE",
                "key": "@id"
            },
            {
                "stringValue": "123456789012",
                "key": "pipelineCreator"
            },
            {
                "stringValue": "PIPELINE",
                "key": "@sphere"
            },
            {
                "stringValue": "123456789012",
                "key": "@userId"
            },
            {
                "stringValue": "123456789012",
                "key": "@accountId"
            },
            {
                "stringValue": "my-pipeline-token",
                "key": "uniqueId"
            }
        ],
        "pipelineId": "df-00627471SOVYZEXAMPLE",
        "name": "my-pipeline",
        "tags": []
    }
  ]
}
```

- Per i dettagli sull'API, consulta DescribePipelines AWS CLICommand Reference.

`get-pipeline-definition`

Il seguente esempio di codice mostra come utilizzare`get-pipeline-definition`.

AWS CLI

Come ottenere una definizione di pipeline

L'esempio seguente riceve la definizione di pipeline per la pipeline specificata:

```
aws datapipeline get-pipeline-definition --pipeline-id df-00627471SOVYZEXAMPLE
```

Di seguito è riportato un output di esempio:

```
{
  "parameters": [
      {
          "type": "AWS::S3::ObjectKey",
          "id": "myS3OutputLoc",
          "description": "S3 output folder"
      },
      {
          "default": "s3://us-east-1.elasticmapreduce.samples/pig-apache-logs/data",
          "type": "AWS::S3::ObjectKey",
          "id": "myS3InputLoc",
          "description": "S3 input folder"
      },
      {
          "default": "grep -rc \"GET\" ${INPUT1_STAGING_DIR}/* >
 ${OUTPUT1_STAGING_DIR}/output.txt",
          "type": "String",
          "id": "myShellCmd",
          "description": "Shell command to run"
      }
  ],
  "objects": [
      {
          "type": "Ec2Resource",
          "terminateAfter": "20 Minutes",
          "instanceType": "t1.micro",
          "id": "EC2ResourceObj",
          "name": "EC2ResourceObj"
      },
      {
```

```
          "name": "Default",
          "failureAndRerunMode": "CASCADE",
          "resourceRole": "DataPipelineDefaultResourceRole",
          "schedule": {
              "ref": "DefaultSchedule"
          },
          "role": "DataPipelineDefaultRole",
          "scheduleType": "cron",
          "id": "Default"
      },
      {
          "directoryPath": "#{myS3OutputLoc}/#{format(@scheduledStartTime, 'YYYY-MM-
dd-HH-mm-ss')}",
          "type": "S3DataNode",
          "id": "S3OutputLocation",
          "name": "S3OutputLocation"
      },
      {
          "directoryPath": "#{myS3InputLoc}",
          "type": "S3DataNode",
          "id": "S3InputLocation",
          "name": "S3InputLocation"
      },
      {
          "startAt": "FIRST_ACTIVATION_DATE_TIME",
          "name": "Every 15 minutes",
          "period": "15 minutes",
          "occurrences": "4",
          "type": "Schedule",
          "id": "DefaultSchedule"
      },
      {
          "name": "ShellCommandActivityObj",
          "command": "#{myShellCmd}",
          "output": {
              "ref": "S3OutputLocation"
          },
          "input": {
              "ref": "S3InputLocation"
          },
          "stage": "true",
          "type": "ShellCommandActivity",
          "id": "ShellCommandActivityObj",
          "runsOn": {
```

```
                "ref": "EC2ResourceObj"
            }
        }
    ],
    "values": {
        "myS3OutputLoc": "s3://amzn-s3-demo-bucket/",
        "myS3InputLoc": "s3://us-east-1.elasticmapreduce.samples/pig-apache-logs/
data",
        "myShellCmd": "grep -rc \"GET\" ${INPUT1_STAGING_DIR}/* >
 ${OUTPUT1_STAGING_DIR}/output.txt"
    }
}
```

- Per i dettagli sull'API, consulta GetPipelineDefinition AWS CLICommand Reference.

`list-pipelines`

Il seguente esempio di codice mostra come utilizzare`list-pipelines`.

AWS CLI

Come elencare le pipeline

Questo esempio elenca le pipeline:

```
aws datapipeline list-pipelines
```

Di seguito è riportato un output di esempio:

```
{
    "pipelineIdList": [
        {
            "id": "df-00627471SOVYZEXAMPLE",
            "name": "my-pipeline"
        },
        {
            "id": "df-09028963KNVMREXAMPLE",
            "name": "ImportDDB"
        },
        {
            "id": "df-0870198233ZYVEXAMPLE",
            "name": "CrossRegionDDB"
```

```
        },
        {
            "id": "df-00189603TB4MZEXAMPLE",
            "name": "CopyRedshift"
        }
    ]
}
```

- Per i dettagli sull'API, consulta ListPipelines AWS CLICommand Reference.

`list-runs`

Il seguente esempio di codice mostra come utilizzare`list-runs`.

AWS CLI

Esempio 1: come elencare le esecuzioni della pipeline

L'esempio `list-runs` seguente elenca le esecuzioni per la pipeline specificata.

```
aws datapipeline list-runs --pipeline-id df-00627471SOVYZEXAMPLE
```

Output:

```
    Name                          Scheduled Start         Status                      ID
                                                 Started                 Ended

 ---------------------------------------------------------------------------------------
1.  EC2ResourceObj              2015-04-12T17:33:02    CREATING
 @EC2ResourceObj_2015-04-12T17:33:02             2015-04-12T17:33:10
2.  S3InputLocation             2015-04-12T17:33:02    FINISHED
      @S3InputLocation_2015-04-12T17:33:02            2015-04-12T17:33:09
 2015-04-12T17:33:09
3.  S3OutputLocation            2015-04-12T17:33:02    WAITING_ON_DEPENDENCIES
 @S3OutputLocation_2015-04-12T17:33:02           2015-04-12T17:33:09
4.  ShellCommandActivityObj     2015-04-12T17:33:02    WAITING_FOR_RUNNER
 @ShellCommandActivityObj_2015-04-12T17:33:02    2015-04-12T17:33:09
```

Esempio 2: come elencare le esecuzioni della pipeline tra le date specificate

L'esempio `list-runs` seguente utilizza `--start-interval` per specificare le date da includere nell'output.

```
aws datapipeline list-runs --pipeline-id df-01434553B58A2SHZUKO5 --start-
interval 2017-10-07T00:00:00,2017-10-08T00:00:00
```

- Per i dettagli sull'API, consulta ListRuns AWS CLICommand Reference.

`put-pipeline-definition`

Il seguente esempio di codice mostra come utilizzare`put-pipeline-definition`.

AWS CLI

Come caricare una definizione di pipeline

Questo esempio carica la definizione della pipeline specificata nella pipeline specificata:

```
aws datapipeline put-pipeline-definition --pipeline-id df-00627471SOVYZEXAMPLE --
pipeline-definition file://my-pipeline-definition.json
```

Di seguito è riportato un output di esempio:

```
{
  "validationErrors": [],
  "errored": false,
  "validationWarnings": []
}
```

- Per i dettagli sull'API, consulta PutPipelineDefinition AWS CLICommand Reference.

`remove-tags`

Il seguente esempio di codice mostra come utilizzare`remove-tags`.

AWS CLI

Come rimuovere un tag da una pipeline

Questo esempio rimuove il tag specificato dalla pipeline specificata:

```
aws datapipeline remove-tags --pipeline-id df-00627471SOVYZEXAMPLE --tag-
keys environment
```

- Per i dettagli sull'API, consulta RemoveTags AWS CLICommand Reference.

DataSync esempi utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with DataSync.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`update-location-azure-blob`

Il seguente esempio di codice mostra come utilizzare`update-location-azure-blob`.

AWS CLI

Come aggiornare la località di trasferimento con un nuovo agente

L'`update-location-object-storage`esempio seguente aggiorna la DataSync posizione per Microsoft Azure Blob Storage con un nuovo agente.

```
aws datasync update-location-azure-blob \
    --location-arn arn:aws:datasync:us-west-2:123456789012:location/loc-
abcdef01234567890 \
    --agent-arns arn:aws:datasync:us-west-2:123456789012:agent/
agent-1234567890abcdef0 \
    --sas-configuration '{ \
        "Token": "sas-token-for-azure-blob-storage-access" \
    }'
```

Questo comando non produce alcun output.

Per ulteriori informazioni, vedere Sostituzione dell'agente nella Guida per l'AWS DataSync utente.

- Per i dettagli sull'API, consulta UpdateLocationAzureBlob AWS CLICommand Reference.

`update-location-hdfs`

Il seguente esempio di codice mostra come utilizzare`update-location-hdfs`.

AWS CLI

Come aggiornare la località di trasferimento con un nuovo agente

L'`update-location-hdfs`esempio seguente aggiorna la posizione DataSync HDFS con un nuovo agente. Sono necessarie le opzioni `--kerberos-keytab` e `--kerberos-krb5-conf` solo se il cluster HDFS utilizza l'autenticazione Kerberos.

```
aws datasync update-location-hdfs \
    --location-arn arn:aws:datasync:us-west-2:123456789012:location/loc-
abcdef01234567890 \
    --agent-arns arn:aws:datasync:us-west-2:123456789012:agent/
agent-1234567890abcdef0 \
    --kerberos-keytab file://hdfs.keytab
    --kerberos-krb5-conf file://krb5.conf
```

Contenuto di `hdfs.keytab`:

```
N/A. The content of this file is encrypted and not human readable.
```

Contenuto di `krb5.conf`:

```
[libdefaults]
    default_realm = EXAMPLE.COM
    dns_lookup_realm = false
    dns_lookup_kdc = false
    rdns = true
    ticket_lifetime = 24h
    forwardable = true
    udp_preference_limit = 1000000
    default_tkt_enctypes = aes256-cts-hmac-sha1-96 aes128-cts-hmac-sha1-96 des3-cbc-
sha1
    default_tgs_enctypes = aes256-cts-hmac-sha1-96 aes128-cts-hmac-sha1-96 des3-cbc-
sha1
```

```
    permitted_enctypes = aes256-cts-hmac-sha1-96 aes128-cts-hmac-sha1-96 des3-cbc-
sha1

[realms]
    EXAMPLE.COM = {
      kdc = kdc1.example.com
      admin_server = krbadmin.example.com
      default_domain = example.com
    }

[domain_realm]
    .example.com = EXAMPLE.COM
    example.com = EXAMPLE.COM

[logging]
    kdc = FILE:/var/log/krb5kdc.log
    admin_server = FILE:/var/log/kerberos/kadmin.log
    default = FILE:/var/log/krb5libs.log
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Sostituzione dell'agente nella Guida per l'AWS DataSync utente.

- Per i dettagli sull'API, consulta UpdateLocationHdfs AWS CLICommand Reference.

`update-location-nfs`

Il seguente esempio di codice mostra come utilizzare`update-location-nfs`.

AWS CLI

Come aggiornare la località di trasferimento con un nuovo agente

L'`update-location-nfs`esempio seguente aggiorna la posizione DataSync NFS con un nuovo agente.

```
aws datasync update-location-nfs \
    --location-arn arn:aws:datasync:us-west-2:123456789012:location/loc-
abcdef01234567890 \
    --on-prem-config AgentArns=arn:aws:datasync:us-west-2:123456789012:agent/
agent-1234567890abcdef0
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Sostituzione dell'agente nella Guida per l'AWS DataSync utente.

- Per i dettagli sull'API, consulta UpdateLocationNfs AWS CLICommand Reference.

`update-location-object-storage`

Il seguente esempio di codice mostra come utilizzare`update-location-object-storage`.

AWS CLI

Come aggiornare la località di trasferimento con un nuovo agente

L'`update-location-object-storage`esempio seguente aggiorna la posizione di archiviazione DataSync degli oggetti con un nuovo agente.

```
aws datasync update-location-object-storage \
    --location-arn arn:aws:datasync:us-west-2:123456789012:location/loc-
abcdef01234567890 \
    --agent-arns arn:aws:datasync:us-west-2:123456789012:agent/
agent-1234567890abcdef0 \
    --secret-key secret-key-for-object-storage
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consultate Sostituzione dell'agente nella Guida AWS DataSync per l'utente.

- Per i dettagli sull'API, consulta UpdateLocationObjectStorage AWS CLICommand Reference.

`update-location-smb`

Il seguente esempio di codice mostra come utilizzare`update-location-smb`.

AWS CLI

Come aggiornare la località di trasferimento con un nuovo agente

L'`update-location-smb`esempio seguente aggiorna la sede DataSync SMB con un nuovo agente.

```
aws datasync update-location-smb \
    --location-arn arn:aws:datasync:us-west-2:123456789012:location/loc-
abcdef01234567890 \
    --agent-arns arn:aws:datasync:us-west-2:123456789012:agent/
agent-1234567890abcdef0 \
    --password smb-file-server-password
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Sostituzione dell'agente nella Guida per l'AWS DataSync utente.

- Per i dettagli sull'API, consulta UpdateLocationSmb AWS CLICommand Reference.

Esempi di utilizzo di DAX AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando il AWS Command Line Interface con DAX.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`create-cluster`

Il seguente esempio di codice mostra come utilizzare`create-cluster`.

AWS CLI

Come creare un cluster DAX

L'esempio `create-cluster` seguente crea un cluster DAX con le impostazioni specificate.

```
aws dax create-cluster \
    --cluster-name daxcluster \
    --node-type dax.r4.large \
    --replication-factor 3 \
    --iam-role-arn roleARN  \
    --sse-specification Enabled=true
```

Output:

```
{
    "Cluster": {
        "ClusterName": "daxcluster",
        "ClusterArn": "arn:aws:dax:us-west-2:123456789012:cache/daxcluster",
        "TotalNodes": 3,
        "ActiveNodes": 0,
        "NodeType": "dax.r4.large",
        "Status": "creating",
        "ClusterDiscoveryEndpoint": {
            "Port": 8111
        },
        "PreferredMaintenanceWindow": "thu:13:00-thu:14:00",
        "SubnetGroup": "default",
        "SecurityGroups": [
            {
                "SecurityGroupIdentifier": "sg-1af6e36e",
                "Status": "active"
            }
        ],
        "IamRoleArn": "arn:aws:iam::123456789012:role/
DAXServiceRoleForDynamoDBAccess",
        "ParameterGroup": {
            "ParameterGroupName": "default.dax1.0",
            "ParameterApplyStatus": "in-sync",
            "NodeIdsToReboot": []
        },
        "SSEDescription": {
            "Status": "ENABLED"
        }
    }
}
```

Per ulteriori informazioni, consulta Fase 3: creare un cluster DAX nella Guida per gli sviluppatori di Amazon DynamoDB.

- Per i dettagli sull'API, consulta CreateCluster AWS CLICommand Reference.

`create-parameter-group`

Il seguente esempio di codice mostra come utilizzare`create-parameter-group`.

AWS CLI

Come creare un gruppo di parametri

Il seguente esempio `` create-parameter-group`` crea un gruppo di parametri con le impostazioni specificate.

```
aws dax create-parameter-group \
    --parameter-group-name daxparametergroup \
    --description "A new parameter group"
```

Output:

```
{
    "ParameterGroup": {
        "ParameterGroupName": "daxparametergroup",
        "Description": "A new parameter group"
    }
}
```

Per ulteriori informazioni, consulta Gestione dei cluster DAX nella Guida per gli sviluppatori di Amazon DynamoDB.

- Per i dettagli sull'API, consulta AWS CLI Command CreateParameterGroupReference.

`create-subnet-group`

Il seguente esempio di codice mostra come utilizzare`create-subnet-group`.

AWS CLI

Come creare un gruppo di sottoreti DAX

L'esempio `create-subnet-group` seguente crea un gruppo di sottoreti con le impostazioni specificate.

```
aws dax create-subnet-group \
    --subnet-group-name daxSubnetGroup \
    --subnet-ids subnet-11111111 subnet-22222222
```

Output:

```
{
    "SubnetGroup": {
        "SubnetGroupName": "daxSubnetGroup",
        "VpcId": "vpc-05a1fa8e00c325226",
        "Subnets": [
            {
                "SubnetIdentifier": "subnet-11111111",
                "SubnetAvailabilityZone": "us-west-2b"
            },
            {
                "SubnetIdentifier": "subnet-22222222",
                "SubnetAvailabilityZone": "us-west-2c"
            }
        ]
    }
}
```

Per ulteriori informazioni, consulta Fase 2: creare un gruppo di sottoreti nella Guida per gli sviluppatori di Amazon DynamoDB.

- Per i dettagli sull'API, consulta CreateSubnetGroup AWS CLICommand Reference.

`decrease-replication-factor`

Il seguente esempio di codice mostra come utilizzare`decrease-replication-factor`.

AWS CLI

Come rimuovere uno o più nodi dal cluster

L'esempio `decrease-replication-factor` seguente riduce a uno il numero di nodi nel cluster DAX specificato.

```
aws dax decrease-replication-factor \
    --cluster-name daxcluster \
    --new-replication-factor 1
```

Output:

```
{
    "Cluster": {
        "ClusterName": "daxcluster",
        "ClusterArn": "arn:aws:dax:us-west-2:123456789012:cache/daxcluster",
        "TotalNodes": 3,
        "ActiveNodes": 3,
        "NodeType": "dax.r4.large",
        "Status": "modifying",
        "ClusterDiscoveryEndpoint": {
            "Address": "daxcluster.ey3o9d.clustercfg.dax.usw2.cache.amazonaws.com",
            "Port": 8111
        },
        "Nodes": [
            {
                "NodeId": "daxcluster-a",
                "Endpoint": {
                    "Address": "daxcluster-
a.ey3o9d.0001.dax.usw2.cache.amazonaws.com",
                    "Port": 8111
                },
                "NodeCreateTime": 1576625059.509,
                "AvailabilityZone": "us-west-2c",
                "NodeStatus": "available",
                "ParameterGroupStatus": "in-sync"
            },
            {
                "NodeId": "daxcluster-b",
                "Endpoint": {
                    "Address": "daxcluster-
b.ey3o9d.0001.dax.usw2.cache.amazonaws.com",
                    "Port": 8111
                },
                "NodeCreateTime": 1576625059.509,
                "AvailabilityZone": "us-west-2a",
                "NodeStatus": "available",
                "ParameterGroupStatus": "in-sync"
            },
```

```
            {
                "NodeId": "daxcluster-c",
                "Endpoint": {
                    "Address": "daxcluster-
c.ey3o9d.0001.dax.usw2.cache.amazonaws.com",
                    "Port": 8111
                },
                "NodeCreateTime": 1576625059.509,
                "AvailabilityZone": "us-west-2b",
                "NodeStatus": "available",
                "ParameterGroupStatus": "in-sync"
            }
        ],
        "PreferredMaintenanceWindow": "thu:13:00-thu:14:00",
        "SubnetGroup": "default",
        "SecurityGroups": [
            {
                "SecurityGroupIdentifier": "sg-1af6e36e",
                "Status": "active"
            }
        ],
        "IamRoleArn": "arn:aws:iam::123456789012:role/
DAXServiceRoleForDynamoDBAccess",
        "ParameterGroup": {
            "ParameterGroupName": "default.dax1.0",
            "ParameterApplyStatus": "in-sync",
            "NodeIdsToReboot": []
        },
        "SSEDescription": {
            "Status": "ENABLED"
        }
    }
}
```

Per ulteriori informazioni, consulta Gestione dei cluster DAX nella Guida per gli sviluppatori di Amazon DynamoDB.

- Per i dettagli sull'API, consulta DecreaseReplicationFactor AWS CLICommand Reference.

`delete-cluster`

Il seguente esempio di codice mostra come utilizzare`delete-cluster`.

AWS CLI

Per eliminare un cluster DAX

L'esempio `delete-cluster` seguente elimina il cluster DAX specificato.

```
aws dax delete-cluster \
    --cluster-name daxcluster
```

Output:

```
{
    "Cluster": {
        "ClusterName": "daxcluster",
        "ClusterArn": "arn:aws:dax:us-west-2:123456789012:cache/daxcluster",
        "TotalNodes": 3,
        "ActiveNodes": 0,
        "NodeType": "dax.r4.large",
        "Status": "deleting",
        "ClusterDiscoveryEndpoint": {
            "Address": "dd.ey3o9d.clustercfg.dax.usw2.cache.amazonaws.com",
            "Port": 8111
        },
        "PreferredMaintenanceWindow": "fri:06:00-fri:07:00",
        "SubnetGroup": "default",
        "SecurityGroups": [
            {
                "SecurityGroupIdentifier": "sg-1af6e36e",
                "Status": "active"
            }
        ],
        "IamRoleArn": "arn:aws:iam::123456789012:role/
DAXServiceRoleForDynamoDBAccess",
        "ParameterGroup": {
            "ParameterGroupName": "default.dax1.0",
            "ParameterApplyStatus": "in-sync",
            "NodeIdsToReboot": []
        },
        "SSEDescription": {
            "Status": "ENABLED"
        }
    }
}
```

Per ulteriori informazioni, consulta Gestione dei cluster DAX nella Guida per gli sviluppatori di Amazon DynamoDB.

- Per i dettagli sull'API, vedere DeleteClusterin AWS CLI Command Reference.

`delete-parameter-group`

Il seguente esempio di codice mostra come utilizzare`delete-parameter-group`.

AWS CLI

Come eliminare un gruppo di parametri

L'esempio `delete-parameter-group` seguente elimina il gruppo di parametri DAX specificato.

```
aws dax  delete-parameter-group \
    --parameter-group-name daxparametergroup
```

Output:

```
{
    "DeletionMessage": "Parameter group daxparametergroup has been deleted."
}
```

Per ulteriori informazioni, consulta Gestione dei cluster DAX nella Guida per gli sviluppatori di Amazon DynamoDB.

- Per i dettagli sull'API, consulta DeleteParameterGroup AWS CLICommand Reference.

`delete-subnet-group`

Il seguente esempio di codice mostra come utilizzare`delete-subnet-group`.

AWS CLI

Come eliminare un gruppo di sottoreti

L'esempio `delete-subnet-group` seguente elimina il gruppo di sottoreti DAX specificato.

```
aws dax delete-subnet-group \
    --subnet-group-name daxSubnetGroup
```

Output:

```
{
    "DeletionMessage": "Subnet group daxSubnetGroup has been deleted."
}
```

Per ulteriori informazioni, consulta Gestione dei cluster DAX nella Guida per gli sviluppatori di Amazon DynamoDB.

- Per i dettagli sull'API, consulta DeleteSubnetGroup AWS CLICommand Reference.

`describe-clusters`

Il seguente esempio di codice mostra come utilizzare`describe-clusters`.

AWS CLI

Come restituire informazioni su tutti i cluster DAX sottoposti a provisioning

L'esempio `describe-clusters` seguente visualizza i dettagli su tutti i cluster DAX di cui è stato effettuato il provisioning.

```
aws dax describe-clusters
```

Output:

```
{
    "Clusters": [
        {
            "ClusterName": "daxcluster",
            "ClusterArn": "arn:aws:dax:us-west-2:123456789012:cache/daxcluster",
            "TotalNodes": 1,
            "ActiveNodes": 1,
            "NodeType": "dax.r4.large",
            "Status": "available",
            "ClusterDiscoveryEndpoint": {
                "Address":
 "daxcluster.ey3o9d.clustercfg.dax.usw2.cache.amazonaws.com",
                "Port": 8111
            },
            "Nodes": [
```

```
                {
                    "NodeId": "daxcluster-a",
                    "Endpoint": {
                        "Address": "daxcluster-
a.ey3o9d.0001.dax.usw2.cache.amazonaws.com",
                        "Port": 8111
                    },
                    "NodeCreateTime": 1576625059.509,
                    "AvailabilityZone": "us-west-2c",
                    "NodeStatus": "available",
                    "ParameterGroupStatus": "in-sync"
                }
            ],
            "PreferredMaintenanceWindow": "thu:13:00-thu:14:00",
            "SubnetGroup": "default",
            "SecurityGroups": [
                {
                    "SecurityGroupIdentifier": "sg-1af6e36e",
                    "Status": "active"
                }
            ],
            "IamRoleArn": "arn:aws:iam::123456789012:role/
DAXServiceRoleForDynamoDBAccess",
            "ParameterGroup": {
                "ParameterGroupName": "default.dax1.0",
                "ParameterApplyStatus": "in-sync",
                "NodeIdsToReboot": []
            },
            "SSEDescription": {
                "Status": "ENABLED"
            }
        }
    ]
}
```

Per ulteriori informazioni, consulta Gestione dei cluster DAX nella Guida per gli sviluppatori di Amazon DynamoDB.

- Per i dettagli sull'API, consulta DescribeClusters AWS CLICommand Reference.

`describe-default-parameters`

Il seguente esempio di codice mostra come utilizzare`describe-default-parameters`.

AWS CLI

Come restituire le informazioni del parametro di sistema predefinito per DAX

L'esempio `describe-default-parameters` seguente visualizza le informazioni sul parametro di sistema di default per DAX.

```
aws dax describe-default-parameters
```

Output:

```
{
    "Parameters": [
        {
            "ParameterName": "query-ttl-millis",
            "ParameterType": "DEFAULT",
            "ParameterValue": "300000",
            "NodeTypeSpecificValues": [],
            "Description": "Duration in milliseconds for queries to remain cached",
            "Source": "user",
            "DataType": "integer",
            "AllowedValues": "0-",
            "IsModifiable": "TRUE",
            "ChangeType": "IMMEDIATE"
        },
        {
            "ParameterName": "record-ttl-millis",
            "ParameterType": "DEFAULT",
            "ParameterValue": "300000",
            "NodeTypeSpecificValues": [],
            "Description": "Duration in milliseconds for records to remain valid in
 cache (Default: 0 = infinite)",
            "Source": "user",
            "DataType": "integer",
            "AllowedValues": "0-",
            "IsModifiable": "TRUE",
            "ChangeType": "IMMEDIATE"
        }
    ]
}
```

Per ulteriori informazioni, consulta Gestione dei cluster DAX nella Guida per gli sviluppatori di Amazon DynamoDB.

- Per i dettagli sull'API, consulta DescribeDefaultParameters AWS CLICommand Reference.

`describe-events`

Il seguente esempio di codice mostra come utilizzare`describe-events`.

AWS CLI

Come restituire gli eventi correlati ai cluster DAX e ai gruppi di parametri

L'esempio `describe-events` seguente visualizza i dettagli degli eventi correlati ai cluster DAX e ai gruppi di parametri.

```
aws dax describe-events
```

Output:

```
{
    "Events": [
        {
            "SourceName": "daxcluster",
            "SourceType": "CLUSTER",
            "Message": "Cluster deleted.",
            "Date": 1576702736.706
        },
        {
            "SourceName": "daxcluster",
            "SourceType": "CLUSTER",
            "Message": "Removed node daxcluster-b.",
            "Date": 1576702691.738
        },
        {
            "SourceName": "daxcluster",
            "SourceType": "CLUSTER",
            "Message": "Removed node daxcluster-a.",
            "Date": 1576702633.498
        },
        {
            "SourceName": "daxcluster",
            "SourceType": "CLUSTER",
            "Message": "Removed node daxcluster-c.",
            "Date": 1576702631.329
```

```
        },
        {
            "SourceName": "daxcluster",
            "SourceType": "CLUSTER",
            "Message": "Cluster created.",
            "Date": 1576626560.057
        }
    ]
}
```

Per ulteriori informazioni, consulta Gestione dei cluster DAX nella Guida per gli sviluppatori di Amazon DynamoDB.

- Per i dettagli sull'API, consulta DescribeEvents AWS CLICommand Reference.

`describe-parameter-groups`

Il seguente esempio di codice mostra come utilizzare`describe-parameter-groups`.

AWS CLI

Come descrivere i gruppi di parametri definiti in DAX

L'esempio `describe-parameter-groups` seguente recupera i dettagli dei gruppi di parametri definiti in DAX.

```
aws dax describe-parameter-groups
```

Output:

```
{
    "ParameterGroups": [
        {
            "ParameterGroupName": "default.dax1.0",
            "Description": "Default parameter group for dax1.0"
        }
    ]
}
```

Per ulteriori informazioni, consulta Gestione dei cluster DAX nella Guida per gli sviluppatori di Amazon DynamoDB.

- Per i dettagli sull'API, consulta DescribeParameterGroups AWS CLICommand Reference.

`describe-parameters`

Il seguente esempio di codice mostra come utilizzare`describe-parameters`.

AWS CLI

Come descrivere i parametri definiti in un gruppo di parametri DAX

L'esempio `describe-parameters` seguente recupera i dettagli dei parametri definiti nel gruppo di parametri DAX specificato.

```
aws dax describe-parameters \
    --parameter-group-name default.dax1.0
```

Output:

```
{
    "Parameters": [
        {
            "ParameterName": "query-ttl-millis",
            "ParameterType": "DEFAULT",
            "ParameterValue": "300000",
            "NodeTypeSpecificValues": [],
            "Description": "Duration in milliseconds for queries to remain cached",
            "Source": "user",
            "DataType": "integer",
            "AllowedValues": "0-",
            "IsModifiable": "TRUE",
            "ChangeType": "IMMEDIATE"
        },
        {
            "ParameterName": "record-ttl-millis",
            "ParameterType": "DEFAULT",
            "ParameterValue": "300000",
            "NodeTypeSpecificValues": [],
            "Description": "Duration in milliseconds for records to remain valid in
 cache (Default: 0 = infinite)",
            "Source": "user",
            "DataType": "integer",
            "AllowedValues": "0-",
            "IsModifiable": "TRUE",
            "ChangeType": "IMMEDIATE"
        }
```

```
    ]
}
```

Per ulteriori informazioni, consulta Gestione dei cluster DAX nella Guida per gli sviluppatori di Amazon DynamoDB.

- Per i dettagli sull'API, consulta DescribeParameters AWS CLICommand Reference.

`describe-subnet-groups`

Il seguente esempio di codice mostra come utilizzare`describe-subnet-groups`.

AWS CLI

Come descrivere i gruppi di sottoreti definiti in DAX

L'esempio `describe-subnet-groups` seguente recupera i dettagli dei gruppi di sottoreti definiti in DAX.

```
aws dax describe-subnet-groups
```

Output:

```
{
    "SubnetGroups": [
        {
            "SubnetGroupName": "default",
            "Description": "Default CacheSubnetGroup",
            "VpcId": "vpc-ee70a196",
            "Subnets": [
                {
                    "SubnetIdentifier": "subnet-874953af",
                    "SubnetAvailabilityZone": "us-west-2d"
                },
                {
                    "SubnetIdentifier": "subnet-bd3d1fc4",
                    "SubnetAvailabilityZone": "us-west-2a"
                },
                {
                    "SubnetIdentifier": "subnet-72c2ff28",
                    "SubnetAvailabilityZone": "us-west-2c"
                },
                {
```

```
                    "SubnetIdentifier": "subnet-09e6aa42",
                    "SubnetAvailabilityZone": "us-west-2b"
                }
            ]
        }
    ]
}
```

Per ulteriori informazioni, consulta Gestione dei cluster DAX nella Guida per gli sviluppatori di Amazon DynamoDB.

- Per i dettagli sull'API, consulta DescribeSubnetGroups AWS CLICommand Reference.

`increase-replication-factor`

Il seguente esempio di codice mostra come utilizzare`increase-replication-factor`.

AWS CLI

Come aumentare il fattore di replica per un cluster DAX

L'esempio `increase-replication-factor` seguente aumenta il fattore di replica del cluster DAX specificato a 3.

```
aws dax increase-replication-factor \
    --cluster-name daxcluster \
    --new-replication-factor 3
```

Output:

```
{
    "Cluster": {
        "ClusterName": "daxcluster",
        "ClusterArn": "arn:aws:dax:us-west-2:123456789012:cache/daxcluster",
        "TotalNodes": 3,
        "ActiveNodes": 1,
        "NodeType": "dax.r4.large",
        "Status": "modifying",
        "ClusterDiscoveryEndpoint": {
            "Address": "daxcluster.ey3o9d.clustercfg.dax.usw2.cache.amazonaws.com",
            "Port": 8111
        },
        "Nodes": [
```

```
            {
                "NodeId": "daxcluster-a",
                "Endpoint": {
                    "Address": "daxcluster-
a.ey3o9d.0001.dax.usw2.cache.amazonaws.com",
                    "Port": 8111
                },
                "NodeCreateTime": 1576625059.509,
                "AvailabilityZone": "us-west-2c",
                "NodeStatus": "available",
                "ParameterGroupStatus": "in-sync"
            },
            {
                "NodeId": "daxcluster-b",
                "NodeStatus": "creating"
            },
            {
                "NodeId": "daxcluster-c",
                "NodeStatus": "creating"
            }
        ],
        "PreferredMaintenanceWindow": "thu:13:00-thu:14:00",
        "SubnetGroup": "default",
        "SecurityGroups": [
            {
                "SecurityGroupIdentifier": "sg-1af6e36e",
                "Status": "active"
            }
        ],
        "IamRoleArn": "arn:aws:iam::123456789012:role/
DAXServiceRoleForDynamoDBAccess",
        "ParameterGroup": {
            "ParameterGroupName": "default.dax1.0",
            "ParameterApplyStatus": "in-sync",
            "NodeIdsToReboot": []
        },
        "SSEDescription": {
            "Status": "ENABLED"
        }
    }
}
```

Per ulteriori informazioni, consulta Gestione dei cluster DAX nella Guida per gli sviluppatori di Amazon DynamoDB.

- Per i dettagli sull'API, consulta IncreaseReplicationFactor AWS CLICommand Reference.

`list-tags`

Il seguente esempio di codice mostra come utilizzare`list-tags`.

AWS CLI

Come elencare i tag per una risorsa DAX

L'esempio `list-tags` seguente elenca le chiavi e i valori dei tag associati al cluster DAX specificato.

```
aws dax list-tags \
    --resource-name arn:aws:dax:us-west-2:123456789012:cache/daxcluster
```

Output:

```
{
    "Tags": [
        {
            "Key": "ClusterUsage",
            "Value": "prod"
        }
    ]
}
```

Per ulteriori informazioni, consulta Gestione dei cluster DAX nella Guida per gli sviluppatori di Amazon DynamoDB.

- Per i dettagli sull'API, consulta ListTags AWS CLICommand Reference.

`tag-resource`

Il seguente esempio di codice mostra come utilizzare`tag-resource`.

AWS CLI

Come taggare una risorsa DAX

L'esempio `tag-resource` seguente allega il nome della chiave del tag specificato e il valore associato al cluster DAX specificato per descrivere l'utilizzo del cluster.

```
aws dax tag-resource \
    --resource-name arn:aws:dax:us-west-2:123456789012:cache/daxcluster \
    --tags="Key=ClusterUsage,Value=prod"
```

Output:

```
{
    "Tags": [
        {
            "Key": "ClusterUsage",
            "Value": "prod"
        }
    ]
}
```

Per ulteriori informazioni, consulta Gestione dei cluster DAX nella Guida per gli sviluppatori di Amazon DynamoDB.

- Per i dettagli sull'API, consulta TagResource AWS CLICommand Reference.

`untag-resource`

Il seguente esempio di codice mostra come utilizzare`untag-resource`.

AWS CLI

Come rimuovere i tag da una risorsa DAX

L'esempio `untag-resource` seguente rimuove il tag con il nome chiave specificato da un cluster DAX.

```
aws dax untag-resource  \
    --resource-name arn:aws:dax:us-west-2:123456789012:cache/daxcluster \
    --tag-keys="ClusterUsage"
```

Output:

```
{
```

```
    "Tags": []
}
```

Per ulteriori informazioni, consulta Gestione dei cluster DAX nella Guida per gli sviluppatori di Amazon DynamoDB.

- Per i dettagli sull'API, consulta UntagResource AWS CLICommand Reference.

Esempi di Detective che utilizzano AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with Detective.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`accept-invitation`

Il seguente esempio di codice mostra come utilizzare`accept-invitation`.

AWS CLI

Come accettare un invito a diventare un account membro in un grafico di comportamento

L'esempio `accept-invitation` seguente accetta un invito a diventare un account membro nel grafico di comportamento arn:aws:detective:us-east- 1:111122223333:graph:123412341234.

```
aws detective accept-invitation \
    --graph-arn arn:aws:detective:us-east-1:111122223333:graph:123412341234
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Risposta a un invito del grafico di comportamento nella Guida per l'amministratore di Amazon Detective.

- Per i dettagli sull'API, consulta AcceptInvitation AWS CLICommand Reference.

`create-graph`

Il seguente esempio di codice mostra come utilizzare`create-graph`.

AWS CLI

Come abilitare Amazon Detective e creare un nuovo grafico di comportamento

L'`create-graph`esempio seguente abilita Detective per l' AWS account che esegue il comando nella regione in cui viene eseguito il comando. Viene creato un nuovo grafico di comportamento con quell'account come account amministratore. Il comando assegna anche il valore Finance al tag Department.

```
aws detective create-graph \
    --tags '{"Department": "Finance"}'
```

Output:

```
{
    "GraphArn": "arn:aws:detective:us-
east-1:111122223333:graph:027c7c4610ea4aacaf0b883093cab899"
}
```

Per ulteriori informazioni, consulta Abilitazione di Amazon Detective nella Guida per l'amministratore di Amazon Detective.

- Per i dettagli sull'API, consulta CreateGraph AWS CLICommand Reference.

`create-members`

Il seguente esempio di codice mostra come utilizzare`create-members`.

AWS CLI

Come invitare gli account membri a un grafico di comportamento

L'`create-members`esempio seguente invita due AWS account a diventare account membro nel grafico di comportamento arn:aws:detective:us-east- 1:111122223333:graph:123412341234. Per ogni account, la richiesta fornisce l'ID dell'account e l'indirizzo e-mail dell' AWS utente root dell'account. La richiesta include un messaggio personalizzato da inserire nell'e-mail di invito.

```
aws detective create-members \
    --
accounts AccountId=444455556666,EmailAddress=mmajor@example.com AccountId=123456789012,Email
 \
    --graph-arn arn:aws:detective:us-east-1:111122223333:graph:123412341234 \
    --message "This is Paul Santos. I need to add your account to the data we use
 for security investigation in Amazon Detective. If you have any questions, contact
 me at psantos@example.com."
```

Output:

```
{
    "Members": [
    {
        "AccountId": "444455556666",
        "AdministratorId": "111122223333",
        "EmailAddress": "mmajor@example.com",
        "GraphArn": "arn:aws:detective:us-east-1:111122223333:graph:123412341234",
        "InvitedTime": 1579826107000,
        "MasterId": "111122223333",
        "Status": "INVITED",
        "UpdatedTime": 1579826107000
   },
   {
        "AccountId": "123456789012",
        "AdministratorId": "111122223333",
        "EmailAddress": "jstiles@example.com",
        "GraphArn": "arn:aws:detective:us-east-1:111122223333:graph:123412341234",
        "InvitedTime": 1579826107000,
        "MasterId": "111122223333",
        "Status": "VERIFICATION_IN_PROGRESS",
        "UpdatedTime": 1579826107000
     }
    ],
    "UnprocessedAccounts": [ ]
}
```

Per ulteriori informazioni, consulta Invitare gli account dei membri a un grafico comportamento< https://docs.aws.amazon.com/detective/ latest/adminguide/graph - admin-add-member-accounts .html> nella Amazon Detective Administration Guide.

Come invitare account membri senza inviare e-mail di invito

L'`create-members`esempio seguente invita due AWS account a diventare account membri nel grafico comportamentale arn:aws:detective:us-east- 1:111122223333:graph:123412341234. Per ogni account, la richiesta fornisce l'ID dell'account e l'indirizzo e-mail dell' AWS utente root dell'account. Gli account membri non ricevono e-mail di invito.

```
aws detective create-members \
    --
accounts AccountId=444455556666,EmailAddress=mmajor@example.com AccountId=123456789012,Email
 \
    --graph-arn arn:aws:detective:us-east-1:111122223333:graph:123412341234 \
    --disable-email-notification
```

Output:

```
{
    "Members": [
    {
        "AccountId": "444455556666",
        "AdministratorId": "111122223333",
        "EmailAddress": "mmajor@example.com",
        "GraphArn": "arn:aws:detective:us-east-1:111122223333:graph:123412341234",
        "InvitedTime": 1579826107000,
        "MasterId": "111122223333",
        "Status": "INVITED",
        "UpdatedTime": 1579826107000
   },
   {
        "AccountId": "123456789012",
        "AdministratorId": "111122223333",
        "EmailAddress": "jstiles@example.com",
        "GraphArn": "arn:aws:detective:us-east-1:111122223333:graph:123412341234",
        "InvitedTime": 1579826107000,
        "MasterId": "111122223333",
        "Status": "VERIFICATION_IN_PROGRESS",
        "UpdatedTime": 1579826107000
     }
```

```
    ],
    "UnprocessedAccounts": [ ]
}
```

Per ulteriori informazioni, consulta Invitare gli account dei membri a un grafico comportamento< https://docs.aws.amazon.com/detective/ latest/adminguide/graph - admin-add-member-accounts .html> nella Amazon Detective Administration Guide.

- Per i dettagli sull'API, consulta Command Reference. CreateMembersAWS CLI

`delete-graph`

Il seguente esempio di codice mostra come utilizzare`delete-graph`.

AWS CLI

Come disabilitare Detective ed eliminare il grafico di comportamento

L'esempio `delete-graph` seguente disattiva Detective ed elimina il grafico comportamentale specificato.

```
aws detective delete-graph \
    --graph-arn arn:aws:detective:us-east-1:111122223333:graph:123412341234
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Disabilitazione di Amazon Detective nella Guida per l'amministratore di Amazon Detective.

- Per i dettagli sull'API, consulta DeleteGraph AWS CLICommand Reference.

`delete-members`

Il seguente esempio di codice mostra come utilizzare`delete-members`.

AWS CLI

Come rimuovere un account membro dal grafico di comportamento

L'esempio `delete-members` seguente rimuove due account membri dal grafico comportamentale arn:aws:detective:us-east-1:111122223333:graph:123412341234. Per identificare gli account, la richiesta fornisce l' AWS account IDs.

```
aws detective delete-members \
    --account-ids 444455556666 123456789012 \
    --graph-arn arn:aws:detective:us-east-1:111122223333:graph:123412341234
```

Output:

```
 {
    "AccountIds": [ "444455556666", "123456789012" ],
    "UnprocessedAccounts": [ ]
}
```

Per ulteriori informazioni, consulta Rimuovere gli account dei membri da un grafico comportamento< https://docs.aws.amazon.com/detective/ latest/adminguide/graph - admin-remove-member-accounts .html> nella Amazon Detective Administration Guide.

- Per i dettagli sull'API, consulta DeleteMembersCommand Reference.AWS CLI

`disassociate-membership`

Il seguente esempio di codice mostra come utilizzare`disassociate-membership`.

AWS CLI

Come rimuovere un account membro dal grafico di comportamento

Il seguente esempio di dissociate-membership rimuove l' AWS account che esegue il comando dal grafico di comportamento arn:aws:detective:us-east- 1:111122223333:graph:123412341234.

```
aws detective disassociate-membership \
     --graph-arn arn:aws:detective:us-east-1:111122223333:graph:123412341234
```

Per ulteriori informazioni, consulta Rimuovere l'account da un grafico comportamento< https://docs.aws.amazon.com/detective/ latest/adminguide/member - remove-self-from-graph .html> nella Amazon Detective Administration Guide.

- Per i dettagli sull'API, consulta DisassociateMembershipCommand Reference.AWS CLI

`get-members`

Il seguente esempio di codice mostra come utilizzare`get-members`.

AWS CLI

Come recuperare informazioni sugli account membri del grafico di comportamento selezionati

L'esempio `get-members` seguente recupera le informazioni su due account nel grafico comportamentale arn:aws:detective:us-east-1:111122223333:graph:123412341234. Per i due account, la richiesta fornisce l' AWS account IDs.

```
aws detective get-members \
    --account-ids 444455556666 123456789012 \
    --graph-arn arn:aws:detective:us-east-1:111122223333:graph:123412341234
```

Output:

```
{
    "MemberDetails": [
    {
        "AccountId": "444455556666",
        "AdministratorId": "111122223333",
        "EmailAddress": "mmajor@example.com",
        "GraphArn": "arn:aws:detective:us-east-1:111122223333:graph:123412341234",
        "InvitedTime": 1579826107000,
        "MasterId": "111122223333",
        "Status": "INVITED",
        "UpdatedTime": 1579826107000
    }
    {
        "AccountId": "123456789012",
        "AdministratorId": "111122223333",
        "EmailAddress": "jstiles@example.com",
        "GraphArn": "arn:aws:detective:us-east-1:111122223333:graph:123412341234",
        "InvitedTime": 1579826107000,
        "MasterId": "111122223333",
        "Status": "INVITED",
        "UpdatedTime": 1579826107000
    }
],
    "UnprocessedAccounts": [ ]
}
```

Per ulteriori informazioni, consulta Visualizzazione dell'elenco degli account in un grafico comportamento< https://docs.aws.amazon.com/detective/ latest/adminguide/graph - admin-view-accounts .html> nella Amazon Detective Administration Guide.

- Per i dettagli sull'API, consulta GetMembersCommand Reference.AWS CLI

`list-graphs`

Il seguente esempio di codice mostra come utilizzare`list-graphs`.

AWS CLI

Come visualizzare un elenco di grafici di comportamento di cui il tuo account è l'amministratore

L'esempio `list-graphs` seguente recupera i grafici comportamentali di cui l'account chiamante è l'amministratore nella Regione corrente.

```
aws detective list-graphs
```

Output:

```
{
    "GraphList": [
        {
            "Arn": "arn:aws:detective:us-east-1:111122223333:graph:123412341234",
            "CreatedTime": 1579736111000
        }
    ]
}
```

- Per i dettagli sull'API, consulta ListGraphs AWS CLICommand Reference.

`list-invitations`

Il seguente esempio di codice mostra come utilizzare`list-invitations`.

AWS CLI

Come visualizzare un elenco di grafici di comportamento di cui un account è membro o a cui è invitato

L'esempio `list-invitations` seguente recupera i grafici comportamentali a cui l'account chiamante è stato invitato. I risultati includono solo gli inviti aperti e accettati. Non includono gli inviti rifiutati o le iscrizioni rimosse.

```
aws detective list-invitations
```

Output:

```
{
    "Invitations": [
    {
        "AccountId": "444455556666",
        "AdministratorId": "111122223333",
        "EmailAddress": "mmajor@example.com",
        "GraphArn": "arn:aws:detective:us-east-1:111122223333:graph:123412341234",
        "InvitedTime": 1579826107000,
        "MasterId": "111122223333",
        "Status": "INVITED",
        "UpdatedTime": 1579826107000
    }
]
}
```

Per ulteriori informazioni, consulta Visualizzazione dell'elenco degli inviti relativi al grafico del comportamento< https://docs.aws.amazon.com/detective/ latest/adminguide/member - view-graph-invitations .html> nella Amazon Detective Administration Guide.

- Per i dettagli sull'API, consulta Command Reference. ListInvitationsAWS CLI

`list-members`

Il seguente esempio di codice mostra come utilizzare`list-members`.

AWS CLI

Come elencare gli account membri a un grafico di comportamento

L'esempio `list-members` seguente recupera gli account membri invitati e abilitati per il grafico comportamentale `arn:aws:detective:us-east-1:111122223333:graph:123412341234`. I risultati non includono gli account membri che sono stati rimossi.

```
aws detective list-members \
    --graph-arn arn:aws:detective:us-east-1:111122223333:graph:123412341234
```

Output:

```
{
    "MemberDetails": [
        {
            "AccountId": "444455556666",
            "AdministratorId": "111122223333",
            "EmailAddress": "mmajor@example.com",
            "GraphArn": "arn:aws:detective:us-
east-1:111122223333:graph:123412341234",
            "InvitedTime": 1579826107000,
            "MasterId": "111122223333",
            "Status": "INVITED",
            "UpdatedTime": 1579826107000
        },
        {
            "AccountId": "123456789012",
            "AdministratorId": "111122223333",
            "EmailAddress": "jstiles@example.com",
            "GraphArn": "arn:aws:detective:us-
east-1:111122223333:graph:123412341234",
            "InvitedTime": 1579826107000,
            "MasterId": "111122223333",
            "PercentOfGraphUtilization": 2,
            "PercentOfGraphUtilizationUpdatedTime": 1586287843,
            "Status": "ENABLED",
            "UpdatedTime": 1579973711000,
            "VolumeUsageInBytes": 200,
            "VolumeUsageUpdatedTime": 1586287843
        }
    ]
}
```

Per ulteriori informazioni, consulta Visualizzazione dell'elenco di account in un grafico di comportamento nella Guida per l'amministratore di Amazon Detective.

- Per i dettagli sull'API, consulta ListMembers AWS CLICommand Reference.

`list-tags-for-resource`

Il seguente esempio di codice mostra come utilizzare`list-tags-for-resource`.

AWS CLI

Come recuperare i tag assegnati a un grafico di comportamento

L'esempio `list-tags-for-resource` seguente riporta i tag assegnati al grafico comportamentale specificato.

```
aws detective list-tags-for-resource \
    --resource-arn arn:aws:detective:us-east-1:111122223333:graph:123412341234
```

Output:

```
{
    "Tags": {
        "Department" : "Finance"
    }
}
```

Per ulteriori informazioni, consulta Gestione dei tag per un grafico di comportamento nella Guida per l'amministratore di Amazon Detective.

- Per i dettagli sull'API, consulta ListTagsForResource AWS CLICommand Reference.

`reject-invitation`

Il seguente esempio di codice mostra come utilizzare`reject-invitation`.

AWS CLI

Come rifiutare un invito a diventare un account membro in un grafico di comportamento

L'esempio `reject-invitation` seguente rifiuta un invito a diventare un account membro nel grafico comportamentale arn:aws:detective:us-east-1:111122223333:graph:123412341234.

```
aws detective reject-invitation \
    --graph-arn arn:aws:detective:us-east-1:111122223333:graph:123412341234
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Rispondere a un invito con grafico comportamento< https://docs.aws.amazon.com/detective/ latest/adminguide/member -invitation-response.html> nella Amazon Detective Administration Guide.

- Per i RejectInvitationdettagli AWS CLI sull'API, consulta Command Reference.

`tag-resource`

Il seguente esempio di codice mostra come utilizzare`tag-resource`.

AWS CLI

Come assegnare un tag a una risorsa

L'esempio `tag-resource` seguente assegna un valore per il tag Department al grafico comportamentale specificato.

```
aws detective tag-resource \
    --resource-arn arn:aws:detective:us-east-1:111122223333:graph:123412341234 \
    --tags '{"Department":"Finance"}'
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Gestione dei tag per un grafico di comportamento nella Guida per l'amministratore di Amazon Detective.

- Per i dettagli sull'API, consulta TagResource AWS CLICommand Reference.

`untag-resource`

Il seguente esempio di codice mostra come utilizzare`untag-resource`.

AWS CLI

Come rimuovere un tag da una risorsa

L'esempio `untag-resource` seguente rimuove il tag Department dal grafico comportamentale specificato.

```
aws detective untag-resource \
    --resource-arn arn:aws:detective:us-east-1:111122223333:graph:123412341234 \
    --tag-keys "Department"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Gestione dei tag per un grafico di comportamento nella Guida per l'amministratore di Amazon Detective.

- Per i dettagli sull'API, consulta UntagResource AWS CLICommand Reference.

Esempi di Device Farm utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with Device Farm.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`create-device-pool`

Il seguente esempio di codice mostra come utilizzare`create-device-pool`.

AWS CLI

Come creare un pool di dispositivi

Il comando seguente crea un pool di dispositivi Android per un progetto:

```
aws devicefarm create-device-pool --name pool1 --rules file://
device-pool-rules.json --project-arn "arn:aws:devicefarm:us-
west-2:123456789012:project:070fc3ca-7ec1-4741-9c1f-d3e044efc506"
```

È possibile ottenere l'ARN del progetto dall'output di `create-project` o `list-projects`. Il file `device-pool-rules.json` è un documento JSON nella cartella corrente che specifica la piattaforma del dispositivo.

```
[
    {
        "attribute": "PLATFORM",
        "operator": "EQUALS",
        "value": "\"ANDROID\""
    }
]
```

Output:

```
{
    "devicePool": {
        "rules": [
            {
                "operator": "EQUALS",
                "attribute": "PLATFORM",
                "value": "\"ANDROID\""
            }
        ],
        "type": "PRIVATE",
        "name": "pool1",
        "arn": "arn:aws:devicefarm:us-
west-2:123456789012:devicepool:070fc3ca-7ec1-4741-9c1f-
d3e044efc506/2aa8d2a9-5e73-47ca-b929-659cb34b7dcd"
    }
}
```

- Per i dettagli sull'API, consulta CreateDevicePool AWS CLICommand Reference.

`create-project`

Il seguente esempio di codice mostra come utilizzare`create-project`.

AWS CLI

Come creare un progetto

Il comando seguente crea un nuovo progetto denominato `my-project`:

```
aws devicefarm create-project --name my-project
```

Output:

```
{
    "project": {
        "name": "myproject",
        "arn": "arn:aws:devicefarm:us-
west-2:123456789012:project:070fc3ca-7ec1-4741-9c1f-d3e044efc506",
        "created": 1503612890.057
    }
}
```

- Per i dettagli sull'API, consulta CreateProject AWS CLICommand Reference.

`create-upload`

Il seguente esempio di codice mostra come utilizzare`create-upload`.

AWS CLI

Come creare un caricamento

Il comando seguente crea un caricamento per un'app Android.

```
aws devicefarm create-upload --project-arn "arn:aws:devicefarm:us-
west-2:123456789012:project:070fc3ca-7ec1-4741-9c1f-d3e044efc506" --name app.apk --
type ANDROID_APP
```

Puoi ottenere l'ARN del progetto dall'output del comando create-project o list-projects.

Output:

```
{
    "upload": {
        "status": "INITIALIZED",
        "name": "app.apk",
        "created": 1503614408.769,
        "url": "https://prod-us-west-2-uploads.s3-us-west-2.amazonaws.com/
arn%3Aaws%3Adevicefarm%3Aus-west-2%3A123456789012%3Aproject%3A070fc3ca-
c7e1-4471-91cf-d3e4efc50604/uploads/arn%3Aaws%3Adevicefarm%3Aus-
west-2%3A123456789012%3Aupload%3A070fc3ca-7ec1-4741-9c1f-d3e044efc506/dd72723a-
ae9e-4087-09e6-f4cea3599514/app.apk?X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-
Date=20170824T224008Z&X-Amz-SignedHeaders=host&X-Amz-Expires=86400&X-Amz-
Credential=AKIAEXAMPLEPBUMBC3GA%2F20170824%2Fus-west-2%2Fs3%2Faws4_request&X-Amz-
Signature=05050370c38894ef5bd09f5d009f36fc8f96fa4bb04e1bba9aca71b8dbe49a0f",
```

```
        "type": "ANDROID_APP",
        "arn": "arn:aws:devicefarm:us-
west-2:123456789012:upload:070fc3ca-7ec1-4741-9c1f-d3e044efc506/dd72723a-
ae9e-4087-09e6-f4cea3599514"
    }
}
```

Utilizza l'URL firmato nell'output per caricare un file in Device Farm:

```
curl -T app.apk "https://prod-us-west-2-uploads.s3-us-west-2.amazonaws.com/
arn%3Aaws%3Adevicefarm%3Aus-west-2%3A123456789012%3Aproject%3A070fc3ca-
c7e1-4471-91cf-d3e4efc50604/uploads/arn%3Aaws%3Adevicefarm%3Aus-
west-2%3A123456789012%3Aupload%3A070fc3ca-7ec1-4741-9c1f-d3e044efc506/dd72723a-
ae9e-4087-09e6-f4cea3599514/app.apk?X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-
Date=20170824T224008Z&X-Amz-SignedHeaders=host&X-Amz-Expires=86400&X-Amz-
Credential=AKIAEXAMPLEPBUMBC3GA%2F20170824%2Fus-west-2%2Fs3%2Faws4_request&X-Amz-
Signature=05050370c38894ef5bd09f5d009f36fc8f96fa4bb04e1bba9aca71b8dbe49a0f"
```

- Per i dettagli sull'API, consulta CreateUpload AWS CLICommand Reference.

`get-upload`

Il seguente esempio di codice mostra come utilizzare`get-upload`.

AWS CLI

Come visualizzare un caricamento

Il comando seguente recupera le informazioni su un caricamento:

```
aws devicefarm get-upload --arn "arn:aws:devicefarm:us-
west-2:123456789012:upload:070fc3ca-7ec1-4741-9c1f-d3e044efc506/dd72723a-
ae9e-4087-09e6-f4cea3599514"
```

È possibile ottenere l'ARN del caricamento dall'output di `create-upload`.

Output:

```
{
    "upload": {
        "status": "SUCCEEDED",
        "name": "app.apk",
```

```
        "created": 1505262773.186,
        "type": "ANDROID_APP",
        "arn": "arn:aws:devicefarm:us-
west-2:123456789012:upload:070fc3ca-7ec1-4741-9c1f-d3e044efc506/dd72723a-
ae9e-4087-09e6-f4cea3599514",
        "metadata": "{\"device_admin\":false,\"activity_name\":
\"ccom.example.client.LauncherActivity\",\"version_name\":\"1.0.2.94\",\"screens
\":[\"small\",\"normal\",\"large\",\"xlarge\"],\"error_type\":null,\"sdk_version
\":\"16\",\"package_name\":\"com.example.client\",\"version_code\":\"20994\",
\"native_code\":[\"armeabi-v7a\"],\"target_sdk_version\":\"25\"}"
    }
}
```

- Per i dettagli sull'API, consulta GetUpload AWS CLICommand Reference.

`list-projects`

Il seguente esempio di codice mostra come utilizzare`list-projects`.

AWS CLI

Come elencare i progetti

Il comando seguente recupera un elenco di progetti:

```
aws devicefarm list-projects
```

Output:

```
{
    "projects": [
        {
            "name": "myproject",
            "arn": "arn:aws:devicefarm:us-
west-2:123456789012:project:070fc3ca-7ec1-4741-9c1f-d3e044efc506",
            "created": 1503612890.057
        },
        {
            "name": "otherproject",
            "arn": "arn:aws:devicefarm:us-
west-2:123456789012:project:a5f5b752-8098-49d1-86bf-5f7682c1c77e",
            "created": 1505257519.337
        }
```

```
    ]
}
```

- Per i dettagli sull'API, consulta ListProjects AWS CLICommand Reference.

Direct Connect esempi utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with Direct Connect.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`accept-direct-connect-gateway-association-proposal`

Il seguente esempio di codice mostra come utilizzare`accept-direct-connect-gateway-association-proposal`.

AWS CLI

Come accettare una proposta di associazione del gateway

Il comando `accept-direct-connect-gateway-association-proposal` seguente accetta la proposta specificata.

```
aws directconnect  accept-direct-connect-gateway-association-proposal \
    --direct-connect-gateway-id 11460968-4ac1-4fd3-bdb2-00599EXAMPLE \
    --proposal-id cb7f41cb-8128-43a5-93b1-dcaedEXAMPLE \
    --associated-gateway-owner-account 111122223333

{
```

```
    "directConnectGatewayAssociation": {
        "directConnectGatewayId": "11460968-4ac1-4fd3-bdb2-00599EXAMPLE",
        "directConnectGatewayOwnerAccount": "111122223333",
        "associationState": "associating",
        "associatedGateway": {
            "id": "tgw-02f776b1a7EXAMPLE",
            "type": "transitGateway",
            "ownerAccount": "111122223333",
            "region": "us-east-1"
        },
        "associationId": "6441f8bf-5917-4279-ade1-9708bEXAMPLE",
        "allowedPrefixesToDirectConnectGateway": [
            {
                "cidr": "192.168.1.0/30"
            }
        ]

    }
}
```

Per ulteriori informazioni, consulta Accettazione o rifiuto di una proposta di associazione per gateway di transito nella Guida per l'utente di AWS Direct Connect.

- Per i dettagli sull'API, consulta AcceptDirectConnectGatewayAssociationProposal AWS CLICommand Reference.

`allocate-connection-on-interconnect`

Il seguente esempio di codice mostra come utilizzare`allocate-connection-on-interconnect`.

AWS CLI

Come creare una connessione ospitata su un'interconnessione

Il comando `allocate-connection-on-interconnect` seguente crea una connessione ospitata su un'interconnessione:

```
aws directconnect allocate-connection-on-interconnect --bandwidth 500Mbps --
connection-name mydcinterconnect --owner-account 123456789012 --interconnect-
id dxcon-fgktov66 --vlan 101
```

Output:

```
{
    "partnerName": "TIVIT",
    "vlan": 101,
    "ownerAccount": "123456789012",
    "connectionId": "dxcon-ffzc51m1",
    "connectionState": "ordering",
    "bandwidth": "500Mbps",
    "location": "TIVIT",
    "connectionName": "mydcinterconnect",
    "region": "sa-east-1"
}
```

- Per i dettagli sull'API, consulta AllocateConnectionOnInterconnect AWS CLICommand Reference.

`allocate-hosted-connection`

Il seguente esempio di codice mostra come utilizzare`allocate-hosted-connection`.

AWS CLI

Come creare una connessione ospitata su un'interconnessione

L'esempio `allocate-hosted-connection` seguente crea una connessione ospitata sull'interconnessione specificata.

```
aws directconnect allocate-hosted-connection \
    --bandwidth 500Mbps \
    --connection-name mydcinterconnect \
    --owner-account 123456789012
    -connection-id dxcon-fgktov66
    -vlan 101
```

Output:

```
{
    "partnerName": "TIVIT",
    "vlan": 101,
    "ownerAccount": "123456789012",
    "connectionId": "dxcon-ffzc51m1",
    "connectionState": "ordering",
    "bandwidth": "500Mbps",
```

```
    "location": "TIVIT",
    "connectionName": "mydcinterconnect",
    "region": "sa-east-1"
}
```

- Per i dettagli sull'API, consulta AllocateHostedConnection AWS CLICommand Reference.

`allocate-private-virtual-interface`

Il seguente esempio di codice mostra come utilizzare`allocate-private-virtual-interface`.

AWS CLI

Come eseguire il provisioning di un'interfaccia virtuale privata

Il comando `allocate-private-virtual-interface` seguente fornisce un'interfaccia virtuale privata di proprietà di un altro cliente:

```
aws directconnect allocate-private-virtual-interface --connection-id dxcon-
ffjrkx17 --owner-account 123456789012 --new-private-virtual-interface-
allocation virtualInterfaceName=PrivateVirtualInterface,vlan=1000,asn=65000,authKey=asdf34ex
```

Output:

```
{
    "virtualInterfaceState": "confirming",
    "asn": 65000,
    "vlan": 1000,
    "customerAddress": "192.168.1.2/30",
    "ownerAccount": "123456789012",
    "connectionId": "dxcon-ffjrkx17",
    "virtualInterfaceId": "dxvif-fgy8orxu",
    "authKey": "asdf34example",
    "routeFilterPrefixes": [],
    "location": "TIVIT",
    "customerRouterConfig": "<?xml version=\"1.0\" encoding=\"UTF-8\"?
>\n  <logical_connection id=\"dxvif-fgy8orxu\">\n  <vlan>1000</
vlan>\n  <customer_address>192.168.1.2/30</customer_address>\n
 <amazon_address>192.168.1.1/30</amazon_address>\n  <bgp_asn>65000</bgp_asn>\n
 <bgp_auth_key>asdf34example</bgp_auth_key>\n  <amazon_bgp_asn>7224</amazon_bgp_asn>
\n  <connection_type>private</connection_type>\n</logical_connection>\n",
    "amazonAddress": "192.168.1.1/30",
```

```
    "virtualInterfaceType": "private",
    "virtualInterfaceName": "PrivateVirtualInterface"
}
```

- Per i dettagli sull'API, consulta AllocatePrivateVirtualInterface AWS CLICommand Reference.

`allocate-public-virtual-interface`

Il seguente esempio di codice mostra come utilizzare`allocate-public-virtual-interface`.

AWS CLI

Come eseguire il provisioning di un'interfaccia virtuale pubblica

Il comando `allocate-public-virtual-interface` seguente fornisce un'interfaccia virtuale pubblica di proprietà di un altro cliente:

```
aws directconnect allocate-public-virtual-interface --connection-id dxcon-
ffjrkx17 --owner-account 123456789012 --new-public-virtual-interface-
allocation virtualInterfaceName=PublicVirtualInterface,vlan=2000,asn=65000,authKey=asdf34exa
{cidr=203.0.113.4/30}]
```

Output:

```
{
    "virtualInterfaceState": "confirming",
    "asn": 65000,
    "vlan": 2000,
    "customerAddress": "203.0.113.2/30",
    "ownerAccount": "123456789012",
    "connectionId": "dxcon-ffjrkx17",
    "virtualInterfaceId": "dxvif-fg9xo9vp",
    "authKey": "asdf34example",
    "routeFilterPrefixes": [
        {
            "cidr": "203.0.113.0/30"
        },
        {
            "cidr": "203.0.113.4/30"
        }
    ],
    "location": "TIVIT",
```

```
    "customerRouterConfig": "<?xml version=\"1.0\" encoding=\"UTF-8\"?
>\n<logical_connection id=\"dxvif-fg9xo9vp\">\n  <vlan>2000</
vlan>\n  <customer_address>203.0.113.2/30</customer_address>\n
 <amazon_address>203.0.113.1/30</amazon_address>\n  <bgp_asn>65000</bgp_asn>\n
 <bgp_auth_key>asdf34example</bgp_auth_key>\n  <amazon_bgp_asn>7224</amazon_bgp_asn>
\n  <connection_type>public</connection_type>\n</logical_connection>\n",
    "amazonAddress": "203.0.113.1/30",
    "virtualInterfaceType": "public",
    "virtualInterfaceName": "PublicVirtualInterface"
}
```

- Per i dettagli sull'API, consulta AllocatePublicVirtualInterface AWS CLICommand Reference.

`allocate-transit-virtual-interface`

Il seguente esempio di codice mostra come utilizzare`allocate-transit-virtual-interface`.

AWS CLI

Fornire un'interfaccia virtuale di transito di proprietà dell' AWS account specificato

L'esempio `allocate-transit-virtual-interface` seguente fornisce un'interfaccia virtuale di transito per l'account specificato.

```
aws directconnect allocate-transit-virtual-interface \
    --connection-id dxlag-fEXAMPLE \
    --owner-account 123456789012 \
    --new-transit-virtual-interface-allocation "virtualInterfaceName=Example Transit
 Virtual
 Interface,vlan=126,asn=65110,mtu=1500,authKey=0xzxgA9YoW9h58u8SEXAMPLE,amazonAddress=192.16
```

Output:

```
{
    "virtualInterface": {
        "ownerAccount": "123456789012",
        "virtualInterfaceId": "dxvif-fEXAMPLE",
        "location": "loc1",
        "connectionId": "dxlag-fEXAMPLE",
        "virtualInterfaceType": "transit",
        "virtualInterfaceName": "Example Transit Virtual Interface",
        "vlan": 126,
```

```
        "asn": 65110,
        "amazonSideAsn": 7224,
        "authKey": "0xzxgA9YoW9h58u8SEXAMPLE",
        "amazonAddress": "192.168.1.1/30",
        "customerAddress": "192.168.1.2/30",
        "addressFamily": "ipv4",
        "virtualInterfaceState": "confirming",
        "customerRouterConfig": "<?xml version=\"1.0\" encoding=
\"UTF-8\"?>\n<logical_connection id=\"dxvif-fEXAMPLE\">\n  <vlan>126</
vlan>\n  <customer_address>192.168.1.2/30</customer_address>\n
 <amazon_address>192.168.1.1/30</amazon_address>\n  <bgp_asn>65110</bgp_asn>\n
 <bgp_auth_key>0xzxgA9YoW9h58u8SEXAMPLE</bgp_auth_key>\n  <amazon_bgp_asn>7224</
amazon_bgp_asn>\n  <connection_type>transit</connection_type>\n</logical_connection>
\n",
        "mtu": 1500,
        "jumboFrameCapable": true,
        "virtualGatewayId": "",
        "directConnectGatewayId": "",
        "routeFilterPrefixes": [],
        "bgpPeers": [
            {
                "bgpPeerId": "dxpeer-fEXAMPLE",
                "asn": 65110,
                "authKey": "0xzxgA9YoW9h58u8EXAMPLE",
                "addressFamily": "ipv4",
                "amazonAddress": "192.168.1.1/30",
                "customerAddress": "192.168.1.2/30",
                "bgpPeerState": "pending",
                "bgpStatus": "down",
                "awsDeviceV2": "loc1-26wz6vEXAMPLE"
            }
        ],
        "region": "sa-east-1",
        "awsDeviceV2": "loc1-26wz6vEXAMPLE",
        "tags": [
            {
                "key": "Tag",
                "value": "Example"
            }
        ]
    }
}
```

Per ulteriori informazioni, consulta Creazione di un'interfaccia virtuale di transito in hosting nella Guida per l'utente di AWS Direct Connect.

- Per i dettagli sull'API, vedere AllocateTransitVirtualInterfacein AWS CLI Command Reference.

`associate-connection-with-lag`

Il seguente esempio di codice mostra come utilizzare`associate-connection-with-lag`.

AWS CLI

Come associare una connessione a un LAG

L'esempio seguente associa la connessione specificata al LAG specificato.

Comando:

```
aws directconnect associate-connection-with-lag --lag-id dxlag-fhccu14t --
connection-id dxcon-fg9607vm
```

Output:

```
{
  "ownerAccount": "123456789012",
  "connectionId": "dxcon-fg9607vm",
  "lagId": "dxlag-fhccu14t",
  "connectionState": "requested",
  "bandwidth": "1Gbps",
  "location": "EqDC2",
  "connectionName": "Con2ForLag",
  "region": "us-east-1"
}
```

- Per i dettagli sull'API, consulta AssociateConnectionWithLag AWS CLICommand Reference.

`associate-hosted-connection`

Il seguente esempio di codice mostra come utilizzare`associate-hosted-connection`.

AWS CLI

Come associare una connessione ospitata a un LAG

L'esempio seguente associa la connessione ospitata specificata al LAG specificato.

Comando:

```
aws directconnect associate-hosted-connection --parent-connection-id dxlag-fhccu14t
 --connection-id dxcon-fg9607vm
```

Output:

```
{
  "partnerName": "TIVIT",
  "vlan": 101,
  "ownerAccount": "123456789012",
  "connectionId": "dxcon-fg9607vm",
  "lagId": "dxlag-fhccu14t",
  "connectionState": "ordering",
  "bandwidth": "500Mbps",
  "location": "TIVIT",
  "connectionName": "mydcinterconnect",
  "region": "sa-east-1"
}
```

- Per i dettagli sull'API, consulta AssociateHostedConnection AWS CLICommand Reference.

`associate-virtual-interface`

Il seguente esempio di codice mostra come utilizzare`associate-virtual-interface`.

AWS CLI

Come associare un'interfaccia virtuale a una connessione

L'esempio seguente associa l'interfaccia virtuale specificata al LAG specificato. In alternativa, per associare l'interfaccia virtuale a una connessione, specificare l'ID di una connessione AWS Direct Connect per`--connection-id`; ad esempio,`dxcon-ffnikghc`.

Comando:

```
aws directconnect associate-virtual-interface --connection-id dxlag-ffjhj9lx --
virtual-interface-id dxvif-fgputw0j
```

Output:

```
{
  "virtualInterfaceState": "pending",
  "asn": 65000,
  "vlan": 123,
  "customerAddress": "169.254.255.2/30",
  "ownerAccount": "123456789012",
  "connectionId": "dxlag-ffjhj9lx",
  "addressFamily": "ipv4",
  "virtualGatewayId": "vgw-38e90b51",
  "virtualInterfaceId": "dxvif-fgputw0j",
  "authKey": "0x123pK5_VBqv.UQ3kJ4123_",
  "routeFilterPrefixes": [],
  "location": "CSVA1",
  "bgpPeers": [
      {
          "bgpStatus": "down",
          "customerAddress": "169.254.255.2/30",
          "addressFamily": "ipv4",
          "authKey": "0x123pK5_VBqv.UQ3kJ4123_",
          "bgpPeerState": "deleting",
          "amazonAddress": "169.254.255.1/30",
          "asn": 65000
      },
      {
          "bgpStatus": "down",
          "customerAddress": "169.254.255.2/30",
          "addressFamily": "ipv4",
          "authKey": "0x123pK5_VBqv.UQ3kJ4123_",
          "bgpPeerState": "pending",
          "amazonAddress": "169.254.255.1/30",
          "asn": 65000
      }
  ],
  "customerRouterConfig": "<?xml version=\"1.0\" encoding=\"UTF-8\"?
>\n<logical_connection id=\"dxvif-fgputw0j\">\n  <vlan>123</vlan>
\n  <customer_address>169.254.255.2/30</customer_address>\n
 <amazon_address>169.254.255.1/30</amazon_address>\n  <bgp_asn>65000</bgp_asn>\n
  <bgp_auth_key>0x123pK5_VBqv.UQ3kJ4123_</bgp_auth_key>\n  <amazon_bgp_asn>7224</
amazon_bgp_asn>\n  <connection_type>private</connection_type>\n</logical_connection>
\n",
  "amazonAddress": "169.254.255.1/30",
  "virtualInterfaceType": "private",
```

```
  "virtualInterfaceName": "VIF1A"
}
```

- Per i dettagli sull'API, consulta AssociateVirtualInterface AWS CLICommand Reference.

`confirm-connection`

Il seguente esempio di codice mostra come utilizzare`confirm-connection`.

AWS CLI

Come confermare la creazione di una connessione ospitata su un'interconnessione

Il comando `confirm-connection` seguente conferma la creazione di una connessione ospitata su un'interconnessione:

```
aws directconnect confirm-connection --connection-id dxcon-fg2wi7hy
```

Output:

```
{
    "connectionState": "pending"
}
```

- Per i dettagli sull'API, consulta ConfirmConnection AWS CLICommand Reference.

`confirm-private-virtual-interface`

Il seguente esempio di codice mostra come utilizzare`confirm-private-virtual-interface`.

AWS CLI

Come accettare la proprietà di un'interfaccia virtuale privata

Il comando `confirm-private-virtual-interface` seguente accetta la proprietà di un'interfaccia virtuale privata creata da un altro cliente:

```
aws directconnect confirm-private-virtual-interface --virtual-interface-id dxvif-
fgy8orxu --virtual-gateway-id vgw-e4a47df9
```

Output:

```
{
    "virtualInterfaceState": "pending"
}
```

- Per i dettagli sull'API, consulta ConfirmPrivateVirtualInterface AWS CLICommand Reference.

`confirm-public-virtual-interface`

Il seguente esempio di codice mostra come utilizzare`confirm-public-virtual-interface`.

AWS CLI

Come accettare la proprietà di un'interfaccia virtuale pubblica

Il comando `confirm-public-virtual-interface` seguente accetta la proprietà di un'interfaccia virtuale pubblica creata da un altro cliente:

```
aws directconnect confirm-public-virtual-interface --virtual-interface-id dxvif-fg9xo9vp
```

Output:

```
{
    "virtualInterfaceState": "verifying"
}
```

- Per i dettagli sull'API, consulta ConfirmPublicVirtualInterface AWS CLICommand Reference.

`confirm-transit-virtual-interface`

Il seguente esempio di codice mostra come utilizzare`confirm-transit-virtual-interface`.

AWS CLI

Come accettare la proprietà di un'interfaccia virtuale di transito

Il comando `confirm-transit-virtual-interface` seguente accetta la proprietà di un'interfaccia virtuale di transito creata da un altro cliente.

```
aws directconnect confirm-transit-virtual-interface \
    --virtual-interface-id dxvif-fEXAMPLE \
    --direct-connect-gateway-id 4112ccf9-25e9-4111-8237-b6c5dEXAMPLE
```

Output:

```
{
    "virtualInterfaceState": "pending"
}
```

Per ulteriori informazioni, consulta Accettazione di un'interfaccia virtuale in hosting nella Guida per l'utente di AWS Direct Connect.

- Per i dettagli sull'API, consulta ConfirmTransitVirtualInterface AWS CLICommand Reference.

`create-bgp-peer`

Il seguente esempio di codice mostra come utilizzare`create-bgp-peer`.

AWS CLI

Per creare una sessione di IPv6 peering BGP

L'esempio seguente crea una sessione di peering IPv6 BGP su un'interfaccia virtuale privata. `dxvif-fg1vuj3d` Gli IPv6 indirizzi peer vengono assegnati automaticamente da Amazon.

Comando:

```
aws directconnect create-bgp-peer --virtual-interface-id dxvif-fg1vuj3d --new-bgp-peer asn=64600,addressFamily=ipv6
```

Output:

```
{
  "virtualInterface": {
      "virtualInterfaceState": "available",
      "asn": 65000,
      "vlan": 125,
      "customerAddress": "169.254.255.2/30",
      "ownerAccount": "123456789012",
```

```
      "connectionId": "dxcon-fguhmqlc",
      "addressFamily": "ipv4",
      "virtualGatewayId": "vgw-f9eb0c90",
      "virtualInterfaceId": "dxvif-fg1vuj3d",
      "authKey": "0xC_ukbCerl6EYA0example",
      "routeFilterPrefixes": [],
      "location": "EqDC2",
      "bgpPeers": [
          {
              "bgpStatus": "down",
              "customerAddress": "169.254.255.2/30",
              "addressFamily": "ipv4",
              "authKey": "0xC_ukbCerl6EYA0uexample",
              "bgpPeerState": "available",
              "amazonAddress": "169.254.255.1/30",
              "asn": 65000
          },
          {
              "bgpStatus": "down",
              "customerAddress": "2001:db8:1100:2f0:0:1:9cb4:4216/125",
              "addressFamily": "ipv6",
              "authKey": "0xS27kAIU_VHPjjAexample",
              "bgpPeerState": "pending",
              "amazonAddress": "2001:db8:1100:2f0:0:1:9cb4:4211/125",
              "asn": 64600
          }
      ],
      "customerRouterConfig": "<?xml version=\"1.0\" encoding=
\"UTF-8\"?>\n<logical_connection id=\"dxvif-fg1vuj3d\">\n  <vlan>125</
vlan>\n  <customer_address>169.254.255.2/30</customer_address>\n
 <amazon_address>169.254.255.1/30</amazon_address>\n  <bgp_asn>65000</
bgp_asn>\n  <bgp_auth_key>0xC_ukbCerl6EYA0uexample</bgp_auth_key>\n
 <ipv6_customer_address>2001:db8:1100:2f0:0:1:9cb4:4216/125</ipv6_customer_address>
\n  <ipv6_amazon_address>2001:db8:1100:2f0:0:1:9cb4:4211/125</ipv6_amazon_address>\n
  <ipv6_bgp_asn>64600</ipv6_bgp_asn>\n  <ipv6_bgp_auth_key>0xS27kAIU_VHPjjAexample</
ipv6_bgp_auth_key>\n  <amazon_bgp_asn>7224</amazon_bgp_asn>\n
 <connection_type>private</connection_type>\n</logical_connection>\n",
      "amazonAddress": "169.254.255.1/30",
      "virtualInterfaceType": "private",
      "virtualInterfaceName": "Test"
  }
}
```

- Per i dettagli sull'API, consulta AWS CLI Command CreateBgpPeerReference.

`create-connection`

Il seguente esempio di codice mostra come utilizzare`create-connection`.

AWS CLI

Per creare una connessione dalla rete a una posizione AWS Direct Connect

Il `create-connection` comando seguente crea una connessione dalla rete a una posizione AWS Direct Connect:

```
aws directconnect create-connection --location TIVIT --bandwidth 1Gbps --connection-name "Connection to AWS"
```

Output:

```
{
    "ownerAccount": "123456789012",
    "connectionId": "dxcon-fg31dyv6",
    "connectionState": "requested",
    "bandwidth": "1Gbps",
    "location": "TIVIT",
    "connectionName": "Connection to AWS",
    "region": "sa-east-1"
}
```

- Per i dettagli sull'API, consulta CreateConnection AWS CLICommand Reference.

`create-direct-connect-gateway-association-proposal`

Il seguente esempio di codice mostra come utilizzare`create-direct-connect-gateway-association-proposal`.

AWS CLI

Come creare una proposta per associare il gateway di transito specificato al gateway Direct Connect specificato

Nell'esempio `create-direct-connect-gateway-association-proposal` seguente viene creata una proposta che associa il gateway di transito specificato al gateway Direct Connect specificato.

```
aws directconnect create-direct-connect-gateway-association-proposal \
    --direct-connect-gateway-id 11460968-4ac1-4fd3-bdb2-00599EXAMPLE \
    --direct-connect-gateway-owner-account 111122223333 \
    --gateway-id tgw-02f776b1a7EXAMPLE \
    --add-allowed-prefixes-to-direct-connect-gateway cidr=192.168.1.0/30
```

Output:

```
{
    "directConnectGatewayAssociationProposal": {
        "proposalId": "cb7f41cb-8128-43a5-93b1-dcaedEXAMPLE",
        "directConnectGatewayId": "11460968-4ac1-4fd3-bdb2-00599EXAMPLE",
        "directConnectGatewayOwnerAccount": "111122223333",
        "proposalState": "requested",
        "associatedGateway": {
            "id": "tgw-02f776b1a7EXAMPLE",
            "type": "transitGateway",
            "ownerAccount": "111122223333",
            "region": "us-east-1"
        },
        "requestedAllowedPrefixesToDirectConnectGateway": [
            {
                "cidr": "192.168.1.0/30"
            }
        ]
    }
}
```

Per ulteriori informazioni, consulta Creazione di una proposta di associazione per gateway di transito nella Guida per l'utente di AWS Direct Connect.

- Per i dettagli sull'API, consulta CreateDirectConnectGatewayAssociationProposal AWS CLICommand Reference.

`create-direct-connect-gateway-association`

Il seguente esempio di codice mostra come utilizzare`create-direct-connect-gateway-association`.

AWS CLI

Come associare un gateway privato virtuale a un gateway Direct Connect

L'esempio seguente associa il gateway privato virtuale `vgw-6efe725e` al gateway Direct Connect `5f294f92-bafb-4011-916d-9b0bexample`. È necessario eseguire il comando nella Regione in cui si trova il gateway privato virtuale.

Comando:

```
aws directconnect create-direct-connect-gateway-association --direct-connect-gateway-id 5f294f92-bafb-4011-916d-9b0bexample --virtual-gateway-id vgw-6efe725e
```

Output:

```
{
  "directConnectGatewayAssociation": {
      "associationState": "associating",
      "virtualGatewayOwnerAccount": "123456789012",
      "directConnectGatewayId": "5f294f92-bafb-4011-916d-9b0bexample",
      "virtualGatewayId": "vgw-6efe725e",
      "virtualGatewayRegion": "us-east-2"
  }
}
```

- Per i dettagli sull'API, consulta CreateDirectConnectGatewayAssociation AWS CLICommand Reference.

`create-direct-connect-gateway`

Il seguente esempio di codice mostra come utilizzare`create-direct-connect-gateway`.

AWS CLI

Come creare un gateway Direct Connect

L'esempio seguente crea un gateway Direct Connect denominato `DxGateway1`.

Comando:

```
aws directconnect create-direct-connect-gateway --direct-connect-gateway-name "DxGateway1"
```

Output:

```
{
```

```
  "directConnectGateway": {
      "amazonSideAsn": 64512,
      "directConnectGatewayId": "5f294f92-bafb-4011-916d-9b0bdexample",
      "ownerAccount": "123456789012",
      "directConnectGatewayName": "DxGateway1",
      "directConnectGatewayState": "available"
  }
}
```

- Per i dettagli sull'API, consulta CreateDirectConnectGateway AWS CLICommand Reference.

`create-interconnect`

Il seguente esempio di codice mostra come utilizzare`create-interconnect`.

AWS CLI

Per creare un'interconnessione tra la rete di un partner e AWS

Il `create-interconnect` comando seguente crea un'interconnessione tra la rete di un partner AWS Direct Connect e una posizione AWS Direct Connect specifica:

```
aws directconnect create-interconnect --interconnect-name "1G Interconnect to AWS"
 --bandwidth 1Gbps --location TIVIT
```

Output:

```
{
    "region": "sa-east-1",
    "bandwidth": "1Gbps",
    "location": "TIVIT",
    "interconnectName": "1G Interconnect to AWS",
    "interconnectId": "dxcon-fgktov66",
    "interconnectState": "requested"
}
```

- Per i dettagli sull'API, consulta CreateInterconnect AWS CLICommand Reference.

`create-lag`

Il seguente esempio di codice mostra come utilizzare`create-lag`.

AWS CLI

Come creare un LAG con nuove connessioni

L'esempio seguente crea un LAG e richiede due nuove connessioni AWS Direct Connect per il LAG con una larghezza di banda di 1 Gbps.

Comando:

```
aws directconnect create-lag --location CSVA1 --number-of-connections 2 --
connections-bandwidth 1Gbps --lag-name 1GBLag
```

Output:

```
{
  "awsDevice": "CSVA1-23u8tlpaz8iks",
  "numberOfConnections": 2,
  "lagState": "pending",
  "ownerAccount": "123456789012",
  "lagName": "1GBLag",
  "connections": [
      {
          "ownerAccount": "123456789012",
          "connectionId": "dxcon-ffqr6x5q",
          "lagId": "dxlag-ffjhj9lx",
          "connectionState": "requested",
          "bandwidth": "1Gbps",
          "location": "CSVA1",
          "connectionName": "Requested Connection 1 for Lag dxlag-ffjhj9lx",
          "region": "us-east-1"
      },
      {
          "ownerAccount": "123456789012",
          "connectionId": "dxcon-fflqyj95",
          "lagId": "dxlag-ffjhj9lx",
          "connectionState": "requested",
          "bandwidth": "1Gbps",
          "location": "CSVA1",
          "connectionName": "Requested Connection 2 for Lag dxlag-ffjhj9lx",
          "region": "us-east-1"
      }
  ],
  "lagId": "dxlag-ffjhj9lx",
```

```
  "minimumLinks": 0,
  "connectionsBandwidth": "1Gbps",
  "region": "us-east-1",
  "location": "CSVA1"
}
```

Come creare un LAG utilizzando una connessione esistente

L'esempio seguente crea un LAG da una connessione esistente nell'account e richiede una seconda nuova connessione per il LAG con la stessa larghezza di banda e la stessa posizione della connessione esistente.

Comando:

```
aws directconnect create-lag --location EqDC2 --number-of-connections 2 --
connections-bandwidth 1Gbps --lag-name 2ConnLAG --connection-id dxcon-fgk145dr
```

Output:

```
{
  "awsDevice": "EqDC2-4h6ce2r1bes6",
  "numberOfConnections": 2,
  "lagState": "pending",
  "ownerAccount": "123456789012",
  "lagName": "2ConnLAG",
  "connections": [
      {
          "ownerAccount": "123456789012",
          "connectionId": "dxcon-fh6ljcvo",
          "lagId": "dxlag-fhccu14t",
          "connectionState": "requested",
          "bandwidth": "1Gbps",
          "location": "EqDC2",
          "connectionName": "Requested Connection 1 for Lag dxlag-fhccu14t",
          "region": "us-east-1"
      },
      {
          "ownerAccount": "123456789012",
          "connectionId": "dxcon-fgk145dr",
          "lagId": "dxlag-fhccu14t",
          "connectionState": "down",
          "bandwidth": "1Gbps",
          "location": "EqDC2",
```

```
            "connectionName": "VAConn1",
            "region": "us-east-1"
        }
    ],
    "lagId": "dxlag-fhccu14t",
    "minimumLinks": 0,
    "connectionsBandwidth": "1Gbps",
    "region": "us-east-1",
    "location": "EqDC2"
}
```

- Per i dettagli sull'API, vedere CreateLagin Command Reference.AWS CLI

`create-private-virtual-interface`

Il seguente esempio di codice mostra come utilizzare`create-private-virtual-interface`.

AWS CLI

Come creare un'interfaccia virtuale privata

Il comando `create-private-virtual-interface` seguente crea un'interfaccia virtuale privata:

```
aws directconnect create-private-virtual-interface --connection-id dxcon-ffjrkx17 --
new-private-virtual-
interface virtualInterfaceName=PrivateVirtualInterface,vlan=101,asn=65000,authKey=asdf34exam
aba37db6
```

Output:

```
{
    "virtualInterfaceState": "pending",
    "asn": 65000,
    "vlan": 101,
    "customerAddress": "192.168.1.2/30",
    "ownerAccount": "123456789012",
    "connectionId": "dxcon-ffjrkx17",
    "virtualGatewayId": "vgw-aba37db6",
    "virtualInterfaceId": "dxvif-ffhhk74f",
    "authKey": "asdf34example",
    "routeFilterPrefixes": [],
```

```
    "location": "TIVIT",
    "customerRouterConfig": "<?xml version=\"1.0\" encoding=
\"UTF-8\"?>\n<logical_connection id=\"dxvif-ffhhk74f\">\n  <vlan>101</
vlan>\n  <customer_address>192.168.1.2/30</customer_address>\n
 <amazon_address>192.168.1.1/30</amazon_address>\n  <bgp_asn>65000</bgp_asn>\n
 <bgp_auth_key>asdf34example</bgp_auth_key>\n  <amazon_bgp_asn>7224</amazon_bgp_asn>
\n  <connection_type>private</connection_type>\n</logical_connection>\n",
    "amazonAddress": "192.168.1.1/30",
    "virtualInterfaceType": "private",
    "virtualInterfaceName": "PrivateVirtualInterface"
}
```

- Per i dettagli sull'API, consulta CreatePrivateVirtualInterface AWS CLICommand Reference.

`create-public-virtual-interface`

Il seguente esempio di codice mostra come utilizzare`create-public-virtual-interface`.

AWS CLI

Come creare un'interfaccia virtuale pubblica

Il comando `create-public-virtual-interface` seguente crea un'interfaccia virtuale pubblica:

```
aws directconnect create-public-virtual-interface --connection-id dxcon-ffjrkx17 --
new-public-virtual-
interface virtualInterfaceName=PublicVirtualInterface,vlan=2000,asn=65000,authKey=asdf34exam
{cidr=203.0.113.4/30}]
```

Output:

```
{
    "virtualInterfaceState": "verifying",
    "asn": 65000,
    "vlan": 2000,
    "customerAddress": "203.0.113.2/30",
    "ownerAccount": "123456789012",
    "connectionId": "dxcon-ffjrkx17",
    "virtualInterfaceId": "dxvif-fgh0hcrk",
    "authKey": "asdf34example",
    "routeFilterPrefixes": [
```

```
        {
            "cidr": "203.0.113.0/30"
        },
        {
            "cidr": "203.0.113.4/30"
        }
    ],
    "location": "TIVIT",
    "customerRouterConfig": "<?xml version=\"1.0\" encoding=\"UTF-8\"?
>\n<logical_connection id=\"dxvif-fgh0hcrk\">\n  <vlan>2000</
vlan>\n  <customer_address>203.0.113.2/30</customer_address>\n
 <amazon_address>203.0.113.1/30</amazon_address>\n  <bgp_asn>65000</bgp_asn>\n
 <bgp_auth_key>asdf34example</bgp_auth_key>\n  <amazon_bgp_asn>7224</amazon_bgp_asn>
\n  <connection_type>public</connection_type>\n</logical_connection>\n",
    "amazonAddress": "203.0.113.1/30",
    "virtualInterfaceType": "public",
    "virtualInterfaceName": "PublicVirtualInterface"
}
```

- Per i dettagli sull'API, consulta CreatePublicVirtualInterface AWS CLICommand Reference.

`create-transit-virtual-interface`

Il seguente esempio di codice mostra come utilizzare`create-transit-virtual-interface`.

AWS CLI

Come creare un'interfaccia virtuale di transito

L'esempio `create-transit-virtual-interface` seguente crea un'interfaccia virtuale di transito per la connessione specificata.

```
aws directconnect create-transit-virtual-interface \
    --connection-id dxlag-fEXAMPLE \
    --new-transit-virtual-interface "virtualInterfaceName=Example Transit Virtual
 Interface,vlan=126,asn=65110,mtu=1500,authKey=0xzxgA9YoW9h58u8SvEXAMPLE,amazonAddress=192.1
aada-5a1baEXAMPLE,tags=[{key=Tag,value=Example}]"
```

Output:

```
{
    "virtualInterface": {
```

```
        "ownerAccount": "1111222233333",
        "virtualInterfaceId": "dxvif-fEXAMPLE",
        "location": "loc1",
        "connectionId": "dxlag-fEXAMPLE",
        "virtualInterfaceType": "transit",
        "virtualInterfaceName": "Example Transit Virtual Interface",
        "vlan": 126,
        "asn": 65110,
        "amazonSideAsn": 4200000000,
        "authKey": "0xzxgA9YoW9h58u8SEXAMPLE",
        "amazonAddress": "192.168.1.1/30",
        "customerAddress": "192.168.1.2/30",
        "addressFamily": "ipv4",
        "virtualInterfaceState": "pending",
        "customerRouterConfig": "<?xml version=\"1.0\" encoding=
\"UTF-8\"?>\n<logical_connection id=\"dxvif-fEXAMPLE\">\n  <vlan>126</
vlan>\n  <customer_address>192.168.1.2/30</customer_address>\n
 <amazon_address>192.168.1.1/30</amazon_address>\n  <bgp_asn>65110</
bgp_asn>\n  <bgp_auth_key>0xzxgA9YoW9h58u8SvOmXRTw</bgp_auth_key>\n
 <amazon_bgp_asn>4200000000</amazon_bgp_asn>\n  <connection_type>transit</
connection_type>\n</logical_connection>\n",
        "mtu": 1500,
        "jumboFrameCapable": true,
        "virtualGatewayId": "",
        "directConnectGatewayId": "8384da05-13ce-4a91-aada-5a1baEXAMPLE",
        "routeFilterPrefixes": [],
        "bgpPeers": [
            {
                "bgpPeerId": "dxpeer-EXAMPLE",
                "asn": 65110,
                "authKey": "0xzxgA9YoW9h58u8SEXAMPLE",
                "addressFamily": "ipv4",
                "amazonAddress": "192.168.1.1/30",
                "customerAddress": "192.168.1.2/30",
                "bgpPeerState": "pending",
                "bgpStatus": "down",
                "awsDeviceV2": "loc1-26wz6vEXAMPLE"
            }
        ],
        "region": "sa-east-1",
        "awsDeviceV2": "loc1-26wz6vEXAMPLE",
        "tags": [
            {
                "key": "Tag",
```

```
                "value": "Example"
            }
        ]
    }
}
```

Per ulteriori informazioni, consulta Creazione di un'interfaccia virtuale di transito per il gateway Direct Connect nella Guida per l'utente di AWS Direct Connect.

- Per i dettagli sull'API, consulta CreateTransitVirtualInterface AWS CLICommand Reference.

`delete-bgp-peer`

Il seguente esempio di codice mostra come utilizzare`delete-bgp-peer`.

AWS CLI

Come eliminare un peer BGP da un'interfaccia virtuale

L'esempio seguente elimina il peer IPv6 BGP dall'interfaccia virtuale. `dxvif-fg1vuj3d`

Comando:

```
aws directconnect delete-bgp-peer --virtual-interface-id dxvif-fg1vuj3d --asn 64600
 --customer-address 2001:db8:1100:2f0:0:1:9cb4:4216/125
```

Output:

```
{
  "virtualInterface": {
      "virtualInterfaceState": "available",
      "asn": 65000,
      "vlan": 125,
      "customerAddress": "169.254.255.2/30",
      "ownerAccount": "123456789012",
      "connectionId": "dxcon-fguhmqlc",
      "addressFamily": "ipv4",
      "virtualGatewayId": "vgw-f9eb0c90",
      "virtualInterfaceId": "dxvif-fg1vuj3d",
      "authKey": "0xC_ukbCerl6EYA0example",
      "routeFilterPrefixes": [],
      "location": "EqDC2",
      "bgpPeers": [
```

```
            {
                "bgpStatus": "down",
                "customerAddress": "169.254.255.2/30",
                "addressFamily": "ipv4",
                "authKey": "0xC_ukbCerl6EYA0uexample",
                "bgpPeerState": "available",
                "amazonAddress": "169.254.255.1/30",
                "asn": 65000
            },
            {
                "bgpStatus": "down",
                "customerAddress": "2001:db8:1100:2f0:0:1:9cb4:4216/125",
                "addressFamily": "ipv6",
                "authKey": "0xS27kAIU_VHPjjAexample",
                "bgpPeerState": "deleting",
                "amazonAddress": "2001:db8:1100:2f0:0:1:9cb4:4211/125",
                "asn": 64600
            }
        ],
        "customerRouterConfig": "<?xml version=\"1.0\" encoding=
\"UTF-8\"?>\n<logical_connection id=\"dxvif-fg1vuj3d\">\n  <vlan>125</
vlan>\n  <customer_address>169.254.255.2/30</customer_address>\n
 <amazon_address>169.254.255.1/30</amazon_address>\n  <bgp_asn>65000</bgp_asn>\n
  <bgp_auth_key>0xC_ukbCerl6EYA0example</bgp_auth_key>\n  <amazon_bgp_asn>7224</
amazon_bgp_asn>\n  <connection_type>private</connection_type>\n</logical_connection>
\n",
        "amazonAddress": "169.254.255.1/30",
        "virtualInterfaceType": "private",
        "virtualInterfaceName": "Test"
    }
}
```

- Per i dettagli sull'API, vedere DeleteBgpPeerin Command Reference.AWS CLI

`delete-connection`

Il seguente esempio di codice mostra come utilizzare`delete-connection`.

AWS CLI

Come eliminare una connessione

Il comando `delete-connection` seguente elimina la connessione specificata:

```
aws directconnect delete-connection --connection-id dxcon-fg31dyv6
```

Output:

```
{
    "ownerAccount": "123456789012",
    "connectionId": "dxcon-fg31dyv6",
    "connectionState": "deleted",
    "bandwidth": "1Gbps",
    "location": "TIVIT",
    "connectionName": "Connection to AWS",
    "region": "sa-east-1"
}
```

- Per i dettagli sull'API, consulta DeleteConnection AWS CLICommand Reference.

`delete-direct-connect-gateway-association`

Il seguente esempio di codice mostra come utilizzare`delete-direct-connect-gateway-association`.

AWS CLI

Come eliminare l'associazione del gateway Direct Connect

L'esempio `delete-direct-connect-gateway-association` seguente elimina l'associazione del gateway Direct Connect a un gateway di transito con l'ID di associazione specificato.

```
aws directconnect delete-direct-connect-gateway-association --association-id
  be85116d-46eb-4b43-a27a-da0c2ad648de
```

Output:

```
{
    "directConnectGatewayAssociation": {
        "directConnectGatewayId": "11460968-4ac1-4fd3-bdb2-00599EXAMPlE",
        "directConnectGatewayOwnerAccount": "123456789012",
        "associationState": "disassociating",
        "associatedGateway": {
```

```
            "id": "tgw-095b3b0b54EXAMPLE",
            "type": "transitGateway",
            "ownerAccount": "123456789012",
            "region": "us-east-1"
        },
         "associationId": " be85116d-46eb-4b43-a27a-da0c2ad648deEXAMPLE ",
         "allowedPrefixesToDirectConnectGateway": [
            {
                "cidr": "192.0.1.0/28"
            }
         ]
    }
}
```

Per ulteriori informazioni, consulta Associazione e annullamento dell'associazione di gateway di transito nella Guida per l'utente di AWS Direct Connect.

- Per i dettagli sull'API, consulta DeleteDirectConnectGatewayAssociation AWS CLICommand Reference.

`delete-direct-connect-gateway`

Il seguente esempio di codice mostra come utilizzare`delete-direct-connect-gateway`.

AWS CLI

Come eliminare un gateway Direct Connect

L'esempio seguente elimina il gateway Direct Connect `5f294f92-bafb-4011-916d-9b0bexample`.

Comando:

```
aws directconnect delete-direct-connect-gateway --direct-connect-gateway-
id 5f294f92-bafb-4011-916d-9b0bexample
```

Output:

```
{
  "directConnectGateway": {
      "amazonSideAsn": 64512,
      "directConnectGatewayId": "5f294f92-bafb-4011-916d-9b0bexample",
```

```
        "ownerAccount": "123456789012",
        "directConnectGatewayName": "DxGateway1",
        "directConnectGatewayState": "deleting"
    }
}
```

- Per i dettagli sull'API, consulta DeleteDirectConnectGateway AWS CLICommand Reference.

`delete-interconnect`

Il seguente esempio di codice mostra come utilizzare`delete-interconnect`.

AWS CLI

Come eliminare un'interconnessione

Il comando `delete-interconnect` seguente elimina l'interconnessione specificata:

```
aws directconnect delete-interconnect --interconnect-id dxcon-fgktov66
```

Output:

```
{
    "interconnectState": "deleted"
}
```

- Per i dettagli sull'API, consulta DeleteInterconnect AWS CLICommand Reference.

`delete-lag`

Il seguente esempio di codice mostra come utilizzare`delete-lag`.

AWS CLI

Come eliminare un LAG

L'esempio seguente elimina il LAG specificato.

Comando:

```
aws directconnect delete-lag --lag-id dxlag-ffrhowd9
```

Output:

```
{
  "awsDevice": "EqDC2-4h6ce2r1bes6",
  "numberOfConnections": 0,
  "lagState": "deleted",
  "ownerAccount": "123456789012",
  "lagName": "TestLAG",
  "connections": [],
  "lagId": "dxlag-ffrhowd9",
  "minimumLinks": 0,
  "connectionsBandwidth": "1Gbps",
  "region": "us-east-1",
  "location": "EqDC2"
}
```

- Per i dettagli sull'API, consulta DeleteLag AWS CLICommand Reference.

`delete-virtual-interface`

Il seguente esempio di codice mostra come utilizzare`delete-virtual-interface`.

AWS CLI

Come eliminare un'interfaccia virtuale

Il comando `delete-virtual-interface` seguente elimina l'interfaccia virtuale specificata:

```
aws directconnect delete-virtual-interface --virtual-interface-id dxvif-ffhhk74f
```

Output:

```
{
    "virtualInterfaceState": "deleting"
}
```

- Per i dettagli sull'API, consulta DeleteVirtualInterface AWS CLICommand Reference.

`describe-connection-loa`

Il seguente esempio di codice mostra come utilizzare`describe-connection-loa`.

AWS CLI

Come descrivere la LOA-CFA per una connessione utilizzando Linux o Mac OS X

L'esempio seguente descrive la LOA-CFA per la connessione `dxcon-fh6ayh1d`. I contenuti della LOA-CFA sono con codifica Base64. Questo comando utilizza i parametri `--output` e `--query` per controllare l'output ed estrarre il contenuto della struttura `loaContent`. La parte finale del comando decodifica il contenuto utilizzando l'utilità `base64` e invia l'output a un file PDF.

```
aws directconnect describe-connection-loa --connection-id dxcon-fh6ayh1d --
output text --query loa.loaContent|base64 --decode > myLoaCfa.pdf
```

Come descrivere la LOA-CFA per una connessione utilizzando Windows

L'esempio precedente richiede l'uso dell'utilità `base64` per decodificare l'output. Su un computer Windows, è possibile utilizzare `certutil`. Nel seguente esempio, il primo comando descrive la LOA-CFA per la connessione `dxcon-fh6ayh1d` e utilizza i parametri `--output` e `--query` per controllare l'output ed estrarre il contenuto della struttura `loaContent` in un file chiamato `myLoaCfa.base64`. Il secondo comando utilizza l'utility `certutil` per decodificare il file e inviare l'output in un file PDF.

```
aws directconnect describe-connection-loa --connection-id dxcon-fh6ayh1d --
output text --query loa.loaContent > myLoaCfa.base64
```

```
certutil -decode myLoaCfa.base64 myLoaCfa.pdf
```

Per ulteriori informazioni sul controllo dell'output della AWS CLI, vedere Controllo dell'output dei comandi dall'interfaccia a riga di AWS comando nella Guida per l'utente dell'interfaccia a riga di AWS comando.

- Per i dettagli sull'API, consulta DescribeConnectionLoa AWS CLICommand Reference.

`describe-connections-on-interconnect`

Il seguente esempio di codice mostra come utilizzare`describe-connections-on-interconnect`.

AWS CLI

Come elencare le connessioni su un'interconnessione

Il comando `describe-connections-on-interconnect` seguente elenca le connessioni che sono state allocate all'interconnessione indicata:

```
aws directconnect describe-connections-on-interconnect --interconnect-id dxcon-fgktov66
```

Output:

```
{
    "connections": [
        {
            "partnerName": "TIVIT",
            "vlan": 101,
            "ownerAccount": "123456789012",
            "connectionId": "dxcon-ffzc51m1",
            "connectionState": "ordering",
            "bandwidth": "500Mbps",
            "location": "TIVIT",
            "connectionName": "mydcinterconnect",
            "region": "sa-east-1"
        }
    ]
}
```

- Per i dettagli sull'API, consulta DescribeConnectionsOnInterconnect AWS CLICommand Reference.

`describe-connections`

Il seguente esempio di codice mostra come utilizzare`describe-connections`.

AWS CLI

Come elencare tutte le connessioni nella Regione corrente

Il comando `describe-connections` seguente elenca tutte le connessioni nella Regione corrente:

```
aws directconnect describe-connections
```

Output:

```
{
  "connections": [
      {
          "awsDevice": "EqDC2-123h49s71dabc",
          "ownerAccount": "123456789012",
          "connectionId": "dxcon-fguhmqlc",
          "lagId": "dxlag-ffrz71kw",
          "connectionState": "down",
          "bandwidth": "1Gbps",
          "location": "EqDC2",
          "connectionName": "My_Connection",
          "loaIssueTime": 1491568964.0,
          "region": "us-east-1"
      }
  ]
}
```

- Per i dettagli sull'API, consulta DescribeConnections AWS CLICommand Reference.

`describe-direct-connect-gateway-association-proposals`

Il seguente esempio di codice mostra come utilizzare`describe-direct-connect-gateway-association-proposals`.

AWS CLI

Come descrivere le proposte di associazione del gateway Direct Connect

L'esempio `describe-direct-connect-gateway-association-proposals` seguente visualizza i dettagli sulle proposte di associazione del gateway Direct Connect.

```
aws directconnect describe-direct-connect-gateway-association-proposals
```

Output:

```
{
    "directConnectGatewayAssociationProposals": [
        {
            "proposalId": "c2ede9b4-bbc6-4d33-923c-bc4feEXAMPLE",
            "directConnectGatewayId": "11460968-4ac1-4fd3-bdb2-00599EXAMPLE",
            "directConnectGatewayOwnerAccount": "111122223333",
            "proposalState": "requested",
```

```
            "associatedGateway": {
                "id": "tgw-02f776b1a7EXAMPLE",
                "type": "transitGateway",
                "ownerAccount": "111122223333",
                "region": "us-east-1"
            },
            "existingAllowedPrefixesToDirectConnectGateway": [
                {
                    "cidr": "192.168.2.0/30"
                },
                {
                    "cidr": "192.168.1.0/30"
                }
            ],
            "requestedAllowedPrefixesToDirectConnectGateway": [
                {
                    "cidr": "192.168.1.0/30"
                }
            ]
        },
        {
            "proposalId": "cb7f41cb-8128-43a5-93b1-dcaedEXAMPLE",
            "directConnectGatewayId": "11560968-4ac1-4fd3-bcb2-00599EXAMPLE",
            "directConnectGatewayOwnerAccount": "111122223333",
            "proposalState": "accepted",
            "associatedGateway": {
                "id": "tgw-045776b1a7EXAMPLE",
                "type": "transitGateway",
                "ownerAccount": "111122223333",
                "region": "us-east-1"
            },
            "existingAllowedPrefixesToDirectConnectGateway": [
                {
                    "cidr": "192.168.4.0/30"
                },
                {
                    "cidr": "192.168.5.0/30"
                }
            ],
            "requestedAllowedPrefixesToDirectConnectGateway": [
                {
                    "cidr": "192.168.5.0/30"
                }
            ]
```

```
        }
    ]
}
```

Per ulteriori informazioni, consulta Associazione e annullamento dell'associazione di gateway di transito nella Guida per l'utente di AWS Direct Connect.

- Per i dettagli sull'API, consulta DescribeDirectConnectGatewayAssociationProposals AWS CLICommand Reference.

`describe-direct-connect-gateway-associations`

Il seguente esempio di codice mostra come utilizzare`describe-direct-connect-gateway-associations`.

AWS CLI

Come descrivere le associazioni del gateway Direct Connect

L'esempio seguente descrive tutte le associazioni con il gateway Direct Connect `5f294f92-bafb-4011-916d-9b0bexample`.

Comando:

```
aws directconnect describe-direct-connect-gateway-associations --direct-connect-gateway-id 5f294f92-bafb-4011-916d-9b0bexample
```

Output:

```
{
  "nextToken":
 "eyJ2IjoxLCJzIjoxLCJpIjoiOU83OTFodzdycnZCbkN4MExHeHVwQT09IiwiYyI6InIxTEN0UEVHV0I1UFlkaWFnNl
  "directConnectGatewayAssociations": [
      {
          "associationState": "associating",
          "virtualGatewayOwnerAccount": "123456789012",
          "directConnectGatewayId": "5f294f92-bafb-4011-916d-9b0bexample",
          "virtualGatewayId": "vgw-6efe725e",
          "virtualGatewayRegion": "us-east-2"
      },
      {
          "associationState": "disassociating",
```

```
            "virtualGatewayOwnerAccount": "123456789012",
            "directConnectGatewayId": "5f294f92-bafb-4011-916d-9b0bexample",
            "virtualGatewayId": "vgw-ebaa27db",
            "virtualGatewayRegion": "us-east-2"
        }
    ]
}
```

- Per i dettagli sull'API, consulta DescribeDirectConnectGatewayAssociations AWS CLICommand Reference.

`describe-direct-connect-gateway-attachments`

Il seguente esempio di codice mostra come utilizzare`describe-direct-connect-gateway-attachments`.

AWS CLI

Come descrivere i collegamenti del gateway Direct Connect

L'esempio seguente descrive le interfacce virtuali collegate al gateway Direct Connect `5f294f92-bafb-4011-916d-9b0bexample`.

Comando:

```
aws directconnect describe-direct-connect-gateway-attachments --direct-connect-gateway-id 5f294f92-bafb-4011-916d-9b0bexample
```

Output:

```
{
    "directConnectGatewayAttachments": [
        {
            "virtualInterfaceOwnerAccount": "123456789012",
            "directConnectGatewayId": "5f294f92-bafb-4011-916d-9b0bexample",
            "virtualInterfaceRegion": "us-east-2",
            "attachmentState": "attaching",
            "virtualInterfaceId": "dxvif-fg9zyabc"
        }
    ],
    "nextToken":
 "eyJ2IjoxLCJzIjoxLCJpIjoibEhXdlNpUXF5RzhoL1JyUW52SlV2QT09IiwiYyI6Im5wQjFHQ0RyQUdRS3puNnNXcU
```

```
}
```

- Per i dettagli sull'API, consulta DescribeDirectConnectGatewayAttachments AWS CLICommand Reference.

`describe-direct-connect-gateways`

Il seguente esempio di codice mostra come utilizzare`describe-direct-connect-gateways`.

AWS CLI

Come descrivere i gateway Direct Connect

L'esempio seguente descrive tutti i gateway Direct Connect.

Comando:

```
aws directconnect describe-direct-connect-gateways
```

Output:

```
{
  "directConnectGateways": [
      {
          "amazonSideAsn": 64512,
          "directConnectGatewayId": "cf68415c-f4ae-48f2-87a7-3b52cexample",
          "ownerAccount": "123456789012",
          "directConnectGatewayName": "DxGateway2",
          "directConnectGatewayState": "available"
      },
      {
          "amazonSideAsn": 64512,
          "directConnectGatewayId": "5f294f92-bafb-4011-916d-9b0bdexample",
          "ownerAccount": "123456789012",
          "directConnectGatewayName": "DxGateway1",
          "directConnectGatewayState": "available"
      }
  ]
}
```

- Per i dettagli sull'API, consulta DescribeDirectConnectGateways AWS CLICommand Reference.

`describe-hosted-connections`

Il seguente esempio di codice mostra come utilizzare`describe-hosted-connections`.

AWS CLI

Come elencare le connessioni su un'interconnessione

L'esempio seguente elenca le connessioni che sono state allocate all'interconnessione indicata.

Comando:

```
aws directconnect describe-hosted-connections --connection-id dxcon-fgktov66
```

Output:

```
{
    "connections": [
        {
            "partnerName": "TIVIT",
            "vlan": 101,
            "ownerAccount": "123456789012",
            "connectionId": "dxcon-ffzc51m1",
            "connectionState": "ordering",
            "bandwidth": "500Mbps",
            "location": "TIVIT",
            "connectionName": "mydcinterconnect",
            "region": "sa-east-1"
        }
    ]
}
```

- Per i dettagli sull'API, consulta DescribeHostedConnections AWS CLICommand Reference.

`describe-interconnect-loa`

Il seguente esempio di codice mostra come utilizzare`describe-interconnect-loa`.

AWS CLI

Come descrivere la LOA-CFA per un'interconnessione utilizzando Linux o Mac OS X

L'esempio seguente descrive la LOA-CFA per l'interconnessione `dxcon-fh6ayh1d`. I contenuti della LOA-CFA sono con codifica Base64. Questo comando utilizza i parametri `--output` e `--query` per controllare l'output ed estrarre il contenuto della struttura `loaContent`. La parte finale del comando decodifica il contenuto utilizzando l'utilità `base64` e invia l'output a un file PDF.

```
aws directconnect describe-interconnect-loa --interconnect-id dxcon-fh6ayh1d --
output text --query loa.loaContent|base64 --decode > myLoaCfa.pdf
```

Come descrivere la LOA-CFA per un'interconnessione utilizzando Windows

L'esempio precedente richiede l'uso dell'utilità `base64` per decodificare l'output. Su un computer Windows, è possibile utilizzare `certutil`. Nel seguente esempio, il primo comando descrive la LOA-CFA per l'interconnessione `dxcon-fh6ayh1d` e utilizza i parametri `--output` e `--query` per controllare l'output ed estrarre il contenuto della struttura `loaContent` in un file chiamato `myLoaCfa.base64`. Il secondo comando utilizza l'utility `certutil` per decodificare il file e inviare l'output in un file PDF.

```
aws directconnect describe-interconnect-loa --interconnect-id dxcon-fh6ayh1d --
output text --query loa.loaContent > myLoaCfa.base64
```

```
certutil -decode myLoaCfa.base64 myLoaCfa.pdf
```

Per ulteriori informazioni sul controllo dell'output della AWS CLI, vedere Controllo dell'output dei comandi dall'interfaccia a riga di AWS comando nella Guida per l'utente dell'interfaccia a riga di AWS comando.

- Per i dettagli sull'API, consulta DescribeInterconnectLoa AWS CLICommand Reference.

`describe-interconnects`

Il seguente esempio di codice mostra come utilizzare`describe-interconnects`.

AWS CLI

Come elencare le interconnessioni

Il `describe-interconnects` comando seguente elenca le interconnessioni di proprietà AWS dell'account:

```
aws directconnect describe-interconnects
```

Output:

```
{
    "interconnects": [
        {
            "region": "sa-east-1",
            "bandwidth": "1Gbps",
            "location": "TIVIT",
            "interconnectName": "1G Interconnect to AWS",
            "interconnectId": "dxcon-fgktov66",
            "interconnectState": "down"
        }
    ]
}
```

- Per i dettagli sull'API, consulta DescribeInterconnects AWS CLICommand Reference.

`describe-lags`

Il seguente esempio di codice mostra come utilizzare`describe-lags`.

AWS CLI

Per descrivere il tuo LAGs

Il comando seguente descrive tutti i dati LAGs relativi alla regione corrente.

Comando:

```
aws directconnect describe-lags
```

Output:

```
{
  "lags": [
      {
          "awsDevice": "EqDC2-19y7z3m17xpuz",
          "numberOfConnections": 2,
          "lagState": "down",
          "ownerAccount": "123456789012",
          "lagName": "DA-LAG",
```

```
            "connections": [
                {
                    "ownerAccount": "123456789012",
                    "connectionId": "dxcon-ffnikghc",
                    "lagId": "dxlag-fgsu9erb",
                    "connectionState": "requested",
                    "bandwidth": "10Gbps",
                    "location": "EqDC2",
                    "connectionName": "Requested Connection 1 for Lag dxlag-fgsu9erb",
                    "region": "us-east-1"
                },
                {
                    "ownerAccount": "123456789012",
                    "connectionId": "dxcon-fglgbdea",
                    "lagId": "dxlag-fgsu9erb",
                    "connectionState": "requested",
                    "bandwidth": "10Gbps",
                    "location": "EqDC2",
                    "connectionName": "Requested Connection 2 for Lag dxlag-fgsu9erb",
                    "region": "us-east-1"
                }
            ],
            "lagId": "dxlag-fgsu9erb",
            "minimumLinks": 0,
            "connectionsBandwidth": "10Gbps",
            "region": "us-east-1",
            "location": "EqDC2"
        }
    ]
}
```


- Per i dettagli sull'API, consulta DescribeLags AWS CLICommand Reference.

`describe-loa`

Il seguente esempio di codice mostra come utilizzare`describe-loa`.

AWS CLI

Come descrivere la LOA-CFA per una connessione utilizzando Linux o Mac OS X

L'esempio seguente descrive la LOA-CFA per la connessione `dxcon-fh6ayh1d`. I contenuti della LOA-CFA sono con codifica Base64. Questo comando utilizza i parametri `--output` e `--`

`query` per controllare l'output ed estrarre il contenuto della struttura `loaContent`. La parte finale del comando decodifica il contenuto utilizzando l'utilità `base64` e invia l'output a un file PDF.

```
aws directconnect describe-loa --connection-id dxcon-fh6ayh1d --output text --
query loa.loaContent|base64 --decode > myLoaCfa.pdf
```

Come descrivere la LOA-CFA per una connessione utilizzando Windows

L'esempio precedente richiede l'uso dell'utilità `base64` per decodificare l'output. Su un computer Windows, è possibile utilizzare `certutil`. Nel seguente esempio, il primo comando descrive la LOA-CFA per la connessione `dxcon-fh6ayh1d` e utilizza i parametri `--output` e `--query` per controllare l'output ed estrarre il contenuto della struttura `loaContent` in un file chiamato `myLoaCfa.base64`. Il secondo comando utilizza l'utility `certutil` per decodificare il file e inviare l'output in un file PDF.

```
aws directconnect describe-loa --connection-id dxcon-fh6ayh1d --output text --
query loa.loaContent > myLoaCfa.base64
```

```
certutil -decode myLoaCfa.base64 myLoaCfa.pdf
```

Per ulteriori informazioni sul controllo dell'output della AWS CLI, vedere Controllo dell'output dei comandi dall'interfaccia a riga di AWS comando nella Guida per l'utente dell'interfaccia a riga di AWS comando.

- Per i dettagli sull'API, consulta DescribeLoa AWS CLICommand Reference.

`describe-locations`

Il seguente esempio di codice mostra come utilizzare`describe-locations`.

AWS CLI

Per elencare i partner e le sedi di AWS Direct Connect

Il `describe-locations` comando seguente elenca i partner e le sedi di AWS Direct Connect nella regione corrente:

```
aws directconnect describe-locations
```

Output:

```
{
    "locations": [
        {
            "locationName": "NAP do Brasil, Barueri, Sao Paulo",
            "locationCode": "TNDB"
        },
        {
            "locationName": "Tivit - Site Transamerica (Sao Paulo)",
            "locationCode": "TIVIT"
        }
    ]
}
```

- Per i dettagli sull'API, consulta DescribeLocations AWS CLICommand Reference.

`describe-tags`

Il seguente esempio di codice mostra come utilizzare`describe-tags`.

AWS CLI

Per descrivere i tag delle risorse AWS Direct Connect

Il comando seguente descrive i tag per la connessione `dxcon-abcabc12`.

Comando:

```
aws directconnect describe-tags --resource-arns arn:aws:directconnect:us-
east-1:123456789012:dxcon/dxcon-abcabc12
```

Output:

```
{
  "resourceTags": [
      {
          "resourceArn": "arn:aws:directconnect:us-east-1:123456789012:dxcon/dxcon-
abcabc12",
          "tags": [
              {
                  "value": "VAConnection",
                  "key": "Name"
              }
```

```
            ]
        }
    ]
}
```

- Per i dettagli sull'API, consulta DescribeTags AWS CLICommand Reference.

`describe-virtual-gateways`

Il seguente esempio di codice mostra come utilizzare`describe-virtual-gateways`.

AWS CLI

Come elencare i gateway privati virtuali

Il `describe-virtual-gateways` comando seguente elenca i gateway privati virtuali di proprietà AWS dell'account:

```
aws directconnect describe-virtual-gateways
```

Output:

```
{
    "virtualGateways": [
        {
            "virtualGatewayId": "vgw-aba37db6",
            "virtualGatewayState": "available"
        }
    ]
}
```

- Per i dettagli sull'API, consulta DescribeVirtualGateways AWS CLICommand Reference.

`describe-virtual-interfaces`

Il seguente esempio di codice mostra come utilizzare`describe-virtual-interfaces`.

AWS CLI

Come elencare tutte le interfacce virtuali

Il `describe-virtual-interfaces` comando seguente elenca le informazioni su tutte le interfacce virtuali associate all' AWS account:

```
aws directconnect describe-virtual-interfaces --connection-id dxcon-ffjrkx17
```

Output:

```
{
    "virtualInterfaces": [
        {
            "virtualInterfaceState": "down",
            "asn": 65000,
            "vlan": 101,
            "customerAddress": "192.168.1.2/30",
            "ownerAccount": "123456789012",
            "connectionId": "dxcon-ffjrkx17",
            "virtualGatewayId": "vgw-aba37db6",
            "virtualInterfaceId": "dxvif-ffhhk74f",
            "authKey": "asdf34example",
            "routeFilterPrefixes": [],
            "location": "TIVIT",
            "customerRouterConfig": "<?xml version=\"1.0\" encoding=
\"UTF-8\"?>\n<logical_connection id=\"dxvif-ffhhk74f\">\n  <vlan>101</
vlan>\n  <customer_address>192.168.1.2/30</customer_address>\n
 <amazon_address>192.168.1.1/30</amazon_address>\n  <bgp_asn>65000</bgp_asn>\n
 <bgp_auth_key>asdf34example</bgp_auth_key>\n  <amazon_bgp_asn>7224</amazon_bgp_asn>
\n  <connection_type>private</connection_type>\n</logical_connection>\n",
            "amazonAddress": "192.168.1.1/30",
            "virtualInterfaceType": "private",
            "virtualInterfaceName": "PrivateVirtualInterface"
        },
        {
            "virtualInterfaceState": "verifying",
            "asn": 65000,
            "vlan": 2000,
            "customerAddress": "203.0.113.2/30",
            "ownerAccount": "123456789012",
            "connectionId": "dxcon-ffjrkx17",
            "virtualGatewayId": "",
            "virtualInterfaceId": "dxvif-fgh0hcrk",
            "authKey": "asdf34example",
            "routeFilterPrefixes": [
                {
```

```
                    "cidr": "203.0.113.4/30"
                },
                {
                    "cidr": "203.0.113.0/30"
                }
            ],
            "location": "TIVIT",
            "customerRouterConfig": "<?xml version=\"1.0\" encoding=
\"UTF-8\"?>\n<logical_connection id=\"dxvif-fgh0hcrk\">\n  <vlan>2000</
vlan>\n  <customer_address>203.0.113.2/30</customer_address>\n
 <amazon_address>203.0.113.1/30</amazon_address>\n  <bgp_asn>65000</bgp_asn>\n
 <bgp_auth_key>asdf34example</bgp_auth_key>\n  <amazon_bgp_asn>7224</amazon_bgp_asn>
\n  <connection_type>public</connection_type>\n</logical_connection>\n",
            "amazonAddress": "203.0.113.1/30",
            "virtualInterfaceType": "public",
            "virtualInterfaceName": "PublicVirtualInterface"
        }
    ]
}
```

- Per i dettagli sull'API, consulta DescribeVirtualInterfaces AWS CLICommand Reference.

`disassociate-connection-from-lag`

Il seguente esempio di codice mostra come utilizzare`disassociate-connection-from-lag`.

AWS CLI

Come annullare l'associazione di una connessione a un LAG

L'esempio seguente annulla l'associazione della connessione specificata al LAG specificato.

Comando:

```
aws directconnect disassociate-connection-from-lag --lag-id dxlag-fhccu14t --
connection-id  dxcon-fg9607vm
```

Output:

```
{
  "ownerAccount": "123456789012",
  "connectionId": "dxcon-fg9607vm",
```

```
  "connectionState": "requested",
  "bandwidth": "1Gbps",
  "location": "EqDC2",
  "connectionName": "Con2ForLag",
  "region": "us-east-1"
}
```

- Per i dettagli sull'API, consulta DisassociateConnectionFromLag AWS CLICommand Reference.

`tag-resource`

Il seguente esempio di codice mostra come utilizzare`tag-resource`.

AWS CLI

Per aggiungere un tag a una risorsa AWS Direct Connect

Il comando seguente aggiunge un tag con la chiave `Name` e il valore `VAConnection` alla connessione `dxcon-abcabc12`. Se il comando va a buon fine, non viene restituito alcun output.

Comando:

```
aws directconnect tag-resource --resource-arn arn:aws:directconnect:us-
east-1:123456789012:dxcon/dxcon-abcabc12 --tags "key=Name,value=VAConnection"
```

- Per i dettagli sull'API, consulta TagResource AWS CLICommand Reference.

`untag-resource`

Il seguente esempio di codice mostra come utilizzare`untag-resource`.

AWS CLI

Per rimuovere un tag da una risorsa AWS Direct Connect

Il comando seguente rimuove il tag con la chiave `Name` dalla connessione `dxcon-abcabc12`. Se il comando va a buon fine, non viene restituito alcun output.

Comando:

```
aws directconnect untag-resource --resource-arn arn:aws:directconnect:us-
east-1:123456789012:dxcon/dxcon-abcabc12 --tag-keys Name
```

- Per i dettagli sull'API, consulta UntagResource AWS CLICommand Reference.

`update-direct-connect-gateway-association`

Il seguente esempio di codice mostra come utilizzare`update-direct-connect-gateway-association`.

AWS CLI

Come aggiornare gli attributi specificati dell'associazione al gateway Direct Connect

L'esempio `update-direct-connect-gateway-association` seguente aggiunge l'intervallo CIDR specificato a un'associazione al gateway Direct Connect.

```
aws directconnect update-direct-connect-gateway-association \
    --association-id 820a6e4f-5374-4004-8317-3f64bEXAMPLE \
    --add-allowed-prefixes-to-direct-connect-gateway cidr=192.168.2.0/30
```

Output:

```
{
    "directConnectGatewayAssociation": {
        "directConnectGatewayId": "11460968-4ac1-4fd3-bdb2-00599EXAMPLE",
        "directConnectGatewayOwnerAccount": "111122223333",
        "associationState": "updating",
        "associatedGateway": {
            "id": "tgw-02f776b1a7EXAMPLE",
            "type": "transitGateway",
            "ownerAccount": "111122223333",
            "region": "us-east-1"
        },
        "associationId": "820a6e4f-5374-4004-8317-3f64bEXAMPLE",
        "allowedPrefixesToDirectConnectGateway": [
            {
                "cidr": "192.168.2.0/30"
            },
            {
                "cidr": "192.168.1.0/30"
            }
```

```
        ]
    }
}
```

Per ulteriori informazioni, consulta Utilizzo dei gateway Direct Connect nella Guida per l'utente di AWS Direct Connect.

- Per i dettagli sull'API, consulta UpdateDirectConnectGatewayAssociation AWS CLICommand Reference.

`update-lag`

Il seguente esempio di codice mostra come utilizzare`update-lag`.

AWS CLI

Come aggiornare un LAG

L'esempio seguente modifica il nome del LAG specificato.

Comando:

```
aws directconnect update-lag --lag-id dxlag-ffjhj9lx --lag-name 2ConnLag
```

Output:

```
{
  "awsDevice": "CSVA1-23u8tlpaz8iks",
  "numberOfConnections": 2,
  "lagState": "down",
  "ownerAccount": "123456789012",
  "lagName": "2ConnLag",
  "connections": [
      {
          "ownerAccount": "123456789012",
          "connectionId": "dxcon-fflqyj95",
          "lagId": "dxlag-ffjhj9lx",
          "connectionState": "requested",
          "bandwidth": "1Gbps",
          "location": "CSVA1",
          "connectionName": "Requested Connection 2 for Lag dxlag-ffjhj9lx",
          "region": "us-east-1"
      },
```

```
        {
            "ownerAccount": "123456789012",
            "connectionId": "dxcon-ffqr6x5q",
            "lagId": "dxlag-ffjhj9lx",
            "connectionState": "requested",
            "bandwidth": "1Gbps",
            "location": "CSVA1",
            "connectionName": "Requested Connection 1 for Lag dxlag-ffjhj9lx",
            "region": "us-east-1"
        }
    ],
    "lagId": "dxlag-ffjhj9lx",
    "minimumLinks": 0,
    "connectionsBandwidth": "1Gbps",
    "region": "us-east-1",
    "location": "CSVA1"
}
```

- Per i dettagli sull'API, consulta UpdateLag AWS CLICommand Reference.

`update-virtual-interface-attributes`

Il seguente esempio di codice mostra come utilizzare`update-virtual-interface-attributes`.

AWS CLI

Come aggiornare la MTU di un'interfaccia virtuale

L'esempio `update-virtual-interface-attributes` seguente aggiorna la MTU dell'interfaccia virtuale specificata.

```
aws directconnect update-virtual-interface-attributes \
    --virtual-interface-id dxvif-fEXAMPLE \
    --mtu 1500
```

Output:

```
{
    "ownerAccount": "1111222233333",
    "virtualInterfaceId": "dxvif-fEXAMPLE",
    "location": "loc1",
    "connectionId": "dxlag-fEXAMPLE",
```

```
    "virtualInterfaceType": "transit",
    "virtualInterfaceName": "example transit virtual interface",
    "vlan": 125,
    "asn": 650001,
    "amazonSideAsn": 64512,
    "authKey": "0xzxgA9YoW9h58u8SEXAMPLE",
    "amazonAddress": "169.254.248.1/30",
    "customerAddress": "169.254.248.2/30",
    "addressFamily": "ipv4",
    "virtualInterfaceState": "down",
    "customerRouterConfig": "<?xml version=\"1.0\" encoding=\"UTF-8\"?
>\n<logical_connection id=\"dxvif-fEXAMPLE\">\n  <vlan>125</vlan>
\n  <customer_address>169.254.248.2/30</customer_address>\n
 <amazon_address>169.254.248.1/30</amazon_address>\n  <bgp_asn>650001</bgp_asn>\n
  <bgp_auth_key>0xzxgA9YoW9h58u8SEXAMPLE</bgp_auth_key>\n  <amazon_bgp_asn>64512</
amazon_bgp_asn>\n  <connection_type>transit</connection_type>\n</logical_connection>
\n",
    "mtu": 1500,
    "jumboFrameCapable": true,
    "virtualGatewayId": "",
    "directConnectGatewayId": "879b76a1-403d-4700-8b53-4a56ed85436e",
    "routeFilterPrefixes": [],
    "bgpPeers": [
        {
            "bgpPeerId": "dxpeer-fEXAMPLE",
            "asn": 650001,
            "authKey": "0xzxgA9YoW9h58u8SEXAMPLE",
            "addressFamily": "ipv4",
            "amazonAddress": "169.254.248.1/30",
            "customerAddress": "169.254.248.2/30",
            "bgpPeerState": "available",
            "bgpStatus": "down",
            "awsDeviceV2": "loc1-26wz6vEXAMPLE"
        }
    ],
    "region": "sa-east-1",
    "awsDeviceV2": "loc1-26wz6vEXAMPLE",
    "tags": []
}
```


Per ulteriori informazioni, consulta Impostazione di MTU di rete per interfacce virtuali private o di transito nella Guida per l'utente di AWS Direct Connect.

- Per i dettagli sull'API, consulta UpdateVirtualInterfaceAttributes AWS CLICommand Reference.

Directory Service esempi utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with Directory Service.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`describe-directories`

Il seguente esempio di codice mostra come utilizzare`describe-directories`.

AWS CLI

Come ottenere i dettagli delle directory

L'esempio `describe-directories` seguente visualizza i dettagli sulla directory specificata.

```
aws ds describe-directories \
   --directory-id d-a1b2c3d4e5
```

Output:

```
{
    "DirectoryDescriptions": [
        {
            "DirectoryId": "d-a1b2c3d4e5",
            "Name": "mydirectory.example.com",
            "ShortName": "mydirectory",
            "Size": "Small",
            "Edition": "Standard",
```

```
            "Alias": "d-a1b2c3d4e5",
            "AccessUrl": "d-a1b2c3d4e5.awsapps.com",
            "Stage": "Active",
            "ShareStatus": "Shared",
            "ShareMethod": "HANDSHAKE",
            "ShareNotes": "These are my share notes",
            "LaunchTime": "2019-07-08T15:33:46.327000-07:00",
            "StageLastUpdatedDateTime": "2019-07-08T15:59:12.307000-07:00",
            "Type": "SharedMicrosoftAD",
            "SsoEnabled": false,
            "DesiredNumberOfDomainControllers": 0,
            "OwnerDirectoryDescription": {
                "DirectoryId": "d-b2c3d4e5f6",
                "AccountId": "123456789111",
                "DnsIpAddrs": [
                    "203.113.0.248",
                    "203.113.0.253"
                ],
                "VpcSettings": {
                    "VpcId": "vpc-a1b2c3d4",
                    "SubnetIds": [
                        "subnet-a1b2c3d4",
                        "subnet-d4c3b2a1"
                    ],
                    "AvailabilityZones": [
                        "us-west-2a",
                        "us-west-2c"
                    ]
                }
            }
        }
    ]
}
```

- Per i dettagli sull'API, consulta DescribeDirectories AWS CLICommand Reference.

`describe-trusts`

Il seguente esempio di codice mostra come utilizzare`describe-trusts`.

AWS CLI

Come ottenere dettagli sulle relazioni di attendibilità

L'esempio `describe-trusts` seguente visualizza i dettagli sulle relazioni di attendibilità per la directory specificata.

```
aws ds describe-trusts \
   --directory-id d-a1b2c3d4e5
```

Output:

```
{
    "Trusts": [
        {
            "DirectoryId": "d-a1b2c3d4e5",
            "TrustId": "t-9a8b7c6d5e",
            "RemoteDomainName": "other.example.com",
            "TrustType": "Forest",
            "TrustDirection": "Two-Way",
            "TrustState": "Verified",
            "CreatedDateTime": "2017-06-20T18:08:45.614000-07:00",
            "LastUpdatedDateTime": "2019-06-04T10:52:12.410000-07:00",
            "StateLastUpdatedDateTime": "2019-06-04T10:52:12.410000-07:00",
            "SelectiveAuth": "Disabled"
        }
    ]
}
```

- Per i dettagli sull'API, consulta DescribeTrusts AWS CLICommand Reference.

Directory Service Esempi di dati utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with Directory Service Data.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`add-group-member`

Il seguente esempio di codice mostra come utilizzare`add-group-member`.

AWS CLI

Come aggiungere un membro del gruppo a una directory

L'esempio `add-group-member` seguente aggiunge l'utente specificato al gruppo specificato nella directory specificata.

```
aws ds-data add-group-member \
    --directory-id d-1234567890 \
    --group-name 'sales' \
    --member-name 'john.doe'
```

Questo comando non produce alcun output.

Per ulteriori informazioni, vedere Aggiungere o rimuovere membri di AWS Managed Microsoft AD dai gruppi e dai gruppi ai gruppi nella AWS Directory Service Administration Guide.

- Per i dettagli sull'API, consulta AddGroupMember AWS CLICommand Reference.

`create-group`

Il seguente esempio di codice mostra come utilizzare`create-group`.

AWS CLI

Come elencare i widget disponibili

L'esempio `create-group` seguente crea un gruppo in una directory specificata.

```
aws ds-data create-group \
    --directory-id d-1234567890 \
    --sam-account-name "sales"
```

Output:

```
{
    "DirectoryId": "d-1234567890",
    "SAMAccountName": "sales",
    "SID": "S-1-2-34-5567891234-5678912345-67891234567-8912"
}
```

Per ulteriori informazioni, vedere Creazione di un gruppo Microsoft AD AWS gestito nella AWS Directory Service Administration Guide.

- Per i dettagli sull'API, consulta CreateGroup AWS CLICommand Reference.

`create-user`

Il seguente esempio di codice mostra come utilizzare`create-user`.

AWS CLI

Come creare un utente

L'esempio `create-user` seguente crea un utente nella directory specificata.

```
aws ds-data create-user \
    --directory-id d-1234567890 \
    --sam-account-name 'john.doe'
```

Output:

```
{
    "DirectoryId": "d-1234567890",
    "SAMAccountName": "john.doe",
    "SID": "S-1-2-34-5567891234-5678912345-67891234567-8912"
}
```

Per ulteriori informazioni, vedere Creazione di un utente Microsoft AD AWS gestito nella AWS Directory Service Administration Guide.

- Per i dettagli sull'API, consulta CreateUser AWS CLICommand Reference.

`delete-group`

Il seguente esempio di codice mostra come utilizzare`delete-group`.

AWS CLI

Come eliminare un gruppo

L'esempio `delete-group` seguente elimina il gruppo specificato dalla directory specificata.

```
aws ds-data delete-group \
    --directory-id d-1234567890 \
    --sam-account-name 'sales'
```

Questo comando non produce alcun output.

Per ulteriori informazioni, vedere Eliminazione di un gruppo Microsoft AD AWS gestito nella AWS Directory Service Administration Guide.

- Per i dettagli sull'API, vedere DeleteGroupin AWS CLI Command Reference.

`delete-user`

Il seguente esempio di codice mostra come utilizzare`delete-user`.

AWS CLI

Come eliminare un utente

L'esempio `delete-user` seguente elimina l'utente specificato dalla directory specificata.

```
aws ds-data delete-user \
    --directory-id d-1234567890 \
    --sam-account-name 'john.doe'
```

Questo comando non produce alcun output.

Per ulteriori informazioni, vedere Eliminazione di un utente Microsoft AD AWS gestito nella AWS Directory Service Administration Guide.

- Per i dettagli sull'API, vedere DeleteUserin AWS CLI Command Reference.

`describe-group`

Il seguente esempio di codice mostra come utilizzare`describe-group`.

AWS CLI

Come elencare i dettagli di un gruppo

L'esempio `describe-group` seguente ottiene informazioni per il gruppo specificato nella directory specificata.

```
aws ds-data describe-group \
    --directory-id d-1234567890 \
    --sam-account-name 'sales'
```

Output:

```
{
    "DirectoryId": "d-1234567890",
    "DistinguishedName": "CN=sales,OU=Users,OU=CORP,DC=corp,DC=example,DC=com",
    "GroupScope": "Global",
    "GroupType": "Security",
    "Realm": "corp.example.com",
    "SAMAccountName": "sales",
    "SID": "S-1-2-34-5567891234-5678912345-67891234567-8912"
}
```

Per ulteriori informazioni, vedere Visualizzazione e aggiornamento dei dettagli di un gruppo AWS Managed Microsoft AD nella AWS Directory Service Administration Guide.

- Per i dettagli sull'API, consulta DescribeGroup AWS CLICommand Reference.

`describe-user`

Il seguente esempio di codice mostra come utilizzare`describe-user`.

AWS CLI

Come elencare le informazioni relative a un utente

L'esempio `describe-user` seguente ottiene informazioni sull'utente specificato nella directory specificata.

```
aws ds-data describe-user command-name \
    --directory-id d-1234567890 \
    --sam-account-name 'john.doe'
```

Output:

```
{
    "DirectoryId": "d-1234567890",
    "DistinguishedName": "CN=john.doe,OU=Users,OU=CORP,DC=corp,DC=example,DC=com",
    "Enabled": false,
    "Realm": "corp.example.com",
    "SAMAccountName": "john.doe",
    "SID": "S-1-2-34-5678901234-5678901234-5678910123-4567",
    "UserPrincipalName": "john.doe@CORP.EXAMPLE.COM"
}
```

Per ulteriori informazioni, vedere Visualizzazione e aggiornamento di un utente di Microsoft AD AWS gestito nella AWS Directory Service Administration Guide.

- Per i dettagli sull'API, vedere DescribeUserin AWS CLI Command Reference.

`disable-directory-data-access`

Il seguente esempio di codice mostra come utilizzare`disable-directory-data-access`.

AWS CLI

Come disabilitare l'API Directory Service Data per una directory

L'esempio `disable-directory-data-access` seguente disabilita l'API Directory Service Data per la directory specificata.

```
aws ds disable-directory-data-access \
    --directory-id d-1234567890
```

Questo comando non produce alcun output.

Per ulteriori informazioni, vedere Abilitazione o disabilitazione della gestione di utenti e gruppi o dei dati del servizio di AWS Directory nella AWS Directory Service Administration Guide.

- Per i dettagli sull'API, vedere DisableDirectoryDataAccessin AWS CLI Command Reference.

`disable-user`

Il seguente esempio di codice mostra come utilizzare`disable-user`.

AWS CLI

Come disabilitare un utente

L'esempio `disable-user` seguente disabilita l'utente specificato dalla directory specificata.

```
aws ds-data disable-user \
    --directory-id d-1234567890 \
    --sam-account-name 'john.doe'
```

Questo comando non produce alcun output.

Per ulteriori informazioni, vedere Disabilitazione di un utente Microsoft AD AWS gestito nella AWS Directory Service Administration Guide.

- Per i dettagli sull'API, vedere DisableUserin AWS CLI Command Reference.

`enable-directory-data-access`

Il seguente esempio di codice mostra come utilizzare`enable-directory-data-access`.

AWS CLI

Come abilitare l'API Directory Service Data per una directory

L'esempio `enable-directory-data-access` seguente abilita l'API Directory Service Data per la directory specificata.

```
aws ds enable-directory-data-access \
    --directory-id d-1234567890
```

Questo comando non produce alcun output.

Per ulteriori informazioni, vedere Abilitazione o disabilitazione della gestione di utenti e gruppi o dei dati del servizio di AWS Directory nella AWS Directory Service Administration Guide.

- Per i dettagli sull'API, vedere EnableDirectoryDataAccessin AWS CLI Command Reference.

`list-group-members`

Il seguente esempio di codice mostra come utilizzare`list-group-members`.

AWS CLI

Come elencare i membri del gruppo di una directory

L'esempio `list-group-members` seguente elenca i membri del gruppo specificato nella directory specificata.

```
aws ds-data list-group-members \
    --directory-id d-1234567890 \
    --sam-account-name 'sales'
```

Output:

```
{
    "Members": [
        {
            "MemberType": "USER",
            "SAMAccountName": "Jane Doe",
            "SID": "S-1-2-34-5678901234-5678901234-5678910123-4568"
        },
        {
            "MemberType": "USER",
            "SAMAccountName": "John Doe",
            "SID": "S-1-2-34-5678901234-5678901234-5678910123-4569"
        }
    ],
    "DirectoryId": "d-1234567890",
    "MemberRealm": "corp.example.com",
    "Realm": "corp.example.com"
}
```

Per ulteriori informazioni, vedere Visualizzazione e aggiornamento dei dettagli di un gruppo AWS Managed Microsoft AD nella AWS Directory Service Administration Guide.

- Per i dettagli sull'API, consulta ListGroupMembers AWS CLICommand Reference.

`list-groups-for-member`

Il seguente esempio di codice mostra come utilizzare`list-groups-for-member`.

AWS CLI

Come elencare l'appartenenza a un gruppo di una directory

L'esempio `list-groups-for-member` seguente elenca l'appartenenza al gruppo per l'utente specificato nella directory specificata.

```
aws ds-data list-groups-for-member \
    --directory-id d-1234567890 \
    --sam-account-name 'john.doe'
```

Output:

```
{
    "Groups": [
        {
            "GroupScope": "Global",
            "GroupType": "Security",
            "SAMAccountName": "Domain Users",
            "SID": "S-1-2-34-5678901234-5678901234-5678910123-4567"
        }
    ],
    "DirectoryId": "d-1234567890",
    "MemberRealm": "corp.example.com",
    "Realm": "corp.example.com"
}
```

Per ulteriori informazioni, vedere Visualizzazione e aggiornamento di un utente di Microsoft AD AWS gestito nella AWS Directory Service Administration Guide.

- Per i dettagli sull'API, vedere ListGroupsForMemberin AWS CLI Command Reference.

`list-groups`

Il seguente esempio di codice mostra come utilizzare`list-groups`.

AWS CLI

Come elencare i gruppi di una directory

L'esempio `list-groups` seguente elenca i gruppi nella directory specificata.

```
aws ds-data list-groups \
    --directory-id d-1234567890
```

Output:


```
{
    "Groups": [
        {
            "GroupScope": "BuiltinLocal",
            "GroupType": "Security",
            "SAMAccountName": "Administrators",
            "SID": "S-1-2-33-441"
        },
        {
            "GroupScope": "BuiltinLocal",
            "GroupType": "Security",
            "SAMAccountName": "Users",
            "SID": "S-1-2-33-442"
        },
        {
            "GroupScope": "BuiltinLocal",
            "GroupType": "Security",
            "SAMAccountName": "Guests",
            "SID": "S-1-2-33-443"
        },
        {
            "GroupScope": "BuiltinLocal",
            "GroupType": "Security",
            "SAMAccountName": "Print Operators",
            "SID": "S-1-2-33-444"
        },
        {
            "GroupScope": "BuiltinLocal",
            "GroupType": "Security",
            "SAMAccountName": "Backup Operators",
            "SID": "S-1-2-33-445"
        },
        {
            "GroupScope": "BuiltinLocal",
            "GroupType": "Security",
            "SAMAccountName": "Replicator",
            "SID": "S-1-2-33-446"
        },
        {
            "GroupScope": "BuiltinLocal",
            "GroupType": "Security",
            "SAMAccountName": "Remote Desktop Users",
```

```
            "SID": "S-1-2-33-447"
        },
        {
            "GroupScope": "BuiltinLocal",
            "GroupType": "Security",
            "SAMAccountName": "Network Configuration Operators",
            "SID": "S-1-2-33-448"
        },
        {
            "GroupScope": "BuiltinLocal",
            "GroupType": "Security",
            "SAMAccountName": "Performance Monitor Users",
            "SID": "S-1-2-33-449"
        },
        {
            "GroupScope": "BuiltinLocal",
            "GroupType": "Security",
            "SAMAccountName": "Performance Log Users",
            "SID": "S-1-2-33-450"
        },
        {
            "GroupScope": "BuiltinLocal",
            "GroupType": "Security",
            "SAMAccountName": "Distributed COM Users",
            "SID": "S-1-2-33-451"
        },
        {
            "GroupScope": "BuiltinLocal",
            "GroupType": "Security",
            "SAMAccountName": "IIS_IUSRS",
            "SID": "S-1-2-33-452"
        },
        {
            "GroupScope": "BuiltinLocal",
            "GroupType": "Security",
            "SAMAccountName": "Cryptographic Operators",
            "SID": "S-1-2-33-453"
        },
        {
            "GroupScope": "BuiltinLocal",
            "GroupType": "Security",
            "SAMAccountName": "Event Log Readers",
            "SID": "S-1-2-33-454"
        },
```

```
        {
            "GroupScope": "BuiltinLocal",
            "GroupType": "Security",
            "SAMAccountName": "Certificate Service DCOM Access",
            "SID": "S-1-2-33-456"
        },
        {
            "GroupScope": "BuiltinLocal",
            "GroupType": "Security",
            "SAMAccountName": "RDS Remote Access Servers",
            "SID": "S-1-2-33-457"
        },
        {
            "GroupScope": "BuiltinLocal",
            "GroupType": "Security",
            "SAMAccountName": "RDS Endpoint Servers",
            "SID": "S-1-2-33-458"
        },
        {
            "GroupScope": "BuiltinLocal",
            "GroupType": "Security",
            "SAMAccountName": "RDS Management Servers",
            "SID": "S-1-2-33-459"
        },
        {
            "GroupScope": "BuiltinLocal",
            "GroupType": "Security",
            "SAMAccountName": "Hyper-V Administrators",
            "SID": "S-1-2-33-460"
        },
        {
            "GroupScope": "BuiltinLocal",
            "GroupType": "Security",
            "SAMAccountName": "Access Control Assistance Operators",
            "SID": "S-1-2-33-461"
        },
        {
            "GroupScope": "BuiltinLocal",
            "GroupType": "Security",
            "SAMAccountName": "Remote Management Users",
            "SID": "S-1-2-33-462"
        },
        {
            "GroupScope": "BuiltinLocal",
```

```
            "GroupType": "Security",
            "SAMAccountName": "Storage Replica Administrators",
            "SID": "S-1-2-33-463"
        },
        {
            "GroupScope": "Global",
            "GroupType": "Security",
            "SAMAccountName": "Domain Computers",
            "SID": "S-1-2-34-56789123456-7891012345-6789123486-789"
        },
        {
            "GroupScope": "Global",
            "GroupType": "Security",
            "SAMAccountName": "Domain Controllers",
            "SID": "S-1-2-34-56789123456-7891012345-6789123486-790"
        },
        {
            "GroupScope": "Universal",
            "GroupType": "Security",
            "SAMAccountName": "Schema Admins",
            "SID": "S-1-2-34-56789123456-7891012345-6789123486-791"
        },
        {
            "GroupScope": "Universal",
            "GroupType": "Security",
            "SAMAccountName": "Enterprise Admins",
            "SID": "S-1-2-34-56789123456-7891012345-6789123486-792"
        },
        {
            "GroupScope": "DomainLocal",
            "GroupType": "Security",
            "SAMAccountName": "Cert Publishers",
            "SID": "S-1-2-34-56789123456-7891012345-6789123486-793"
        },
        {
            "GroupScope": "Global",
            "GroupType": "Security",
            "SAMAccountName": "Domain Admins",
            "SID": "S-1-2-34-56789123456-7891012345-6789123486-794"
        },
        {
            "GroupScope": "Global",
            "GroupType": "Security",
            "SAMAccountName": "Domain Users",
```

```
            "SID": "S-1-2-34-56789123456-7891012345-6789123486-795"
        },
        {
            "GroupScope": "Global",
            "GroupType": "Security",
            "SAMAccountName": "Domain Guests",
            "SID": "S-1-2-34-56789123456-7891012345-6789123486-796"
        },
        {
            "GroupScope": "Global",
            "GroupType": "Security",
            "SAMAccountName": "Group Policy Creator Owners",
            "SID": "S-1-2-34-56789123456-7891012345-6789123486-797"
        },
        {
            "GroupScope": "DomainLocal",
            "GroupType": "Security",
            "SAMAccountName": "RAS and IAS Servers",
            "SID": "S-1-2-34-56789123456-7891012345-6789123486-798"
        },
        {
            "GroupScope": "BuiltinLocal",
            "GroupType": "Security",
            "SAMAccountName": "Server Operators",
            "SID": "S-1-2-33-464"
        },
        {
            "GroupScope": "BuiltinLocal",
            "GroupType": "Security",
            "SAMAccountName": "Account Operators",
            "SID": "S-1-2-33-465"
        },
        {
            "GroupScope": "BuiltinLocal",
            "GroupType": "Security",
            "SAMAccountName": "Pre-Windows 2000 Compatible Access",
            "SID": "S-1-2-33-466"
        },
        {
            "GroupScope": "BuiltinLocal",
            "GroupType": "Security",
            "SAMAccountName": "Incoming Forest Trust Builders",
            "SID": "S-1-2-33-467"
        },
```

```
        {
            "GroupScope": "BuiltinLocal",
            "GroupType": "Security",
            "SAMAccountName": "Windows Authorization Access Group",
            "SID": "S-1-2-33-468"
        },
        {
            "GroupScope": "BuiltinLocal",
            "GroupType": "Security",
            "SAMAccountName": "Terminal Server License Servers",
            "SID": "S-1-2-33-469"
        },
        {
            "GroupScope": "DomainLocal",
            "GroupType": "Security",
            "SAMAccountName": "Allowed RODC Password Replication Group",
            "SID": "S-1-2-34-56789123456-7891012345-6789123486-798"
        },
        {
            "GroupScope": "DomainLocal",
            "GroupType": "Security",
            "SAMAccountName": "Denied RODC Password Replication Group",
            "SID": "S-1-2-34-56789123456-7891012345-6789123486-799"
        },
        {
            "GroupScope": "Global",
            "GroupType": "Security",
            "SAMAccountName": "Read-only Domain Controllers",
            "SID": "S-1-2-34-56789123456-7891012345-6789123486-800"
        },
        {
            "GroupScope": "Universal",
            "GroupType": "Security",
            "SAMAccountName": "Enterprise Read-only Domain Controllers",
            "SID": "S-1-2-34-56789123456-7891012345-6789123486-801"
        },
        {
            "GroupScope": "Global",
            "GroupType": "Security",
            "SAMAccountName": "Cloneable Domain Controllers",
            "SID": "S-1-2-34-56789123456-7891012345-6789123486-802"
        },
        {
            "GroupScope": "Global",
```

```
            "GroupType": "Security",
            "SAMAccountName": "Protected Users",
            "SID": "S-1-2-34-56789123456-7891012345-6789123486-803"
        },
        {
            "GroupScope": "Global",
            "GroupType": "Security",
            "SAMAccountName": "Key Admins",
            "SID": "S-1-2-34-56789123456-7891012345-6789123486-804"
        },
        {
            "GroupScope": "Universal",
            "GroupType": "Security",
            "SAMAccountName": "Enterprise Key Admins",
            "SID": "S-1-2-34-56789123456-7891012345-6789123486-805"
        },
        {
            "GroupScope": "DomainLocal",
            "GroupType": "Security",
            "SAMAccountName": "DnsAdmins",
            "SID": "S-1-2-34-5678901234-5678901234-5678910123-4567"
        },
        {
            "GroupScope": "Global",
            "GroupType": "Security",
            "SAMAccountName": "DnsUpdateProxy",
            "SID": "S-1-2-34-5678901234-5678901234-5678910123-4568"
        },
        {
            "GroupScope": "DomainLocal",
            "GroupType": "Security",
            "SAMAccountName": "Admins",
            "SID": "S-1-2-34-5678901234-5678901234-5678910123-4569"
        },
        {
            "GroupScope": "DomainLocal",
            "GroupType": "Security",
            "SAMAccountName": "AWSAdministrators",
            "SID": "S-1-2-34-5678901234-5678901234-5678910123-4570"
        },
        {
            "GroupScope": "DomainLocal",
            "GroupType": "Security",
            "SAMAccountName": "AWS Object Management Service Accounts",
```

```
            "SID": "S-1-2-34-5678901234-5678901234-5678910123-4571"
        },
        {
            "GroupScope": "DomainLocal",
            "GroupType": "Security",
            "SAMAccountName": "AWS Private CA Connector for AD Delegated Group",
            "SID": "S-1-2-34-5678901234-5678901234-5678910123-4572"
        },
        {
            "GroupScope": "DomainLocal",
            "GroupType": "Security",
            "SAMAccountName": "AWS Application and Service Delegated Group",
            "SID": "S-1-2-34-5678901234-5678901234-5678910123-4573"
        },
        {
            "GroupScope": "DomainLocal",
            "GroupType": "Security",
            "SAMAccountName": "AWS Delegated Administrators",
            "SID": "S-1-2-34-5678901234-5678901234-5678910123-4574"
        },
        {
            "GroupScope": "DomainLocal",
            "GroupType": "Security",
            "SAMAccountName": "AWS Delegated FSx Administrators",
            "SID": "S-1-2-34-5678901234-5678901234-5678910123-4575"
        },
        {
            "GroupScope": "DomainLocal",
            "GroupType": "Security",
            "SAMAccountName": "AWS Delegated Account Operators",
            "SID": "S-1-2-34-5678901234-5678901234-5678910123-4576"
        },
        {
            "GroupScope": "DomainLocal",
            "GroupType": "Security",
            "SAMAccountName": "AWS Delegated Active Directory Based Activation
Administrators",
            "SID": "S-1-2-34-5678901234-5678901234-5678910123-4577"
        },
        {
            "GroupScope": "DomainLocal",
            "GroupType": "Security",
            "SAMAccountName": "AWS Delegated Allowed to Authenticate Objects",
            "SID": "S-1-2-34-5678901234-5678901234-5678910123-4578"
```

```
        },
        {
            "GroupScope": "DomainLocal",
            "GroupType": "Security",
            "SAMAccountName": "AWS Delegated Allowed to Authenticate to Domain
Controllers",
            "SID": "S-1-2-34-5678901234-5678901234-5678910123-4579"
        },
        {
            "GroupScope": "DomainLocal",
            "GroupType": "Security",
            "SAMAccountName": "AWS Delegated Deleted Object Lifetime
Administrators",
            "SID": "S-1-2-34-5678901234-5678901234-5678910123-4580"
        },
        {
            "GroupScope": "DomainLocal",
            "GroupType": "Security",
            "SAMAccountName": "AWS Delegated Distributed File System
Administrators",
            "SID": "S-1-2-34-5678901234-5678901234-5678910123-4581"
        },
        {
            "GroupScope": "DomainLocal",
            "GroupType": "Security",
            "SAMAccountName": "AWS Delegated Dynamic Host Configuration Protocol
Administrators",
            "SID": "S-1-2-34-5678901234-5678901234-5678910123-4582"
        },
        {
            "GroupScope": "DomainLocal",
            "GroupType": "Security",
            "SAMAccountName": "AWS Delegated Enterprise Certificate Authority
Administrators",
            "SID": "S-1-2-34-5678901234-5678901234-5678910123-4583"
        },
        {
            "GroupScope": "DomainLocal",
            "GroupType": "Security",
            "SAMAccountName": "AWS Delegated Fine Grained Password Policy
Administrators",
            "SID": "S-1-2-34-5678901234-5678901234-5678910123-4584"
        },
        {
```

```
            "GroupScope": "DomainLocal",
            "GroupType": "Security",
            "SAMAccountName": "AWS Delegated Group Policy Administrators",
            "SID": "S-1-2-34-5678901234-5678901234-5678910123-4585"
        },
        {
            "GroupScope": "DomainLocal",
            "GroupType": "Security",
            "SAMAccountName": "AWS Delegated Managed Service Account
Administrators",
            "SID": "S-1-2-34-5678901234-5678901234-5678910123-4586"
        },
        {
            "GroupScope": "DomainLocal",
            "GroupType": "Security",
            "SAMAccountName": "AWS Delegated Read Foreign Security Principals",
            "SID": "S-1-2-34-5678901234-5678901234-5678910123-4587"
        },
        {
            "GroupScope": "DomainLocal",
            "GroupType": "Security",
            "SAMAccountName": "AWS Delegated Remote Access Service Administrators",
            "SID": "S-1-2-34-5678901234-5678901234-5678910123-4588"
        },
        {
            "GroupScope": "DomainLocal",
            "GroupType": "Security",
            "SAMAccountName": "AWS Delegated Replicate Directory Changes
Administrators",
            "SID": "S-1-2-34-5678901234-5678901234-5678910123-4588"
        },
        {
            "GroupScope": "DomainLocal",
            "GroupType": "Security",
            "SAMAccountName": "AWS Delegated Sites and Services Administrators",
            "SID": "S-1-2-34-5678901234-5678901234-5678910123-4589"
        },
        {
            "GroupScope": "DomainLocal",
            "GroupType": "Security",
            "SAMAccountName": "AWS Delegated System Management Administrators",
            "SID": "S-1-2-34-5678901234-5678901234-5678910123-4590"
        },
        {
```

```
            "GroupScope": "DomainLocal",
            "GroupType": "Security",
            "SAMAccountName": "AWS Delegated Terminal Server Licensing
Administrators",
            "SID": "S-1-2-34-5678901234-5678901234-5678910123-4591"
        },
        {
            "GroupScope": "DomainLocal",
            "GroupType": "Security",
            "SAMAccountName": "AWS Delegated User Principal Name Suffix
Administrators",
            "SID": "S-1-2-34-5678901234-5678901234-5678910123-4592"
        },
        {
            "GroupScope": "DomainLocal",
            "GroupType": "Security",
            "SAMAccountName": "AWS Delegated Add Workstations To Domain Users",
            "SID": "S-1-2-34-5678901234-5678901234-5678910123-4593"
        },
        {
            "GroupScope": "DomainLocal",
            "GroupType": "Security",
            "SAMAccountName": "AWS Delegated Domain Name System Administrators",
            "SID": "S-1-2-34-5678901234-5678901234-5678910123-4594"
        },
        {
            "GroupScope": "DomainLocal",
            "GroupType": "Security",
            "SAMAccountName": "AWS Delegated Kerberos Delegation Administrators",
            "SID": "S-1-2-34-5678901234-5678901234-5678910123-4595"
        },
        {
            "GroupScope": "DomainLocal",
            "GroupType": "Security",
            "SAMAccountName": "AWS Delegated Server Administrators",
            "SID": "S-1-2-34-5678901234-5678901234-5678910123-4596"
        },
        {
            "GroupScope": "DomainLocal",
            "GroupType": "Security",
            "SAMAccountName": "AWS Delegated MS-NPRC Non-Compliant Devices",
            "SID": "S-1-2-34-5678901234-5678901234-5678910123-4597"
        },
        {
```

```
            "GroupScope": "Global",
            "GroupType": "Security",
            "SAMAccountName": "Remote Access",
            "SID": "S-1-2-34-5678901234-5678901234-5678910123-4598"
        },
        {
            "GroupScope": "Global",
            "GroupType": "Security",
            "SAMAccountName": "Accounting",
            "SID": "S-1-2-34-5678901234-5678901234-5678910123-4599"
        },
        {
            "GroupScope": "Global",
            "GroupType": "Distribution",
            "SAMAccountName": "sales",
            "SID": "S-1-2-34-5678901234-5678901234-5678910123-4567"
        }
    ],
    "DirectoryId": "d-1234567890",
    "Realm": "corp.example.com"
}
```

Per ulteriori informazioni, vedere Visualizzazione e aggiornamento dei dettagli di un gruppo AWS Managed Microsoft AD nella AWS Directory Service Administration Guide.

- Per i dettagli sull'API, consulta ListGroups AWS CLICommand Reference.

`list-users`

Il seguente esempio di codice mostra come utilizzare`list-users`.

AWS CLI

Come elencare gli utenti di una directory

L'esempio `list-users` seguente elenca gli utenti nella directory specificata.

```
aws ds-data list-users \
    --directory-id d-1234567890
```

Output:

```
{
```

```
    "Users": [
        {
            "Enabled": true,
            "SAMAccountName": "Administrator",
            "SID": "S-1-2-34-5678910123-4567895012-3456789012-345"
        },
        {
            "Enabled": false,
            "SAMAccountName": "Guest",
            "SID": "S-1-2-34-5678910123-4567895012-3456789012-345"
        },
        {
            "Enabled": false,
            "SAMAccountName": "krbtgt",
            "SID": "S-1-2-34-5678910123-4567895012-3456789012-346"
        },
        {
            "Enabled": true,
            "SAMAccountName": "Admin",
            "SID": "S-1-2-34-5678910123-4567895012-3456789012-347"
        },
        {
            "Enabled": true,
            "SAMAccountName": "Richard Roe",
            "SID": "S-1-2-34-5678910123-4567895012-3456789012-348"
        },
        {
            "Enabled": true,
            "SAMAccountName": "Jane Doe",
            "SID": "S-1-2-34-5678910123-4567895012-3456789012-349"
        },
        {
            "Enabled": true,
            "SAMAccountName": "AWS_WGnzYlN6YyY",
            "SID": "S-1-2-34-5678901234-5678901234-5678910123-4567"
        },
        {
            "Enabled": true,
            "SAMAccountName": "john.doe",
            "SID": "S-1-2-34-5678901234-5678901234-5678910123-4568"
        }
    ],
    "DirectoryId": "d-1234567890",
    "Realm": "corp.example.com"
```

```
}
```

Per ulteriori informazioni, vedere Visualizzazione e aggiornamento di un utente di Microsoft AD AWS gestito nella AWS Directory Service Administration Guide.

- Per i dettagli sull'API, vedere ListUsersin AWS CLI Command Reference.

`remove-group-member`

Il seguente esempio di codice mostra come utilizzare`remove-group-member`.

AWS CLI

Come rimuovere il membro di un gruppo da una directory

L'esempio `remove-group-member` seguente rimuove un membro del gruppo specificato dal gruppo specificato nella directory specificata.

```
aws ds-data remove-group-member \
    --directory-id d-1234567890 \
    --group-name 'sales' \
    --member-name 'john.doe'
```

Questo comando non produce alcun output.

Per ulteriori informazioni, vedere Aggiungere e rimuovere membri di AWS Managed Microsoft AD dai gruppi e dai gruppi ai gruppi nella AWS Directory Service Administration Guide.

- Per i dettagli sull'API, consulta RemoveGroupMember AWS CLICommand Reference.

`reset-user-password`

Il seguente esempio di codice mostra come utilizzare`reset-user-password`.

AWS CLI

Come reimpostare la password di un utente in una directory

L'esempio `reset-user-password` seguente reimposta e abilita l'utente specificato nella directory specificata.

```
aws ds reset-user-password \
    --directory-id d-1234567890 \
    --user-name 'john.doe' \
    --new-password 'password'
```

Questo comando non produce alcun output.

Per ulteriori informazioni, vedere Reimpostazione e attivazione della password di un utente AWS Microsoft AD gestito nella AWS Directory Service Administration Guide.

- Per i dettagli sull'API, consulta AWS CLI Command ResetUserPasswordReference.

`search-groups`

Il seguente esempio di codice mostra come utilizzare`search-groups`.

AWS CLI

Come cercare un gruppo in una directory

L'esempio `search-groups` seguente cerca il gruppo specificato nella directory specificata.

```
aws ds-data search-groups \
    --directory-id d-1234567890 \
    --search-attributes 'SamAccountName' \
    --search-string 'sales'
```

Output:

```
{
    "Groups": [
        {
            "GroupScope": "Global",
            "GroupType": "Distribution",
            "SAMAccountName": "sales",
            "SID": "S-1-2-34-5678901234-5678901234-5678910123-4567"
        }
    ],
    "DirectoryId": "d-1234567890",
    "Realm": "corp.example.com"
}
```

Per ulteriori informazioni, vedere Visualizzazione e aggiornamento dei dettagli di un gruppo AWS Managed Microsoft AD nella AWS Directory Service Administration Guide.

- Per i dettagli sull'API, consulta SearchGroups AWS CLICommand Reference.

`search-users`

Il seguente esempio di codice mostra come utilizzare`search-users`.

AWS CLI

Come cercare un utente in una directory

L'esempio `search-users` seguente cerca l'utente specificato nella directory specificata.

```
aws ds-data search-users \
    --directory-id d-1234567890 \
    --search-attributes 'SamAccountName' \
    --Search-string 'john.doe'
```

Output:

```
{
    "Users": [
        {
            "Enabled": true,
            "SAMAccountName": "john.doe",
            "SID": "S-1-2-34-5678901234-5678901234-5678910123-4567"
        }
    ],
    "DirectoryId": "d-1234567890",
    "Realm": "corp.example.com"
}
```

Per ulteriori informazioni, vedere Visualizzazione e aggiornamento di un utente di Microsoft AD AWS gestito nella AWS Directory Service Administration Guide.

- Per i dettagli sull'API, vedere SearchUsersin AWS CLI Command Reference.

`update-group`

Il seguente esempio di codice mostra come utilizzareu`pdate-group`.

AWS CLI

Come aggiornare l'attributo di un gruppo in una directory

L'esempio `update-group` seguente aggiorna l'attributo specificato per il gruppo specificato nella directory specificata.

```
aws ds-data update-group \
    --directory-id d-1234567890 \
    --sam-account-name 'sales' \
    --update-type 'REPLACE' \
    --group-type 'Distribution'
```

Questo comando non produce alcun output.

Per ulteriori informazioni, vedere Visualizzazione e aggiornamento dei dettagli di un gruppo AWS Managed Microsoft AD nella AWS Directory Service Administration Guide.

- Per i dettagli sull'API, consulta UpdateGroup AWS CLICommand Reference.

`update-user`

Il seguente esempio di codice mostra come utilizzare`update-user`.

AWS CLI

Come aggiornare l'attributo di un utente in una directory

L'esempio `update-user` seguente aggiorna l'attributo specificato per l'utente specificato nella directory specificata.

```
aws ds-data update-user \
    --directory-id d-1234567890 \
    --sam-account-name 'john.doe' \
    --update-type 'ADD' \
    --email-address 'example.corp.com'
```

Questo comando non produce alcun output.

Per ulteriori informazioni, vedere Visualizzazione e aggiornamento di un utente di Microsoft AD AWS gestito nella AWS Directory Service Administration Guide.

- Per i dettagli sull'API, vedere UpdateUserin AWS CLI Command Reference.

AWS DMS esempi utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with AWS DMS.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`add-tags-to-resource`

Il seguente esempio di codice mostra come utilizzare`add-tags-to-resource`.

AWS CLI

Come aggiungere tag a una risorsa

L'esempio `add-tags-to-resource` seguente aggiunge dei tag all'istanza di replica.

```
aws dms add-tags-to-resource \
    --resource-arn arn:aws:dms:us-east-1:123456789012:rep:T3OM7OUB5NM2LCVZF7JPGJRNUE
 \
    --tags Key=Environment,Value=PROD Key=Project,Value=dbMigration
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Assegnazione di tag alle risorse nella Guida per l'utente di AWS Database Migration Service.

- Per i dettagli sull'API, consulta AddTagsToResource AWS CLICommand Reference.

`create-endpoint`

Il seguente esempio di codice mostra come utilizzare`create-endpoint`.

AWS CLI

Come creare un endpoint

L'esempio `create-endpoint` seguente crea un endpoint per un'origine Amazon S3.

```
aws dms create-endpoint \
    --endpoint-type source \
    --engine-name s3 \
    --endpoint-identifier src-endpoint \
    --s3-settings file://s3-settings.json
```

Contenuto di `s3-settings.json`:

```
{
    "BucketName":"my-corp-data",
    "BucketFolder":"sourcedata",
    "ServiceAccessRoleArn":"arn:aws:iam::123456789012:role/my-s3-access-role"
}
```

Output:

```
{
    "Endpoint": {
        "EndpointIdentifier": "src-endpoint",
        "EndpointType": "SOURCE",
        "EngineName": "s3",
        "EngineDisplayName": "Amazon S3",
        "ExtraConnectionAttributes": "bucketFolder=sourcedata;bucketName=my-corp-
data;compressionType=NONE;csvDelimiter=,;csvRowDelimiter=\\n;",
        "Status": "active",
        "EndpointArn": "arn:aws:dms:us-
east-1:123456789012:endpoint:GUVAFG34EECUOJ6QVZ56DAHT3U",
        "SslMode": "none",
        "ServiceAccessRoleArn": "arn:aws:iam::123456789012:role/my-s3-access-role",
        "S3Settings": {
            "ServiceAccessRoleArn": "arn:aws:iam::123456789012:role/my-s3-access-
role",
```

```
            "CsvRowDelimiter": "\\n",
            "CsvDelimiter": ",",
            "BucketFolder": "sourcedata",
            "BucketName": "my-corp-data",
            "CompressionType": "NONE",
            "EnableStatistics": true
        }
    }
}
```

Per ulteriori informazioni, vedere Working with AWS DMS Endpoints nella AWS Database Migration Service User Guide.

- Per i dettagli sull'API, consulta AWS CLI Command CreateEndpointReference.

`create-event-subscription`

Il seguente esempio di codice mostra come utilizzare`create-event-subscription`.

AWS CLI

Come elencare le sottoscrizioni agli eventi

L'`create-event-subscription`esempio seguente crea una sottoscrizione a un evento a un argomento Amazon SNS ()my-sns-topic.

```
aws dms create-event-subscription \
    --subscription-name my-dms-events \
    --sns-topic-arn arn:aws:sns:us-east-1:123456789012:my-sns-topic
```

Output:

```
{
    "EventSubscription": {
        "CustomerAwsId": "123456789012",
        "CustSubscriptionId": "my-dms-events",
        "SnsTopicArn": "arn:aws:sns:us-east-1:123456789012:my-sns-topic",
        "Status": "creating",
        "SubscriptionCreationTime": "2020-05-21 21:58:38.598",
        "Enabled": true
    }
}
```

Per ulteriori informazioni, consulta Utilizzo di eventi e notifiche nella Guida per l'utente di AWS Database Migration Service.

- Per i dettagli sull'API, consulta CreateEventSubscription AWS CLICommand Reference.

`create-replication-instance`

Il seguente esempio di codice mostra come utilizzare`create-replication-instance`.

AWS CLI

Come creare un'istanza di replica

L'esempio `create-replication-instance` seguente crea un'istanza di replica.

```
aws dms create-replication-instance \
    --replication-instance-identifier my-repl-instance \
    --replication-instance-class dms.t2.micro \
    --allocated-storage 5
```

Output:

```
{
    "ReplicationInstance": {
        "ReplicationInstanceIdentifier": "my-repl-instance",
        "ReplicationInstanceClass": "dms.t2.micro",
        "ReplicationInstanceStatus": "creating",
        "AllocatedStorage": 5,
        "VpcSecurityGroups": [
            {
                "VpcSecurityGroupId": "sg-f839b688",
                "Status": "active"
            }
        ],
        "ReplicationSubnetGroup": {
            "ReplicationSubnetGroupIdentifier": "default",
            "ReplicationSubnetGroupDescription": "default",
            "VpcId": "vpc-136a4c6a",
            "SubnetGroupStatus": "Complete",
            "Subnets": [
                {
                    "SubnetIdentifier": "subnet-da327bf6",
                    "SubnetAvailabilityZone": {
```

```
                    "Name": "us-east-1a"
                },
                "SubnetStatus": "Active"
            },
            {
                "SubnetIdentifier": "subnet-42599426",
                "SubnetAvailabilityZone": {
                    "Name": "us-east-1d"
                },
                "SubnetStatus": "Active"
            },
            {
                "SubnetIdentifier": "subnet-bac383e0",
                "SubnetAvailabilityZone": {
                    "Name": "us-east-1c"
                },
                "SubnetStatus": "Active"
            },
            {
                "SubnetIdentifier": "subnet-6746046b",
                "SubnetAvailabilityZone": {
                    "Name": "us-east-1f"
                },
                "SubnetStatus": "Active"
            },
            {
                "SubnetIdentifier": "subnet-d7c825e8",
                "SubnetAvailabilityZone": {
                    "Name": "us-east-1e"
                },
                "SubnetStatus": "Active"
            },
            {
                "SubnetIdentifier": "subnet-cbfff283",
                "SubnetAvailabilityZone": {
                    "Name": "us-east-1b"
                },
                "SubnetStatus": "Active"
            }
        ]
    },
    "PreferredMaintenanceWindow": "sat:12:35-sat:13:05",
    "PendingModifiedValues": {},
    "MultiAZ": false,
```

```
        "EngineVersion": "3.3.2",
        "AutoMinorVersionUpgrade": true,
        "KmsKeyId": "arn:aws:kms:us-east-1:123456789012:key/f7bc0f8e-1a3a-4ace-9faa-
e8494fa3921a",
        "ReplicationInstanceArn": "arn:aws:dms:us-
east-1:123456789012:rep:ZK2VQBUWFDBAWHIXHAYG5G2PKY",
        "PubliclyAccessible": true
    }
}
```

Per ulteriori informazioni, vedere Working with an AWS DMS Replication Instance nella AWS Database Migration Service User Guide.

- Per i dettagli sull'API, consulta AWS CLI Command CreateReplicationInstanceReference.

`create-replication-subnet-group`

Il seguente esempio di codice mostra come utilizzare`create-replication-subnet-group`.

AWS CLI

Come creare un gruppo di sottoreti

L'esempio `create-replication-subnet-group` seguente crea un gruppo composto da 3 sottoreti.

```
aws dms create-replication-subnet-group \
    --replication-subnet-group-identifier my-subnet-group \
    --replication-subnet-group-description "my subnet group" \
    --subnet-ids subnet-da327bf6 subnet-bac383e0 subnet-d7c825e8
```

Output:

```
{
    "ReplicationSubnetGroup": {
        "ReplicationSubnetGroupIdentifier": "my-subnet-group",
        "ReplicationSubnetGroupDescription": "my subnet group",
        "VpcId": "vpc-136a4c6a",
        "SubnetGroupStatus": "Complete",
        "Subnets": [
            {
```

```
                "SubnetIdentifier": "subnet-da327bf6",
                "SubnetAvailabilityZone": {
                    "Name": "us-east-1a"
                },
                "SubnetStatus": "Active"
            },
            {
                "SubnetIdentifier": "subnet-bac383e0",
                "SubnetAvailabilityZone": {
                    "Name": "us-east-1c"
                },
                "SubnetStatus": "Active"
            },
            {
                "SubnetIdentifier": "subnet-d7c825e8",
                "SubnetAvailabilityZone": {
                    "Name": "us-east-1e"
                },
                "SubnetStatus": "Active"
            }
        ]
    }
}
```

Per ulteriori informazioni, consulta Impostazione di una rete per un'istanza di replica nella Guida per l'utente di AWS Database Migration Service.

- Per i dettagli sull'API, consulta CreateReplicationSubnetGroup AWS CLICommand Reference.

`create-replication-task`

Il seguente esempio di codice mostra come utilizzare`create-replication-task`.

AWS CLI

Come creare un'attività di replica

L'esempio `create-replication-task` seguente crea un'attività di replica.

```
aws dms create-replication-task \
    --replication-task-identifier movedata \
    --source-endpoint-arn arn:aws:dms:us-
east-1:123456789012:endpoint:6GGI6YPWWGAYUVLKIB732KEVWA \
```

```
    --target-endpoint-arn arn:aws:dms:us-
east-1:123456789012:endpoint:EOM4SFKCZEYHZBFGAGZT3QEC5U \
    --replication-instance-arn $RI_ARN \
    --migration-type full-load \
    --table-mappings file://table-mappings.json
```

Contenuto di `table-mappings.json`:

```
{
    "rules": [
        {
            "rule-type": "selection",
            "rule-id": "1",
            "rule-name": "1",
            "object-locator": {
                "schema-name": "prodrep",
                "table-name": "%"
            },
            "rule-action": "include",
            "filters": []
        }
    ]
}
```

Output:

```
{
    "ReplicationTask": {
        "ReplicationTaskIdentifier": "moveit2",
        "SourceEndpointArn": "arn:aws:dms:us-
east-1:123456789012:endpoint:6GGI6YPWWGAYUVLKIB732KEVWA",
        "TargetEndpointArn": "arn:aws:dms:us-
east-1:123456789012:endpoint:EOM4SFKCZEYHZBFGAGZT3QEC5U",
        "ReplicationInstanceArn": "arn:aws:dms:us-
east-1:123456789012:rep:T3OM7OUB5NM2LCVZF7JPGJRNUE",
        "MigrationType": "full-load",
        "TableMappings": ...output omitted... ,
        "ReplicationTaskSettings": ...output omitted... ,
        "Status": "creating",
        "ReplicationTaskCreationDate": 1590524772.505,
        "ReplicationTaskArn": "arn:aws:dms:us-
east-1:123456789012:task:K55IUCGBASJS5VHZJIINA45FII"
    }
```

```
}
```

Per ulteriori informazioni, vedere Working with AWS DMS Tasks nella AWS Database Migration Service User Guide.

- Per i dettagli sull'API, consulta CreateReplicationTask AWS CLICommand Reference.

`delete-connection`

Il seguente esempio di codice mostra come utilizzare`delete-connection`.

AWS CLI

Come eliminare una connessione

L'esempio `delete-connection` seguente dissocia un endpoint da un'istanza di replica.

```
aws dms delete-connection \
    --endpoint-arn arn:aws:dms:us-
east-1:123456789012:endpoint:6GGI6YPWWGAYUVLKIB732KEVWA \
    --replication-instance-arn arn:aws:dms:us-
east-1:123456789012:rep:T3OM7OUB5NM2LCVZF7JPGJRNUE
```

Output:

```
{
    "Connection": {
        "ReplicationInstanceArn": "arn:aws:dms:us-
east-1:123456789012:rep:T3OM7OUB5NM2LCVZF7JPGJRNUE",
        "EndpointArn": "arn:aws:dms:us-
east-1:123456789012:endpoint:6GGI6YPWWGAYUVLKIB732KEVWA",
        "Status": "deleting",
        "EndpointIdentifier": "src-database-1",
        "ReplicationInstanceIdentifier": "my-repl-instance"
    }
}
```

Per ulteriori informazioni, vedere https://docs.aws.amazon.com/dms/latest/userguide/CHAP_endpoints.creating.html nella Database Migration AWS Service User Guide.

- Per i dettagli sull'API, consulta Command Reference. DeleteConnectionAWS CLI

delete-endpoint

Il seguente esempio di codice mostra come utilizzaredelete-endpoint.

AWS CLI

Come eliminare un endpoint

L'esempio delete-endpoint seguente elimina un endpoint.

```
aws dms delete-endpoint \
    --endpoint-arn arn:aws:dms:us-
east-1:123456789012:endpoint:OUJJVXO4XZ4CYTSEG5XGMN2R3Y
```

Output:

```
{
    "Endpoint": {
        "EndpointIdentifier": "src-endpoint",
        "EndpointType": "SOURCE",
        "EngineName": "s3",
        "EngineDisplayName": "Amazon S3",
        "ExtraConnectionAttributes": "bucketFolder=sourcedata;bucketName=my-corp-
data;compressionType=NONE;csvDelimiter=,;csvRowDelimiter=\\n;",
        "Status": "deleting",
        "EndpointArn": "arn:aws:dms:us-
east-1:123456789012:endpoint:OUJJVXO4XZ4CYTSEG5XGMN2R3Y",
        "SslMode": "none",
        "ServiceAccessRoleArn": "arn:aws:iam::123456789012:role/my-s3-access-role",
        "S3Settings": {
            "ServiceAccessRoleArn": "arn:aws:iam::123456789012:role/my-s3-access-
role",
            "CsvRowDelimiter": "\\n",
            "CsvDelimiter": ",",
            "BucketFolder": "sourcedata",
            "BucketName": "my-corp-data",
            "CompressionType": "NONE",
            "EnableStatistics": true
        }
    }
}
```

Per ulteriori informazioni, vedere Working with AWS DMS Endpoints nella AWS Database Migration Service User Guide.

- Per i dettagli sull'API, consulta AWS CLI Command DeleteEndpointReference.

`delete-event-subscription`

Il seguente esempio di codice mostra come utilizzare`delete-event-subscription`.

AWS CLI

Come eliminare un abbonamento all'evento

L'esempio `delete-event-subscription` seguente elimina un abbonamento a un argomento Amazon SNS.

```
aws dms delete-event-subscription \
    --subscription-name "my-dms-events"
```

Output:

```
{
    "EventSubscription": {
        "CustomerAwsId": "123456789012",
        "CustSubscriptionId": "my-dms-events",
        "SnsTopicArn": "arn:aws:sns:us-east-1:123456789012:my-sns-topic",
        "Status": "deleting",
        "SubscriptionCreationTime": "2020-05-21 21:58:38.598",
        "Enabled": true
    }
}
```

Per ulteriori informazioni, consulta Utilizzo di eventi e notifiche nella Guida per l'utente di AWS Database Migration Service.

- Per i dettagli sull'API, consulta DeleteEventSubscription AWS CLICommand Reference.

`delete-replication-instance`

Il seguente esempio di codice mostra come utilizzare`delete-replication-instance`.

AWS CLI

Come eliminare un'istanza di replica

L'esempio `delete-replication-instance` seguente elimina un'istanza di replica.

```
aws dms delete-replication-instance \
    --replication-instance-arn arn:aws:dms:us-
east-1:123456789012:rep:T3OM7OUB5NM2LCVZF7JPGJRNUE
```

Output:

```
{
    "ReplicationInstance": {
        "ReplicationInstanceIdentifier": "my-repl-instance",
        "ReplicationInstanceClass": "dms.t2.micro",
        "ReplicationInstanceStatus": "deleting",
        "AllocatedStorage": 5,
        "InstanceCreateTime": 1590011235.952,
        "VpcSecurityGroups": [
            {
                "VpcSecurityGroupId": "sg-f839b688",
                "Status": "active"
            }
        ],
        "AvailabilityZone": "us-east-1e",
        "ReplicationSubnetGroup": {
            "ReplicationSubnetGroupIdentifier": "default",
            "ReplicationSubnetGroupDescription": "default",
            "VpcId": "vpc-136a4c6a",
            "SubnetGroupStatus": "Complete",
            "Subnets": [
                {
                    "SubnetIdentifier": "subnet-da327bf6",
                    "SubnetAvailabilityZone": {
                        "Name": "us-east-1a"
                    },
                    "SubnetStatus": "Active"
                },
                {
                    "SubnetIdentifier": "subnet-42599426",
                    "SubnetAvailabilityZone": {
                        "Name": "us-east-1d"
```

```
                    },
                    "SubnetStatus": "Active"
                },
                {
                    "SubnetIdentifier": "subnet-bac383e0",
                    "SubnetAvailabilityZone": {
                        "Name": "us-east-1c"
                    },
                    "SubnetStatus": "Active"
                },
                {
                    "SubnetIdentifier": "subnet-6746046b",
                    "SubnetAvailabilityZone": {
                        "Name": "us-east-1f"
                    },
                    "SubnetStatus": "Active"
                },
                {
                    "SubnetIdentifier": "subnet-d7c825e8",
                    "SubnetAvailabilityZone": {
                        "Name": "us-east-1e"
                    },
                    "SubnetStatus": "Active"
                },
                {
                    "SubnetIdentifier": "subnet-cbfff283",
                    "SubnetAvailabilityZone": {
                        "Name": "us-east-1b"
                    },
                    "SubnetStatus": "Active"
                }
            ]
        },
        "PreferredMaintenanceWindow": "wed:11:42-wed:12:12",
        "PendingModifiedValues": {},
        "MultiAZ": true,
        "EngineVersion": "3.3.2",
        "AutoMinorVersionUpgrade": true,
        "KmsKeyId": "arn:aws:kms:us-east-1:123456789012:key/f7bc0f8e-1a3a-4ace-9faa-
e8494fa3921a",
        "ReplicationInstanceArn": "arn:aws:dms:us-
east-1:123456789012:rep:T3OM7OUB5NM2LCVZF7JPGJRNUE",
        "ReplicationInstancePublicIpAddress": "54.225.120.92",
        "ReplicationInstancePrivateIpAddress": "172.31.30.121",
```

```
        "ReplicationInstancePublicIpAddresses": [
            "54.225.120.92",
            "3.230.18.248"
        ],
        "ReplicationInstancePrivateIpAddresses": [
            "172.31.30.121",
            "172.31.75.90"
        ],
        "PubliclyAccessible": true,
        "SecondaryAvailabilityZone": "us-east-1b"
    }
}
```

Per ulteriori informazioni, vedere Working with an AWS DMS Replication Instance nella AWS Database Migration Service User Guide.

- Per i dettagli sull'API, consulta AWS CLI Command DeleteReplicationInstanceReference.

`delete-replication-subnet-group`

Il seguente esempio di codice mostra come utilizzare`delete-replication-subnet-group`.

AWS CLI

Come eliminare un gruppo di sottoreti

L'esempio `delete-replication-subnet-group` seguente elimina un gruppo di sottoreti.

```
aws dms delete-replication-subnet-group \
--replication-subnet-group-identifier my-subnet-group
```

Output:

```
(none)
```

Per ulteriori informazioni, consulta Impostazione di una rete per un'istanza di replica nella Guida per l'utente di AWS Database Migration Service.

- Per i dettagli sull'API, consulta DeleteReplicationSubnetGroup AWS CLICommand Reference.

`delete-replication-task`

Il seguente esempio di codice mostra come utilizzare`delete-replication-task`.

AWS CLI

Come eliminare un'attività di replica

L'esempio `delete-replication-task` seguente elimina un'istanza di replica.

```
aws dms delete-replication-task \
    --replication-task-arn arn:aws:dms:us-
east-1:123456789012:task:K55IUCGBASJS5VHZJIINA45FII
```

Output:

```
{
    "ReplicationTask": {
        "ReplicationTaskIdentifier": "moveit2",
        "SourceEndpointArn": "arn:aws:dms:us-
east-1:123456789012:endpoint:6GGI6YPWWGAYUVLKIB732KEVWA",
        "TargetEndpointArn": "arn:aws:dms:us-
east-1:123456789012:endpoint:EOM4SFKCZEYHZBFGAGZT3QEC5U",
        "ReplicationInstanceArn": "arn:aws:dms:us-
east-1:123456789012:rep:T3OM7OUB5NM2LCVZF7JPGJRNUE",
        "MigrationType": "full-load",
        "TableMappings": ...output omitted...,
        "ReplicationTaskSettings": ...output omitted...,
        "Status": "deleting",
        "StopReason": "Stop Reason FULL_LOAD_ONLY_FINISHED",
        "ReplicationTaskCreationDate": 1590524772.505,
        "ReplicationTaskStartDate": 1590789988.677,
        "ReplicationTaskArn": "arn:aws:dms:us-
east-1:123456789012:task:K55IUCGBASJS5VHZJIINA45FII"
    }
}
```

Per ulteriori informazioni, vedere Working with AWS DMS Tasks nella AWS Database Migration Service User Guide.

- Per i dettagli sull'API, consulta DeleteReplicationTask AWS CLICommand Reference.

`describe-account-attributes`

Il seguente esempio di codice mostra come utilizzare`describe-account-attributes`.

AWS CLI

Come descrivere gli attributi dell'account

L'`describe-account-attributes`esempio seguente elenca gli attributi del tuo AWS account.

```
aws dms describe-account-attributes
```

Output:

```
{
    "AccountQuotas": [
        {
            "AccountQuotaName": "ReplicationInstances",
            "Used": 1,
            "Max": 20
        },
        {
            "AccountQuotaName": "AllocatedStorage",
            "Used": 5,
            "Max": 10000
        },

        ...remaining output omitted...

    ],
    "UniqueAccountIdentifier": "cqahfbfy5xee"
}
```

- Per i dettagli sull'API, consulta DescribeAccountAttributes AWS CLICommand Reference.

`describe-certificates`

Il seguente esempio di codice mostra come utilizzare`describe-certificates`.

AWS CLI

Come elencare i certificati disponibili

L'`describe-certificates`esempio seguente elenca i certificati disponibili nel tuo AWS account.

```
aws dms describe-certificates
```

Output:

```
{
    "Certificates": [
        {
            "CertificateIdentifier": "my-cert",
            "CertificateCreationDate": 1543259542.506,
            "CertificatePem": "-----BEGIN CERTIFICATE-----
\nMIID9DCCAtygAwIBAgIBQjANBgkqhkiG9w0BAQ ...U"

            ... remaining output omittted ...

        }
    ]
}
```

Per ulteriori informazioni, consulta Utilizzo di SSL nella Guida per l'utente di AWS Database Migration Service.

- Per i dettagli sull'API, consulta DescribeCertificates AWS CLICommand Reference.

`describe-connections`

Il seguente esempio di codice mostra come utilizzare`describe-connections`.

AWS CLI

Come descrivere le connessioni

L'esempio `describe-connections` seguente elenca le connessioni che sono state testate tra un'istanza di replica e un endpoint.

```
aws dms describe-connections
```

Output:

```
{
```

```
    "Connections": [
        {
            "Status": "successful",
            "ReplicationInstanceIdentifier": "test",
            "EndpointArn": "arn:aws:dms:us-east-arn:aws:dms:us-
east-1:123456789012:endpoint:ZW5UAN6P4E77EC7YWHK4RZZ3BE",
            "EndpointIdentifier": "testsrc1",
            "ReplicationInstanceArn": "arn:aws:dms:us-
east-1:123456789012:rep:6UTDJGBOUS3VI3SUWA66XFJCJQ"
        }
    ]
}
```

Per ulteriori informazioni, consulta Creazione di endpoint sorgente e obiettivo nella Guida per l'utente di AWS Database Migration Service.

- Per i dettagli sull'API, consulta DescribeConnections AWS CLICommand Reference.

`describe-endpoint-types`

Il seguente esempio di codice mostra come utilizzare`describe-endpoint-types`.

AWS CLI

Come elencare i tipi di endpoint disponibili

L'esempio `describe-endpoint-types` seguente elenca i tipi di endpoint MySQL disponibili.

```
aws dms describe-endpoint-types \
    --filters "Name=engine-name,Values=mysql"
```

Output:

```
{
    "SupportedEndpointTypes": [
        {
            "EngineName": "mysql",
            "SupportsCDC": true,
            "EndpointType": "source",
            "EngineDisplayName": "MySQL"
        },
        {
            "EngineName": "mysql",
```

```
            "SupportsCDC": true,
            "EndpointType": "target",
            "EngineDisplayName": "MySQL"
        }
    ]
}
```

Per ulteriori informazioni, vedere Working with AWS DMS Endpoints < https://docs.aws.amazon.com/dms/ latest/userguide/CHAP _endpoints.html>`__ nella Database Migration Service User Guide.AWS

- Per AWS CLI i dettagli sull'API, consulta Command DescribeEndpointTypesReference.

`describe-endpoints`

Il seguente esempio di codice mostra come utilizzare`describe-endpoints`.

AWS CLI

Come descrivere gli endpoint

L'`describe-endpoints`esempio seguente elenca gli endpoint del tuo AWS account.

```
aws dms describe-endpoints
```

Output:

```
{
    "Endpoints": [
        {
            "Username": "dms",
            "Status": "active",
            "EndpointArn": "arn:aws:dms:us-
east-1:123456789012:endpoint:SF2WOFLWYWKVEOHID2EKLP3SJI",
            "ServerName": "ec2-52-32-48-61.us-west-2.compute.amazonaws.com",
            "EndpointType": "SOURCE",
            "KmsKeyId": "arn:aws:kms:us-east-1:123456789012:key/94d5c4e7-4e4c-44be-
b58a-c8da7adf57cd",
            "DatabaseName": "test",
            "EngineName": "mysql",
            "EndpointIdentifier": "pri100",
            "Port": 8193
        },
```

```
        {
            "Username": "admin",
            "Status": "active",
            "EndpointArn": "arn:aws:dms:us-
east-1:123456789012:endpoint:TJJZCIH3CJ24TJRU4VC32WEWFR",
            "ServerName": "test.example.com",
            "EndpointType": "SOURCE",
            "KmsKeyId": "arn:aws:kms:us-east-1:123456789012:key/2431021b-1cf2-
a2d4-77b2-59a9e4bce323",
            "DatabaseName": "EMPL",
            "EngineName": "oracle",
            "EndpointIdentifier": "test",
            "Port": 1521
        }
    ]
}
```

Per ulteriori informazioni, vedere Working with AWS DMS Endpoints nella AWS Database Migration Service User Guide.

- Per i dettagli sull'API, consulta AWS CLI Command DescribeEndpointsReference.

`describe-event-categories`

Il seguente esempio di codice mostra come utilizzare`describe-event-categories`.

AWS CLI

Come descrivere le categorie di eventi

L'esempio `describe-event-categories` seguente elenca le categorie di eventi disponibili.

```
aws dms describe-event-categories
```

Output:

```
{
    "EventCategoryGroupList": [
        {
            "SourceType": "replication-instance",
            "EventCategories": [
                "low storage",
                "configuration change",
```

```
                "maintenance",
                "deletion",
                "creation",
                "failover",
                "failure"
            ]
        },
        {
            "SourceType": "replication-task",
            "EventCategories": [
                "configuration change",
                "state change",
                "deletion",
                "creation",
                "failure"
            ]
        }
    ]
}
```

Per ulteriori informazioni, consulta Utilizzo di eventi e notifiche nella Guida per l'utente di AWS Database Migration Service.

- Per i dettagli sull'API, consulta DescribeEventCategories AWS CLICommand Reference.

`describe-event-subscriptions`

Il seguente esempio di codice mostra come utilizzare`describe-event-subscriptions`.

AWS CLI

Come descrivere le sottoscrizioni a eventi

L'esempio `describe-event-subscriptions` seguente elenca gli abbonamenti agli eventi di un argomento Amazon SNS.

```
aws dms describe-event-subscriptions
```

Output:

```
{
    "EventSubscriptionsList": [
        {
```

```
            "CustomerAwsId": "123456789012",
            "CustSubscriptionId": "my-dms-events",
            "SnsTopicArn": "arn:aws:sns:us-east-1:123456789012:my-sns-topic",
            "Status": "deleting",
            "SubscriptionCreationTime": "2020-05-21 22:28:51.924",
            "Enabled": true
        }
    ]
}
```

Per ulteriori informazioni, consulta Utilizzo di eventi e notifiche nella Guida per l'utente di AWS Database Migration Service.

- Per i dettagli sull'API, consulta DescribeEventSubscriptions AWS CLICommand Reference.

`describe-events`

Il seguente esempio di codice mostra come utilizzare`describe-events`.

AWS CLI

Come elencare gli eventi DMS

L'esempio `describe-events` seguente elenca gli eventi che hanno avuto origine da un'istanza di replica.

```
aws dms describe-events \
    --source-type "replication-instance"
```

Output:

```
{
    "Events": [
        {
            "SourceIdentifier": "my-repl-instance",
            "SourceType": "replication-instance",
            "Message": "Replication application shutdown",
            "EventCategories": [],
            "Date": 1590771645.776
        }
    ]
}
```

Per ulteriori informazioni, consulta Utilizzo di eventi e notifiche nella Guida per l'utente di AWS Database Migration Service.

- Per i dettagli sull'API, consulta DescribeEvents AWS CLICommand Reference.

`describe-orderable-replication-instances`

Il seguente esempio di codice mostra come utilizzare`describe-orderable-replication-instances`.

AWS CLI

Come descrivere le istanze di replica ordinabili

L'esempio `describe-orderable-replication-instances` seguente elenca i tipi di istanze di replica che è possibile ordinare.

```
aws dms describe-orderable-replication-instances
```

Output:

```
{
    "OrderableReplicationInstances": [
        {
            "EngineVersion": "3.3.2",
            "ReplicationInstanceClass": "dms.c4.2xlarge",
            "StorageType": "gp2",
            "MinAllocatedStorage": 5,
            "MaxAllocatedStorage": 6144,
            "DefaultAllocatedStorage": 100,
            "IncludedAllocatedStorage": 100,
            "AvailabilityZones": [
                "us-east-1a",
                "us-east-1b",
                "us-east-1c",
                "us-east-1d",
                "us-east-1e",
                "us-east-1f"
            ]
        },
        {
            "EngineVersion": "3.3.2",
            "ReplicationInstanceClass": "dms.c4.4xlarge",
```

```
            "StorageType": "gp2",
            "MinAllocatedStorage": 5,
            "MaxAllocatedStorage": 6144,
            "DefaultAllocatedStorage": 100,
            "IncludedAllocatedStorage": 100,
            "AvailabilityZones": [
                "us-east-1a",
                "us-east-1b",
                "us-east-1c",
                "us-east-1d",
                "us-east-1e",
                "us-east-1f"
            ]
        },

        ...remaining output omitted...

    }
```

Per ulteriori informazioni, vedere Working with an AWS DMS Replication Instance nella AWS Database Migration Service User Guide.

- Per i dettagli sull'API, consulta AWS CLI Command DescribeOrderableReplicationInstancesReference.

`describe-refresh-schemas-status`

Il seguente esempio di codice mostra come utilizzare`describe-refresh-schemas-status`.

AWS CLI

Come elencare lo stato di aggiornamento di un endpoint

L'esempio `describe-refresh-schemas-status` seguente restituisce lo stato di una precedente richiesta di aggiornamento.

```
aws dms describe-refresh-schemas-status \
    --endpoint-arn arn:aws:dms:us-
east-1:123456789012:endpoint:6GGI6YPWWGAYUVLKIB732KEVWA
```

Output:

```
{
```

```
    "RefreshSchemasStatus": {
        "EndpointArn": "arn:aws:dms:us-
east-1:123456789012:endpoint:6GGI6YPWWGAYUVLKIB732KEVWA",
        "ReplicationInstanceArn": "arn:aws:dms:us-
east-1:123456789012:rep:T3OM7OUB5NM2LCVZF7JPGJRNUE",
        "Status": "successful",
        "LastRefreshDate": 1590786544.605
    }
}
```

- Per i dettagli sull'API, consulta DescribeRefreshSchemasStatus AWS CLICommand Reference.

`describe-replication-instances`

Il seguente esempio di codice mostra come utilizzare`describe-replication-instances`.

AWS CLI

Come descrivere le istanze di replica

L'`describe-replication-instances`esempio seguente elenca le istanze di replica presenti nell'account AWS .

```
aws dms describe-replication-instances
```

Output:

```
{
    "ReplicationInstances": [
        {
            "ReplicationInstanceIdentifier": "my-repl-instance",
            "ReplicationInstanceClass": "dms.t2.micro",
            "ReplicationInstanceStatus": "available",
            "AllocatedStorage": 5,
            "InstanceCreateTime": 1590011235.952,
            "VpcSecurityGroups": [
                {
                    "VpcSecurityGroupId": "sg-f839b688",
                    "Status": "active"
                }
            ],
            "AvailabilityZone": "us-east-1e",
```

```
            "ReplicationSubnetGroup": {
                "ReplicationSubnetGroupIdentifier": "default",
                "ReplicationSubnetGroupDescription": "default",
                "VpcId": "vpc-136a4c6a",
                "SubnetGroupStatus": "Complete",
                "Subnets": [
                    {
                        "SubnetIdentifier": "subnet-da327bf6",
                        "SubnetAvailabilityZone": {
                            "Name": "us-east-1a"
                        },
                        "SubnetStatus": "Active"
                    },
                    {
                        "SubnetIdentifier": "subnet-42599426",
                        "SubnetAvailabilityZone": {
                            "Name": "us-east-1d"
                        },
                        "SubnetStatus": "Active"
                    },
                    {
                        "SubnetIdentifier": "subnet-bac383e0",
                        "SubnetAvailabilityZone": {
                            "Name": "us-east-1c"
                        },
                        "SubnetStatus": "Active"
                    },
                    {
                        "SubnetIdentifier": "subnet-6746046b",
                        "SubnetAvailabilityZone": {
                            "Name": "us-east-1f"
                        },
                        "SubnetStatus": "Active"
                    },
                    {
                        "SubnetIdentifier": "subnet-d7c825e8",
                        "SubnetAvailabilityZone": {
                            "Name": "us-east-1e"
                        },
                        "SubnetStatus": "Active"
                    },
                    {
                        "SubnetIdentifier": "subnet-cbfff283",
                        "SubnetAvailabilityZone": {
```

```
                            "Name": "us-east-1b"
                        },
                        "SubnetStatus": "Active"
                    }
                ]
            },
            "PreferredMaintenanceWindow": "wed:11:42-wed:12:12",
            "PendingModifiedValues": {
                "MultiAZ": true
            },
            "MultiAZ": false,
            "EngineVersion": "3.3.2",
            "AutoMinorVersionUpgrade": true,
            "KmsKeyId": "arn:aws:kms:us-east-1:123456789012:key/
f7bc0f8e-1a3a-4ace-9faa-e8494fa3921a",
            "ReplicationInstanceArn": "arn:aws:dms:us-
east-1:123456789012:rep:T3OM7OUB5NM2LCVZF7JPGJRNUE",
            "ReplicationInstancePublicIpAddress": "3.230.18.248",
            "ReplicationInstancePrivateIpAddress": "172.31.75.90",
            "ReplicationInstancePublicIpAddresses": [
                "3.230.18.248"
            ],
            "ReplicationInstancePrivateIpAddresses": [
                "172.31.75.90"
            ],
            "PubliclyAccessible": true,
            "FreeUntil": 1590194829.267
        }
    ]
}
```


Per ulteriori informazioni, vedere Working with an AWS DMS Replication Instance nella AWS Database Migration Service User Guide.

- Per i dettagli sull'API, consulta AWS CLI Command DescribeReplicationInstancesReference.

`describe-replication-subnet-groups`

Il seguente esempio di codice mostra come utilizzare`describe-replication-subnet-groups`.

AWS CLI

Come visualizzare i gruppi di sottoreti disponibili

L'esempio `describe-replication-subnet-groups` seguente elenca i gruppi di sottoreti disponibili.

```
aws dms describe-replication-subnet-groups \
    --filter "Name=replication-subnet-group-id,Values=my-subnet-group"
```

Output:

```
{
    "ReplicationSubnetGroups": [
        {
            "ReplicationSubnetGroupIdentifier": "my-subnet-group",
            "ReplicationSubnetGroupDescription": "my subnet group",
            "VpcId": "vpc-136a4c6a",
            "SubnetGroupStatus": "Complete",
            "Subnets": [
                {
                    "SubnetIdentifier": "subnet-da327bf6",
                    "SubnetAvailabilityZone": {
                        "Name": "us-east-1a"
                    },
                    "SubnetStatus": "Active"
                },
                {
                    "SubnetIdentifier": "subnet-bac383e0",
                    "SubnetAvailabilityZone": {
                        "Name": "us-east-1c"
                    },
                    "SubnetStatus": "Active"
                },
                {
                    "SubnetIdentifier": "subnet-d7c825e8",
                    "SubnetAvailabilityZone": {
                        "Name": "us-east-1e"
                    },
                    "SubnetStatus": "Active"
                }
            ]
        }
    ]
}
```

Per ulteriori informazioni, consulta Impostazione di una rete per un'istanza di replica nella Guida per l'utente di AWS Database Migration Service.

- Per i dettagli sull'API, consulta DescribeReplicationSubnetGroups AWS CLICommand Reference.

`describe-replication-task-assessment-results`

Il seguente esempio di codice mostra come utilizzare`describe-replication-task-assessment-results`.

AWS CLI

Come elencare i risultati delle valutazioni delle attività di replica

L'esempio `describe-replication-task-assessment-results` seguente elenca i risultati di una precedente valutazione delle attività.

```
aws dms describe-replication-task-assessment-results
```

Output:

```
{
    "ReplicationTaskAssessmentResults": [
        {
            "ReplicationTaskIdentifier": "moveit2",
            "ReplicationTaskArn": "arn:aws:dms:us-
east-1:123456789012:task:K55IUCGBASJS5VHZJIINA45FII",
            "ReplicationTaskLastAssessmentDate": 1590790230.0,
            "AssessmentStatus": "No issues found",
            "AssessmentResultsFile": "moveit2/2020-05-29-22-10"
        }
    ]
}
```

Per ulteriori informazioni, consulta Creazione di un report di valutazione del task nella Guida per l'utente di AWS Database Migration Service.

- Per i dettagli sull'API, consulta DescribeReplicationTaskAssessmentResults AWS CLICommand Reference.

describe-replication-tasks

Il seguente esempio di codice mostra come utilizzaredescribe-replication-tasks.

AWS CLI

Come descrivere un'attività di replica

L'esempio describe-replication-tasks seguente descrive le attività di replica correnti.

```
aws dms describe-replication-tasks
```

Output:

```
{
    "ReplicationTasks": [
        {
            "ReplicationTaskIdentifier": "moveit2",
            "SourceEndpointArn": "arn:aws:dms:us-
east-1:123456789012:endpoint:6GGI6YPWWGAYUVLKIB732KEVWA",
            "TargetEndpointArn": "arn:aws:dms:us-
east-1:123456789012:endpoint:EOM4SFKCZEYHZBFGAGZT3QEC5U",
            "ReplicationInstanceArn": "arn:aws:dms:us-
east-1:123456789012:rep:T3OM7OUB5NM2LCVZF7JPGJRNUE",
            "MigrationType": "full-load",
            "TableMappings": ...output omitted... ,
            "ReplicationTaskSettings": ...output omitted... ,
            "Status": "stopped",
            "StopReason": "Stop Reason FULL_LOAD_ONLY_FINISHED",
            "ReplicationTaskCreationDate": 1590524772.505,
            "ReplicationTaskStartDate": 1590619805.212,
            "ReplicationTaskArn": "arn:aws:dms:us-
east-1:123456789012:task:K55IUCGBASJS5VHZJIINA45FII",
            "ReplicationTaskStats": {
                "FullLoadProgressPercent": 100,
                "ElapsedTimeMillis": 0,
                "TablesLoaded": 0,
                "TablesLoading": 0,
                "TablesQueued": 0,
                "TablesErrored": 0,
                "FreshStartDate": 1590619811.528,
                "StartDate": 1590619811.528,
                "StopDate": 1590619842.068
```

```
            }
        }
    ]
}
```

Per ulteriori informazioni, vedere Working with AWS DMS Tasks nella AWS Database Migration Service User Guide.

- Per i dettagli sull'API, consulta DescribeReplicationTasks AWS CLICommand Reference.

`describe-schemas`

Il seguente esempio di codice mostra come utilizzare`describe-schemas`.

AWS CLI

Come descrivere gli schemi di database

L'esempio `describe-schemas` seguente elenca le tabelle disponibili in un endpoint.

```
aws dms describe-schemas \
    --endpoint-arn "arn:aws:dms:us-
east-1:123456789012:endpoint:6GGI6YPWWGAYUVLKIB732KEVWA"
```

Output:

```
{
    "Schemas": [
        "prodrep"
    ]
}
```

Per ulteriori informazioni, consulta il seguente argomento nella Guida per l'utente di AWS Database Migration Service.

- Per i dettagli sull'API, consulta DescribeSchemas AWS CLICommand Reference.

`list-tags-for-resource`

Il seguente esempio di codice mostra come utilizzare`list-tags-for-resource`.

AWS CLI

Come elencare i tag per una risorsa

L'esempio `list-tags-for-resource` seguente elenca tutti i tag associati a un'istanza di replica.

```
aws dms list-tags-for-resource \
    --resource-arn arn:aws:dms:us-east-1:123456789012:rep:T3OM7OUB5NM2LCVZF7JPGJRNUE
```

Output:

```
{
    "TagList": [
        {
            "Key": "Project",
            "Value": "dbMigration"
        },
        {
            "Key": "Environment",
            "Value": "PROD"
        }
    ]
}
```

Per ulteriori informazioni, consulta Assegnazione di tag alle risorse nella Guida per l'utente di AWS Database Migration Service.

- Per i dettagli sull'API, consulta ListTagsForResource AWS CLICommand Reference.

`modify-endpoint`

Il seguente esempio di codice mostra come utilizzare`modify-endpoint`.

AWS CLI

Come modificare un endpoint

L'esempio `modify-endpoint` seguente aggiunge un attributo di connessione aggiuntivo a un endpoint.

```
aws dms modify-endpoint \
```

```
    --endpoint-arn "arn:aws:dms:us-
east-1:123456789012:endpoint:GUVAFG34EECUOJ6QVZ56DAHT3U" \
    --extra-connection-attributes "compressionType=GZIP"
```

Output:

```
{
    "Endpoint": {
        "EndpointIdentifier": "src-endpoint",
        "EndpointType": "SOURCE",
        "EngineName": "s3",
        "EngineDisplayName": "Amazon S3",
        "ExtraConnectionAttributes":
 "compressionType=GZIP;csvDelimiter=,;csvRowDelimiter=\\n;",
        "Status": "active",
        "EndpointArn": "arn:aws:dms:us-
east-1:123456789012:endpoint:GUVAFG34EECUOJ6QVZ56DAHT3U",
        "SslMode": "none",
        "ServiceAccessRoleArn": "arn:aws:iam::123456789012:role/my-s3-access-role",
        "S3Settings": {
            "ServiceAccessRoleArn": "arn:aws:iam::123456789012:role/my-s3-access-
role",
            "CsvRowDelimiter": "\\n",
            "CsvDelimiter": ",",
            "BucketFolder": "",
            "BucketName": "",
            "CompressionType": "GZIP",
            "EnableStatistics": true
        }
    }
}
```

Per ulteriori informazioni, vedere Working with AWS DMS Endpoints < https://docs.aws.amazon.com/dms/ latest/userguide/CHAP _endpoints.html>`__ nella Database Migration Service User Guide.AWS

- Per AWS CLI i dettagli sull'API, consulta Command ModifyEndpointReference.

`modify-event-subscription`

Il seguente esempio di codice mostra come utilizzare`modify-event-subscription`.

AWS CLI

Come modificare un abbonamento all'evento

L'esempio `modify-event-subscription` seguente modifica il tipo di origine di una sottoscrizione a un evento.

```
aws dms modify-event-subscription \
    --subscription-name "my-dms-events" \
    --source-type replication-task
```

Output:

```
{
    "EventSubscription": {
        "CustomerAwsId": "123456789012",
        "CustSubscriptionId": "my-dms-events",
        "SnsTopicArn": "arn:aws:sns:us-east-1:123456789012:my-sns-topic",
        "Status": "modifying",
        "SubscriptionCreationTime": "2020-05-29 17:04:40.262",
        "SourceType": "replication-task",
        "Enabled": true
    }
}
```

Per ulteriori informazioni, consulta Utilizzo di eventi e notifiche nella Guida per l'utente di AWS Database Migration Service.

- Per i dettagli sull'API, consulta ModifyEventSubscription AWS CLICommand Reference.

`modify-replication-instance`

Il seguente esempio di codice mostra come utilizzare`modify-replication-instance`.

AWS CLI

Come modificare un'istanza di replica

L'esempio `modify-replication-instance` seguente modifica un'istanza di replica in modo che utilizzi un'implementazione Multi-AZ.

```
aws dms modify-replication-instance \
```

```
    --replication-instance-arn arn:aws:dms:us-
east-1:123456789012:rep:T3OM7OUB5NM2LCVZF7JPGJRNUE \
    --multi-az
```

Output:

```
{
    "ReplicationInstance": {
        "ReplicationInstanceIdentifier": "my-repl-instance",
        "ReplicationInstanceClass": "dms.t2.micro",
        "ReplicationInstanceStatus": "available",
        "AllocatedStorage": 5,
        "InstanceCreateTime": 1590011235.952,

        ...output omitted...

        "PendingModifiedValues": {
            "MultiAZ": true
        },
        "MultiAZ": false,
        "EngineVersion": "3.3.2",
        "AutoMinorVersionUpgrade": true,
        "KmsKeyId": "arn:aws:kms:us-east-1:123456789012:key/f7bc0f8e-1a3a-4ace-9faa-
e8494fa3921a",

        ...output omitted...

    }
}
```

Per ulteriori informazioni, vedere Working with an AWS DMS Replication Instance nella AWS Database Migration Service User Guide.

- Per i dettagli sull'API, consulta AWS CLI Command ModifyReplicationInstanceReference.

`modify-replication-subnet-group`

Il seguente esempio di codice mostra come utilizzare`modify-replication-subnet-group`.

AWS CLI

Come modificare un gruppo di sottoreti

L'esempio `modify-replication-subnet-group` seguente modifica gli elenchi delle sottoreti associate a un gruppo di sottoreti.

```
aws dms modify-replication-subnet-group \
    --replication-subnet-group-identifier my-subnet-group \
    --subnet-id subnet-da327bf6 subnet-bac383e0
```

Output:

```
{
    "ReplicationSubnetGroup": {
        "ReplicationSubnetGroupIdentifier": "my-subnet-group",
        "ReplicationSubnetGroupDescription": "my subnet group",
        "VpcId": "vpc-136a4c6a",
        "SubnetGroupStatus": "Complete",
        "Subnets": [
            {
                "SubnetIdentifier": "subnet-da327bf6",
                "SubnetAvailabilityZone": {
                    "Name": "us-east-1a"
                },
                "SubnetStatus": "Active"
            },
            {
                "SubnetIdentifier": "subnet-bac383e0",
                "SubnetAvailabilityZone": {
                    "Name": "us-east-1c"
                },
                "SubnetStatus": "Active"
            }
        ]
    }
}
```

Per ulteriori informazioni, consulta Impostazione di una rete per un'istanza di replica nella Guida per l'utente di AWS Database Migration Service.

- Per i dettagli sull'API, consulta ModifyReplicationSubnetGroup AWS CLICommand Reference.

`modify-replication-task`

Il seguente esempio di codice mostra come utilizzare`modify-replication-task`.

AWS CLI

Come modificare un'attività di replica

L'esempio `modify-replication-task` seguente modifica i mapping delle tabelle per un'attività.

```
aws dms modify-replication-task \
    --replication-task-arn "arn:aws:dms:us-
east-1:123456789012:task:K55IUCGBASJS5VHZJIINA45FII" \
    --table-mappings file://table-mappings.json
```

Contenuto di `table-mappings.json`:

```
{
    "rules": [
        {
            "rule-type": "selection",
            "rule-id": "1",
            "rule-name": "1",
            "object-locator": {
                "schema-name": "prodrep",
                "table-name": "ACCT_%"
            },
            "rule-action": "include",
            "filters": []
        }
    ]
}
```

Output:

```
{
    "ReplicationTask": {
        "ReplicationTaskIdentifier": "moveit2",
        "SourceEndpointArn": "arn:aws:dms:us-
east-1:123456789012:endpoint:6GGI6YPWWGAYUVLKIB732KEVWA",
        "TargetEndpointArn": "arn:aws:dms:us-
east-1:123456789012:endpoint:EOM4SFKCZEYHZBFGAGZT3QEC5U",
        "ReplicationInstanceArn": "arn:aws:dms:us-
east-1:123456789012:rep:T3OM7OUB5NM2LCVZF7JPGJRNUE",
        "MigrationType": "full-load",
```

```
        "TableMappings": ...output omitted...,
        "ReplicationTaskSettings": ...output omitted...,
        "Status": "modifying",
        "StopReason": "Stop Reason FULL_LOAD_ONLY_FINISHED",
        "ReplicationTaskCreationDate": 1590524772.505,
        "ReplicationTaskStartDate": 1590789424.653,
        "ReplicationTaskArn": "arn:aws:dms:us-
east-1:123456789012:task:K55IUCGBASJS5VHZJIINA45FII"
    }
}
```

Per ulteriori informazioni, vedere Working with AWS DMS Tasks nella AWS Database Migration Service User Guide.

- Per i dettagli sull'API, consulta ModifyReplicationTask AWS CLICommand Reference.

`reboot-replication-instance`

Il seguente esempio di codice mostra come utilizzare`reboot-replication-instance`.

AWS CLI

Come riavviare un'istanza di replica

L'esempio `reboot-replication-instance` seguente riavvia un'istanza di replica.

```
aws dms reboot-replication-instance \
    --replication-instance-arn arn:aws:dms:us-
east-1:123456789012:rep:T3OM7OUB5NM2LCVZF7JPGJRNUE
```

Output:

```
{
    "ReplicationInstance": {
        "ReplicationInstanceIdentifier": "my-repl-instance",
        "ReplicationInstanceClass": "dms.t2.micro",
        "ReplicationInstanceStatus": "rebooting",
        "AllocatedStorage": 5,
        "InstanceCreateTime": 1590011235.952,
    ... output omitted ...
    }
}
```

Per ulteriori informazioni, vedere Working with an AWS DMS Replication Instance nella AWS Database Migration Service User Guide.

- Per i dettagli sull'API, consulta AWS CLI Command RebootReplicationInstanceReference.

`refresh-schemas`

Il seguente esempio di codice mostra come utilizzare`refresh-schemas`.

AWS CLI

Come aggiornare gli schemi di database

L'`refresh-schemas`esempio seguente richiede che AWS DMS aggiorni l'elenco degli schemi su un endpoint.

```
aws dms refresh-schemas \
    --replication-instance-arn arn:aws:dms:us-
east-1:123456789012:rep:T3OM7OUB5NM2LCVZF7JPGJRNUE \
    --endpoint-arn "arn:aws:dms:us-
east-1:123456789012:endpoint:6GGI6YPWWGAYUVLKIB732KEVWA"
```

Output:

```
{
    "RefreshSchemasStatus": {
        "EndpointArn": "arn:aws:dms:us-
east-1:123456789012:endpoint:6GGI6YPWWGAYUVLKIB732KEVWA",
        "ReplicationInstanceArn": "arn:aws:dms:us-
east-1:123456789012:rep:T3OM7OUB5NM2LCVZF7JPGJRNUE",
        "Status": "refreshing",
        "LastRefreshDate": 1590019949.103
    }
}
```

- Per i dettagli sull'API, vedere RefreshSchemasin Command Reference.AWS CLI

`reload-tables`

Il seguente esempio di codice mostra come utilizzare`reload-tables`.

AWS CLI

Come aggiornare l'elenco delle tabelle disponibili in un endpoint

L'esempio `reload-tables` seguente ricarica l'elenco delle tabelle disponibili in un endpoint.

```
aws dms reload-tables \
    --replication-task-arn "arn:aws:dms:us-
east-1:123456789012:task:K55IUCGBASJS5VHZJIINA45FII" \
    --tables-to-reload "SchemaName=prodrep,TableName=ACCT_BAL"
```

Output:

```
{
    "ReplicationTaskArn": "arn:aws:dms:us-
east-1:123456789012:task:K55IUCGBASJS5VHZJIINA45FII"
}
```

- Per i dettagli sull'API, consulta ReloadTables AWS CLICommand Reference.

`remove-tags-from-resource`

Il seguente esempio di codice mostra come utilizzare`remove-tags-from-resource`.

AWS CLI

Come rimuovere i tag da un'istanza di replica

L'esempio `remove-tags-from-resource` seguente elimina i tag da un'istanza di replica.

```
aws dms remove-tags-from-resource \
    --resource-arn arn:aws:dms:us-east-1:123456789012:rep:T3OM7OUB5NM2LCVZF7JPGJRNUE
 \
    --tag-keys Environment Project
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Assegnazione di tag alle risorse nella Guida per l'utente di AWS Database Migration Service.

- Per i dettagli sull'API, consulta RemoveTagsFromResource AWS CLICommand Reference.

`start-replication-task-assessment`

Il seguente esempio di codice mostra come utilizzare`start-replication-task-assessment`.

AWS CLI

Come avviare una valutazione delle attività

L'esempio `start-replication-task-assessment` seguente avvia una valutazione dell'attività di replica.

```
aws dms start-replication-task-assessment \
    --replication-task-arn arn:aws:dms:us-
east-1:123456789012:task:K55IUCGBASJS5VHZJIINA45FII
```

Output:

```
{
    "ReplicationTask": {
        "ReplicationTaskIdentifier": "moveit2",
        "SourceEndpointArn": "arn:aws:dms:us-
east-1:123456789012:endpoint:6GGI6YPWWGAYUVLKIB732KEVWA",
        "TargetEndpointArn": "arn:aws:dms:us-
east-1:123456789012:endpoint:EOM4SFKCZEYHZBFGAGZT3QEC5U",
        "ReplicationInstanceArn": "arn:aws:dms:us-
east-1:123456789012:rep:T3OM7OUB5NM2LCVZF7JPGJRNUE",
        "MigrationType": "full-load",
        "TableMappings": ...output omitted...,
        "ReplicationTaskSettings": ...output omitted...,
        "Status": "testing",
        "StopReason": "Stop Reason FULL_LOAD_ONLY_FINISHED",
        "ReplicationTaskCreationDate": 1590524772.505,
        "ReplicationTaskStartDate": 1590789988.677,
        "ReplicationTaskArn": "arn:aws:dms:us-
east-1:123456789012:task:K55IUCGBASJS5VHZJIINA45FII"
    }
}
```

Per ulteriori informazioni, consulta Creazione di un report di valutazione del task nella Guida per l'utente di AWS Database Migration Service.

- Per i dettagli sull'API, consulta StartReplicationTaskAssessment AWS CLICommand Reference.

`start-replication-task`

Il seguente esempio di codice mostra come utilizzare`start-replication-task`.

AWS CLI

Come avviare un'attività di replica

L'`command-name`esempio seguente elenca i widget disponibili nel tuo AWS account.

```
aws dms start-replication-task \
    --replication-task-arn arn:aws:dms:us-
east-1:123456789012:task:K55IUCGBASJS5VHZJIINA45FII \
    --start-replication-task-type reload-target
```

Output:

```
{
    "ReplicationTask": {
        "ReplicationTaskIdentifier": "moveit2",
        "SourceEndpointArn": "arn:aws:dms:us-
east-1:123456789012:endpoint:6GGI6YPWWGAYUVLKIB732KEVWA",
        "TargetEndpointArn": "arn:aws:dms:us-
east-1:123456789012:endpoint:EOM4SFKCZEYHZBFGAGZT3QEC5U",
        "ReplicationInstanceArn": "arn:aws:dms:us-
east-1:123456789012:rep:T3OM7OUB5NM2LCVZF7JPGJRNUE",
        "MigrationType": "full-load",
        "TableMappings": ...output omitted... ,
        "ReplicationTaskSettings": ...output omitted... ,
        "Status": "starting",
        "ReplicationTaskCreationDate": 1590524772.505,
        "ReplicationTaskStartDate": 1590619805.212,
        "ReplicationTaskArn": "arn:aws:dms:us-
east-1:123456789012:task:K55IUCGBASJS5VHZJIINA45FII"
    }
}
```

Per ulteriori informazioni, vedere Working with AWS DMS Tasks nella AWS Database Migration Service User Guide.

- Per i dettagli sull'API, consulta StartReplicationTask AWS CLICommand Reference.

`stop-replication-task`

Il seguente esempio di codice mostra come utilizzare`stop-replication-task`.

AWS CLI

Come arrestare un'attività

L'esempio `stop-replication-task` seguente arresta un'attività.

```
aws dms stop-replication-task \
    --replication-task-arn arn:aws:dms:us-
east-1:123456789012:task:K55IUCGBASJS5VHZJIINA45FII
```

Output:

```
{
    "ReplicationTask": {
        "ReplicationTaskIdentifier": "moveit2",
        "SourceEndpointArn": "arn:aws:dms:us-
east-1:123456789012:endpoint:6GGI6YPWWGAYUVLKIB732KEVWA",
        "TargetEndpointArn": "arn:aws:dms:us-
east-1:123456789012:endpoint:EOM4SFKCZEYHZBFGAGZT3QEC5U",
        "ReplicationInstanceArn": "arn:aws:dms:us-
east-1:123456789012:rep:T3OM7OUB5NM2LCVZF7JPGJRNUE",
        "MigrationType": "full-load",
        "TableMappings": ...output omitted...,
        "ReplicationTaskSettings": ...output omitted...,
        "Status": "stopping",
        "ReplicationTaskCreationDate": 1590524772.505,
        "ReplicationTaskStartDate": 1590789424.653,
        "ReplicationTaskArn": "arn:aws:dms:us-
east-1:123456789012:task:K55IUCGBASJS5VHZJIINA45FII"
    }
}
```

Per ulteriori informazioni, vedere Working with AWS DMS Tasks nella AWS Database Migration Service User Guide.

- Per i dettagli sull'API, consulta StopReplicationTask AWS CLICommand Reference.

`test-connection`

Il seguente esempio di codice mostra come utilizzare`test-connection`.

AWS CLI

Come testare una connessione a un endpoint

L'esempio `test-connection` seguente verifica se è possibile accedere a un endpoint da un'istanza di replica.

```
aws dms test-connection \
    --replication-instance-arn arn:aws:dms:us-
east-1:123456789012:rep:T3OM7OUB5NM2LCVZF7JPGJRNUE \
    --endpoint-arn arn:aws:dms:us-
east-1:123456789012:endpoint:6GGI6YPWWGAYUVLKIB732KEVWA
```

Output:

```
{
    "Connection": {
        "ReplicationInstanceArn": "arn:aws:dms:us-
east-1:123456789012:rep:T3OM7OUB5NM2LCVZF7JPGJRNUE",
        "EndpointArn": "arn:aws:dms:us-
east-1:123456789012:endpoint:6GGI6YPWWGAYUVLKIB732KEVWA",
        "Status": "testing",
        "EndpointIdentifier": "src-database-1",
        "ReplicationInstanceIdentifier": "my-repl-instance"
    }
}
```

Per ulteriori informazioni, consulta Creazione di endpoint sorgente e obiettivo nella Guida per l'utente di AWS Database Migration Service.

- Per i dettagli sull'API, consulta TestConnection AWS CLICommand Reference.

Esempi di utilizzo di Amazon DocumentDB AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando Amazon DocumentDB. AWS Command Line Interface

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`add-tags-to-resource`

Il seguente esempio di codice mostra come usare`add-tags-to-resource`.

AWS CLI

Come aggiungere uno o più tag a una risorsa specificata

L'esempio `add-tags-to-resource` seguente aggiunge tre tag a `sample-cluster`. Un tag (`CropB`) ha un nome di chiave ma nessun valore.

```
aws docdb add-tags-to-resource \
    --resource-name arn:aws:rds:us-west-2:123456789012:cluster:sample-cluster \
    --tags Key="CropA",Value="Apple" Key="CropB" Key="CropC",Value="Corn"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Tagging delle risorse Amazon DocumentDB nella Guida per gli sviluppatori di Amazon DocumentDB.

- Per i dettagli sull'API, consulta AddTagsToResource AWS CLICommand Reference.

`apply-pending-maintenance-action`

Il seguente esempio di codice mostra come utilizzare`apply-pending-maintenance-action`.

AWS CLI

Come fare in modo che le operazioni di manutenzione in sospeso vengano eseguite durante la finestra di manutenzione successiva

L'esempio `apply-pending-maintenance-action` seguente fa in modo che tutte le operazioni di aggiornamento del sistema vengano eseguite durante la successiva finestra di manutenzione pianificata.

```
aws docdb apply-pending-maintenance-action \
--resource-identifier arn:aws:rds:us-west-2:123456789012:cluster:sample-cluster \
--apply-action system-update \
--opt-in-type next-maintenance
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Applicazione degli aggiornamenti di Amazon DocumentDB nella Guida per gli sviluppatori di Amazon DocumentDB.

- Per i dettagli sull'API, consulta ApplyPendingMaintenanceAction AWS CLICommand Reference.

`copy-db-cluster-parameter-group`

Il seguente esempio di codice mostra come utilizzare`copy-db-cluster-parameter-group`.

AWS CLI

Come duplicare un gruppo di parametri di un cluster di database esistente

L'esempio `copy-db-cluster-parameter-group` seguente crea una copia del gruppo di parametri `custom-docdb3-6` denominata `custom-docdb3-6-copy`. Quando crea la copia, aggiunge tag al nuovo gruppo di parametri.

```
aws docdb copy-db-cluster-parameter-group \
    --source-db-cluster-parameter-group-identifier custom-docdb3-6 \
    --target-db-cluster-parameter-group-identifier custom-docdb3-6-copy \
    --target-db-cluster-parameter-group-description "Copy of custom-docdb3-6" \
    --tags Key="CopyNumber",Value="1" Key="Modifiable",Value="Yes"
```

Output:

```
{
    "DBClusterParameterGroup": {
        "DBParameterGroupFamily": "docdb3.6",
        "DBClusterParameterGroupArn": "arn:aws:rds:us-east-1:12345678901:cluster-
pg:custom-docdb3-6-copy",
```

```
        "DBClusterParameterGroupName": "custom-docdb3-6-copy",
        "Description": "Copy of custom-docdb3-6"
    }
}
```

Per ulteriori informazioni, consulta Copia di un gruppo di parametri del cluster Amazon DocumentDB nella Guida per gli sviluppatori di Amazon DocumentDB.

- Per i dettagli sull'API, consulta CopyDbClusterParameterGroup AWS CLICommand Reference.

`copy-db-cluster-snapshot`

Il seguente esempio di codice mostra come utilizzare`copy-db-cluster-snapshot`.

AWS CLI

Come creare una copia di uno snapshot

L'esempio seguente dell'`copy-db-cluster-snapshot` copia `sample-cluster-snapshot` e assegna il nome `sample-cluster-snapshot-copy`. La copia contiene tutti i tag dell'originale più un nuovo tag con il nome della chiave `CopyNumber`.

```
aws docdb copy-db-cluster-snapshot \
    --source-db-cluster-snapshot-identifier sample-cluster-snapshot \
    --target-db-cluster-snapshot-identifier sample-cluster-snapshot-copy \
    --copy-tags \
    --tags Key="CopyNumber",Value="1"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Copia di una snapshot del cluster nella Guida per gli sviluppatori di Amazon DocumentDB.

- Per i dettagli sull'API, consulta CopyDbClusterSnapshot AWS CLICommand Reference.

`create-db-cluster-parameter-group`

Il seguente esempio di codice mostra come utilizzare`create-db-cluster-parameter-group`.

AWS CLI

Come creare un gruppo di parametri di un cluster Amazon DocumentDB

L'esempio `create-db-cluster-parameter-group` seguente crea il gruppo di parametri del cluster di database `sample-parameter-group` utilizzando la famiglia `docdb3.6`.

```
aws docdb create-db-cluster-parameter-group \
    --db-cluster-parameter-group-name sample-parameter-group \
    --db-parameter-group-family docdb3.6 \
    --description "Sample parameter group based on docdb3.6"
```

Output:

```
{
    "DBClusterParameterGroup": {
        "Description": "Sample parameter group based on docdb3.6",
        "DBParameterGroupFamily": "docdb3.6",
        "DBClusterParameterGroupArn": "arn:aws:rds:us-west-2:123456789012:cluster-
pg:sample-parameter-group",
        "DBClusterParameterGroupName": "sample-parameter-group"
    }
}
```

Per ulteriori informazioni, consulta Creazione di un gruppo di parametri del cluster Amazon DocumentDB nella Guida per gli sviluppatori di Amazon DocumentDB.

- Per i dettagli sull'API, consulta CreateDbClusterParameterGroup AWS CLICommand Reference.

`create-db-cluster-snapshot`

Il seguente esempio di codice mostra come utilizzare`create-db-cluster-snapshot`.

AWS CLI

Come creare uno snapshot manuale di un cluster Amazon DocumentDB

L'`create-db-cluster-snapshot`esempio seguente crea uno snapshot del cluster Amazon DB denominato sample-cluster-snapshot.

```
aws docdb create-db-cluster-snapshot \
   --db-cluster-identifier sample-cluster \
   --db-cluster-snapshot-identifier sample-cluster-snapshot
```

Output:

```
{
    "DBClusterSnapshot": {
        "MasterUsername": "master-user",
        "SnapshotCreateTime": "2019-03-18T18:27:14.794Z",
        "AvailabilityZones": [
            "us-west-2a",
            "us-west-2b",
            "us-west-2c",
            "us-west-2d",
            "us-west-2e",
            "us-west-2f"
        ],
        "SnapshotType": "manual",
        "DBClusterSnapshotArn": "arn:aws:rds:us-west-2:123456789012:cluster-
snapshot:sample-cluster-snapshot",
        "EngineVersion": "3.6.0",
        "PercentProgress": 0,
        "DBClusterSnapshotIdentifier": "sample-cluster-snapshot",
        "Engine": "docdb",
        "DBClusterIdentifier": "sample-cluster",
        "Status": "creating",
        "ClusterCreateTime": "2019-03-15T20:29:58.836Z",
        "Port": 0,
        "StorageEncrypted": false,
        "VpcId": "vpc-91280df6"
    }
}
```

Per ulteriori informazioni, consulta Creazione manuale di uno snapshot del cluster nella Guida per gli sviluppatori di Amazon DocumentDB.

- Per i dettagli sull'API, consulta CreateDbClusterSnapshot AWS CLICommand Reference.

`create-db-cluster`

Il seguente esempio di codice mostra come utilizzare`create-db-cluster`.

AWS CLI

Come creare un cluster Amazon DocumentDB

L'esempio `create-db-cluster` seguente crea un cluster Amazon DocumentDB denominato `sample-cluster` con la finestra di manutenzione preferita impostata sulla domenica tra le 20:30 e le 11:00.

```
aws docdb create-db-cluster \
    --db-cluster-identifier sample-cluster \
    --engine docdb \
    --master-username master-user \
    --master-user-password password \
    --preferred-maintenance-window Sun:20:30-Sun:21:00
```

Output:

```
{
    "DBCluster": {
        "DBClusterParameterGroup": "default.docdb3.6",
        "AssociatedRoles": [],
        "DBSubnetGroup": "default",
        "ClusterCreateTime": "2019-03-18T18:06:34.616Z",
        "Status": "creating",
        "Port": 27017,
        "PreferredMaintenanceWindow": "sun:20:30-sun:21:00",
        "HostedZoneId": "ZNKXH85TT8WVW",
        "DBClusterMembers": [],
        "Engine": "docdb",
        "DBClusterIdentifier": "sample-cluster",
        "PreferredBackupWindow": "10:12-10:42",
        "AvailabilityZones": [
            "us-west-2d",
            "us-west-2f",
            "us-west-2e"
        ],
        "MasterUsername": "master-user",
        "BackupRetentionPeriod": 1,
        "ReaderEndpoint": "sample-cluster.cluster-ro-corcjozrlsfc.us-
west-2.docdb.amazonaws.com",
        "VpcSecurityGroups": [
            {
                "VpcSecurityGroupId": "sg-77186e0d",
                "Status": "active"
            }
        ],
        "StorageEncrypted": false,
```

```
        "DBClusterArn": "arn:aws:rds:us-west-2:123456789012:cluster:sample-cluster",
        "DbClusterResourceId": "cluster-L3R4YRSBUYDP4GLMTJ2WF5GH5Q",
        "MultiAZ": false,
        "Endpoint": "sample-cluster.cluster-corcjozrlsfc.us-
west-2.docdb.amazonaws.com",
        "EngineVersion": "3.6.0"
    }
}
```

Per ulteriori informazioni, consulta Creazione di un cluster Amazon DocumentDB nella Guida per gli sviluppatori di Amazon DocumentDB.

- Per i dettagli sull'API, consulta CreateDbCluster AWS CLICommand Reference.

`create-db-instance`

Il seguente esempio di codice mostra come utilizzare`create-db-instance`.

AWS CLI

Come creare un'istanza in un cluster Amazon DocumentDB

Il codice di esempio `create-db-instance` seguente crea l'istanza `sample-cluster-instance-2` nel cluster Amazon DocumentDB `sample-cluster`.

```
aws docdb create-db-instance \
    --db-cluster-identifier sample-cluster \
    --db-instance-class db.r4.xlarge \
    --db-instance-identifier sample-cluster-instance-2 \
    --engine docdb
```

Output:

```
{
    "DBInstance": {
        "DBInstanceStatus": "creating",
        "PendingModifiedValues": {
            "PendingCloudwatchLogsExports": {
                "LogTypesToEnable": [
                    "audit"
                ]
            }
        },
```

```
        "PubliclyAccessible": false,
        "PreferredBackupWindow": "00:00-00:30",
        "PromotionTier": 1,
        "EngineVersion": "3.6.0",
        "BackupRetentionPeriod": 3,
        "DBInstanceIdentifier": "sample-cluster-instance-2",
        "PreferredMaintenanceWindow": "tue:10:28-tue:10:58",
        "StorageEncrypted": false,
        "Engine": "docdb",
        "DBClusterIdentifier": "sample-cluster",
        "DBSubnetGroup": {
            "Subnets": [
                {
                    "SubnetAvailabilityZone": {
                        "Name": "us-west-2a"
                    },
                    "SubnetStatus": "Active",
                    "SubnetIdentifier": "subnet-4e26d263"
                },
                {
                    "SubnetAvailabilityZone": {
                        "Name": "us-west-2c"
                    },
                    "SubnetStatus": "Active",
                    "SubnetIdentifier": "subnet-afc329f4"
                },
                {
                    "SubnetAvailabilityZone": {
                        "Name": "us-west-2d"
                    },
                    "SubnetStatus": "Active",
                    "SubnetIdentifier": "subnet-53ab3636"
                },
                {
                    "SubnetAvailabilityZone": {
                        "Name": "us-west-2b"
                    },
                    "SubnetStatus": "Active",
                    "SubnetIdentifier": "subnet-991cb8d0"
                }
            ],
            "DBSubnetGroupDescription": "default",
            "SubnetGroupStatus": "Complete",
            "VpcId": "vpc-91280df6",
```

```
            "DBSubnetGroupName": "default"
        },
        "DBInstanceClass": "db.r4.xlarge",
        "VpcSecurityGroups": [
            {
                "Status": "active",
                "VpcSecurityGroupId": "sg-77186e0d"
            }
        ],
        "DBInstanceArn": "arn:aws:rds:us-west-2:123456789012:db:sample-cluster-
instance-2",
        "DbiResourceId": "db-XEKJLEMGRV5ZKCARUVA4HO3ITE"
    }
}
```

Per ulteriori informazioni, consulta Aggiunta di un'istanza Amazon DocumentDB a un cluster nella Guida per gli sviluppatori di Amazon DocumentDB.

- Per i dettagli sull'API, consulta CreateDbInstance AWS CLICommand Reference.

`create-db-subnet-group`

Il seguente esempio di codice mostra come utilizzare`create-db-subnet-group`.

AWS CLI

Come creare un gruppo di sottoreti di Amazon DocumentDB

L'esempio `create-db-subnet-group` seguente crea un gruppo di sottoreti di Amazon DocumentDB denominato `sample-subnet-group`.

```
aws docdb create-db-subnet-group \
    --db-subnet-group-description "a sample subnet group" \
    --db-subnet-group-name sample-subnet-group \
    --subnet-ids "subnet-29ab1025" "subnet-991cb8d0" "subnet-53ab3636"
```

Output:

```
{
    "DBSubnetGroup": {
        "SubnetGroupStatus": "Complete",
```

```
        "DBSubnetGroupName": "sample-subnet-group",
        "DBSubnetGroupDescription": "a sample subnet group",
        "VpcId": "vpc-91280df6",
        "DBSubnetGroupArn": "arn:aws:rds:us-west-2:123456789012:subgrp:sample-
subnet-group",
        "Subnets": [
            {
                "SubnetStatus": "Active",
                "SubnetIdentifier": "subnet-53ab3636",
                "SubnetAvailabilityZone": {
                    "Name": "us-west-2d"
                }
            },
            {
                "SubnetStatus": "Active",
                "SubnetIdentifier": "subnet-991cb8d0",
                "SubnetAvailabilityZone": {
                    "Name": "us-west-2b"
                }
            },
            {
                "SubnetStatus": "Active",
                "SubnetIdentifier": "subnet-29ab1025",
                "SubnetAvailabilityZone": {
                    "Name": "us-west-2c"
                }
            }
        ]
    }
}
```

Per ulteriori informazioni, consulta Creazione di un gruppo di sottoreti Amazon DocumentDB nella Guida per gli sviluppatori di Amazon DocumentDB.

- Per i dettagli sull'API, consulta CreateDbSubnetGroup AWS CLICommand Reference.

`delete-db-cluster-parameter-group`

Il seguente esempio di codice mostra come utilizzare`delete-db-cluster-parameter-group`.

AWS CLI

Come eliminare un gruppo di parametri di un cluster Amazon DocumentDB

L'esempio `delete-db-cluster-parameter-group` seguente elimina il gruppo di parametri `sample-parameter-group` di Amazon DocumentDB.

```
aws docdb delete-db-cluster-parameter-group \
    --db-cluster-parameter-group-name sample-parameter-group
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Eliminazione di un gruppo di parametri del cluster Amazon DocumentDB nella Guida per gli sviluppatori di Amazon DocumentDB.

- Per i dettagli sull'API, consulta DeleteDbClusterParameterGroup AWS CLICommand Reference.

`delete-db-cluster-snapshot`

Il seguente esempio di codice mostra come utilizzare`delete-db-cluster-snapshot`.

AWS CLI

Come eliminare uno snapshot di un cluster Amazon DocumentDB

L'esempio `delete-db-cluster-snapshot` seguente elimina lo snapshot `sample-cluster-snapshot` del cluster Amazon DocumentDB.

```
aws docdb delete-db-cluster-snapshot \
    --db-cluster-snapshot-identifier sample-cluster-snapshot
```

Output:

```
{
    "DBClusterSnapshot": {
        "DBClusterIdentifier": "sample-cluster",
        "AvailabilityZones": [
            "us-west-2a",
            "us-west-2b",
            "us-west-2c",
            "us-west-2d"
        ],
        "DBClusterSnapshotIdentifier": "sample-cluster-snapshot",
```

```
        "VpcId": "vpc-91280df6",
        "DBClusterSnapshotArn": "arn:aws:rds:us-west-2:123456789012:cluster-
snapshot:sample-cluster-snapshot",
        "EngineVersion": "3.6.0",
        "Engine": "docdb",
        "SnapshotCreateTime": "2019-03-18T18:27:14.794Z",
        "Status": "available",
        "MasterUsername": "master-user",
        "ClusterCreateTime": "2019-03-15T20:29:58.836Z",
        "PercentProgress": 100,
        "StorageEncrypted": false,
        "SnapshotType": "manual",
        "Port": 0
    }
}
```

Per ulteriori informazioni, consulta Eliminazione di una snapshot del cluster nella Guida per gli sviluppatori di Amazon DocumentDB.

- Per i dettagli sull'API, consulta DeleteDbClusterSnapshot AWS CLICommand Reference.

`delete-db-cluster`

Il seguente esempio di codice mostra come utilizzare`delete-db-cluster`.

AWS CLI

Come eliminare un cluster Amazon DocumentDB

L'esempio `delete-db-cluster` seguente elimina il cluster Amazon DocumentDB `sample-cluster`. Non viene eseguito alcun backup del cluster prima di eliminarlo. NOTA: è necessario scaricare tutte le istanze associate al cluster prima di poterlo eliminare.

```
aws docdb delete-db-cluster \
    --db-cluster-identifier sample-cluster \
    --skip-final-snapshot
```

Output:

```
{
    "DBCluster": {
```

```
        "DBClusterIdentifier": "sample-cluster",
        "DBSubnetGroup": "default",
        "EngineVersion": "3.6.0",
        "Engine": "docdb",
        "LatestRestorableTime": "2019-03-18T18:07:24.610Z",
        "PreferredMaintenanceWindow": "sun:20:30-sun:21:00",
        "StorageEncrypted": false,
        "EarliestRestorableTime": "2019-03-18T18:07:24.610Z",
        "Port": 27017,
        "VpcSecurityGroups": [
            {
                "Status": "active",
                "VpcSecurityGroupId": "sg-77186e0d"
            }
        ],
        "MultiAZ": false,
        "MasterUsername": "master-user",
        "DBClusterArn": "arn:aws:rds:us-west-2:123456789012:cluster:sample-cluster",
        "Status": "available",
        "PreferredBackupWindow": "10:12-10:42",
        "ReaderEndpoint": "sample-cluster.cluster-ro-corcjozrlsfc.us-
west-2.docdb.amazonaws.com",
        "AvailabilityZones": [
            "us-west-2c",
            "us-west-2b",
            "us-west-2a"
        ],
        "Endpoint": "sample-cluster.cluster-corcjozrlsfc.us-
west-2.docdb.amazonaws.com",
        "DbClusterResourceId": "cluster-L3R4YRSBUYDP4GLMTJ2WF5GH5Q",
        "ClusterCreateTime": "2019-03-18T18:06:34.616Z",
        "AssociatedRoles": [],
        "DBClusterParameterGroup": "default.docdb3.6",
        "HostedZoneId": "ZNKXH85TT8WVW",
        "BackupRetentionPeriod": 1,
        "DBClusterMembers": []
    }
}
```

Per ulteriori informazioni, consulta Eliminazione di un cluster Amazon DocumentDB nella Guida per gli sviluppatori di Amazon DocumentDB.

- Per i dettagli sull'API, consulta DeleteDbCluster AWS CLICommand Reference.

`delete-db-instance`

Il seguente esempio di codice mostra come utilizzare`delete-db-instance`.

AWS CLI

Come eliminare un'istanza Amazon DocumentDB

L'esempio `delete-db-instance` seguente elimina l'istanza Amazon DocumentDB `sample-cluster-instance-2`.

```
aws docdb delete-db-instance \
    --db-instance-identifier sample-cluster-instance-2
```

Output:

```
{
    "DBInstance": {
        "DBSubnetGroup": {
            "Subnets": [
                {
                    "SubnetAvailabilityZone": {
                        "Name": "us-west-2a"
                    },
                    "SubnetStatus": "Active",
                    "SubnetIdentifier": "subnet-4e26d263"
                },
                {
                    "SubnetAvailabilityZone": {
                        "Name": "us-west-2c"
                    },
                    "SubnetStatus": "Active",
                    "SubnetIdentifier": "subnet-afc329f4"
                },
                {
                    "SubnetAvailabilityZone": {
                        "Name": "us-west-2d"
                    },
                    "SubnetStatus": "Active",
                    "SubnetIdentifier": "subnet-53ab3636"
                },
                {
                    "SubnetAvailabilityZone": {
```

```
                        "Name": "us-west-2b"
                    },
                    "SubnetStatus": "Active",
                    "SubnetIdentifier": "subnet-991cb8d0"
                }
            ],
            "DBSubnetGroupName": "default",
            "DBSubnetGroupDescription": "default",
            "VpcId": "vpc-91280df6",
            "SubnetGroupStatus": "Complete"
        },
        "PreferredBackupWindow": "00:00-00:30",
        "InstanceCreateTime": "2019-03-18T18:37:33.709Z",
        "DBInstanceClass": "db.r4.xlarge",
        "DbiResourceId": "db-XEKJLEMGRV5ZKCARUVA4HO3ITE",
        "BackupRetentionPeriod": 3,
        "Engine": "docdb",
        "VpcSecurityGroups": [
            {
                "Status": "active",
                "VpcSecurityGroupId": "sg-77186e0d"
            }
        ],
        "AutoMinorVersionUpgrade": true,
        "PromotionTier": 1,
        "EngineVersion": "3.6.0",
        "Endpoint": {
            "Address": "sample-cluster-instance-2.corcjozrlsfc.us-
west-2.docdb.amazonaws.com",
            "HostedZoneId": "ZNKXH85TT8WVW",
            "Port": 27017
        },
        "DBInstanceIdentifier": "sample-cluster-instance-2",
        "PreferredMaintenanceWindow": "tue:10:28-tue:10:58",
        "EnabledCloudwatchLogsExports": [
            "audit"
        ],
        "PendingModifiedValues": {},
        "DBInstanceStatus": "deleting",
        "PubliclyAccessible": false,
        "DBInstanceArn": "arn:aws:rds:us-west-2:123456789012:db:sample-cluster-
instance-2",
        "DBClusterIdentifier": "sample-cluster",
        "AvailabilityZone": "us-west-2c",
```

```
        "StorageEncrypted": false
    }
}
```

Per ulteriori informazioni, consulta Eliminazione di un'istanza Amazon DocumentDB nella Guida per gli sviluppatori di Amazon DocumentDB.

- Per i dettagli sull'API, consulta DeleteDbInstance AWS CLICommand Reference.

`delete-db-subnet-group`

Il seguente esempio di codice mostra come utilizzare`delete-db-subnet-group`.

AWS CLI

Come eliminare un gruppo di sottoreti di Amazon DocumentDB

L'esempio `delete-db-subnet-group` seguente elimina il gruppo di sottoreti `sample-subnet-group` di Amazon DocumentDB.

```
aws docdb delete-db-subnet-group \
    --db-subnet-group-name sample-subnet-group
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Eliminazione di un gruppo di sottoreti Amazon DocumentDB nella Guida per gli sviluppatori di Amazon DocumentDB.

- Per i dettagli sull'API, consulta DeleteDbSubnetGroup AWS CLICommand Reference.

`describe-db-cluster-parameter-groups`

Il seguente esempio di codice mostra come utilizzare`describe-db-cluster-parameter-groups`.

AWS CLI

Come visualizzare i dettagli di uno o più gruppi di parametri del cluster Amazon DocumentDB

L'esempio `describe-db-cluster-parameter-groups` seguente visualizza i dettagli del gruppo di parametri `custom3-6-param-grp` del cluster Amazon DocumentDB.

```
aws docdb describe-db-cluster-parameter-groups \
    --db-cluster-parameter-group-name custom3-6-param-grp
```

Output:

```
{
    "DBClusterParameterGroups": [
        {
            "DBParameterGroupFamily": "docdb3.6",
            "DBClusterParameterGroupArn": "arn:aws:rds:us-
east-1:123456789012:cluster-pg:custom3-6-param-grp",
            "Description": "Custom docdb3.6 parameter group",
            "DBClusterParameterGroupName": "custom3-6-param-grp"
        }
    ]
}
```

Per ulteriori informazioni, consulta Visualizzazione di un gruppo di parametri del cluster Amazon DocumentDB nella Guida per gli sviluppatori di Amazon DocumentDB.

- Per i dettagli sull'API, consulta DescribeDbClusterParameterGroups AWS CLICommand Reference.

`describe-db-cluster-parameters`

Il seguente esempio di codice mostra come utilizzare`describe-db-cluster-parameters`.

AWS CLI

Come visualizzare l'elenco dettagliato dei parametri di un gruppo di parametri del cluster Amazon DocumentDB.

L'esempio `describe-db-cluster-parameters` seguente elenca i parametri per il gruppo di parametri custom3-6-param-grp di Amazon DocumentDB.

```
aws docdb describe-db-cluster-parameters \
     --db-cluster-parameter-group-name custom3-6-param-grp
```

Output:

```
{
```

```
    "Parameters": [
        {
            "DataType": "string",
            "ParameterName": "audit_logs",
            "IsModifiable": true,
            "ApplyMethod": "pending-reboot",
            "Source": "system",
            "ApplyType": "dynamic",
            "AllowedValues": "enabled,disabled",
            "Description": "Enables auditing on cluster.",
            "ParameterValue": "disabled"
        },
        {
            "DataType": "string",
            "ParameterName": "tls",
            "IsModifiable": true,
            "ApplyMethod": "pending-reboot",
            "Source": "system",
            "ApplyType": "static",
            "AllowedValues": "disabled,enabled",
            "Description": "Config to enable/disable TLS",
            "ParameterValue": "enabled"
        },
        {
            "DataType": "string",
            "ParameterName": "ttl_monitor",
            "IsModifiable": true,
            "ApplyMethod": "pending-reboot",
            "Source": "user",
            "ApplyType": "dynamic",
            "AllowedValues": "disabled,enabled",
            "Description": "Enables TTL Monitoring",
            "ParameterValue": "enabled"
        }
    ]
}
```

Per ulteriori informazioni, consulta Visualizzazione dei parametri del cluster Amazon DocumentDB nella Guida per gli sviluppatori di Amazon DocumentDB.

- Per i dettagli sull'API, consulta DescribeDbClusterParameters AWS CLICommand Reference.

`describe-db-cluster-snapshot-attributes`

Il seguente esempio di codice mostra come utilizzare`describe-db-cluster-snapshot-attributes`.

AWS CLI

Come elencare i nomi e i valori degli attributi di uno snapshot di Amazon DocumentDB

L'esempio `describe-db-cluster-snapshot-attributes` seguente elenca i nomi e i valori degli attributi per lo snapshot `sample-cluster-snapshot` di Amazon DocumentDB.

```
aws docdb describe-db-cluster-snapshot-attributes \
    --db-cluster-snapshot-identifier sample-cluster-snapshot
```

Output:

```
{
    "DBClusterSnapshotAttributesResult": {
        "DBClusterSnapshotAttributes": [
            {
                "AttributeName": "restore",
                "AttributeValues": []
            }
        ],
        "DBClusterSnapshotIdentifier": "sample-cluster-snapshot"
    }
}
```

Per ulteriori informazioni, consulta Descrivi DBCluster SnapshotAttributes nella Amazon DocumentDB Developer Guide.

- Per i dettagli sull'API, consulta DescribeDbClusterSnapshotAttributes AWS CLICommand Reference.

`describe-db-cluster-snapshots`

Il seguente esempio di codice mostra come utilizzare`describe-db-cluster-snapshots`.

AWS CLI

Come descrivere gli snapshot di Amazon DocumentDB

L'esempio `describe-db-cluster-snapshots` seguente visualizza i dettagli dello snapshot `sample-cluster-snapshot` di Amazon DocumentDB.

```
aws docdb describe-db-cluster-snapshots \
    --db-cluster-snapshot-identifier sample-cluster-snapshot
```

Output:

```
{
    "DBClusterSnapshots": [
        {
            "AvailabilityZones": [
                "us-west-2a",
                "us-west-2b",
                "us-west-2c",
                "us-west-2d"
            ],
            "Status": "available",
            "DBClusterSnapshotArn": "arn:aws:rds:us-west-2:123456789012:cluster-
snapshot:sample-cluster-snapshot",
            "SnapshotCreateTime": "2019-03-15T20:41:26.515Z",
            "SnapshotType": "manual",
            "DBClusterSnapshotIdentifier": "sample-cluster-snapshot",
            "DBClusterIdentifier": "sample-cluster",
            "MasterUsername": "master-user",
            "StorageEncrypted": false,
            "VpcId": "vpc-91280df6",
            "EngineVersion": "3.6.0",
            "PercentProgress": 100,
            "Port": 0,
            "Engine": "docdb",
            "ClusterCreateTime": "2019-03-15T20:29:58.836Z"
        }
    ]
}
```

Per ulteriori informazioni, consulta Descrivi le DBCluster istantanee nella Amazon DocumentDB Developer Guide.

- Per i dettagli sull'API, consulta AWS CLI Command DescribeDbClusterSnapshotsReference.

`describe-db-clusters`

Il seguente esempio di codice mostra come utilizzare`describe-db-clusters`.

AWS CLI

Come ottenere informazioni dettagliate su uno o più cluster Amazon DocumentDB.

L'esempio `describe-db-clusters` seguente visualizza i dettagli del cluster Amazon DocumentDB `sample-cluster`. Se si omette il parametro `--db-cluster-identifier`, è possibile ottenere informazioni su un massimo di 100 cluster.

```
aws docdb describe-db-clusters
    --db-cluster-identifier sample-cluster
```

Output:

```
{
    "DBClusters": [
        {
            "DBClusterParameterGroup": "default.docdb3.6",
            "Endpoint": "sample-cluster.cluster-corcjozrlsfc.us-
west-2.docdb.amazonaws.com",
            "PreferredBackupWindow": "00:00-00:30",
            "DBClusterIdentifier": "sample-cluster",
            "ClusterCreateTime": "2019-03-15T20:29:58.836Z",
            "LatestRestorableTime": "2019-03-18T20:28:03.239Z",
            "MasterUsername": "master-user",
            "DBClusterMembers": [
                {
                    "PromotionTier": 1,
                    "DBClusterParameterGroupStatus": "in-sync",
                    "IsClusterWriter": false,
                    "DBInstanceIdentifier": "sample-cluster"
                },
                {
                    "PromotionTier": 1,
                    "DBClusterParameterGroupStatus": "in-sync",
                    "IsClusterWriter": true,
                    "DBInstanceIdentifier": "sample-cluster2"
                }
            ],
```

```
            "PreferredMaintenanceWindow": "sat:04:30-sat:05:00",
            "VpcSecurityGroups": [
                {
                    "VpcSecurityGroupId": "sg-77186e0d",
                    "Status": "active"
                }
            ],
            "Engine": "docdb",
            "ReaderEndpoint": "sample-cluster.cluster-ro-corcjozrlsfc.us-
west-2.docdb.amazonaws.com",
            "DBSubnetGroup": "default",
            "MultiAZ": true,
            "AvailabilityZones": [
                "us-west-2a",
                "us-west-2c",
                "us-west-2b"
            ],
            "EarliestRestorableTime": "2019-03-15T20:30:47.020Z",
            "DbClusterResourceId": "cluster-UP4EF2PVDDFVHHDJQTYDAIGHLE",
            "DBClusterArn": "arn:aws:rds:us-west-2:123456789012:cluster:sample-
cluster",
            "BackupRetentionPeriod": 3,
            "HostedZoneId": "ZNKXH85TT8WVW",
            "StorageEncrypted": false,
            "EnabledCloudwatchLogsExports": [
                "audit"
            ],
            "AssociatedRoles": [],
            "EngineVersion": "3.6.0",
            "Port": 27017,
            "Status": "available"
        }
    ]
}
```

Per ulteriori informazioni, consulta Descrizione dei cluster Amazon DocumentDB nella Guida per gli sviluppatori di Amazon DocumentDB.

- Per i dettagli sull'API, consulta DescribeDbClusters AWS CLICommand Reference.

`describe-db-engine-versions`

Il seguente esempio di codice mostra come utilizzare`describe-db-engine-versions`.

AWS CLI

Come elencare le versioni disponibili del motore Amazon DocumentDB

L'esempio `describe-db-engine-versions` seguente elenca tutte le versioni disponibili del motore Amazon DocumentDB.

```
aws docdb describe-db-engine-versions \
    --engine docdb
```

Output:

```
{
    "DBEngineVersions": [
        {
            "DBEngineVersionDescription": "DocDB version 1.0.200837",
            "DBParameterGroupFamily": "docdb3.6",
            "EngineVersion": "3.6.0",
            "ValidUpgradeTarget": [],
            "DBEngineDescription": "Amazon DocumentDB (with MongoDB compatibility)",
            "SupportsLogExportsToCloudwatchLogs": true,
            "Engine": "docdb",
            "ExportableLogTypes": [
                "audit"
            ]
        }
    ]
}
```

Per ulteriori informazioni, consulta Descrivi DBEngine le versioni nella Amazon DocumentDB Developer Guide.

- Per i dettagli sull'API, consulta DescribeDbEngineVersions AWS CLICommand Reference.

`describe-db-instances`

Il seguente esempio di codice mostra come utilizzare`describe-db-instances`.

AWS CLI

Come ottenere informazioni sulle istanze Amazon DocumentDB con provisioning

L'esempio `describe-db-instances` seguente visualizza informazioni dettagliate sull'istanza Amazon DocumentDB `sample-cluster-instance`. Se si omette il parametro `--db-instance-identifier`, vengono restituite informazioni su un massimo di 100 istanze.

```
aws docdb describe-db-instances \
    --db-instance-identifier sample-cluster-instance
```

Output:

```
{
    "DBInstances": [
        {
            "Endpoint": {
                "HostedZoneId": "ZNKXH85TT8WVW",
                "Address": "sample-cluster-instance.corcjozrlsfc.us-
west-2.docdb.amazonaws.com",
                "Port": 27017
            },
            "PreferredBackupWindow": "00:00-00:30",
            "DBInstanceStatus": "available",
            "DBInstanceClass": "db.r4.large",
            "EnabledCloudwatchLogsExports": [
                "audit"
            ],
            "DBInstanceIdentifier": "sample-cluster-instance",
            "DBSubnetGroup": {
                "Subnets": [
                    {
                        "SubnetStatus": "Active",
                        "SubnetIdentifier": "subnet-4e26d263",
                        "SubnetAvailabilityZone": {
                            "Name": "us-west-2a"
                        }
                    },
                    {
                        "SubnetStatus": "Active",
                        "SubnetIdentifier": "subnet-afc329f4",
                        "SubnetAvailabilityZone": {
                            "Name": "us-west-2c"
                        }
                    },
                    {
                        "SubnetStatus": "Active",
```

```
                        "SubnetIdentifier": "subnet-53ab3636",
                        "SubnetAvailabilityZone": {
                            "Name": "us-west-2d"
                        }
                    },
                    {
                        "SubnetStatus": "Active",
                        "SubnetIdentifier": "subnet-991cb8d0",
                        "SubnetAvailabilityZone": {
                            "Name": "us-west-2b"
                        }
                    }
                ],
                "DBSubnetGroupName": "default",
                "SubnetGroupStatus": "Complete",
                "DBSubnetGroupDescription": "default",
                "VpcId": "vpc-91280df6"
            },
            "InstanceCreateTime": "2019-03-15T20:36:06.338Z",
            "Engine": "docdb",
            "StorageEncrypted": false,
            "AutoMinorVersionUpgrade": true,
            "DBInstanceArn": "arn:aws:rds:us-west-2:123456789012:db:sample-cluster-
instance",
            "PreferredMaintenanceWindow": "tue:08:39-tue:09:09",
            "VpcSecurityGroups": [
                {
                    "Status": "active",
                    "VpcSecurityGroupId": "sg-77186e0d"
                }
            ],
            "DBClusterIdentifier": "sample-cluster",
            "PendingModifiedValues": {},
            "BackupRetentionPeriod": 3,
            "PubliclyAccessible": false,
            "EngineVersion": "3.6.0",
            "PromotionTier": 1,
            "AvailabilityZone": "us-west-2c",
            "DbiResourceId": "db-A2GIKUV6KPOHITGGKI2NHVISZA"
        }
    ]
}
```

Per ulteriori informazioni, consulta Descrizione delle istanze Amazon DocumentDB nella Guida per gli sviluppatori di Amazon DocumentDB.

- Per i dettagli sull'API, consulta DescribeDbInstances AWS CLICommand Reference.

`describe-db-subnet-groups`

Il seguente esempio di codice mostra come utilizzare`describe-db-subnet-groups`.

AWS CLI

Come recuperare un elenco di descrizioni delle sottoreti Amazon DocumentDB

L'esempio `describe-db-subnet-groups` seguente descrive i dettagli per la sottorete Amazon DocumentDB denominata `default`.

```
aws docdb describe-db-subnet-groups \
    --db-subnet-group-name default
```

Output:

```
{
    "DBSubnetGroups": [
        {
            "VpcId": "vpc-91280df6",
            "DBSubnetGroupArn": "arn:aws:rds:us-west-2:123456789012:subgrp:default",
            "Subnets": [
                {
                    "SubnetIdentifier": "subnet-4e26d263",
                    "SubnetStatus": "Active",
                    "SubnetAvailabilityZone": {
                        "Name": "us-west-2a"
                    }
                },
                {
                    "SubnetIdentifier": "subnet-afc329f4",
                    "SubnetStatus": "Active",
                    "SubnetAvailabilityZone": {
                        "Name": "us-west-2c"
                    }
                },
                {
```

```
                    "SubnetIdentifier": "subnet-53ab3636",
                    "SubnetStatus": "Active",
                    "SubnetAvailabilityZone": {
                        "Name": "us-west-2d"
                    }
                },
                {
                    "SubnetIdentifier": "subnet-991cb8d0",
                    "SubnetStatus": "Active",
                    "SubnetAvailabilityZone": {
                        "Name": "us-west-2b"
                    }
                }
            ],
            "DBSubnetGroupName": "default",
            "SubnetGroupStatus": "Complete",
            "DBSubnetGroupDescription": "default"
        }
    ]
}
```

Per ulteriori informazioni, consulta Descrizione dei gruppi di sottoreti nella Guida per gli sviluppatori di Amazon DocumentDB.

- Per i dettagli sull'API, consulta DescribeDbSubnetGroups AWS CLICommand Reference.

`describe-engine-default-cluster-parameters`

Il seguente esempio di codice mostra come utilizzare`describe-engine-default-cluster-parameters`.

AWS CLI

Come descrivere il motore predefinito e ottenere informazioni sui parametri di sistema per Amazon DocumentDB

L'esempio `describe-engine-default-cluster-parameters` seguente visualizza i dettagli del motore predefinito e informazioni sui parametri di sistema per il gruppo di parametri `docdb3.6` di Amazon DocumentDB.

```
aws docdb describe-engine-default-cluster-parameters \
    --db-parameter-group-family docdb3.6
```

Output:

```
{
    "EngineDefaults": {
        "DBParameterGroupFamily": "docdb3.6",
        "Parameters": [
            {
                "ApplyType": "dynamic",
                "ParameterValue": "disabled",
                "Description": "Enables auditing on cluster.",
                "Source": "system",
                "DataType": "string",
                "MinimumEngineVersion": "3.6.0",
                "AllowedValues": "enabled,disabled",
                "ParameterName": "audit_logs",
                "IsModifiable": true
            },
            {
                "ApplyType": "static",
                "ParameterValue": "enabled",
                "Description": "Config to enable/disable TLS",
                "Source": "system",
                "DataType": "string",
                "MinimumEngineVersion": "3.6.0",
                "AllowedValues": "disabled,enabled",
                "ParameterName": "tls",
                "IsModifiable": true
            },
            {
                "ApplyType": "dynamic",
                "ParameterValue": "enabled",
                "Description": "Enables TTL Monitoring",
                "Source": "system",
                "DataType": "string",
                "MinimumEngineVersion": "3.6.0",
                "AllowedValues": "disabled,enabled",
                "ParameterName": "ttl_monitor",
                "IsModifiable": true
            }
        ]
    }
}
```

Per ulteriori informazioni, consulta DescribeEngineDefaultClusterParameterslа Amazon DocumentDB Developer Guide.

- Per i dettagli sull'API, consulta DescribeEngineDefaultClusterParameters AWS CLICommand Reference.

`describe-event-categories`

Il seguente esempio di codice mostra come utilizzare`describe-event-categories`.

AWS CLI

Come descrivere tutte le categorie di eventi di Amazon DocumentDB

L'esempio `describe-event-categories` seguente elenca tutte le categorie per il tipo di origine evento `db-instance` di Amazon DocumentDB.

```
aws docdb describe-event-categories \
    --source-type db-cluster
```

Output:

```
{
    "EventCategoriesMapList": [
        {
            "SourceType": "db-cluster",
            "EventCategories": [
                "failover",
                "maintenance",
                "notification",
                "failure"
            ]
        }
    ]
}
```

Per ulteriori informazioni, consulta Visualizzazione delle categorie di eventi nella Guida per gli sviluppatori di Amazon DocumentDB.

- Per i dettagli sull'API, consulta DescribeEventCategories AWS CLICommand Reference.

`describe-events`

Il seguente esempio di codice mostra come utilizzare`describe-events`.

AWS CLI

Come elencare gli eventi di Amazon DocumentDB

L'esempio `describe-events` seguente elenca tutti gli eventi di Amazon DocumentDB delle ultime 24 ore (1440 minuti).

```
aws docdb describe-events \
    --duration 1440
```

Questo comando non produce alcun output. Output:

```
{
    "Events": [
        {
            "EventCategories": [
                "failover"
            ],
            "Message": "Started cross AZ failover to DB instance: sample-cluster",
            "Date": "2019-03-18T21:36:29.807Z",
            "SourceArn": "arn:aws:rds:us-west-2:123456789012:cluster:sample-
cluster",
            "SourceIdentifier": "sample-cluster",
            "SourceType": "db-cluster"
        },
        {
            "EventCategories": [
                "availability"
            ],
            "Message": "DB instance restarted",
            "Date": "2019-03-18T21:36:40.793Z",
            "SourceArn": "arn:aws:rds:us-west-2:123456789012:db:sample-cluster",
            "SourceIdentifier": "sample-cluster",
            "SourceType": "db-instance"
        },
        {
            "EventCategories": [],
            "Message": "A new writer was promoted. Restarting database as a
 reader.",
```

```
            "Date": "2019-03-18T21:36:43.873Z",
            "SourceArn": "arn:aws:rds:us-west-2:123456789012:db:sample-cluster2",
            "SourceIdentifier": "sample-cluster2",
            "SourceType": "db-instance"
        },
        {
            "EventCategories": [
                "availability"
            ],
            "Message": "DB instance restarted",
            "Date": "2019-03-18T21:36:51.257Z",
            "SourceArn": "arn:aws:rds:us-west-2:123456789012:db:sample-cluster2",
            "SourceIdentifier": "sample-cluster2",
            "SourceType": "db-instance"
        },
        {
            "EventCategories": [
                "failover"
            ],
            "Message": "Completed failover to DB instance: sample-cluster",
            "Date": "2019-03-18T21:36:53.462Z",
            "SourceArn": "arn:aws:rds:us-west-2:123456789012:cluster:sample-
cluster",
            "SourceIdentifier": "sample-cluster",
            "SourceType": "db-cluster"
        },
        {
            "Date": "2019-03-19T16:51:48.847Z",
            "EventCategories": [
                "configuration change"
            ],
            "Message": "Updated parameter audit_logs to enabled with apply method
 pending-reboot",
            "SourceIdentifier": "custom3-6-param-grp",
            "SourceType": "db-parameter-group"
        },
        {
            "EventCategories": [
                "configuration change"
            ],
            "Message": "Applying modification to database instance class",
            "Date": "2019-03-19T17:55:20.095Z",
            "SourceArn": "arn:aws:rds:us-west-2:123456789012:db:sample-cluster2",
            "SourceIdentifier": "sample-cluster2",
```

```
            "SourceType": "db-instance"
        },
        {
            "EventCategories": [
                "availability"
            ],
            "Message": "DB instance shutdown",
            "Date": "2019-03-19T17:56:31.127Z",
            "SourceArn": "arn:aws:rds:us-west-2:123456789012:db:sample-cluster2",
            "SourceIdentifier": "sample-cluster2",
            "SourceType": "db-instance"
        },
        {
            "EventCategories": [
                "configuration change"
            ],
            "Message": "Finished applying modification to DB instance class",
            "Date": "2019-03-19T18:00:45.822Z",
            "SourceArn": "arn:aws:rds:us-west-2:123456789012:db:sample-cluster2",
            "SourceIdentifier": "sample-cluster2",
            "SourceType": "db-instance"
        },
        {
            "EventCategories": [
                "availability"
            ],
            "Message": "DB instance restarted",
            "Date": "2019-03-19T18:00:53.397Z",
            "SourceArn": "arn:aws:rds:us-west-2:123456789012:db:sample-cluster2",
            "SourceIdentifier": "sample-cluster2",
            "SourceType": "db-instance"
        },
        {
            "EventCategories": [
                "availability"
            ],
            "Message": "DB instance shutdown",
            "Date": "2019-03-19T18:23:36.045Z",
            "SourceArn": "arn:aws:rds:us-west-2:123456789012:db:sample-cluster2",
            "SourceIdentifier": "sample-cluster2",
            "SourceType": "db-instance"
        },
        {
            "EventCategories": [
```

```
                "availability"
            ],
            "Message": "DB instance restarted",
            "Date": "2019-03-19T18:23:46.209Z",
            "SourceArn": "arn:aws:rds:us-west-2:123456789012:db:sample-cluster2",
            "SourceIdentifier": "sample-cluster2",
            "SourceType": "db-instance"
        },
        {
            "Date": "2019-03-19T18:39:05.822Z",
            "EventCategories": [
                "configuration change"
            ],
            "Message": "Updated parameter ttl_monitor to enabled with apply method
 immediate",
            "SourceIdentifier": "custom3-6-param-grp",
            "SourceType": "db-parameter-group"
        },
        {
            "Date": "2019-03-19T18:39:48.067Z",
            "EventCategories": [
                "configuration change"
            ],
            "Message": "Updated parameter audit_logs to disabled with apply method
 immediate",
            "SourceIdentifier": "custom3-6-param-grp",
            "SourceType": "db-parameter-group"
        }
    ]
}
```

Per ulteriori informazioni, consulta Visualizzazione degli eventi Amazon DocumentDB nella Guida per gli sviluppatori di Amazon DocumentDB.

- Per i dettagli sull'API, consulta DescribeEvents AWS CLICommand Reference.

`describe-orderable-db-instance-options`

Il seguente esempio di codice mostra come utilizzare`describe-orderable-db-instance-options`.

AWS CLI

Come trovare le opzioni delle istanze Amazon DocumentDB che è possibile ordinare

L'esempio `describe-orderable-db-instance-options` seguente elenca tutte le opzioni delle istanze per Amazon DocumentDB per una Regione.

```
aws docdb describe-orderable-db-instance-options \
    --engine docdb \
    --region us-east-1
```

Output:

```
{
    "OrderableDBInstanceOptions": [
        {
            "Vpc": true,
            "AvailabilityZones": [
                {
                    "Name": "us-east-1a"
                },
                {
                    "Name": "us-east-1b"
                },
                {
                    "Name": "us-east-1c"
                },
                {
                    "Name": "us-east-1d"
                }
            ],
            "EngineVersion": "3.6.0",
            "DBInstanceClass": "db.r4.16xlarge",
            "LicenseModel": "na",
            "Engine": "docdb"
        },
        {
            "Vpc": true,
            "AvailabilityZones": [
                {
                    "Name": "us-east-1a"
                },
                {
```

```
                    "Name": "us-east-1b"
                },
                {
                    "Name": "us-east-1c"
                },
                {
                    "Name": "us-east-1d"
                }
                }
            ],
            "EngineVersion": "3.6.0",
            "DBInstanceClass": "db.r4.2xlarge",
            "LicenseModel": "na",
            "Engine": "docdb"
        },
        {
            "Vpc": true,
            "AvailabilityZones": [
                {
                    "Name": "us-east-1a"
                },
                {
                    "Name": "us-east-1b"
                },
                {
                    "Name": "us-east-1c"
                },
                {
                    "Name": "us-east-1d"
                }
            ],
            "EngineVersion": "3.6.0",
            "DBInstanceClass": "db.r4.4xlarge",
            "LicenseModel": "na",
            "Engine": "docdb"
        },
        {
            "Vpc": true,
            "AvailabilityZones": [
                {
                    "Name": "us-east-1a"
                },
                {
                    "Name": "us-east-1b"
```

```
                },
                {
                    "Name": "us-east-1c"
                },
                {
                    "Name": "us-east-1d"
                }
            ],
            "EngineVersion": "3.6.0",
            "DBInstanceClass": "db.r4.8xlarge",
            "LicenseModel": "na",
            "Engine": "docdb"
        },
        {
            "Vpc": true,
            "AvailabilityZones": [
                {
                    "Name": "us-east-1a"
                },
                {
                    "Name": "us-east-1b"
                },
                {
                    "Name": "us-east-1c"
                },
                {
                    "Name": "us-east-1d"
                }
            ],
            "EngineVersion": "3.6.0",
            "DBInstanceClass": "db.r4.large",
            "LicenseModel": "na",
            "Engine": "docdb"
        },
        {
            "Vpc": true,
            "AvailabilityZones": [
                {
                    "Name": "us-east-1a"
                },
                {
                    "Name": "us-east-1b"
                },
                {
```

```
                    "Name": "us-east-1c"
                },
                {
                    "Name": "us-east-1d"
                }
            ],
            "EngineVersion": "3.6.0",
            "DBInstanceClass": "db.r4.xlarge",
            "LicenseModel": "na",
            "Engine": "docdb"
        }
    ]
}
```

Per ulteriori informazioni, consulta Aggiunta di un'istanza Amazon DocumentDB a un cluster nella Guida per gli sviluppatori di Amazon DocumentDB.

- Per i dettagli sull'API, consulta DescribeOrderableDbInstanceOptions AWS CLICommand Reference.

`describe-pending-maintenance-actions`

Il seguente esempio di codice mostra come utilizzare`describe-pending-maintenance-actions`.

AWS CLI

Come elencare le operazioni di manutenzione di Amazon DocumentDB in sospeso

L'esempio `describe-pending-maintenance-actions` seguente elenca tutte le operazioni di manutenzione di Amazon DocumentDB in sospeso.

```
aws docdb describe-pending-maintenance-actions
```

Output:

```
{
    "PendingMaintenanceActions": []
}
```

Per ulteriori informazioni, consulta Gestione di Amazon DocumentDB nella Guida per gli sviluppatori di Amazon DocumentDB.

- Per i dettagli sull'API, consulta DescribePendingMaintenanceActions AWS CLICommand Reference.

`failover-db-cluster`

Il seguente esempio di codice mostra come utilizzare`failover-db-cluster`.

AWS CLI

Come forzare il failover di un cluster Amazon DocumentDB su una replica

L'esempio `failover-db-cluster` seguente determina il failover dell'istanza primaria nel cluster Amazon DocumentDB sample-cluster su una replica.

```
aws docdb failover-db-cluster \
    --db-cluster-identifier sample-cluster
```

Output:

```
{
    "DBCluster": {
        "AssociatedRoles": [],
        "DBClusterIdentifier": "sample-cluster",
        "EngineVersion": "3.6.0",
        "DBSubnetGroup": "default",
        "MasterUsername": "master-user",
        "EarliestRestorableTime": "2019-03-15T20:30:47.020Z",
        "Endpoint": "sample-cluster.cluster-corcjozrlsfc.us-
west-2.docdb.amazonaws.com",
        "AvailabilityZones": [
            "us-west-2a",
            "us-west-2c",
            "us-west-2b"
        ],
        "LatestRestorableTime": "2019-03-18T21:35:23.548Z",
        "PreferredMaintenanceWindow": "sat:04:30-sat:05:00",
        "PreferredBackupWindow": "00:00-00:30",
        "Port": 27017,
        "VpcSecurityGroups": [
            {
                "VpcSecurityGroupId": "sg-77186e0d",
                "Status": "active"
```

```
            }
        ],
        "StorageEncrypted": false,
        "ClusterCreateTime": "2019-03-15T20:29:58.836Z",
        "MultiAZ": true,
        "Status": "available",
        "DBClusterMembers": [
            {
                "DBClusterParameterGroupStatus": "in-sync",
                "IsClusterWriter": false,
                "DBInstanceIdentifier": "sample-cluster",
                "PromotionTier": 1
            },
            {
                "DBClusterParameterGroupStatus": "in-sync",
                "IsClusterWriter": true,
                "DBInstanceIdentifier": "sample-cluster2",
                "PromotionTier": 2
            }
        ],
        "EnabledCloudwatchLogsExports": [
            "audit"
        ],
        "DBClusterParameterGroup": "default.docdb3.6",
        "HostedZoneId": "ZNKXH85TT8WVW",
        "DBClusterArn": "arn:aws:rds:us-west-2:123456789012:cluster:sample-cluster",
        "BackupRetentionPeriod": 3,
        "DbClusterResourceId": "cluster-UP4EF2PVDDFVHHDJQTYDAIGHLE",
        "ReaderEndpoint": "sample-cluster.cluster-ro-corcjozrlsfc.us-
west-2.docdb.amazonaws.com",
        "Engine": "docdb"
    }
}
```

Per ulteriori informazioni, consulta Failover di Amazon DocumentDB nella Guida per gli sviluppatori di Amazon DocumentDB.

- Per i dettagli sull'API, consulta FailoverDbCluster AWS CLICommand Reference.

`list-tags-for-resource`

Il seguente esempio di codice mostra come utilizzare`list-tags-for-resource`.

AWS CLI

Come elencare tutti i tag su una risorsa Amazon DocumentDB

L'esempio `list-tags-for-resource` seguente elimina tutti i tag sul cluster Amazon DocumentDB `sample-cluster`.

```
aws docdb list-tags-for-resource \
    --resource-name arn:aws:rds:us-west-2:123456789012:cluster:sample-cluster
```

Output:

```
{
    "TagList": [
        {
            "Key": "A",
            "Value": "ALPHA"
        },
        {
            "Key": "B",
            "Value": ""
        },
        {
            "Key": "C",
            "Value": "CHARLIE"
        }
    ]
}
```

Per ulteriori informazioni, consulta Elenco dei tag in una risorsa Amazon DocumentDB nella Guida per gli sviluppatori di Amazon DocumentDB.

- Per i dettagli sull'API, consulta ListTagsForResource AWS CLICommand Reference.

`modify-db-cluster-parameter-group`

Il seguente esempio di codice mostra come utilizzare`modify-db-cluster-parameter-group`.

AWS CLI

Come modificare un gruppo di parametri del cluster di database Amazon DocumentDB

L'esempio `modify-db-cluster-parameter-group` seguente modifica il gruppo di parametri `custom3-6-param-grp` del cluster Amazon DocumentDB impostando i due parametri `audit_logs` e `ttl_monitor` come abilitati. Le modifiche vengono applicate al riavvio successivo.

```
aws docdb modify-db-cluster-parameter-group \
    --db-cluster-parameter-group-name custom3-6-param-grp \
    --
parameters ParameterName=audit_logs,ParameterValue=enabled,ApplyMethod=pending-
reboot \

  ParameterName=ttl_monitor,ParameterValue=enabled,ApplyMethod=pending-reboot
```

Output:

```
{
    "DBClusterParameterGroupName": "custom3-6-param-grp"
}
```

Per ulteriori informazioni, consulta Modifica di un gruppo di parametri del cluster Amazon DocumentDB nella Guida per gli sviluppatori di Amazon DocumentDB.

- Per i dettagli sull'API, consulta ModifyDbClusterParameterGroup AWS CLICommand Reference.

`modify-db-cluster-snapshot-attribute`

Il seguente esempio di codice mostra come utilizzare`modify-db-cluster-snapshot-attribute`.

AWS CLI

Esempio 1: come aggiungere un attributo a uno snapshot di Amazon DocumentDB

L'esempio `modify-db-cluster-snapshot-attribute` seguente aggiunge quattro valori di attributo a uno snapshot del cluster Amazon DocumentDB.

```
aws docdb modify-db-cluster-snapshot-attribute \
    --db-cluster-snapshot-identifier sample-cluster-snapshot \
    --attribute-name restore \
    --values-to-add 123456789011 123456789012 123456789013
```

Output:

```
{
    "DBClusterSnapshotAttributesResult": {
        "DBClusterSnapshotAttributes": [
            {
                "AttributeName": "restore",
                "AttributeValues": [
                    "123456789011",
                    "123456789012",
                    "123456789013"
                ]
            }
        ],
        "DBClusterSnapshotIdentifier": "sample-cluster-snapshot"
    }
}
```

Esempio 2: come rimuovere un attributo da uno snapshot di Amazon DocumentDB

L'esempio `modify-db-cluster-snapshot-attribute` seguente rimuove due valori di attributo da uno snapshot del cluster Amazon DocumentDB.

```
aws docdb modify-db-cluster-snapshot-attribute \
    --db-cluster-snapshot-identifier sample-cluster-snapshot \
    --attribute-name restore \
    --values-to-remove 123456789012
```

Output:

```
{
    "DBClusterSnapshotAttributesResult": {
        "DBClusterSnapshotAttributes": [
            {
                "AttributeName": "restore",
                "AttributeValues": [
                    "123456789011",
                    "123456789013"
                ]
            }
        ],
        "DBClusterSnapshotIdentifier": "sample-cluster-snapshot"
    }
```

```
}
```

Per ulteriori informazioni, consulta Modify DBCluster SnapshotAttribute nella Amazon DocumentDB Developer Guide.

- Per i dettagli sull'API, consulta ModifyDbClusterSnapshotAttribute AWS CLICommand Reference.

`modify-db-cluster`

Il seguente esempio di codice mostra come utilizzare`modify-db-cluster`.

AWS CLI

Come modificare un cluster Amazon DocumentDB

L'esempio `modify-db-cluster` seguente modifica il cluster Amazon DocumentDB `sample-cluster` impostando un periodo di conservazione per i backup automatici di 7 giorni e cambiando le finestre preferite per i backup e per la manutenzione. Tutte le modifiche vengono applicate nella finestra di manutenzione successiva.

```
aws docdb modify-db-cluster \
    --db-cluster-identifier sample-cluster \
    --no-apply-immediately \
    --backup-retention-period 7 \
    --preferred-backup-window 18:00-18:30 \
    --preferred-maintenance-window sun:20:00-sun:20:30
```

Output:

```
{
    "DBCluster": {
        "Endpoint": "sample-cluster.cluster-corcjozrlsfc.us-
west-2.docdb.amazonaws.com",
        "DBClusterMembers": [
            {
                "DBClusterParameterGroupStatus": "in-sync",
                "DBInstanceIdentifier": "sample-cluster",
                "IsClusterWriter": true,
                "PromotionTier": 1
            },
            {
```

```
                "DBClusterParameterGroupStatus": "in-sync",
                "DBInstanceIdentifier": "sample-cluster2",
                "IsClusterWriter": false,
                "PromotionTier": 2
            }
        ],
        "HostedZoneId": "ZNKXH85TT8WVW",
        "StorageEncrypted": false,
        "PreferredBackupWindow": "18:00-18:30",
        "MultiAZ": true,
        "EngineVersion": "3.6.0",
        "MasterUsername": "master-user",
        "ReaderEndpoint": "sample-cluster.cluster-ro-corcjozrlsfc.us-
west-2.docdb.amazonaws.com",
        "DBSubnetGroup": "default",
        "LatestRestorableTime": "2019-03-18T22:08:13.408Z",
        "EarliestRestorableTime": "2019-03-15T20:30:47.020Z",
        "PreferredMaintenanceWindow": "sun:20:00-sun:20:30",
        "AssociatedRoles": [],
        "EnabledCloudwatchLogsExports": [
            "audit"
        ],
        "Engine": "docdb",
        "DBClusterParameterGroup": "default.docdb3.6",
        "DBClusterArn": "arn:aws:rds:us-west-2:123456789012:cluster:sample-cluster",
        "BackupRetentionPeriod": 7,
        "DBClusterIdentifier": "sample-cluster",
        "AvailabilityZones": [
            "us-west-2a",
            "us-west-2c",
            "us-west-2b"
        ],
        "Status": "available",
        "DbClusterResourceId": "cluster-UP4EF2PVDDFVHHDJQTYDAIGHLE",
        "ClusterCreateTime": "2019-03-15T20:29:58.836Z",
        "VpcSecurityGroups": [
            {
                "VpcSecurityGroupId": "sg-77186e0d",
                "Status": "active"
            }
        ],
        "Port": 27017
    }
```

```
}
```

Per ulteriori informazioni, consulta Modifica di un cluster Amazon DocumentDB nella Guida per gli sviluppatori di Amazon DocumentDB.

- Per i dettagli sull'API, consulta ModifyDbCluster AWS CLICommand Reference.

`modify-db-instance`

Il seguente esempio di codice mostra come utilizzare`modify-db-instance`.

AWS CLI

Come modificare un'istanza Amazon DocumentDB

L'esempio `modify-db-instance` seguente modifica l'istanza `sample-cluster2` di Amazon DocumentDB cambiando la relativa classe di istanza in `db.r4.4xlarge` e il relativo livello di promozione in 5. Le modifiche vengono applicate immediatamente, ma sono visibili solo quando lo stato dell'istanza è disponibile.

```
aws docdb modify-db-instance \
    --db-instance-identifier sample-cluster2 \
    --apply-immediately \
    --db-instance-class db.r4.4xlarge \
    --promotion-tier 5
```

Output:

```
{
    "DBInstance": {
        "EngineVersion": "3.6.0",
        "StorageEncrypted": false,
        "DBInstanceClass": "db.r4.large",
        "PreferredMaintenanceWindow": "mon:08:39-mon:09:09",
        "AutoMinorVersionUpgrade": true,
        "VpcSecurityGroups": [
            {
                "VpcSecurityGroupId": "sg-77186e0d",
                "Status": "active"
            }
        ],
        "PreferredBackupWindow": "18:00-18:30",
```

```
        "EnabledCloudwatchLogsExports": [
            "audit"
        ],
        "AvailabilityZone": "us-west-2f",
        "DBInstanceIdentifier": "sample-cluster2",
        "InstanceCreateTime": "2019-03-15T20:36:06.338Z",
        "Engine": "docdb",
        "BackupRetentionPeriod": 7,
        "DBSubnetGroup": {
            "DBSubnetGroupName": "default",
            "DBSubnetGroupDescription": "default",
            "SubnetGroupStatus": "Complete",
            "Subnets": [
                {
                    "SubnetIdentifier": "subnet-4e26d263",
                    "SubnetAvailabilityZone": {
                        "Name": "us-west-2a"
                    },
                    "SubnetStatus": "Active"
                },
                {
                    "SubnetIdentifier": "subnet-afc329f4",
                    "SubnetAvailabilityZone": {
                        "Name": "us-west-2c"
                    },
                    "SubnetStatus": "Active"
                },
                {
                    "SubnetIdentifier": "subnet-53ab3636",
                    "SubnetAvailabilityZone": {
                        "Name": "us-west-2d"
                    },
                    "SubnetStatus": "Active"
                },
                {
                    "SubnetIdentifier": "subnet-991cb8d0",
                    "SubnetAvailabilityZone": {
                        "Name": "us-west-2b"
                    },
                    "SubnetStatus": "Active"
                }
            ],
            "VpcId": "vpc-91280df6"
        },
```

```
        "PromotionTier": 2,
        "Endpoint": {
            "Address": "sample-cluster2.corcjozrlsfc.us-west-2.docdb.amazonaws.com",
            "HostedZoneId": "ZNKXH85TT8WVW",
            "Port": 27017
        },
        "DbiResourceId": "db-A2GIKUV6KPOHITGGKI2NHVISZA",
        "DBClusterIdentifier": "sample-cluster",
        "DBInstanceArn": "arn:aws:rds:us-west-2:123456789012:db:sample-cluster2",
        "PendingModifiedValues": {
            "DBInstanceClass": "db.r4.4xlarge"
        },
        "PubliclyAccessible": false,
        "DBInstanceStatus": "available"
    }
}
```

Per ulteriori informazioni, consulta Modifica di un'istanza Amazon DocumentDB nella Guida per gli sviluppatori di Amazon DocumentDB.

- Per i dettagli sull'API, consulta ModifyDbInstance AWS CLICommand Reference.

`modify-db-subnet-group`

Il seguente esempio di codice mostra come utilizzare`modify-db-subnet-group`.

AWS CLI

Come modificare un gruppo di sottoreti di Amazon DocumentDB

L'esempio `modify-db-subnet-group` seguente modifica il gruppo di sottoreti `sample-subnet-group` aggiungendo le sottoreti specificate e una nuova descrizione.

```
aws docdb modify-db-subnet-group \
    --db-subnet-group-name sample-subnet-group \
    --subnet-ids subnet-b3806e8f subnet-53ab3636 subnet-991cb8d0 \
    --db-subnet-group-description "New subnet description"
```

Output:

```
{
    "DBSubnetGroup": {
        "DBSubnetGroupName": "sample-subnet-group",
```

```
        "SubnetGroupStatus": "Complete",
        "DBSubnetGroupArn": "arn:aws:rds:us-west-2:123456789012:subgrp:sample-
subnet-group",
        "VpcId": "vpc-91280df6",
        "DBSubnetGroupDescription": "New subnet description",
        "Subnets": [
            {
                "SubnetIdentifier": "subnet-b3806e8f",
                "SubnetStatus": "Active",
                "SubnetAvailabilityZone": {
                    "Name": "us-west-2a"
                }
            },
            {
                "SubnetIdentifier": "subnet-53ab3636",
                "SubnetStatus": "Active",
                "SubnetAvailabilityZone": {
                    "Name": "us-west-2c"
                }
            },
            {
                "SubnetIdentifier": "subnet-991cb8d0",
                "SubnetStatus": "Active",
                "SubnetAvailabilityZone": {
                    "Name": "us-west-2b"
                }
            }
        ]
    }
}
```

Per ulteriori informazioni, consulta Modifica di un gruppo di sottoreti Amazon DocumentDB nella Guida per gli sviluppatori di Amazon DocumentDB.

- Per i dettagli sull'API, consulta ModifyDbSubnetGroup AWS CLICommand Reference.

`reboot-db-instance`

Il seguente esempio di codice mostra come utilizzare`reboot-db-instance`.

AWS CLI

Come riavviare un'istanza Amazon DocumentDB

L'esempio `reboot-db-instance` seguente riavvia l'istanza Amazon DocumentDB `sample-cluster2`.

```
aws docdb reboot-db-instance \
    --db-instance-identifier sample-cluster2
```

Questo comando non produce alcun output. Output:

```
{
    "DBInstance": {
        "PreferredBackupWindow": "18:00-18:30",
        "DBInstanceIdentifier": "sample-cluster2",
        "VpcSecurityGroups": [
            {
                "Status": "active",
                "VpcSecurityGroupId": "sg-77186e0d"
            }
        ],
        "DBSubnetGroup": {
            "VpcId": "vpc-91280df6",
            "Subnets": [
                {
                    "SubnetStatus": "Active",
                    "SubnetAvailabilityZone": {
                        "Name": "us-west-2a"
                    },
                    "SubnetIdentifier": "subnet-4e26d263"
                },
                {
                    "SubnetStatus": "Active",
                    "SubnetAvailabilityZone": {
                        "Name": "us-west-2c"
                    },
                    "SubnetIdentifier": "subnet-afc329f4"
                },
                {
                    "SubnetStatus": "Active",
                    "SubnetAvailabilityZone": {
                        "Name": "us-west-2d"
                    },
                    "SubnetIdentifier": "subnet-53ab3636"
                },
                {
```

```
                    "SubnetStatus": "Active",
                    "SubnetAvailabilityZone": {
                        "Name": "us-west-2b"
                    },
                    "SubnetIdentifier": "subnet-991cb8d0"
                }
            ],
            "SubnetGroupStatus": "Complete",
            "DBSubnetGroupName": "default",
            "DBSubnetGroupDescription": "default"
        },
        "PendingModifiedValues": {},
        "Endpoint": {
            "Address": "sample-cluster2.corcjozrlsfc.us-west-2.docdb.amazonaws.com",
            "HostedZoneId": "ZNKXH85TT8WVW",
            "Port": 27017
        },
        "EnabledCloudwatchLogsExports": [
            "audit"
        ],
        "StorageEncrypted": false,
        "DbiResourceId": "db-A2GIKUV6KPOHITGGKI2NHVISZA",
        "AutoMinorVersionUpgrade": true,
        "Engine": "docdb",
        "InstanceCreateTime": "2019-03-15T20:36:06.338Z",
        "EngineVersion": "3.6.0",
        "PromotionTier": 5,
        "BackupRetentionPeriod": 7,
        "DBClusterIdentifier": "sample-cluster",
        "PreferredMaintenanceWindow": "mon:08:39-mon:09:09",
        "PubliclyAccessible": false,
        "DBInstanceClass": "db.r4.4xlarge",
        "AvailabilityZone": "us-west-2d",
        "DBInstanceArn": "arn:aws:rds:us-west-2:123456789012:db:sample-cluster2",
        "DBInstanceStatus": "rebooting"
    }
}
```

Per ulteriori informazioni, consulta Rebooting an Amazon DocumentDB nella Amazon ILnstance DocumentDB Developer Guide.

- Per i dettagli sull'API, consulta RebootDbInstanceCommand Reference.AWS CLI

`remove-tags-from-resource`

Il seguente esempio di codice mostra come utilizzare`remove-tags-from-resource`.

AWS CLI

Come rimuovere tag da una risorsa Amazon DocumentDB

L'esempio `remove-tags-from-resource` seguente rimuove il tag con la chiave denominata `B` dal cluster Amazon DocumentDB `sample-cluster`.

```
aws docdb remove-tags-from-resource \
    --resource-name arn:aws:rds:us-west-2:123456789012:cluster:sample-cluster \
    --tag-keys B
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Removing tags from an Amazon Document DBResource nella Amazon DocumentDB Developer Guide.

- Per i dettagli sull'API, consulta RemoveTagsFromResource AWS CLICommand Reference.

`reset-db-cluster-parameter-group`

Il seguente esempio di codice mostra come utilizzare`reset-db-cluster-parameter-group`.

AWS CLI

Come ripristinare il valore predefinito del parametro specificato in un gruppo di parametri di Amazon DocumentDB

L'esempio `reset-db-cluster-parameter-group` seguente ripristina il valore predefinito del parametro `ttl_monitor` nel gruppo di parametri `custom3-6-param-grp` di Amazon DocumentDB.

```
aws docdb reset-db-cluster-parameter-group \
    --db-cluster-parameter-group-name custom3-6-param-grp \
    --parameters ParameterName=ttl_monitor,ApplyMethod=immediate
```

Output:

```
{
```

```
    "DBClusterParameterGroupName": "custom3-6-param-grp"
}
```

Per ulteriori informazioni, consulta la Guida per gli sviluppatori di Amazon DocumentDB.

Come ripristinare il valore predefinito del parametro specificato o di tutti i parametri in un gruppo di parametri di Amazon DocumentDB

L'esempio `reset-db-cluster-parameter-group` seguente ripristina il valore predefinito di tutti i parametri nel gruppo di parametri `custom3-6-param-grp` di Amazon DocumentDB.

```
aws docdb reset-db-cluster-parameter-group \
    --db-cluster-parameter-group-name custom3-6-param-grp \
    --reset-all-parameters
```

Output:

```
{
    "DBClusterParameterGroupName": "custom3-6-param-grp"
}
```

Per ulteriori informazioni, consulta Reimpostazione di un gruppo di parametri del cluster Amazon DocumentDB nella Guida per gli sviluppatori di Amazon DocumentDB.

- Per i dettagli sull'API, consulta ResetDbClusterParameterGroup AWS CLICommand Reference.

`restore-db-cluster-from-snapshot`

Il seguente esempio di codice mostra come utilizzare`restore-db-cluster-from-snapshot`.

AWS CLI

Come ripristinare un cluster Amazon DocumentDB da uno snapshot automatico o manuale

L'esempio `restore-db-cluster-from-snapshot` seguente crea un nuovo cluster Amazon DocumentDB denominato `sample-cluster-2019-03-16-00-01-restored` dallo snapshot `rds:sample-cluster-2019-03-16-00-01`.

```
aws docdb restore-db-cluster-from-snapshot \
    --db-cluster-identifier sample-cluster-2019-03-16-00-01-restored \
    --engine docdb \
```

```
    --snapshot-identifier rds:sample-cluster-2019-03-16-00-01
```

Output:

```
{
    "DBCluster": {
        "ClusterCreateTime": "2019-03-19T18:45:01.857Z",
        "HostedZoneId": "ZNKXH85TT8WVW",
        "Engine": "docdb",
        "DBClusterMembers": [],
        "MultiAZ": false,
        "AvailabilityZones": [
            "us-west-2a",
            "us-west-2c",
            "us-west-2b"
        ],
        "StorageEncrypted": false,
        "ReaderEndpoint": "sample-cluster-2019-03-16-00-01-restored.cluster-ro-
corcjozrlsfc.us-west-2.docdb.amazonaws.com",
        "Endpoint": "sample-cluster-2019-03-16-00-01-restored.cluster-
corcjozrlsfc.us-west-2.docdb.amazonaws.com",
        "Port": 27017,
        "PreferredBackupWindow": "00:00-00:30",
        "DBSubnetGroup": "default",
        "DBClusterIdentifier": "sample-cluster-2019-03-16-00-01-restored",
        "PreferredMaintenanceWindow": "sat:04:30-sat:05:00",
        "DBClusterArn": "arn:aws:rds:us-west-2:123456789012:cluster:sample-
cluster-2019-03-16-00-01-restored",
        "DBClusterParameterGroup": "default.docdb3.6",
        "DbClusterResourceId": "cluster-XOO46Q3RH4LWSYNH3NMZKXPISU",
        "MasterUsername": "master-user",
        "EngineVersion": "3.6.0",
        "BackupRetentionPeriod": 3,
        "AssociatedRoles": [],
        "Status": "creating",
        "VpcSecurityGroups": [
            {
                "Status": "active",
                "VpcSecurityGroupId": "sg-77186e0d"
            }
        ]
    }
}
```

Per ulteriori informazioni, consulta Ripristino da uno snapshot del cluster nella Guida per gli sviluppatori di Amazon DocumentDB.

- Per i dettagli sull'API, consulta RestoreDbClusterFromSnapshot AWS CLICommand Reference.

`restore-db-cluster-to-point-in-time`

Il seguente esempio di codice mostra come utilizzare`restore-db-cluster-to-point-in-time`.

AWS CLI

Per ripristinare un cluster Amazon DocumentDB point-in-time da uno snapshot manuale

L'esempio `restore-db-cluster-to-point-in-time` seguente utilizza `sample-cluster-snapshot` per creare un nuovo cluster Amazon DocumentDB`sample-cluster-pit` utilizzando l'orario di ripristino più recente.

```
aws docdb restore-db-cluster-to-point-in-time \
    --db-cluster-identifier sample-cluster-pit \
    --source-db-cluster-identifier arn:aws:rds:us-
west-2:123456789012:cluster:sample-cluster \
    --use-latest-restorable-time
```

Output:

```
{
    "DBCluster": {
        "StorageEncrypted": false,
        "BackupRetentionPeriod": 3,
        "MasterUsername": "master-user",
        "HostedZoneId": "ZNKXH85TT8WVW",
        "PreferredBackupWindow": "00:00-00:30",
        "MultiAZ": false,
        "DBClusterIdentifier": "sample-cluster-pit",
        "DBSubnetGroup": "default",
        "ClusterCreateTime": "2019-04-03T15:55:21.320Z",
        "AssociatedRoles": [],
        "DBClusterParameterGroup": "default.docdb3.6",
        "DBClusterMembers": [],
        "Status": "creating",
        "AvailabilityZones": [
            "us-west-2a",
            "us-west-2d",
```

```
            "us-west-2b"
        ],
        "ReaderEndpoint": "sample-cluster-pit.cluster-ro-corcjozrlsfc.us-
west-2.docdb.amazonaws.com",
        "Port": 27017,
        "Engine": "docdb",
        "EngineVersion": "3.6.0",
        "VpcSecurityGroups": [
            {
                "VpcSecurityGroupId": "sg-77186e0d",
                "Status": "active"
            }
        ],
        "PreferredMaintenanceWindow": "sat:04:30-sat:05:00",
        "Endpoint": "sample-cluster-pit.cluster-corcjozrlsfc.us-
west-2.docdb.amazonaws.com",
        "DbClusterResourceId": "cluster-NLCABBXOSE2QPQ4GOLZIFWEPLM",
        "DBClusterArn": "arn:aws:rds:us-west-2:123456789012:cluster:sample-cluster-
pit"
    }
}
```

Per ulteriori informazioni, consulta Ripristino di uno snapshot a un point-in-time (PITR) nella Guida per gli sviluppatori di Amazon DocumentDB.

- Per i dettagli sull'API, consulta AWS CLI Command RestoreDbClusterToPointInTimeReference.

`start-db-cluster`

Il seguente esempio di codice mostra come utilizzare`start-db-cluster`.

AWS CLI

Come avviare un cluster Amazon DocumentDB arrestato

L'esempio `start-db-cluster` seguente elimina il cluster Amazon DocumentDB specificato.

```
aws docdb start-db-cluster \
    --db-cluster-identifier sample-cluster
```

Output:

```
{
```

```
    "DBCluster": {
        "ClusterCreateTime": "2019-03-19T18:45:01.857Z",
        "HostedZoneId": "ZNKXH85TT8WVW",
        "Engine": "docdb",
        "DBClusterMembers": [],
        "MultiAZ": false,
        "AvailabilityZones": [
            "us-east-1a",
            "us-east-1c",
            "us-east-1f"
        ],
        "StorageEncrypted": false,
        "ReaderEndpoint": "sample-cluster-2019-03-16-00-01-restored.cluster-ro-
corcjozrlsfc.us-east-1.docdb.amazonaws.com",
        "Endpoint": "sample-cluster-2019-03-16-00-01-restored.cluster-
corcjozrlsfc.us-east-1.docdb.amazonaws.com",
        "Port": 27017,
        "PreferredBackupWindow": "00:00-00:30",
        "DBSubnetGroup": "default",
        "DBClusterIdentifier": "sample-cluster-2019-03-16-00-01-restored",
        "PreferredMaintenanceWindow": "sat:04:30-sat:05:00",
        "DBClusterArn": "arn:aws:rds:us-east-1:123456789012:cluster:sample-
cluster-2019-03-16-00-01-restored",
        "DBClusterParameterGroup": "default.docdb3.6",
        "DbClusterResourceId": "cluster-XOO46Q3RH4LWSYNH3NMZKXPISU",
        "MasterUsername": "master-user",
        "EngineVersion": "3.6.0",
        "BackupRetentionPeriod": 3,
        "AssociatedRoles": [],
        "Status": "creating",
        "VpcSecurityGroups": [
            {
                "Status": "active",
                "VpcSecurityGroupId": "sg-77186e0d"
            }
        ]
    }
}
```

Per ulteriori informazioni, consulta Arresto e avvio di un cluster Amazon DocumentDB nella Guida per gli sviluppatori di Amazon DocumentDB.

- Per i dettagli sull'API, consulta StartDbCluster AWS CLICommand Reference.

`stop-db-cluster`

Il seguente esempio di codice mostra come utilizzare`stop-db-cluster`.

AWS CLI

Come arrestare un cluster Amazon DocumentDB in esecuzione

L'esempio `stop-db-cluster` seguente arresta il cluster Amazon DocumentDB specificato.

```
aws docdb stop-db-cluster \
    --db-cluster-identifier sample-cluster
```

Output:

```
{
    "DBCluster": {
        "ClusterCreateTime": "2019-03-19T18:45:01.857Z",
        "HostedZoneId": "ZNKXH85TT8WVW",
        "Engine": "docdb",
        "DBClusterMembers": [],
        "MultiAZ": false,
        "AvailabilityZones": [
            "us-east-1a",
            "us-east-1c",
            "us-east-1f"
        ],
        "StorageEncrypted": false,
        "ReaderEndpoint": "sample-cluster-2019-03-16-00-01-restored.cluster-ro-
corcjozrlsfc.us-east-1.docdb.amazonaws.com",
        "Endpoint": "sample-cluster-2019-03-16-00-01-restored.cluster-
corcjozrlsfc.us-east-1.docdb.amazonaws.com",
        "Port": 27017,
        "PreferredBackupWindow": "00:00-00:30",
        "DBSubnetGroup": "default",
        "DBClusterIdentifier": "sample-cluster-2019-03-16-00-01-restored",
        "PreferredMaintenanceWindow": "sat:04:30-sat:05:00",
        "DBClusterArn": "arn:aws:rds:us-east-1:123456789012:cluster:sample-
cluster-2019-03-16-00-01-restored",
        "DBClusterParameterGroup": "default.docdb3.6",
        "DbClusterResourceId": "cluster-XOO46Q3RH4LWSYNH3NMZKXPISU",
        "MasterUsername": "master-user",
        "EngineVersion": "3.6.0",
```

```
        "BackupRetentionPeriod": 3,
        "AssociatedRoles": [],
        "Status": "creating",
        "VpcSecurityGroups": [
            {
                "Status": "active",
                "VpcSecurityGroupId": "sg-77186e0d"
            }
        ]
    }
}
```

Per ulteriori informazioni, consulta Arresto e avvio di un cluster Amazon DocumentDB nella Guida per gli sviluppatori di Amazon DocumentDB.

- Per i dettagli sull'API, consulta StopDbCluster AWS CLICommand Reference.

Esempi di utilizzo di DynamoDB AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface con DynamoDB.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`batch-get-item`

Il seguente esempio di codice mostra come utilizzare. `batch-get-item`

AWS CLI

Come recuperare più elementi da una tabella

L'esempio `batch-get-items` seguente legge più elementi dalla tabella `MusicCollection` utilizzando un batch di tre richieste `GetItem` e richiede il numero di unità di capacità di lettura utilizzate dall'operazione. Il comando restituisce solo l'attributo `AlbumTitle`.

```
aws dynamodb batch-get-item \
    --request-items file://request-items.json \
    --return-consumed-capacity TOTAL
```

Contenuto di `request-items.json`:

```
{
    "MusicCollection": {
        "Keys": [
            {
                "Artist": {"S": "No One You Know"},
                "SongTitle": {"S": "Call Me Today"}
            },
            {
                "Artist": {"S": "Acme Band"},
                "SongTitle": {"S": "Happy Day"}
            },
            {
                "Artist": {"S": "No One You Know"},
                "SongTitle": {"S": "Scared of My Shadow"}
            }
        ],
        "ProjectionExpression":"AlbumTitle"
    }
}
```

Output:

```
{
    "Responses": {
        "MusicCollection": [
            {
                "AlbumTitle": {
                    "S": "Somewhat Famous"
                }
            },
            {
                "AlbumTitle": {
```

```
                    "S": "Blue Sky Blues"
                }
            },
            {
                "AlbumTitle": {
                    "S": "Louder Than Ever"
                }
            }
        ]
    },
    "UnprocessedKeys": {},
    "ConsumedCapacity": [
        {
            "TableName": "MusicCollection",
            "CapacityUnits": 1.5
        }
    ]
}
```

Per ulteriori informazioni, consulta Operazioni batch nella Guida per gli sviluppatori di Amazon DynamoDB.

- Per i dettagli sull'API, consulta BatchGetItem AWS CLICommand Reference.

`batch-write-item`

Il seguente esempio di codice mostra come utilizzare`batch-write-item`.

AWS CLI

Come aggiungere più elementi a una tabella

L'esempio `batch-write-item` seguente aggiunge tre nuovi elementi alla tabella `MusicCollection` utilizzando un batch di tre richieste `PutItem`. Richiede inoltre informazioni sul numero di unità di capacità di scrittura utilizzate dall'operazione e su eventuali raccolte di elementi modificate dall'operazione.

```
aws dynamodb batch-write-item \
    --request-items file://request-items.json \
    --return-consumed-capacity INDEXES \
    --return-item-collection-metrics SIZE
```

Contenuto di `request-items.json`:

```
{
    "MusicCollection": [
        {
            "PutRequest": {
                "Item": {
                    "Artist": {"S": "No One You Know"},
                    "SongTitle": {"S": "Call Me Today"},
                    "AlbumTitle": {"S": "Somewhat Famous"}
                }
            }
        },
        {
            "PutRequest": {
                "Item": {
                    "Artist": {"S": "Acme Band"},
                    "SongTitle": {"S": "Happy Day"},
                    "AlbumTitle": {"S": "Songs About Life"}
                }
            }
        },
        {
            "PutRequest": {
                "Item": {
                    "Artist": {"S": "No One You Know"},
                    "SongTitle": {"S": "Scared of My Shadow"},
                    "AlbumTitle": {"S": "Blue Sky Blues"}
                }
            }
        }
    ]
}
```

Output:

```
{
    "UnprocessedItems": {},
    "ItemCollectionMetrics": {
        "MusicCollection": [
            {
                "ItemCollectionKey": {
                    "Artist": {
```

```
                        "S": "No One You Know"
                    }
                },
                "SizeEstimateRangeGB": [
                    0.0,
                    1.0
                ]
            },
            {
                "ItemCollectionKey": {
                    "Artist": {
                        "S": "Acme Band"
                    }
                },
                "SizeEstimateRangeGB": [
                    0.0,
                    1.0
                ]
            }
        ]
    },
    "ConsumedCapacity": [
        {
            "TableName": "MusicCollection",
            "CapacityUnits": 6.0,
            "Table": {
                "CapacityUnits": 3.0
            },
            "LocalSecondaryIndexes": {
                "AlbumTitleIndex": {
                    "CapacityUnits": 3.0
                }
            }
        }
    ]
}
```

Per ulteriori informazioni, consulta Operazioni batch nella Guida per gli sviluppatori di Amazon DynamoDB.

- Per i dettagli sull'API, consulta BatchWriteItem AWS CLICommand Reference.

`create-backup`

Il seguente esempio di codice mostra come utilizzare`create-backup`.

AWS CLI

Come creare un backup per una tabella DynamoDB esistente

L'esempio `create-backup` seguente crea un backup della tabella `MusicCollection`.

```
aws dynamodb create-backup \
    --table-name MusicCollection \
    --backup-name MusicCollectionBackup
```

Output:

```
{
    "BackupDetails": {
        "BackupArn": "arn:aws:dynamodb:us-west-2:123456789012:table/MusicCollection/
backup/01576616366715-b4e58d3a",
        "BackupName": "MusicCollectionBackup",
        "BackupSizeBytes": 0,
        "BackupStatus": "CREATING",
        "BackupType": "USER",
        "BackupCreationDateTime": 1576616366.715
    }
}
```

Per ulteriori informazioni, consulta Backup e ripristino on demand per DynamoDB nella Guida per gli sviluppatori di Amazon DynamoDB.

- Per i dettagli sull'API, consulta CreateBackup AWS CLICommand Reference.

`create-global-table`

Il seguente esempio di codice mostra come utilizzare`create-global-table`.

AWS CLI

Come creare una tabella globale

L'`create-global-table`esempio seguente crea una tabella globale da due tabelle identiche nelle AWS regioni separate specificate.

```
aws dynamodb create-global-table \
    --global-table-name MusicCollection \
    --replication-group RegionName=us-east-2 RegionName=us-east-1 \
    --region us-east-2
```

Output:

```
{
    "GlobalTableDescription": {
        "ReplicationGroup": [
            {
                "RegionName": "us-east-2"
            },
            {
                "RegionName": "us-east-1"
            }
        ],
        "GlobalTableArn": "arn:aws:dynamodb::123456789012:global-table/
MusicCollection",
        "CreationDateTime": 1576625818.532,
        "GlobalTableStatus": "CREATING",
        "GlobalTableName": "MusicCollection"
    }
}
```

Per ulteriori informazioni, consulta Tabelle globali DynamoDB nella Guida per gli sviluppatori di Amazon DynamoDB.

- Per i dettagli sull'API, vedere CreateGlobalTablein AWS CLI Command Reference.

`create-table`

Il seguente esempio di codice mostra come utilizzare`create-table`.

AWS CLI

Esempio 1: come creare una tabella con tag

L'esempio `create-table` seguente utilizza gli attributi e lo schema di chiavi specificati per creare una tabella denominata `MusicCollection`. Questa tabella utilizza la velocità effettiva assegnata ed è crittografata a riposo utilizzando la CMK di AWS proprietà predefinita. Il comando applica inoltre un tag alla tabella, con una chiave `Owner` e un valore `blueTeam`.

```
aws dynamodb create-table \
    --table-name MusicCollection \
    --attribute-
definitions AttributeName=Artist,AttributeType=S AttributeName=SongTitle,AttributeType=S
 \
    --key-
schema AttributeName=Artist,KeyType=HASH AttributeName=SongTitle,KeyType=RANGE \
    --provisioned-throughput ReadCapacityUnits=5,WriteCapacityUnits=5 \
    --tags Key=Owner,Value=blueTeam
```

Output:

```
{
    "TableDescription": {
        "AttributeDefinitions": [
            {
                "AttributeName": "Artist",
                "AttributeType": "S"
            },
            {
                "AttributeName": "SongTitle",
                "AttributeType": "S"
            }
        ],
        "ProvisionedThroughput": {
            "NumberOfDecreasesToday": 0,
            "WriteCapacityUnits": 5,
            "ReadCapacityUnits": 5
        },
        "TableSizeBytes": 0,
        "TableName": "MusicCollection",
        "TableStatus": "CREATING",
        "KeySchema": [
            {
                "KeyType": "HASH",
                "AttributeName": "Artist"
            },
            {
                "KeyType": "RANGE",
                "AttributeName": "SongTitle"
            }
        ],
        "ItemCount": 0,
```

```
        "CreationDateTime": "2020-05-26T16:04:41.627000-07:00",
        "TableArn": "arn:aws:dynamodb:us-west-2:123456789012:table/MusicCollection",
        "TableId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
    }
}
```

Per ulteriori informazioni, consulta Operazioni di base per le tabelle nella Guida per gli sviluppatori di Amazon DynamoDB.

Esempio 2: come creare una tabella in modalità on demand

L'esempio seguente crea una tabella denominata `MusicCollection` utilizzando la modalità on demand, anziché la modalità basata su throughput allocato. Ciò risulta utile per le tabelle con carichi di lavoro imprevedibili.

```
aws dynamodb create-table \
    --table-name MusicCollection \
    --attribute-
definitions AttributeName=Artist,AttributeType=S AttributeName=SongTitle,AttributeType=S
 \
    --key-
schema AttributeName=Artist,KeyType=HASH AttributeName=SongTitle,KeyType=RANGE \
    --billing-mode PAY_PER_REQUEST
```

Output:

```
{
    "TableDescription": {
        "AttributeDefinitions": [
            {
                "AttributeName": "Artist",
                "AttributeType": "S"
            },
            {
                "AttributeName": "SongTitle",
                "AttributeType": "S"
            }
        ],
        "TableName": "MusicCollection",
        "KeySchema": [
            {
                "AttributeName": "Artist",
                "KeyType": "HASH"
```

```
            },
            {
                "AttributeName": "SongTitle",
                "KeyType": "RANGE"
            }
        ],
        "TableStatus": "CREATING",
        "CreationDateTime": "2020-05-27T11:44:10.807000-07:00",
        "ProvisionedThroughput": {
            "NumberOfDecreasesToday": 0,
            "ReadCapacityUnits": 0,
            "WriteCapacityUnits": 0
        },
        "TableSizeBytes": 0,
        "ItemCount": 0,
        "TableArn": "arn:aws:dynamodb:us-west-2:123456789012:table/MusicCollection",
        "TableId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
        "BillingModeSummary": {
            "BillingMode": "PAY_PER_REQUEST"
        }
    }
}
```

Per ulteriori informazioni, consulta Operazioni di base per le tabelle nella Guida per gli sviluppatori di Amazon DynamoDB.

Esempio 3: come creare una tabella e crittografarla con una chiave gestita dal cliente (CMK).

L'esempio seguente crea una tabella denominata `MusicCollection` e la crittografa utilizzando una chiave gestita dal cliente (CMK).

```
aws dynamodb create-table \
    --table-name MusicCollection \
    --attribute-
definitions AttributeName=Artist,AttributeType=S AttributeName=SongTitle,AttributeType=S
 \
    --key-
schema AttributeName=Artist,KeyType=HASH AttributeName=SongTitle,KeyType=RANGE \
    --provisioned-throughput ReadCapacityUnits=5,WriteCapacityUnits=5 \
    --sse-specification Enabled=true,SSEType=KMS,KMSMasterKeyId=abcd1234-abcd-1234-
a123-ab1234a1b234
```

Output:

```
{
    "TableDescription": {
        "AttributeDefinitions": [
            {
                "AttributeName": "Artist",
                "AttributeType": "S"
            },
            {
                "AttributeName": "SongTitle",
                "AttributeType": "S"
            }
        ],
        "TableName": "MusicCollection",
        "KeySchema": [
            {
                "AttributeName": "Artist",
                "KeyType": "HASH"
            },
            {
                "AttributeName": "SongTitle",
                "KeyType": "RANGE"
            }
        ],
        "TableStatus": "CREATING",
        "CreationDateTime": "2020-05-27T11:12:16.431000-07:00",
        "ProvisionedThroughput": {
            "NumberOfDecreasesToday": 0,
            "ReadCapacityUnits": 5,
            "WriteCapacityUnits": 5
        },
        "TableSizeBytes": 0,
        "ItemCount": 0,
        "TableArn": "arn:aws:dynamodb:us-west-2:123456789012:table/MusicCollection",
        "TableId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
        "SSEDescription": {
            "Status": "ENABLED",
            "SSEType": "KMS",
            "KMSMasterKeyArn": "arn:aws:kms:us-west-2:123456789012:key/abcd1234-
abcd-1234-a123-ab1234a1b234"
        }
    }
}
```

Per ulteriori informazioni, consulta Operazioni di base per le tabelle nella Guida per gli sviluppatori di Amazon DynamoDB.

Esempio 4: come creare una tabella con un indice secondario locale

Nell'esempio seguente vengono utilizzati gli attributi e lo schema di chiavi specificati per creare una tabella denominata `MusicCollection` con un indice secondario locale denominato `AlbumTitleIndex`.

```
aws dynamodb create-table \
    --table-name MusicCollection \
    --attribute-
definitions AttributeName=Artist,AttributeType=S AttributeName=SongTitle,AttributeType=S Att
 \
    --key-
schema AttributeName=Artist,KeyType=HASH AttributeName=SongTitle,KeyType=RANGE \
    --provisioned-throughput ReadCapacityUnits=10,WriteCapacityUnits=5 \
    --local-secondary-indexes \
        "[
            {
                \"IndexName\": \"AlbumTitleIndex\",
                \"KeySchema\": [
                    {\"AttributeName\": \"Artist\",\"KeyType\":\"HASH\"},
                    {\"AttributeName\": \"AlbumTitle\",\"KeyType\":\"RANGE\"}
                ],
                \"Projection\": {
                    \"ProjectionType\": \"INCLUDE\",
                    \"NonKeyAttributes\": [\"Genre\", \"Year\"]
                }
            }
        ]"
```

Output:

```
{
    "TableDescription": {
        "AttributeDefinitions": [
            {
                "AttributeName": "AlbumTitle",
                "AttributeType": "S"
            },
            {
                "AttributeName": "Artist",
```

```
                "AttributeType": "S"
            },
            {
                "AttributeName": "SongTitle",
                "AttributeType": "S"
            }
        ],
        "TableName": "MusicCollection",
        "KeySchema": [
            {
                "AttributeName": "Artist",
                "KeyType": "HASH"
            },
            {
                "AttributeName": "SongTitle",
                "KeyType": "RANGE"
            }
        ],
        "TableStatus": "CREATING",
        "CreationDateTime": "2020-05-26T15:59:49.473000-07:00",
        "ProvisionedThroughput": {
            "NumberOfDecreasesToday": 0,
            "ReadCapacityUnits": 10,
            "WriteCapacityUnits": 5
        },
        "TableSizeBytes": 0,
        "ItemCount": 0,
        "TableArn": "arn:aws:dynamodb:us-west-2:123456789012:table/MusicCollection",
        "TableId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
        "LocalSecondaryIndexes": [
            {
                "IndexName": "AlbumTitleIndex",
                "KeySchema": [
                    {
                        "AttributeName": "Artist",
                        "KeyType": "HASH"
                    },
                    {
                        "AttributeName": "AlbumTitle",
                        "KeyType": "RANGE"
                    }
                ],
                "Projection": {
                    "ProjectionType": "INCLUDE",
```

```
                    "NonKeyAttributes": [
                        "Genre",
                        "Year"
                    ]
                },
                "IndexSizeBytes": 0,
                "ItemCount": 0,
                "IndexArn": "arn:aws:dynamodb:us-west-2:123456789012:table/
MusicCollection/index/AlbumTitleIndex"
            }
        ]
    }
}
```

Per ulteriori informazioni, consulta Operazioni di base per le tabelle nella Guida per gli sviluppatori di Amazon DynamoDB.

Esempio 5: come creare una tabella con un indice secondario globale

L'esempio seguente crea una tabella denominata `GameScores` con un indice secondario globale denominato `GameTitleIndex`. La tabella di base ha una chiave di partizione di `UserId` e una chiave di ordinamento di `GameTitle`, permettendo di trovare il miglior punteggio di un singolo utente per un gioco specifico in modo efficiente, mentre il GSI ha una chiave di partizione di `GameTitle` e una chiave di ordinamento di `TopScore`, permettendo di trovare rapidamente il punteggio più alto complessivo per un determinato gioco.

```
aws dynamodb create-table \
    --table-name GameScores \
    --attribute-
definitions AttributeName=UserId,AttributeType=S AttributeName=GameTitle,AttributeType=S Att
 \
    --key-schema AttributeName=UserId,KeyType=HASH \
                 AttributeName=GameTitle,KeyType=RANGE \
    --provisioned-throughput ReadCapacityUnits=10,WriteCapacityUnits=5 \
    --global-secondary-indexes \
        "[
            {
                \"IndexName\": \"GameTitleIndex\",
                \"KeySchema\": [
                    {\"AttributeName\":\"GameTitle\",\"KeyType\":\"HASH\"},
                    {\"AttributeName\":\"TopScore\",\"KeyType\":\"RANGE\"}
                ],
```

```
                \"Projection\": {
                    \"ProjectionType\":\"INCLUDE\",
                    \"NonKeyAttributes\":[\"UserId\"]
                },
                \"ProvisionedThroughput\": {
                    \"ReadCapacityUnits\": 10,
                    \"WriteCapacityUnits\": 5
                }
            }
        ]"
```

Output:

```
{
    "TableDescription": {
        "AttributeDefinitions": [
            {
                "AttributeName": "GameTitle",
                "AttributeType": "S"
            },
            {
                "AttributeName": "TopScore",
                "AttributeType": "N"
            },
            {
                "AttributeName": "UserId",
                "AttributeType": "S"
            }
        ],
        "TableName": "GameScores",
        "KeySchema": [
            {
                "AttributeName": "UserId",
                "KeyType": "HASH"
            },
            {
                "AttributeName": "GameTitle",
                "KeyType": "RANGE"
            }
        ],
        "TableStatus": "CREATING",
        "CreationDateTime": "2020-05-26T17:28:15.602000-07:00",
        "ProvisionedThroughput": {
```

```
            "NumberOfDecreasesToday": 0,
            "ReadCapacityUnits": 10,
            "WriteCapacityUnits": 5
        },
        "TableSizeBytes": 0,
        "ItemCount": 0,
        "TableArn": "arn:aws:dynamodb:us-west-2:123456789012:table/GameScores",
        "TableId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
        "GlobalSecondaryIndexes": [
            {
                "IndexName": "GameTitleIndex",
                "KeySchema": [
                    {
                        "AttributeName": "GameTitle",
                        "KeyType": "HASH"
                    },
                    {
                        "AttributeName": "TopScore",
                        "KeyType": "RANGE"
                    }
                ],
                "Projection": {
                    "ProjectionType": "INCLUDE",
                    "NonKeyAttributes": [
                        "UserId"
                    ]
                },
                "IndexStatus": "CREATING",
                "ProvisionedThroughput": {
                    "NumberOfDecreasesToday": 0,
                    "ReadCapacityUnits": 10,
                    "WriteCapacityUnits": 5
                },
                "IndexSizeBytes": 0,
                "ItemCount": 0,
                "IndexArn": "arn:aws:dynamodb:us-west-2:123456789012:table/
GameScores/index/GameTitleIndex"
            }
        ]
    }
}
```

Per ulteriori informazioni, consulta Operazioni di base per le tabelle nella Guida per gli sviluppatori di Amazon DynamoDB.

Esempio 6: come creare una tabella con più indici secondari globali contemporaneamente

Nell'esempio seguente viene creata una tabella denominata `GameScores` con due indici secondari globali. Gli schemi GSI vengono passati tramite un file, anziché nella riga di comando.

```
aws dynamodb create-table \
    --table-name GameScores \
    --attribute-
definitions AttributeName=UserId,AttributeType=S AttributeName=GameTitle,AttributeType=S Att
 \
    --key-
schema AttributeName=UserId,KeyType=HASH AttributeName=GameTitle,KeyType=RANGE \
    --provisioned-throughput ReadCapacityUnits=10,WriteCapacityUnits=5 \
    --global-secondary-indexes file://gsi.json
```

Contenuto di `gsi.json`:

```
[
    {
        "IndexName": "GameTitleIndex",
        "KeySchema": [
            {
                "AttributeName": "GameTitle",
                "KeyType": "HASH"
            },
            {
                "AttributeName": "TopScore",
                "KeyType": "RANGE"
            }
        ],
        "Projection": {
            "ProjectionType": "ALL"
        },
        "ProvisionedThroughput": {
            "ReadCapacityUnits": 10,
            "WriteCapacityUnits": 5
        }
    },
    {
        "IndexName": "GameDateIndex",
```

```
        "KeySchema": [
            {
                "AttributeName": "GameTitle",
                "KeyType": "HASH"
            },
            {
                "AttributeName": "Date",
                "KeyType": "RANGE"
            }
        ],
        "Projection": {
            "ProjectionType": "ALL"
        },
        "ProvisionedThroughput": {
            "ReadCapacityUnits": 5,
            "WriteCapacityUnits": 5
        }
    }
]
```

Output:

```
{
    "TableDescription": {
        "AttributeDefinitions": [
            {
                "AttributeName": "Date",
                "AttributeType": "S"
            },
            {
                "AttributeName": "GameTitle",
                "AttributeType": "S"
            },
            {
                "AttributeName": "TopScore",
                "AttributeType": "N"
            },
            {
                "AttributeName": "UserId",
                "AttributeType": "S"
            }
        ],
        "TableName": "GameScores",
```

```
        "KeySchema": [
            {
                "AttributeName": "UserId",
                "KeyType": "HASH"
            },
            {
                "AttributeName": "GameTitle",
                "KeyType": "RANGE"
            }
        ],
        "TableStatus": "CREATING",
        "CreationDateTime": "2020-08-04T16:40:55.524000-07:00",
        "ProvisionedThroughput": {
            "NumberOfDecreasesToday": 0,
            "ReadCapacityUnits": 10,
            "WriteCapacityUnits": 5
        },
        "TableSizeBytes": 0,
        "ItemCount": 0,
        "TableArn": "arn:aws:dynamodb:us-west-2:123456789012:table/GameScores",
        "TableId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
        "GlobalSecondaryIndexes": [
            {
                "IndexName": "GameTitleIndex",
                "KeySchema": [
                    {
                        "AttributeName": "GameTitle",
                        "KeyType": "HASH"
                    },
                    {
                        "AttributeName": "TopScore",
                        "KeyType": "RANGE"
                    }
                ],
                "Projection": {
                    "ProjectionType": "ALL"
                },
                "IndexStatus": "CREATING",
                "ProvisionedThroughput": {
                    "NumberOfDecreasesToday": 0,
                    "ReadCapacityUnits": 10,
                    "WriteCapacityUnits": 5
                },
                "IndexSizeBytes": 0,
```

```
                "ItemCount": 0,
                "IndexArn": "arn:aws:dynamodb:us-west-2:123456789012:table/
GameScores/index/GameTitleIndex"
            },
            {
                "IndexName": "GameDateIndex",
                "KeySchema": [
                    {
                        "AttributeName": "GameTitle",
                        "KeyType": "HASH"
                    },
                    {
                        "AttributeName": "Date",
                        "KeyType": "RANGE"
                    }
                ],
                "Projection": {
                    "ProjectionType": "ALL"
                },
                "IndexStatus": "CREATING",
                "ProvisionedThroughput": {
                    "NumberOfDecreasesToday": 0,
                    "ReadCapacityUnits": 5,
                    "WriteCapacityUnits": 5
                },
                "IndexSizeBytes": 0,
                "ItemCount": 0,
                "IndexArn": "arn:aws:dynamodb:us-west-2:123456789012:table/
GameScores/index/GameDateIndex"
            }
        ]
    }
}
```

Per ulteriori informazioni, consulta Operazioni di base per le tabelle nella Guida per gli sviluppatori di Amazon DynamoDB.

Esempio 7: come creare una tabella con i flussi abilitati

L'esempio seguente crea una tabella denominata `GameScores` con i flussi DynamoDB abilitati. Le immagini vecchie e nuove di ciascun elemento verranno scritte nel flusso.

```
aws dynamodb create-table \
```

```
    --table-name GameScores \
    --attribute-
definitions AttributeName=UserId,AttributeType=S AttributeName=GameTitle,AttributeType=S
 \
    --key-
schema AttributeName=UserId,KeyType=HASH AttributeName=GameTitle,KeyType=RANGE \
    --provisioned-throughput ReadCapacityUnits=10,WriteCapacityUnits=5 \
    --stream-specification StreamEnabled=TRUE,StreamViewType=NEW_AND_OLD_IMAGES
```

Output:

```
{
    "TableDescription": {
        "AttributeDefinitions": [
            {
                "AttributeName": "GameTitle",
                "AttributeType": "S"
            },
            {
                "AttributeName": "UserId",
                "AttributeType": "S"
            }
        ],
        "TableName": "GameScores",
        "KeySchema": [
            {
                "AttributeName": "UserId",
                "KeyType": "HASH"
            },
            {
                "AttributeName": "GameTitle",
                "KeyType": "RANGE"
            }
        ],
        "TableStatus": "CREATING",
        "CreationDateTime": "2020-05-27T10:49:34.056000-07:00",
        "ProvisionedThroughput": {
            "NumberOfDecreasesToday": 0,
            "ReadCapacityUnits": 10,
            "WriteCapacityUnits": 5
        },
        "TableSizeBytes": 0,
        "ItemCount": 0,
```

```
        "TableArn": "arn:aws:dynamodb:us-west-2:123456789012:table/GameScores",
        "TableId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
        "StreamSpecification": {
            "StreamEnabled": true,
            "StreamViewType": "NEW_AND_OLD_IMAGES"
        },
        "LatestStreamLabel": "2020-05-27T17:49:34.056",
        "LatestStreamArn": "arn:aws:dynamodb:us-west-2:123456789012:table/
GameScores/stream/2020-05-27T17:49:34.056"
    }
}
```

Per ulteriori informazioni, consulta Operazioni di base per le tabelle nella Guida per gli sviluppatori di Amazon DynamoDB.

Esempio 8: come creare una tabella con il flusso KEYS_ONLY abilitato

L'esempio seguente crea una tabella denominata `GameScores` con i flussi DynamoDB abilitati. Solo gli attributi chiave dell'elemento modificato vengono scritti nel flusso.

```
aws dynamodb create-table \
    --table-name GameScores \
    --attribute-
definitions AttributeName=UserId,AttributeType=S AttributeName=GameTitle,AttributeType=S
 \
    --key-
schema AttributeName=UserId,KeyType=HASH AttributeName=GameTitle,KeyType=RANGE \
    --provisioned-throughput ReadCapacityUnits=10,WriteCapacityUnits=5 \
    --stream-specification StreamEnabled=TRUE,StreamViewType=KEYS_ONLY
```

Output:

```
{
    "TableDescription": {
        "AttributeDefinitions": [
            {
                "AttributeName": "GameTitle",
                "AttributeType": "S"
            },
            {
                "AttributeName": "UserId",
                "AttributeType": "S"
            }
```

```
        ],
        "TableName": "GameScores",
        "KeySchema": [
            {
                "AttributeName": "UserId",
                "KeyType": "HASH"
            },
            {
                "AttributeName": "GameTitle",
                "KeyType": "RANGE"
            }
        ],
        "TableStatus": "CREATING",
        "CreationDateTime": "2023-05-25T18:45:34.140000+00:00",
        "ProvisionedThroughput": {
            "NumberOfDecreasesToday": 0,
            "ReadCapacityUnits": 10,
            "WriteCapacityUnits": 5
        },
        "TableSizeBytes": 0,
        "ItemCount": 0,
        "TableArn": "arn:aws:dynamodb:us-west-2:123456789012:table/GameScores",
        "TableId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
        "StreamSpecification": {
            "StreamEnabled": true,
            "StreamViewType": "KEYS_ONLY"
        },
        "LatestStreamLabel": "2023-05-25T18:45:34.140",
        "LatestStreamArn": "arn:aws:dynamodb:us-west-2:123456789012:table/
GameScores/stream/2023-05-25T18:45:34.140",
        "DeletionProtectionEnabled": false
    }
}
```

Per ulteriori informazioni, consulta Acquisizione dei dati di modifica per DynamoDB Streams nella Guida per gli sviluppatori di Amazon DynamoDB.

Esempio 9: come creare una tabella con la classe Accesso infrequente standard

L'esempio seguente crea una tabella denominata `GameScores` e assegna la classe di tabella Accesso Infrequente Amazon DynamoDB Standard (AI Amazon DynamoDB Standard). Questa classe di tabelle è ottimizzata a livello di archiviazione in quanto l'archiviazione rappresenta il costo principale.

```
aws dynamodb create-table \
    --table-name GameScores \
    --attribute-
definitions AttributeName=UserId,AttributeType=S AttributeName=GameTitle,AttributeType=S
 \
    --key-
schema AttributeName=UserId,KeyType=HASH AttributeName=GameTitle,KeyType=RANGE \
    --provisioned-throughput ReadCapacityUnits=10,WriteCapacityUnits=5 \
    --table-class STANDARD_INFREQUENT_ACCESS
```

Output:

```
{
    "TableDescription": {
        "AttributeDefinitions": [
            {
                "AttributeName": "GameTitle",
                "AttributeType": "S"
            },
            {
                "AttributeName": "UserId",
                "AttributeType": "S"
            }
        ],
        "TableName": "GameScores",
        "KeySchema": [
            {
                "AttributeName": "UserId",
                "KeyType": "HASH"
            },
            {
                "AttributeName": "GameTitle",
                "KeyType": "RANGE"
            }
        ],
        "TableStatus": "CREATING",
        "CreationDateTime": "2023-05-25T18:33:07.581000+00:00",
        "ProvisionedThroughput": {
            "NumberOfDecreasesToday": 0,
            "ReadCapacityUnits": 10,
            "WriteCapacityUnits": 5
        },
        "TableSizeBytes": 0,
```

```
        "ItemCount": 0,
        "TableArn": "arn:aws:dynamodb:us-west-2:123456789012:table/GameScores",
        "TableId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
        "TableClassSummary": {
            "TableClass": "STANDARD_INFREQUENT_ACCESS"
        },
        "DeletionProtectionEnabled": false
    }
}
```

Per ulteriori informazioni, consulta Classi di tabelle nella Guida per gli sviluppatori di Amazon DynamoDB.

Esempio 10: come creare una tabella con l'opzione Protezione da eliminazione abilitata

L'esempio seguente crea una tabella denominata `GameScores` e abilita la protezione da eliminazione.

```
aws dynamodb create-table \
    --table-name GameScores \
    --attribute-
definitions AttributeName=UserId,AttributeType=S AttributeName=GameTitle,AttributeType=S
 \
    --key-
schema AttributeName=UserId,KeyType=HASH AttributeName=GameTitle,KeyType=RANGE \
    --provisioned-throughput ReadCapacityUnits=10,WriteCapacityUnits=5 \
    --deletion-protection-enabled
```

Output:

```
{
    "TableDescription": {
        "AttributeDefinitions": [
            {
                "AttributeName": "GameTitle",
                "AttributeType": "S"
            },
            {
                "AttributeName": "UserId",
                "AttributeType": "S"
            }
        ],
        "TableName": "GameScores",
```

```
        "KeySchema": [
            {
                "AttributeName": "UserId",
                "KeyType": "HASH"
            },
            {
                "AttributeName": "GameTitle",
                "KeyType": "RANGE"
            }
        ],
        "TableStatus": "CREATING",
        "CreationDateTime": "2023-05-25T23:02:17.093000+00:00",
        "ProvisionedThroughput": {
            "NumberOfDecreasesToday": 0,
            "ReadCapacityUnits": 10,
            "WriteCapacityUnits": 5
        },
        "TableSizeBytes": 0,
        "ItemCount": 0,
        "TableArn": "arn:aws:dynamodb:us-west-2:123456789012:table/GameScores",
        "TableId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
        "DeletionProtectionEnabled": true
    }
}
```

Per ulteriori informazioni, consulta Uso della protezione da eliminazione nella Guida per gli sviluppatori di Amazon DynamoDB.

- Per i dettagli sull'API, consulta Command CreateTableReference AWS CLI .

`delete-backup`

Il seguente esempio di codice mostra come utilizzare`delete-backup`.

AWS CLI

Come eliminare un backup DynamoDB esistente

L'esempio `delete-backup` seguente elimina il backup esistente specificato.

```
aws dynamodb delete-backup \
    --backup-arn arn:aws:dynamodb:us-west-2:123456789012:table/MusicCollection/
backup/01576616366715-b4e58d3a
```

Output:

```
{
    "BackupDescription": {
        "BackupDetails": {
            "BackupArn": "arn:aws:dynamodb:us-west-2:123456789012:table/
MusicCollection/backup/01576616366715-b4e58d3a",
            "BackupName": "MusicCollectionBackup",
            "BackupSizeBytes": 0,
            "BackupStatus": "DELETED",
            "BackupType": "USER",
            "BackupCreationDateTime": 1576616366.715
        },
        "SourceTableDetails": {
            "TableName": "MusicCollection",
            "TableId": "b0c04bcc-309b-4352-b2ae-9088af169fe2",
            "TableArn": "arn:aws:dynamodb:us-west-2:123456789012:table/
MusicCollection",
            "TableSizeBytes": 0,
            "KeySchema": [
                {
                    "AttributeName": "Artist",
                    "KeyType": "HASH"
                },
                {
                    "AttributeName": "SongTitle",
                    "KeyType": "RANGE"
                }
            ],
            "TableCreationDateTime": 1576615228.571,
            "ProvisionedThroughput": {
                "ReadCapacityUnits": 5,
                "WriteCapacityUnits": 5
            },
            "ItemCount": 0,
            "BillingMode": "PROVISIONED"
        },
        "SourceTableFeatureDetails": {}
    }
}
```

Per ulteriori informazioni, consulta Backup e ripristino on demand per DynamoDB nella Guida per gli sviluppatori di Amazon DynamoDB.

- Per i dettagli sull'API, consulta DeleteBackup AWS CLICommand Reference.

`delete-item`

Il seguente esempio di codice mostra come utilizzare`delete-item`.

AWS CLI

Esempio 1: come eliminare un elemento

L'esempio `delete-item` seguente elimina un elemento dalla tabella `MusicCollection` e richiede i dettagli sull'elemento eliminato e sulla capacità utilizzata dalla richiesta.

```
aws dynamodb delete-item \
    --table-name MusicCollection \
    --key file://key.json \
    --return-values ALL_OLD \
    --return-consumed-capacity TOTAL \
    --return-item-collection-metrics SIZE
```

Contenuto di `key.json`:

```
{
    "Artist": {"S": "No One You Know"},
    "SongTitle": {"S": "Scared of My Shadow"}
}
```

Output:

```
{
    "Attributes": {
        "AlbumTitle": {
            "S": "Blue Sky Blues"
        },
        "Artist": {
            "S": "No One You Know"
        },
        "SongTitle": {
            "S": "Scared of My Shadow"
        }
    },
```

```
    "ConsumedCapacity": {
        "TableName": "MusicCollection",
        "CapacityUnits": 2.0
    },
    "ItemCollectionMetrics": {
        "ItemCollectionKey": {
            "Artist": {
                "S": "No One You Know"
            }
        },
        "SizeEstimateRangeGB": [
            0.0,
            1.0
        ]
    }
}
```

Per ulteriori informazioni, consulta Scrittura di un elemento nella Guida per gli sviluppatori di Amazon DynamoDB.

Esempio 2: come eliminare un elemento in base a una condizione

L'esempio seguente elimina un elemento dalla tabella `ProductCatalog` solo se `ProductCategory` è `Sporting Goods` o `Gardening Supplies` e se il relativo prezzo è compreso tra 500 e 600. Restituisce i dettagli sull'elemento eliminato.

```
aws dynamodb delete-item \
    --table-name ProductCatalog \
    --key '{"Id":{"N":"456"}}' \
    --condition-expression "(ProductCategory IN (:cat1, :cat2)) and (#P between :lo
 and :hi)" \
    --expression-attribute-names file://names.json \
    --expression-attribute-values file://values.json \
    --return-values ALL_OLD
```

Contenuto di `names.json`:

```
{
    "#P": "Price"
}
```

Contenuto di `values.json`:

```
{
    ":cat1": {"S": "Sporting Goods"},
    ":cat2": {"S": "Gardening Supplies"},
    ":lo": {"N": "500"},
    ":hi": {"N": "600"}
}
```

Output:

```
{
    "Attributes": {
        "Id": {
            "N": "456"
        },
        "Price": {
            "N": "550"
        },
        "ProductCategory": {
            "S": "Sporting Goods"
        }
    }
}
```

Per ulteriori informazioni, consulta Scrittura di un elemento nella Guida per gli sviluppatori di Amazon DynamoDB.

- Per i dettagli sull'API, consulta DeleteItem AWS CLICommand Reference.

`delete-table`

Il seguente esempio di codice mostra come utilizzare`delete-table`.

AWS CLI

Come eliminare una tabella

L'esempio `delete-table` seguente elimina la tabella `MusicCollection`.

```
aws dynamodb delete-table \
    --table-name MusicCollection
```

Output:

```
{
    "TableDescription": {
        "TableStatus": "DELETING",
        "TableSizeBytes": 0,
        "ItemCount": 0,
        "TableName": "MusicCollection",
        "ProvisionedThroughput": {
            "NumberOfDecreasesToday": 0,
            "WriteCapacityUnits": 5,
            "ReadCapacityUnits": 5
        }
    }
}
```

Per ulteriori informazioni, consulta Eliminazione di una tabella nella Guida per gli sviluppatori di Amazon DynamoDB.

- Per i dettagli sull'API, consulta DeleteTable AWS CLICommand Reference.

`describe-backup`

Il seguente esempio di codice mostra come utilizzare`describe-backup`.

AWS CLI

Come ottenere informazioni su un backup esistente di una tabella

L'esempio `describe-backup` seguente mostra le informazioni relative al backup esistente specificato.

```
aws dynamodb describe-backup \
    --backup-arn arn:aws:dynamodb:us-west-2:123456789012:table/MusicCollection/
backup/01576616366715-b4e58d3a
```

Output:

```
{
    "BackupDescription": {
        "BackupDetails": {
            "BackupArn": "arn:aws:dynamodb:us-west-2:123456789012:table/
MusicCollection/backup/01576616366715-b4e58d3a",
```

```
            "BackupName": "MusicCollectionBackup",
            "BackupSizeBytes": 0,
            "BackupStatus": "AVAILABLE",
            "BackupType": "USER",
            "BackupCreationDateTime": 1576616366.715
        },
        "SourceTableDetails": {
            "TableName": "MusicCollection",
            "TableId": "b0c04bcc-309b-4352-b2ae-9088af169fe2",
            "TableArn": "arn:aws:dynamodb:us-west-2:123456789012:table/
MusicCollection",
            "TableSizeBytes": 0,
            "KeySchema": [
                {
                    "AttributeName": "Artist",
                    "KeyType": "HASH"
                },
                {
                    "AttributeName": "SongTitle",
                    "KeyType": "RANGE"
                }
            ],
            "TableCreationDateTime": 1576615228.571,
            "ProvisionedThroughput": {
                "ReadCapacityUnits": 5,
                "WriteCapacityUnits": 5
            },
            "ItemCount": 0,
            "BillingMode": "PROVISIONED"
        },
        "SourceTableFeatureDetails": {}
    }
}
```

Per ulteriori informazioni, consulta Backup e ripristino on demand per DynamoDB nella Guida per gli sviluppatori di Amazon DynamoDB.

- Per i dettagli sull'API, consulta DescribeBackup AWS CLICommand Reference.

`describe-continuous-backups`

Il seguente esempio di codice mostra come utilizzare`describe-continuous-backups`.

AWS CLI

Come ottenere informazioni sui backup continui per una tabella DynamoDB

L'esempio `describe-continuous-backups` seguente mostra i dettagli sulle impostazioni dei backup continui per la tabella `MusicCollection`.

```
aws dynamodb describe-continuous-backups \
    --table-name MusicCollection
```

Output:

```
{
    "ContinuousBackupsDescription": {
        "ContinuousBackupsStatus": "ENABLED",
        "PointInTimeRecoveryDescription": {
            "PointInTimeRecoveryStatus": "DISABLED"
        }
    }
}
```

Per ulteriori informazioni, consulta Point-in-Time Recovery for DynamoDB nella Amazon DynamoDB Developer Guide.

- Per i dettagli sull'API, consulta DescribeContinuousBackupsCommand Reference.AWS CLI

`describe-contributor-insights`

Il seguente esempio di codice mostra come utilizzare`describe-contributor-insights`.

AWS CLI

Come visualizzare le impostazioni di Contributor Insights per una tabella DynamoDB

L'esempio `describe-contributor-insights` seguente mostra le impostazioni di Contributor Insights per la tabella `MusicCollection` e l'indice secondario globale `AlbumTitle-index`.

```
aws dynamodb describe-contributor-insights \
    --table-name MusicCollection \
    --index-name AlbumTitle-index
```

Output:

```
{
    "TableName": "MusicCollection",
    "IndexName": "AlbumTitle-index",
    "ContributorInsightsRuleList": [
        "DynamoDBContributorInsights-PKC-MusicCollection-1576629651520",
        "DynamoDBContributorInsights-SKC-MusicCollection-1576629651520",
        "DynamoDBContributorInsights-PKT-MusicCollection-1576629651520",
        "DynamoDBContributorInsights-SKT-MusicCollection-1576629651520"
    ],
    "ContributorInsightsStatus": "ENABLED",
    "LastUpdateDateTime": 1576629654.78
}
```

Per ulteriori informazioni, consulta Analyzing Data Access Using CloudWatch Contributor Insights for DynamoDB nella Amazon DynamoDB Developer Guide.

- Per i dettagli sull'API, consulta Command Reference. DescribeContributorInsightsAWS CLI

`describe-endpoints`

Il seguente esempio di codice mostra come utilizzare`describe-endpoints`.

AWS CLI

Come visualizzare le informazioni sugli endpoint regionali

L'`describe-endpoints`esempio seguente mostra i dettagli sugli endpoint per la AWS regione corrente.

```
aws dynamodb describe-endpoints
```

Output:

```
{
    "Endpoints": [
        {
            "Address": "dynamodb.us-west-2.amazonaws.com",
            "CachePeriodInMinutes": 1440
        }
    ]
}
```

Per ulteriori informazioni, consulta Endpoint e quote di Amazon DynamoDB nella documentazione generale di riferimento di AWS .

- Per i dettagli sull'API, vedere DescribeEndpointsin AWS CLI Command Reference.

`describe-global-table-settings`

Il seguente esempio di codice mostra come utilizzare`describe-global-table-settings`.

AWS CLI

Come ottenere informazioni sulle impostazioni di una tabella globale DynamoDB

L'esempio `describe-global-table-settings` seguente mostra le impostazioni per la tabella globale `MusicCollection`.

```
aws dynamodb describe-global-table-settings \
    --global-table-name MusicCollection
```

Output:

```
{
    "GlobalTableName": "MusicCollection",
    "ReplicaSettings": [
        {
            "RegionName": "us-east-1",
            "ReplicaStatus": "ACTIVE",
            "ReplicaProvisionedReadCapacityUnits": 10,
            "ReplicaProvisionedReadCapacityAutoScalingSettings": {
                "AutoScalingDisabled": true
            },
            "ReplicaProvisionedWriteCapacityUnits": 5,
            "ReplicaProvisionedWriteCapacityAutoScalingSettings": {
                "AutoScalingDisabled": true
            }
        },
        {
            "RegionName": "us-east-2",
            "ReplicaStatus": "ACTIVE",
            "ReplicaProvisionedReadCapacityUnits": 10,
            "ReplicaProvisionedReadCapacityAutoScalingSettings": {
                "AutoScalingDisabled": true
```

```
            },
            "ReplicaProvisionedWriteCapacityUnits": 5,
            "ReplicaProvisionedWriteCapacityAutoScalingSettings": {
                "AutoScalingDisabled": true
            }
        }
    ]
}
```

Per ulteriori informazioni, consulta Tabelle globali DynamoDB nella Guida per gli sviluppatori di Amazon DynamoDB.

- Per i dettagli sull'API, consulta DescribeGlobalTableSettings AWS CLICommand Reference.

`describe-global-table`

Il seguente esempio di codice mostra come utilizzare`describe-global-table`.

AWS CLI

Come visualizzare le informazioni su una tabella globale DynamoDB

L'esempio `describe-global-table` seguente mostra i dettagli relativi alla tabella globale `MusicCollection`.

```
aws dynamodb describe-global-table \
    --global-table-name MusicCollection
```

Output:

```
{
    "GlobalTableDescription": {
        "ReplicationGroup": [
            {
                "RegionName": "us-east-2"
            },
            {
                "RegionName": "us-east-1"
            }
        ],
        "GlobalTableArn": "arn:aws:dynamodb::123456789012:global-table/
MusicCollection",
```

```
        "CreationDateTime": 1576625818.532,
        "GlobalTableStatus": "ACTIVE",
        "GlobalTableName": "MusicCollection"
    }
}
```

Per ulteriori informazioni, consulta Tabelle globali DynamoDB nella Guida per gli sviluppatori di Amazon DynamoDB.

- Per i dettagli sull'API, consulta DescribeGlobalTable AWS CLICommand Reference.

`describe-limits`

Il seguente esempio di codice mostra come utilizzare`describe-limits`.

AWS CLI

Come visualizzare i limiti di capacità con provisioning

L'`describe-limits`esempio seguente mostra i limiti di capacità prevista per il tuo account nella regione corrente. AWS

```
aws dynamodb describe-limits
```

Output:

```
{
    "AccountMaxReadCapacityUnits": 80000,
    "AccountMaxWriteCapacityUnits": 80000,
    "TableMaxReadCapacityUnits": 40000,
    "TableMaxWriteCapacityUnits": 40000
}
```

Per ulteriori informazioni, consulta Limiti in DynamoDB nella Guida per gli sviluppatori di Amazon DynamoDB.

- Per i dettagli sull'API, consulta Command DescribeLimitsReference AWS CLI .

`describe-table-replica-auto-scaling`

Il seguente esempio di codice mostra come utilizzare`describe-table-replica-auto-scaling`.

AWS CLI

Come visualizzare le impostazioni di dimensionamento automatico tra le repliche di una tabella globale

L'esempio `describe-table-replica-auto-scaling` seguente mostra le impostazioni di dimensionamento automatico tra le repliche della tabella globale `MusicCollection`.

```
aws dynamodb describe-table-replica-auto-scaling \
    --table-name MusicCollection
```

Output:

```
{
    "TableAutoScalingDescription": {
        "TableName": "MusicCollection",
        "TableStatus": "ACTIVE",
        "Replicas": [
            {
                "RegionName": "us-east-1",
                "GlobalSecondaryIndexes": [],
                "ReplicaProvisionedReadCapacityAutoScalingSettings": {
                    "MinimumUnits": 5,
                    "MaximumUnits": 40000,
                    "AutoScalingRoleArn": "arn:aws:iam::123456789012:role/
aws-service-role/dynamodb.application-autoscaling.amazonaws.com/
AWSServiceRoleForApplicationAutoScaling_DynamoDBTable",
                    "ScalingPolicies": [
                        {
                            "PolicyName": "DynamoDBReadCapacityUtilization:table/
MusicCollection",
                            "TargetTrackingScalingPolicyConfiguration": {
                                "TargetValue": 70.0
                            }
                        }
                    ]
                },
                "ReplicaProvisionedWriteCapacityAutoScalingSettings": {
                    "MinimumUnits": 5,
                    "MaximumUnits": 40000,
                    "AutoScalingRoleArn": "arn:aws:iam::123456789012:role/
aws-service-role/dynamodb.application-autoscaling.amazonaws.com/
AWSServiceRoleForApplicationAutoScaling_DynamoDBTable",
```

```
                    "ScalingPolicies": [
                        {
                            "PolicyName": "DynamoDBWriteCapacityUtilization:table/
MusicCollection",
                            "TargetTrackingScalingPolicyConfiguration": {
                                "TargetValue": 70.0
                            }
                        }
                    ]
                },
                "ReplicaStatus": "ACTIVE"
            },
            {
                "RegionName": "us-east-2",
                "GlobalSecondaryIndexes": [],
                "ReplicaProvisionedReadCapacityAutoScalingSettings": {
                    "MinimumUnits": 5,
                    "MaximumUnits": 40000,
                    "AutoScalingRoleArn": "arn:aws:iam::123456789012:role/
aws-service-role/dynamodb.application-autoscaling.amazonaws.com/
AWSServiceRoleForApplicationAutoScaling_DynamoDBTable",
                    "ScalingPolicies": [
                        {
                            "PolicyName": "DynamoDBReadCapacityUtilization:table/
MusicCollection",
                            "TargetTrackingScalingPolicyConfiguration": {
                                "TargetValue": 70.0
                            }
                        }
                    ]
                },
                "ReplicaProvisionedWriteCapacityAutoScalingSettings": {
                    "MinimumUnits": 5,
                    "MaximumUnits": 40000,
                    "AutoScalingRoleArn": "arn:aws:iam::123456789012:role/
aws-service-role/dynamodb.application-autoscaling.amazonaws.com/
AWSServiceRoleForApplicationAutoScaling_DynamoDBTable",
                    "ScalingPolicies": [
                        {
                            "PolicyName": "DynamoDBWriteCapacityUtilization:table/
MusicCollection",
                            "TargetTrackingScalingPolicyConfiguration": {
                                "TargetValue": 70.0
                            }
```

```
                        }
                    ]
                },
                "ReplicaStatus": "ACTIVE"
            }
        ]
    }
}
```

Per ulteriori informazioni, consulta Tabelle globali DynamoDB nella Guida per gli sviluppatori di Amazon DynamoDB.

- Per i dettagli sull'API, consulta DescribeTableReplicaAutoScaling AWS CLICommand Reference.

`describe-table`

Il seguente esempio di codice mostra come utilizzare`describe-table`.

AWS CLI

Come descrivere una tabella

L'esempio `describe-table` seguente descrive la tabella `MusicCollection`.

```
aws dynamodb describe-table \
    --table-name MusicCollection
```

Output:

```
{
    "Table": {
        "AttributeDefinitions": [
            {
                "AttributeName": "Artist",
                "AttributeType": "S"
            },
            {
                "AttributeName": "SongTitle",
                "AttributeType": "S"
            }
        ],
```

```
        "ProvisionedThroughput": {
            "NumberOfDecreasesToday": 0,
            "WriteCapacityUnits": 5,
            "ReadCapacityUnits": 5
        },
        "TableSizeBytes": 0,
        "TableName": "MusicCollection",
        "TableStatus": "ACTIVE",
        "KeySchema": [
            {
                "KeyType": "HASH",
                "AttributeName": "Artist"
            },
            {
                "KeyType": "RANGE",
                "AttributeName": "SongTitle"
            }
        ],
        "ItemCount": 0,
        "CreationDateTime": 1421866952.062
    }
}
```

Per ulteriori informazioni, consulta Descrizione di una tabella nella Guida per gli sviluppatori di Amazon DynamoDB.

- Per i dettagli sull'API, consulta DescribeTable AWS CLICommand Reference.

`describe-time-to-live`

Il seguente esempio di codice mostra come utilizzare`describe-time-to-live`.

AWS CLI

Come visualizzare le impostazioni Time to Live per una tabella

L'esempio `describe-time-to-live` seguente visualizza le impostazioni Time to Live per la tabella `MusicCollection`.

```
aws dynamodb describe-time-to-live \
    --table-name MusicCollection
```

Output:

```
{
    "TimeToLiveDescription": {
        "TimeToLiveStatus": "ENABLED",
        "AttributeName": "ttl"
    }
}
```

Per ulteriori informazioni, consulta Time to Live nella Guida per gli sviluppatori di Amazon DynamoDB.

- Per i dettagli sull'API, consulta DescribeTimeToLive AWS CLICommand Reference.

`get-item`

Il seguente esempio di codice mostra come utilizzare`get-item`.

AWS CLI

Esempio 1: come leggere un elemento in una tabella

L'esempio `get-item` seguente recupera un elemento dalla tabella `MusicCollection`. La tabella ha una chiave hash-and-range primaria (`ArtisteSongTitle`), quindi è necessario specificare entrambi questi attributi. Il comando richiede anche informazioni sulla capacità di scrittura utilizzata dall'operazione.

```
aws dynamodb get-item \
    --table-name MusicCollection \
    --key file://key.json \
    --return-consumed-capacity TOTAL
```

Contenuto di `key.json`:

```
{
    "Artist": {"S": "Acme Band"},
    "SongTitle": {"S": "Happy Day"}
}
```

Output:

```
{
    "Item": {
```

```
        "AlbumTitle": {
            "S": "Songs About Life"
        },
        "SongTitle": {
            "S": "Happy Day"
        },
        "Artist": {
            "S": "Acme Band"
        }
    },
    "ConsumedCapacity": {
        "TableName": "MusicCollection",
        "CapacityUnits": 0.5
    }
}
```

Per ulteriori informazioni, consulta Lettura di un elemento nella Guida per gli sviluppatori di Amazon DynamoDB.

Esempio 2: come leggere un elemento utilizzando una lettura consistente

L'esempio seguente recupera un elemento dalla tabella `MusicCollection` tramite operazioni a elevata consistenza di lettura.

```
aws dynamodb get-item \
    --table-name MusicCollection \
    --key file://key.json \
    --consistent-read \
    --return-consumed-capacity TOTAL
```

Contenuto di `key.json`:

```
{
    "Artist": {"S": "Acme Band"},
    "SongTitle": {"S": "Happy Day"}
}
```

Output:

```
{
    "Item": {
        "AlbumTitle": {
```

```
            "S": "Songs About Life"
        },
        "SongTitle": {
            "S": "Happy Day"
        },
        "Artist": {
            "S": "Acme Band"
        }
    },
    "ConsumedCapacity": {
        "TableName": "MusicCollection",
        "CapacityUnits": 1.0
    }
}
```

Per ulteriori informazioni, consulta Lettura di un elemento nella Guida per gli sviluppatori di Amazon DynamoDB.

Esempio 3: come recuperare attributi specifici di un elemento

L'esempio seguente utilizza un'espressione di proiezione per recuperare solo tre attributi dell'elemento desiderato.

```
aws dynamodb get-item \
    --table-name ProductCatalog \
    --key '{"Id": {"N": "102"}}' \
    --projection-expression "#T, #C, #P" \
    --expression-attribute-names file://names.json
```

Contenuto di `names.json`:

```
{
    "#T": "Title",
    "#C": "ProductCategory",
    "#P": "Price"
}
```

Output:

```
{
    "Item": {
        "Price": {
```

```
            "N": "20"
        },
        "Title": {
            "S": "Book 102 Title"
        },
        "ProductCategory": {
            "S": "Book"
        }
    }
}
```

Per ulteriori informazioni, consulta Lettura di un elemento nella Guida per gli sviluppatori di Amazon DynamoDB.

- Per i dettagli sull'API, consulta GetItem AWS CLICommand Reference.

`list-backups`

Il seguente esempio di codice mostra come utilizzare`list-backups`.

AWS CLI

Esempio 1: come elencare tutti i backup DynamoDB esistenti

L'esempio `list-backups` seguente elenca tutti i backup esistenti.

```
aws dynamodb list-backups
```

Output:

```
{
    "BackupSummaries": [
        {
            "TableName": "MusicCollection",
            "TableId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
            "TableArn": "arn:aws:dynamodb:us-west-2:123456789012:table/
MusicCollection",
            "BackupArn": "arn:aws:dynamodb:us-west-2:123456789012:table/
MusicCollection/backup/01234567890123-a1bcd234",
            "BackupName": "MusicCollectionBackup1",
            "BackupCreationDateTime": "2020-02-12T14:41:51.617000-08:00",
            "BackupStatus": "AVAILABLE",
            "BackupType": "USER",
```

```
            "BackupSizeBytes": 170
        },
        {
            "TableName": "MusicCollection",
            "TableId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
            "TableArn": "arn:aws:dynamodb:us-west-2:123456789012:table/
MusicCollection",
            "BackupArn": "arn:aws:dynamodb:us-west-2:123456789012:table/
MusicCollection/backup/01234567890123-b2abc345",
            "BackupName": "MusicCollectionBackup2",
            "BackupCreationDateTime": "2020-06-26T11:08:35.431000-07:00",
            "BackupStatus": "AVAILABLE",
            "BackupType": "USER",
            "BackupSizeBytes": 400
        }
    ]
}
```

Per ulteriori informazioni, consulta Backup e ripristino on demand per DynamoDB nella Guida per gli sviluppatori di Amazon DynamoDB.

Esempio 2: come elencare i backup creati dall'utente in un intervallo di tempo specifico

L'esempio seguente elenca solo i backup della tabella `MusicCollection` creati dall'utente (non quelli creati automaticamente da DynamoDB) con una data di creazione compresa tra il 1° gennaio 2020 e il 1° marzo 2020.

```
aws dynamodb list-backups \
    --table-name MusicCollection \
    --time-range-lower-bound 1577836800 \
    --time-range-upper-bound 1583020800 \
    --backup-type USER
```

Output:

```
{
    "BackupSummaries": [
        {
            "TableName": "MusicCollection",
            "TableId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
            "TableArn": "arn:aws:dynamodb:us-west-2:123456789012:table/
MusicCollection",
```

```
            "BackupArn": "arn:aws:dynamodb:us-west-2:123456789012:table/
MusicCollection/backup/01234567890123-a1bcd234",
            "BackupName": "MusicCollectionBackup1",
            "BackupCreationDateTime": "2020-02-12T14:41:51.617000-08:00",
            "BackupStatus": "AVAILABLE",
            "BackupType": "USER",
            "BackupSizeBytes": 170
        }
    ]
}
```

Per ulteriori informazioni, consulta Backup e ripristino on demand per DynamoDB nella Guida per gli sviluppatori di Amazon DynamoDB.

Esempio 3: come limitare le dimensioni della pagina

L'esempio seguente restituisce un elenco di tutti i backup esistenti, ma recupera solo un elemento in ogni chiamata, eseguendo più chiamate, se necessario, per ottenere l'intero elenco. La limitazione delle dimensioni della pagina è utile quando si eseguono comandi di tipo elenco su un numero elevato di risorse, il che può causare un errore di "timeout" quando si utilizza la dimensione di pagina predefinita pari a 1.000.

```
aws dynamodb list-backups \
    --page-size 1
```

Output:

```
{
    "BackupSummaries": [
        {
            "TableName": "MusicCollection",
            "TableId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
            "TableArn": "arn:aws:dynamodb:us-west-2:123456789012:table/
MusicCollection",
            "BackupArn": "arn:aws:dynamodb:us-west-2:123456789012:table/
MusicCollection/backup/01234567890123-a1bcd234",
            "BackupName": "MusicCollectionBackup1",
            "BackupCreationDateTime": "2020-02-12T14:41:51.617000-08:00",
            "BackupStatus": "AVAILABLE",
            "BackupType": "USER",
            "BackupSizeBytes": 170
        },
```

```
        {
            "TableName": "MusicCollection",
            "TableId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
            "TableArn": "arn:aws:dynamodb:us-west-2:123456789012:table/
MusicCollection",
            "BackupArn": "arn:aws:dynamodb:us-west-2:123456789012:table/
MusicCollection/backup/01234567890123-b2abc345",
            "BackupName": "MusicCollectionBackup2",
            "BackupCreationDateTime": "2020-06-26T11:08:35.431000-07:00",
            "BackupStatus": "AVAILABLE",
            "BackupType": "USER",
            "BackupSizeBytes": 400
        }
    ]
}
```

Per ulteriori informazioni, consulta Backup e ripristino on demand per DynamoDB nella Guida per gli sviluppatori di Amazon DynamoDB.

Esempio 4: come limitare il numero di elementi restituiti

L'esempio seguente limita il numero di righe restituite a 1. La risposta include un valore `NextToken` con cui recuperare la pagina di risultati successiva.

```
aws dynamodb list-backups \
    --max-items 1
```

Output:

```
{
    "BackupSummaries": [
        {
            "TableName": "MusicCollection",
            "TableId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
            "TableArn": "arn:aws:dynamodb:us-west-2:123456789012:table/
MusicCollection",
            "BackupArn": "arn:aws:dynamodb:us-west-2:123456789012:table/
MusicCollection/backup/01234567890123-a1bcd234",
            "BackupName": "MusicCollectionBackup1",
            "BackupCreationDateTime": "2020-02-12T14:41:51.617000-08:00",
            "BackupStatus": "AVAILABLE",
            "BackupType": "USER",
```

```
            "BackupSizeBytes": 170
        }
    ],
    "NextToken":
 "abCDeFGhiJKlmnOPqrSTuvwxYZ1aBCdEFghijK7LM51nOpqRSTuv3WxY3ZabC5dEFGhI2Jk3LmnoPQ6RST9"
}
```

Per ulteriori informazioni, consulta Backup e ripristino on demand per DynamoDB nella Guida per gli sviluppatori di Amazon DynamoDB.

Esempio 5: come recuperare la pagina di risultati successiva

Il comando seguente utilizza il valore `NextToken` di una precedente chiamata al comando `list-backups` per recuperare un'altra pagina di risultati. Poiché la risposta in questo caso non include un valore `NextToken`, ciò significa che è stata raggiunta la fine dei risultati.

```
aws dynamodb list-backups \
    --starting-
token abCDeFGhiJKlmnOPqrSTuvwxYZ1aBCdEFghijK7LM51nOpqRSTuv3WxY3ZabC5dEFGhI2Jk3LmnoPQ6RST9
```

Output

```
{
    "BackupSummaries": [
        {
            "TableName": "MusicCollection",
            "TableId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
            "TableArn": "arn:aws:dynamodb:us-west-2:123456789012:table/
MusicCollection",
            "BackupArn": "arn:aws:dynamodb:us-west-2:123456789012:table/
MusicCollection/backup/01234567890123-b2abc345",
            "BackupName": "MusicCollectionBackup2",
            "BackupCreationDateTime": "2020-06-26T11:08:35.431000-07:00",
            "BackupStatus": "AVAILABLE",
            "BackupType": "USER",
            "BackupSizeBytes": 400
        }
    ]
}
```

Per ulteriori informazioni, consulta Backup e ripristino on demand per DynamoDB nella Guida per gli sviluppatori di Amazon DynamoDB.

- Per i dettagli sull'API, consulta ListBackups AWS CLICommand Reference.

`list-contributor-insights`

Il seguente esempio di codice mostra come utilizzare`list-contributor-insights`.

AWS CLI

Esempio 1: come visualizzare un elenco di riepiloghi di Contributor Insights

L'esempio `list-contributor-insights` seguente mostra un elenco di riepiloghi di Contributor Insights.

```
aws dynamodb list-contributor-insights
```

Output:

```
{
    "ContributorInsightsSummaries": [
        {
            "TableName": "MusicCollection",
            "IndexName": "AlbumTitle-index",
            "ContributorInsightsStatus": "ENABLED"
        },
        {
            "TableName": "ProductCatalog",
            "ContributorInsightsStatus": "ENABLED"
        },
        {
            "TableName": "Forum",
            "ContributorInsightsStatus": "ENABLED"
        },
        {
            "TableName": "Reply",
            "ContributorInsightsStatus": "ENABLED"
        },
        {
            "TableName": "Thread",
            "ContributorInsightsStatus": "ENABLED"
        }
    ]
}
```

Per ulteriori informazioni, consulta Analyzing Data Access Using CloudWatch Contributor Insights for DynamoDB nella Amazon DynamoDB Developer Guide.

Esempio 2: come limitare il numero di elementi restituiti

L'esempio seguente limita il numero di righe restituite a 4. La risposta include un valore `NextToken` con cui recuperare la pagina di risultati successiva.

```
aws dynamodb list-contributor-insights \
    --max-results 4
```

Output:

```
{
    "ContributorInsightsSummaries": [
        {
            "TableName": "MusicCollection",
            "IndexName": "AlbumTitle-index",
            "ContributorInsightsStatus": "ENABLED"
        },
        {
            "TableName": "ProductCatalog",
            "ContributorInsightsStatus": "ENABLED"
        },
        {
            "TableName": "Forum",
            "ContributorInsightsStatus": "ENABLED"
        }
    ],
    "NextToken":
 "abCDeFGhiJKlmnOPqrSTuvwxYZ1aBCdEFghijK7LM51nOpqRSTuv3WxY3ZabC5dEFGhI2Jk3LmnoPQ6RST9"
}
```

Per ulteriori informazioni, consulta Analyzing Data Access Using CloudWatch Contributor Insights for DynamoDB nella Amazon DynamoDB Developer Guide.

Esempio 3: come recuperare la pagina di risultati successiva

Il comando seguente utilizza il valore `NextToken` di una precedente chiamata al comando `list-contributor-insights` per recuperare un'altra pagina di risultati. Poiché la risposta in questo caso non include un valore `NextToken`, ciò significa che è stata raggiunta la fine dei risultati.

```
aws dynamodb list-contributor-insights \
    --max-results 4 \
    --next-
token abCDeFGhiJKlmnOPqrSTuvwxYZ1aBCdEFghijK7LM51nOpqRSTuv3WxY3ZabC5dEFGhI2Jk3LmnoPQ6RST9
```

Output:

```
{
    "ContributorInsightsSummaries": [
        {
            "TableName": "Reply",
            "ContributorInsightsStatus": "ENABLED"
        },
        {
            "TableName": "Thread",
            "ContributorInsightsStatus": "ENABLED"
        }
    ]
}
```

Per ulteriori informazioni, consulta Analyzing Data Access Using CloudWatch Contributor Insights for DynamoDB nella Amazon DynamoDB Developer Guide.

- Per i dettagli sull'API, consulta Command Reference. ListContributorInsightsAWS CLI

`list-global-tables`

Il seguente esempio di codice mostra come utilizzare`list-global-tables`.

AWS CLI

Come elencare le tabelle globali DynamoDB esistenti

L'esempio `list-global-tables` seguente elenca tutte le tabelle globali esistenti.

```
aws dynamodb list-global-tables
```

Output:

```
{
    "GlobalTables": [
```

```
        {
            "GlobalTableName": "MusicCollection",
            "ReplicationGroup": [
                {
                    "RegionName": "us-east-2"
                },
                {
                    "RegionName": "us-east-1"
                }
            ]
        }
    ]
}
```

Per ulteriori informazioni, consulta Tabelle globali DynamoDB nella Guida per gli sviluppatori di Amazon DynamoDB.

- Per i dettagli sull'API, consulta ListGlobalTables AWS CLICommand Reference.

`list-tables`

Il seguente esempio di codice mostra come utilizzare`list-tables`.

AWS CLI

Esempio 1: come elencare le tabelle

L'`list-tables`esempio seguente elenca tutte le tabelle associate all' AWS account corrente e alla regione.

```
aws dynamodb list-tables
```

Output:

```
{
    "TableNames": [
        "Forum",
        "ProductCatalog",
        "Reply",
        "Thread"
    ]
}
```

Per ulteriori informazioni, consulta Elenco dei nomi delle tabelle nella Guida per gli sviluppatori di Amazon DynamoDB.

Esempio 2: come limitare le dimensioni della pagina

L'esempio seguente restituisce un elenco di tutte le tabelle esistenti, ma recupera solo un elemento in ogni chiamata, eseguendo più chiamate se necessario per ottenere l'intero elenco. La limitazione delle dimensioni della pagina è utile quando si eseguono comandi di tipo elenco su un numero elevato di risorse, il che può causare un errore di "timeout" quando si utilizza la dimensione di pagina predefinita di 1000.

```
aws dynamodb list-tables \
    --page-size 1
```

Output:

```
{
    "TableNames": [
        "Forum",
        "ProductCatalog",
        "Reply",
        "Thread"
    ]
}
```

Per ulteriori informazioni, consulta Elenco dei nomi delle tabelle nella Guida per gli sviluppatori di Amazon DynamoDB.

Esempio 3: come limitare il numero di elementi restituiti

L'esempio seguente limita il numero di righe restituite a 2. La risposta include un valore `NextToken` con cui recuperare la pagina di risultati successiva.

```
aws dynamodb list-tables \
    --max-items 2
```

Output:

```
{
    "TableNames": [
        "Forum",
```

```
        "ProductCatalog"
    ],
    "NextToken":
 "abCDeFGhiJKlmnOPqrSTuvwxYZ1aBCdEFghijK7LM51nOpqRSTuv3WxY3ZabC5dEFGhI2Jk3LmnoPQ6RST9"
}
```

Per ulteriori informazioni, consulta Elenco dei nomi delle tabelle nella Guida per gli sviluppatori di Amazon DynamoDB.

Esempio 4: come recuperare la pagina di risultati successiva

Il comando seguente utilizza il valore `NextToken` di una precedente chiamata al comando `list-tables` per recuperare un'altra pagina di risultati. Poiché la risposta in questo caso non include un valore `NextToken`, ciò significa che è stata raggiunta la fine dei risultati.

```
aws dynamodb list-tables \
    --starting-
token abCDeFGhiJKlmnOPqrSTuvwxYZ1aBCdEFghijK7LM51nOpqRSTuv3WxY3ZabC5dEFGhI2Jk3LmnoPQ6RST9
```

Output:

```
{
    "TableNames": [
        "Reply",
        "Thread"
    ]
}
```

Per ulteriori informazioni, consulta Elenco dei nomi delle tabelle nella Guida per gli sviluppatori di Amazon DynamoDB.

- Per i dettagli sull'API, vedere ListTablesin AWS CLI Command Reference.

`list-tags-of-resource`

Il seguente esempio di codice mostra come utilizzare`list-tags-of-resource`.

AWS CLI

Esempio 1: come elencare i tag di una risorsa DynamoDB

L'esempio `list-tags-of-resource` seguente mostra i tag per la tabella `MusicCollection`.

```
aws dynamodb list-tags-of-resource \
    --resource-arn arn:aws:dynamodb:us-west-2:123456789012:table/MusicCollection
```

Output:

```
{
    "Tags": [
        {
            "Key": "Owner",
            "Value": "blueTeam"
        },
        {
            "Key": "Environment",
            "Value": "Production"
        }
    ]
}
```

Per ulteriori informazioni, consulta Tagging per Amazon DynamoDB nella Guida per gli sviluppatori di Amazon DynamoDB.

Esempio 2: come limitare il numero di tag restituiti

L'esempio seguente limita a 1 il numero di tag restituiti. La risposta include un valore `NextToken` con cui recuperare la pagina di risultati successiva.

```
aws dynamodb list-tags-of-resource \
    --resource-arn arn:aws:dynamodb:us-west-2:123456789012:table/MusicCollection \
    --max-items 1
```

Output:

```
{
    "Tags": [
        {
            "Key": "Owner",
            "Value": "blueTeam"
        }
    ],
    "NextToken":
 "abCDeFGhiJKlmnOPqrSTuvwxYZ1aBCdEFghijK7LM51nOpqRSTuv3WxY3ZabC5dEFGhI2Jk3LmnoPQ6RST9"
```

```
}
```

Per ulteriori informazioni, consulta Tagging per Amazon DynamoDB nella Guida per gli sviluppatori di Amazon DynamoDB.

Esempio 3: come recuperare la pagina di risultati successiva

Il comando seguente utilizza il valore `NextToken` di una precedente chiamata al comando `list-tags-of-resource` per recuperare un'altra pagina di risultati. Poiché la risposta in questo caso non include un valore `NextToken`, ciò significa che è stata raggiunta la fine dei risultati.

```
aws dynamodb list-tags-of-resource \
    --resource-arn arn:aws:dynamodb:us-west-2:123456789012:table/MusicCollection \
    --starting-
token abCDeFGhiJKlmnOPqrSTuvwxYZ1aBCdEFghijK7LM51nOpqRSTuv3WxY3ZabC5dEFGhI2Jk3LmnoPQ6RST9
```

Output:

```
{
    "Tags": [
        {
            "Key": "Environment",
            "Value": "Production"
        }
    ]
}
```

Per ulteriori informazioni, consulta Tagging per Amazon DynamoDB nella Guida per gli sviluppatori di Amazon DynamoDB.

- Per i dettagli sull'API, consulta ListTagsOfResource AWS CLICommand Reference.

`put-item`

Il seguente esempio di codice mostra come utilizzare`put-item`.

AWS CLI

Esempio 1: come aggiungere un elemento a una tabella

L'`put-item`esempio seguente aggiunge un nuovo elemento alla MusicCollectiontabella.

```
aws dynamodb put-item \
    --table-name MusicCollection \
    --item file://item.json \
    --return-consumed-capacity TOTAL \
    --return-item-collection-metrics SIZE
```

Contenuto di `item.json`:

```
{
    "Artist": {"S": "No One You Know"},
    "SongTitle": {"S": "Call Me Today"},
    "AlbumTitle": {"S": "Greatest Hits"}
}
```

Output:

```
{
    "ConsumedCapacity": {
        "TableName": "MusicCollection",
        "CapacityUnits": 1.0
    },
    "ItemCollectionMetrics": {
        "ItemCollectionKey": {
            "Artist": {
                "S": "No One You Know"
            }
        },
        "SizeEstimateRangeGB": [
            0.0,
            1.0
        ]
    }
}
```

Per ulteriori informazioni, consulta Scrittura di un elemento nella Guida per gli sviluppatori di Amazon DynamoDB.

Esempio 2: come sovrascrivere in base a una condizione un elemento in una tabella

L'esempio `put-item` seguente sovrascrive un elemento esistente nella tabella `MusicCollection` solo se tale elemento include un attributo `AlbumTitle` con il valore `Greatest Hits`. Il comando restituisce il valore precedente dell'elemento.

```
aws dynamodb put-item \
    --table-name MusicCollection \
    --item file://item.json \
    --condition-expression "#A = :A" \
    --expression-attribute-names file://names.json \
    --expression-attribute-values file://values.json \
    --return-values ALL_OLD
```

Contenuto di `item.json`:

```
{
    "Artist": {"S": "No One You Know"},
    "SongTitle": {"S": "Call Me Today"},
    "AlbumTitle": {"S": "Somewhat Famous"}
}
```

Contenuto di `names.json`:

```
{
    "#A": "AlbumTitle"
}
```

Contenuto di `values.json`:

```
{
    ":A": {"S": "Greatest Hits"}
}
```

Output:

```
{
    "Attributes": {
        "AlbumTitle": {
            "S": "Greatest Hits"
        },
        "Artist": {
            "S": "No One You Know"
        },
        "SongTitle": {
            "S": "Call Me Today"
```

```
        }
    }
}
```

Se la chiave esiste già, verrà visualizzato l'output seguente:

```
A client error (ConditionalCheckFailedException) occurred when calling the PutItem
 operation: The conditional request failed.
```

Per ulteriori informazioni, consulta Scrittura di un elemento nella Guida per gli sviluppatori di Amazon DynamoDB.

- Per i dettagli sull'API, vedere PutItemin AWS CLI Command Reference.

`query`

Il seguente esempio di codice mostra come utilizzare`query`.

AWS CLI

Esempio 1: come eseguire una query su una tabella

L'esempio `query` seguente esegue query sulle voce della tabella `MusicCollection`. La tabella ha una chiave hash-and-range primaria (`Artist`and`SongTitle`), ma questa query specifica solo il valore della chiave hash. Restituisce i titoli dei brani dell'artista "No One You Know".

```
aws dynamodb query \
    --table-name MusicCollection \
    --projection-expression "SongTitle" \
    --key-condition-expression "Artist = :v1" \
    --expression-attribute-values file://expression-attributes.json \
    --return-consumed-capacity TOTAL
```

Contenuto di `expression-attributes.json`:

```
{
    ":v1": {"S": "No One You Know"}
}
```

Output:

```
{
    "Items": [
        {
            "SongTitle": {
                "S": "Call Me Today"
            },
            "SongTitle": {
                "S": "Scared of My Shadow"
            }
        }
    ],
    "Count": 2,
    "ScannedCount": 2,
    "ConsumedCapacity": {
        "TableName": "MusicCollection",
        "CapacityUnits": 0.5
    }
}
```

Per ulteriori informazioni, consulta Utilizzo delle query in DynamoDB nella Guida per gli sviluppatori di Amazon DynamoDB.

Esempio 2: come eseguire una query su una tabella utilizzando l'elevata consistenza di lettura e attraversare l'indice in ordine decrescente

L'esempio seguente esegue la stessa query del primo esempio, ma restituisce i risultati in ordine inverso e utilizza operazioni a elevata consistenza di lettura.

```
aws dynamodb query \
    --table-name MusicCollection \
    --projection-expression "SongTitle" \
    --key-condition-expression "Artist = :v1" \
    --expression-attribute-values file://expression-attributes.json \
    --consistent-read \
    --no-scan-index-forward \
    --return-consumed-capacity TOTAL
```

Contenuto di `expression-attributes.json`:

```
{
    ":v1": {"S": "No One You Know"}
```

```
}
```

Output:

```
{
    "Items": [
        {
            "SongTitle": {
                "S": "Scared of My Shadow"
            }
        },
        {
            "SongTitle": {
                "S": "Call Me Today"
            }
        }
    ],
    "Count": 2,
    "ScannedCount": 2,
    "ConsumedCapacity": {
        "TableName": "MusicCollection",
        "CapacityUnits": 1.0
    }
}
```

Per ulteriori informazioni, consulta Utilizzo delle query in DynamoDB nella Guida per gli sviluppatori di Amazon DynamoDB.

Esempio 3: come filtrare risultati specifici

L'esempio seguente esegue una query su `MusicCollection` ma esclude i risultati con valori specifici nell'attributo `AlbumTitle`. Nota che ciò non influisce su `ScannedCount` o `ConsumedCapacity`, poiché il filtro viene applicato dopo la lettura degli elementi.

```
aws dynamodb query \
    --table-name MusicCollection \
    --key-condition-expression "#n1 = :v1" \
    --filter-expression "NOT (#n2 IN (:v2, :v3))" \
    --expression-attribute-names file://names.json \
    --expression-attribute-values file://values.json \
    --return-consumed-capacity TOTAL
```

Contenuto di `values.json`:

```
{
    ":v1": {"S": "No One You Know"},
    ":v2": {"S": "Blue Sky Blues"},
    ":v3": {"S": "Greatest Hits"}
}
```

Contenuto di `names.json`:

```
{
    "#n1": "Artist",
    "#n2": "AlbumTitle"
}
```

Output:

```
{
    "Items": [
        {
            "AlbumTitle": {
                "S": "Somewhat Famous"
            },
            "Artist": {
                "S": "No One You Know"
            },
            "SongTitle": {
                "S": "Call Me Today"
            }
        }
    ],
    "Count": 1,
    "ScannedCount": 2,
    "ConsumedCapacity": {
        "TableName": "MusicCollection",
        "CapacityUnits": 0.5
    }
}
```

Per ulteriori informazioni, consulta Utilizzo delle query in DynamoDB nella Guida per gli sviluppatori di Amazon DynamoDB.

Esempio 4: come recuperare solo il conteggio degli elementi

L'esempio seguente recupera il conteggio degli elementi corrispondenti alla query, ma non recupera nessuno degli elementi stessi.

```
aws dynamodb query \
    --table-name MusicCollection \
    --select COUNT \
    --key-condition-expression "Artist = :v1" \
    --expression-attribute-values file://expression-attributes.json
```

Contenuto di `expression-attributes.json`:

```
{
    ":v1": {"S": "No One You Know"}
}
```

Output:

```
{
    "Count": 2,
    "ScannedCount": 2,
    "ConsumedCapacity": null
}
```

Per ulteriori informazioni, consulta Utilizzo delle query in DynamoDB nella Guida per gli sviluppatori di Amazon DynamoDB.

Esempio 5: come eseguire una query su un indice

L'esempio seguente esegue una query sull'indice globale secondario `AlbumTitleIndex`. La query restituisce tutti gli attributi della tabella di base proiettati nell'indice secondario locale. Nota che durante l'esecuzione di query su un indice secondario locale o su un indice secondario globale è necessario fornire anche il nome della tabella di base mediante il parametro `table-name`.

```
aws dynamodb query \
    --table-name MusicCollection \
    --index-name AlbumTitleIndex \
    --key-condition-expression "Artist = :v1" \
    --expression-attribute-values file://expression-attributes.json \
```

```
    --select ALL_PROJECTED_ATTRIBUTES \
    --return-consumed-capacity INDEXES
```

Contenuto di `expression-attributes.json`:

```
{
    ":v1": {"S": "No One You Know"}
}
```

Output:

```
{
    "Items": [
        {
            "AlbumTitle": {
                "S": "Blue Sky Blues"
            },
            "Artist": {
                "S": "No One You Know"
            },
            "SongTitle": {
                "S": "Scared of My Shadow"
            }
        },
        {
            "AlbumTitle": {
                "S": "Somewhat Famous"
            },
            "Artist": {
                "S": "No One You Know"
            },
            "SongTitle": {
                "S": "Call Me Today"
            }
        }
    ],
    "Count": 2,
    "ScannedCount": 2,
    "ConsumedCapacity": {
        "TableName": "MusicCollection",
        "CapacityUnits": 0.5,
        "Table": {
            "CapacityUnits": 0.0
```

```
        },
        "LocalSecondaryIndexes": {
            "AlbumTitleIndex": {
                "CapacityUnits": 0.5
            }
        }
    }
}
```

Per ulteriori informazioni, consulta Utilizzo delle query in DynamoDB nella Guida per gli sviluppatori di Amazon DynamoDB.

- Per informazioni dettagliate sull'API, consulta Query nella documentazione di riferimento di AWS CLI .

`restore-table-from-backup`

Il seguente esempio di codice mostra come utilizzare. `restore-table-from-backup`

AWS CLI

Come ripristinare una tabella DynamoDB da un backup esistente

L'esempio `restore-table-from-backup` seguente ripristina la tabella specificata da un backup esistente.

```
aws dynamodb restore-table-from-backup \
    --target-table-name MusicCollection \
    --backup-arnarn:aws:dynamodb:us-west-2:123456789012:table/MusicCollection/
backup/01576616366715-b4e58d3a
```

Output:

```
{
    "TableDescription": {
        "AttributeDefinitions": [
            {
                "AttributeName": "Artist",
                "AttributeType": "S"
            },
            {
                "AttributeName": "SongTitle",
```

```
                "AttributeType": "S"
            }
        ],
        "TableName": "MusicCollection2",
        "KeySchema": [
            {
                "AttributeName": "Artist",
                "KeyType": "HASH"
            },
            {
                "AttributeName": "SongTitle",
                "KeyType": "RANGE"
            }
        ],
        "TableStatus": "CREATING",
        "CreationDateTime": 1576618274.326,
        "ProvisionedThroughput": {
            "NumberOfDecreasesToday": 0,
            "ReadCapacityUnits": 5,
            "WriteCapacityUnits": 5
        },
        "TableSizeBytes": 0,
        "ItemCount": 0,
        "TableArn": "arn:aws:dynamodb:us-west-2:123456789012:table/
MusicCollection2",
        "TableId": "114865c9-5ef3-496c-b4d1-c4cbdd2d44fb",
        "BillingModeSummary": {
            "BillingMode": "PROVISIONED"
        },
        "RestoreSummary": {
            "SourceBackupArn": "arn:aws:dynamodb:us-west-2:123456789012:table/
MusicCollection/backup/01576616366715-b4e58d3a",
            "SourceTableArn": "arn:aws:dynamodb:us-west-2:123456789012:table/
MusicCollection",
            "RestoreDateTime": 1576616366.715,
            "RestoreInProgress": true
        }
    }
}
```

Per ulteriori informazioni, consulta Backup e ripristino on demand per DynamoDB nella Guida per gli sviluppatori di Amazon DynamoDB.

- Per i dettagli sull'API, consulta RestoreTableFromBackup AWS CLICommand Reference.

`restore-table-to-point-in-time`

Il seguente esempio di codice mostra come utilizzare`restore-table-to-point-in-time`.

AWS CLI

Come ripristinare una tabella DynamoDB a un determinato momento

L'esempio `restore-table-to-point-in-time` seguente ripristina la tabella `MusicCollection` al momento specificato.

```
aws dynamodb restore-table-to-point-in-time \
    --source-table-name MusicCollection \
    --target-table-name MusicCollectionRestore \
    --restore-date-time 1576622404.0
```

Output:

```
{
    "TableDescription": {
        "AttributeDefinitions": [
            {
                "AttributeName": "Artist",
                "AttributeType": "S"
            },
            {
                "AttributeName": "SongTitle",
                "AttributeType": "S"
            }
        ],
        "TableName": "MusicCollectionRestore",
        "KeySchema": [
            {
                "AttributeName": "Artist",
                "KeyType": "HASH"
            },
            {
                "AttributeName": "SongTitle",
                "KeyType": "RANGE"
            }
        ],
        "TableStatus": "CREATING",
        "CreationDateTime": 1576623311.86,
```

```
        "ProvisionedThroughput": {
            "NumberOfDecreasesToday": 0,
            "ReadCapacityUnits": 5,
            "WriteCapacityUnits": 5
        },
        "TableSizeBytes": 0,
        "ItemCount": 0,
        "TableArn": "arn:aws:dynamodb:us-west-2:123456789012:table/
MusicCollectionRestore",
        "TableId": "befd9e0e-1843-4dc6-a147-d6d00e85cb1f",
        "BillingModeSummary": {
            "BillingMode": "PROVISIONED"
        },
        "RestoreSummary": {
            "SourceTableArn": "arn:aws:dynamodb:us-west-2:123456789012:table/
MusicCollection",
            "RestoreDateTime": 1576622404.0,
            "RestoreInProgress": true
        }
    }
}
```

Per ulteriori informazioni, consulta Point-in-Time Recovery for DynamoDB nella Amazon DynamoDB Developer Guide.

- Per i dettagli sull'API, consulta RestoreTableToPointInTimeCommand Reference.AWS CLI

`scan`

Il seguente esempio di codice mostra come utilizzare`scan`.

AWS CLI

Come eseguire la scansione di una tabella

L'esempio `scan` seguente analizza l'intera tabella `MusicCollection`, quindi restringe i risultati alle canzoni dell'artista "No One You Know". Per ogni elemento, vengono restituiti solo il titolo dell'album e il titolo del brano.

```
aws dynamodb scan \
    --table-name MusicCollection \
    --filter-expression "Artist = :a" \
    --projection-expression "#ST, #AT" \
```

```
    --expression-attribute-names file://expression-attribute-names.json \
    --expression-attribute-values file://expression-attribute-values.json
```

Contenuto di `expression-attribute-names.json`:

```
{
    "#ST": "SongTitle",
    "#AT":"AlbumTitle"
}
```

Contenuto di `expression-attribute-values.json`:

```
{
    ":a": {"S": "No One You Know"}
}
```

Output:

```
{
    "Count": 2,
    "Items": [
        {
            "SongTitle": {
                "S": "Call Me Today"
            },
            "AlbumTitle": {
                "S": "Somewhat Famous"
            }
        },
        {
            "SongTitle": {
                "S": "Scared of My Shadow"
            },
            "AlbumTitle": {
                "S": "Blue Sky Blues"
            }
        }
    ],
    "ScannedCount": 3,
    "ConsumedCapacity": null
}
```

Per ulteriori informazioni, consulta Utilizzo delle scansioni in DynamoDB nella Guida per gli sviluppatori di Amazon DynamoDB.

- Per informazioni dettagliate sull'API, consulta Scansione nella documentazione di riferimento di AWS CLI .

`tag-resource`

Il seguente esempio di codice mostra come utilizzare`tag-resource`.

AWS CLI

Come aggiungere tag a una risorsa DynamoDB

L'`tag-resource`esempio seguente aggiunge una key/value coppia di tag alla `MusicCollection` tabella.

```
aws dynamodb tag-resource \
    --resource-arn arn:aws:dynamodb:us-west-2:123456789012:table/MusicCollection \
    --tags Key=Owner,Value=blueTeam
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Tagging per Amazon DynamoDB nella Guida per gli sviluppatori di Amazon DynamoDB.

- Per i dettagli sull'API, vedere TagResourcein AWS CLI Command Reference.

`transact-get-items`

Il seguente esempio di codice mostra come utilizzare`transact-get-items`.

AWS CLI

Come recuperare più elementi in modo atomico da una o più tabelle

L'esempio `transact-get-items` seguente recupera più elementi in modo atomico.

```
aws dynamodb transact-get-items \
    --transact-items file://transact-items.json \
    --return-consumed-capacity TOTAL
```

Contenuto di `transact-items.json`:

```
[
    {
        "Get": {
            "Key": {
                "Artist": {"S": "Acme Band"},
                "SongTitle": {"S": "Happy Day"}
            },
            "TableName": "MusicCollection"
        }
    },
    {
        "Get": {
            "Key": {
                "Artist": {"S": "No One You Know"},
                "SongTitle": {"S": "Call Me Today"}
            },
            "TableName": "MusicCollection"
        }
    }
]
```

Output:

```
{
    "ConsumedCapacity": [
        {
            "TableName": "MusicCollection",
            "CapacityUnits": 4.0,
            "ReadCapacityUnits": 4.0
        }
    ],
    "Responses": [
        {
            "Item": {
                "AlbumTitle": {
                    "S": "Songs About Life"
                },
                "Artist": {
                    "S": "Acme Band"
                },
                "SongTitle": {
```

```
                    "S": "Happy Day"
                }
            }
        },
        {
            "Item": {
                "AlbumTitle": {
                    "S": "Somewhat Famous"
                },
                "Artist": {
                    "S": "No One You Know"
                },
                "SongTitle": {
                    "S": "Call Me Today"
                }
            }
        }
    ]
}
```

Per ulteriori informazioni, consulta Gestione di flussi di lavoro complessi con transazioni DynamoDB nella Guida per gli sviluppatori di Amazon DynamoDB.

- Per i dettagli sull'API, consulta TransactGetItems AWS CLICommand Reference.

`transact-write-items`

Il seguente esempio di codice mostra come utilizzare`transact-write-items`.

AWS CLI

Esempio 1: come scrivere elementi in modo atomico su una o più tabelle

L'esempio `transact-write-items` seguente aggiorna un elemento e ne elimina un altro. L'operazione ha esito negativo se una delle operazioni fallisce o se uno degli elementi contiene un attributo `Rating`.

```
aws dynamodb transact-write-items \
    --transact-items file://transact-items.json \
    --return-consumed-capacity TOTAL \
    --return-item-collection-metrics SIZE
```

Contenuto del file `transact-items.json`:

```
[
    {
        "Update": {
            "Key": {
                "Artist": {"S": "Acme Band"},
                "SongTitle": {"S": "Happy Day"}
            },
            "UpdateExpression": "SET AlbumTitle = :newval",
            "ExpressionAttributeValues": {
                ":newval": {"S": "Updated Album Title"}
            },
            "TableName": "MusicCollection",
            "ConditionExpression": "attribute_not_exists(Rating)"
        }
    },
    {
        "Delete": {
            "Key": {
                "Artist": {"S": "No One You Know"},
                "SongTitle": {"S": "Call Me Today"}
            },
            "TableName": "MusicCollection",
            "ConditionExpression": "attribute_not_exists(Rating)"
        }
    }
]
```

Output:

```
{
    "ConsumedCapacity": [
        {
            "TableName": "MusicCollection",
            "CapacityUnits": 10.0,
            "WriteCapacityUnits": 10.0
        }
    ],
    "ItemCollectionMetrics": {
        "MusicCollection": [
            {
                "ItemCollectionKey": {
                    "Artist": {
                        "S": "No One You Know"
```

```
                    }
                },
                "SizeEstimateRangeGB": [
                    0.0,
                    1.0
                ]
            },
            {
                "ItemCollectionKey": {
                    "Artist": {
                        "S": "Acme Band"
                    }
                },
                "SizeEstimateRangeGB": [
                    0.0,
                    1.0
                ]
            }
        ]
    }
}
```

Per ulteriori informazioni, consulta Gestione di flussi di lavoro complessi con transazioni DynamoDB nella Guida per gli sviluppatori di Amazon DynamoDB.

Esempio 2: come scrivere elementi in modo atomico utilizzando un token di richiesta del client

Il comando seguente utilizza un token di richiesta del client per effettuare la chiamata a `transact-write-items` idempotente, il che significa che più chiamate hanno lo stesso effetto di una singola chiamata.

```
aws dynamodb transact-write-items \
    --transact-items file://transact-items.json \
    --client-request-token abc123
```

Contenuto del file `transact-items.json`:

```
[
    {
        "Update": {
            "Key": {
                "Artist": {"S": "Acme Band"},
```

```
                "SongTitle": {"S": "Happy Day"}
            },
            "UpdateExpression": "SET AlbumTitle = :newval",
            "ExpressionAttributeValues": {
                ":newval": {"S": "Updated Album Title"}
            },
            "TableName": "MusicCollection",
            "ConditionExpression": "attribute_not_exists(Rating)"
        }
    },
    {
        "Delete": {
            "Key": {
                "Artist": {"S": "No One You Know"},
                "SongTitle": {"S": "Call Me Today"}
            },
            "TableName": "MusicCollection",
            "ConditionExpression": "attribute_not_exists(Rating)"
        }
    }
]
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Gestione di flussi di lavoro complessi con transazioni DynamoDB nella Guida per gli sviluppatori di Amazon DynamoDB.

- Per i dettagli sull'API, consulta TransactWriteItems AWS CLICommand Reference.

`untag-resource`

Il seguente esempio di codice mostra come utilizzare`untag-resource`.

AWS CLI

Come rimuovere un tag da una risorsa DynamoDB

L'esempio `untag-resource` seguente rimuove il tag con la chiave `Owner` dalla tabella `MusicCollection`.

```
aws dynamodb untag-resource \
    --resource-arn arn:aws:dynamodb:us-west-2:123456789012:table/MusicCollection \
```

```
    --tag-keys Owner
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Tagging per Amazon DynamoDB nella Guida per gli sviluppatori di Amazon DynamoDB.

- Per i dettagli sull'API, consulta UntagResource AWS CLICommand Reference.

`update-continuous-backups`

Il seguente esempio di codice mostra come utilizzare`update-continuous-backups`.

AWS CLI

Come aggiornare le impostazioni di backup continuo per una tabella DynamoDB

L'`update-continuous-backups`esempio seguente abilita il point-in-time ripristino della `MusicCollection` tabella.

```
aws dynamodb update-continuous-backups \
    --table-name MusicCollection \
    --point-in-time-recovery-specification PointInTimeRecoveryEnabled=true
```

Output:

```
{
    "ContinuousBackupsDescription": {
        "ContinuousBackupsStatus": "ENABLED",
        "PointInTimeRecoveryDescription": {
            "PointInTimeRecoveryStatus": "ENABLED",
            "EarliestRestorableDateTime": 1576622404.0,
            "LatestRestorableDateTime": 1576622404.0
        }
    }
}
```

Per ulteriori informazioni, consulta Point-in-Time Recovery for DynamoDB nella Amazon DynamoDB Developer Guide.

- Per i dettagli sull'API, consulta UpdateContinuousBackupsCommand Reference.AWS CLI

`update-contributor-insights`

Il seguente esempio di codice mostra come utilizzare`update-contributor-insights`.

AWS CLI

Come abilitare Contributor Insights su una tabella

L'esempio `update-contributor-insights` seguente abilita Contributor Insights sulla tabella `MusicCollection` e l'indice secondario globale `AlbumTitle-index`.

```
aws dynamodb update-contributor-insights \
    --table-name MusicCollection \
    --index-name AlbumTitle-index \
    --contributor-insights-action ENABLE
```

Output:

```
{
    "TableName": "MusicCollection",
    "IndexName": "AlbumTitle-index",
    "ContributorInsightsStatus": "ENABLING"
}
```

Per ulteriori informazioni, consulta Analyzing Data Access Using CloudWatch Contributor Insights for DynamoDB nella Amazon DynamoDB Developer Guide.

- Per i dettagli sull'API, consulta Command Reference. UpdateContributorInsightsAWS CLI

`update-global-table-settings`

Il seguente esempio di codice mostra come utilizzare`update-global-table-settings`.

AWS CLI

Come aggiornare le impostazioni della capacità di scrittura con provisioning su una tabella globale DynamoDB

L'esempio `update-global-table-settings` seguente imposta su 15 la capacità di scrittura con provisioning della tabella globale `MusicCollection`.

```
aws dynamodb update-global-table-settings \
    --global-table-name MusicCollection \
    --global-table-provisioned-write-capacity-units 15
```

Output:

```
{
    "GlobalTableName": "MusicCollection",
    "ReplicaSettings": [
        {
            "RegionName": "eu-west-1",
            "ReplicaStatus": "UPDATING",
            "ReplicaProvisionedReadCapacityUnits": 10,
            "ReplicaProvisionedReadCapacityAutoScalingSettings": {
                "AutoScalingDisabled": true
            },
            "ReplicaProvisionedWriteCapacityUnits": 10,
            "ReplicaProvisionedWriteCapacityAutoScalingSettings": {
                "AutoScalingDisabled": true
            }
        },
        {
            "RegionName": "us-east-1",
            "ReplicaStatus": "UPDATING",
            "ReplicaProvisionedReadCapacityUnits": 10,
            "ReplicaProvisionedReadCapacityAutoScalingSettings": {
                "AutoScalingDisabled": true
            },
            "ReplicaProvisionedWriteCapacityUnits": 10,
            "ReplicaProvisionedWriteCapacityAutoScalingSettings": {
                "AutoScalingDisabled": true
            }
        },
        {
            "RegionName": "us-east-2",
            "ReplicaStatus": "UPDATING",
            "ReplicaProvisionedReadCapacityUnits": 10,
            "ReplicaProvisionedReadCapacityAutoScalingSettings": {
                "AutoScalingDisabled": true
            },
            "ReplicaProvisionedWriteCapacityUnits": 10,
            "ReplicaProvisionedWriteCapacityAutoScalingSettings": {
                "AutoScalingDisabled": true
```

```
            }
        }
    ]
}
```

Per ulteriori informazioni, consulta Tabelle globali DynamoDB nella Guida per gli sviluppatori di Amazon DynamoDB.

- Per i dettagli sull'API, consulta UpdateGlobalTableSettings AWS CLICommand Reference.

`update-global-table`

Il seguente esempio di codice mostra come utilizzare`update-global-table`.

AWS CLI

Come aggiornare una tabella globale DynamoDB

L'esempio `update-global-table` seguente aggiunge una replica nella Regione specificata alla tabella globale `MusicCollection`.

```
aws dynamodb update-global-table \
    --global-table-name MusicCollection \
    --replica-updates Create={RegionName=eu-west-1}
```

Output:

```
{
    "GlobalTableDescription": {
        "ReplicationGroup": [
            {
                "RegionName": "eu-west-1"
            },
            {
                "RegionName": "us-east-2"
            },
            {
                "RegionName": "us-east-1"
            }
        ],
        "GlobalTableArn": "arn:aws:dynamodb::123456789012:global-table/
MusicCollection",
        "CreationDateTime": 1576625818.532,
```

```
        "GlobalTableStatus": "ACTIVE",
        "GlobalTableName": "MusicCollection"
    }
}
```

Per ulteriori informazioni, consulta Tabelle globali DynamoDB nella Guida per gli sviluppatori di Amazon DynamoDB.

- Per i dettagli sull'API, consulta UpdateGlobalTable AWS CLICommand Reference.

`update-item`

Il seguente esempio di codice mostra come utilizzare`update-item`.

AWS CLI

Esempio 1: come aggiornare un elemento in una tabella

L'esempio `update-item` seguente legge una voce dalla tabella `MusicCollection`. Aggiunge un nuovo attributo (`Year`) e modifica l'attributo `AlbumTitle`. Tutti gli attributi dell'elemento come appaiono dopo l'aggiornamento vengono restituiti nella risposta.

```
aws dynamodb update-item \
    --table-name MusicCollection \
    --key file://key.json \
    --update-expression "SET #Y = :y, #AT = :t" \
    --expression-attribute-names file://expression-attribute-names.json \
    --expression-attribute-values file://expression-attribute-values.json  \
    --return-values ALL_NEW \
    --return-consumed-capacity TOTAL \
    --return-item-collection-metrics SIZE
```

Contenuto di `key.json`:

```
{
    "Artist": {"S": "Acme Band"},
    "SongTitle": {"S": "Happy Day"}
}
```

Contenuto di `expression-attribute-names.json`:

```
{
```

```
    "#Y":"Year", "#AT":"AlbumTitle"
}
```

Contenuto di `expression-attribute-values.json`:

```
{
    ":y":{"N": "2015"},
    ":t":{"S": "Louder Than Ever"}
}
```

Output:

```
{
    "Attributes": {
        "AlbumTitle": {
            "S": "Louder Than Ever"
        },
        "Awards": {
            "N": "10"
        },
        "Artist": {
            "S": "Acme Band"
        },
        "Year": {
            "N": "2015"
        },
        "SongTitle": {
            "S": "Happy Day"
        }
    },
    "ConsumedCapacity": {
        "TableName": "MusicCollection",
        "CapacityUnits": 3.0
    },
    "ItemCollectionMetrics": {
        "ItemCollectionKey": {
            "Artist": {
                "S": "Acme Band"
            }
        },
        "SizeEstimateRangeGB": [
            0.0,
            1.0
```

```
        ]
    }
}
```

Per ulteriori informazioni, consulta Scrittura di un elemento nella Guida per gli sviluppatori di Amazon DynamoDB.

Esempio 2: come aggiornare un elemento in base a una condizione

L'esempio seguente aggiorna un elemento nella tabella `MusicCollection`, ma solo se l'elemento esistente non dispone già di un attributo `Year`.

```
aws dynamodb update-item \
    --table-name MusicCollection \
    --key file://key.json \
    --update-expression "SET #Y = :y, #AT = :t" \
    --expression-attribute-names file://expression-attribute-names.json \
    --expression-attribute-values file://expression-attribute-values.json  \
    --condition-expression "attribute_not_exists(#Y)"
```

Contenuto di `key.json`:

```
{
    "Artist": {"S": "Acme Band"},
    "SongTitle": {"S": "Happy Day"}
}
```

Contenuto di `expression-attribute-names.json`:

```
{
    "#Y":"Year",
    "#AT":"AlbumTitle"
}
```

Contenuto di `expression-attribute-values.json`:

```
{
    ":y":{"N": "2015"},
    ":t":{"S": "Louder Than Ever"}
}
```

Se l'elemento include già un attributo `Year`, DynamoDB restituisce il seguente output.

```
An error occurred (ConditionalCheckFailedException) when calling the UpdateItem
 operation: The conditional request failed
```

Per ulteriori informazioni, consulta Scrittura di un elemento nella Guida per gli sviluppatori di Amazon DynamoDB.

- Per i dettagli sull'API, consulta UpdateItem AWS CLICommand Reference.

`update-table-replica-auto-scaling`

Il seguente esempio di codice mostra come utilizzare`update-table-replica-auto-scaling`.

AWS CLI

Come aggiornare le impostazioni di dimensionamento automatico tra le repliche di una tabella globale

L'esempio `update-table-replica-auto-scaling` seguente aggiorna le impostazioni di dimensionamento automatico della capacità di scrittura tra le repliche della tabella globale specificata.

```
aws dynamodb update-table-replica-auto-scaling \
    --table-name MusicCollection \
    --provisioned-write-capacity-auto-scaling-update file://auto-scaling-policy.json
```

Contenuto di `auto-scaling-policy.json`:

```
{
    "MinimumUnits": 10,
    "MaximumUnits": 100,
    "AutoScalingDisabled": false,
    "ScalingPolicyUpdate": {
        "PolicyName": "DynamoDBWriteCapacityUtilization:table/MusicCollection",
        "TargetTrackingScalingPolicyConfiguration": {
            "TargetValue": 80
        }
    }
}
```

Output:

```
{
    "TableAutoScalingDescription": {
        "TableName": "MusicCollection",
        "TableStatus": "ACTIVE",
        "Replicas": [
            {
                "RegionName": "eu-central-1",
                "GlobalSecondaryIndexes": [],
                "ReplicaProvisionedReadCapacityAutoScalingSettings": {
                    "MinimumUnits": 5,
                    "MaximumUnits": 40000,
                    "AutoScalingRoleArn": "arn:aws:iam::123456789012:role/
aws-service-role/dynamodb.application-autoscaling.amazonaws.com/
AWSServiceRoleForApplicationAutoScaling_DynamoDBTable",
                    "ScalingPolicies": [
                        {
                            "PolicyName": "DynamoDBReadCapacityUtilization:table/
MusicCollection",
                            "TargetTrackingScalingPolicyConfiguration": {
                                "TargetValue": 70.0
                            }
                        }
                    ]
                },
                "ReplicaProvisionedWriteCapacityAutoScalingSettings": {
                    "MinimumUnits": 10,
                    "MaximumUnits": 100,
                    "AutoScalingRoleArn": "arn:aws:iam::123456789012:role/
aws-service-role/dynamodb.application-autoscaling.amazonaws.com/
AWSServiceRoleForApplicationAutoScaling_DynamoDBTable",
                    "ScalingPolicies": [
                        {
                            "PolicyName": "DynamoDBWriteCapacityUtilization:table/
MusicCollection",
                            "TargetTrackingScalingPolicyConfiguration": {
                                "TargetValue": 80.0
                            }
                        }
                    ]
                },
                "ReplicaStatus": "ACTIVE"
            },
```

```
            {
                "RegionName": "us-east-1",
                "GlobalSecondaryIndexes": [],
                "ReplicaProvisionedReadCapacityAutoScalingSettings": {
                    "MinimumUnits": 5,
                    "MaximumUnits": 40000,
                    "AutoScalingRoleArn": "arn:aws:iam::123456789012:role/
aws-service-role/dynamodb.application-autoscaling.amazonaws.com/
AWSServiceRoleForApplicationAutoScaling_DynamoDBTable",
                    "ScalingPolicies": [
                        {
                            "PolicyName": "DynamoDBReadCapacityUtilization:table/
MusicCollection",
                            "TargetTrackingScalingPolicyConfiguration": {
                                "TargetValue": 70.0
                            }
                        }
                    ]
                },
                "ReplicaProvisionedWriteCapacityAutoScalingSettings": {
                    "MinimumUnits": 10,
                    "MaximumUnits": 100,
                    "AutoScalingRoleArn": "arn:aws:iam::123456789012:role/
aws-service-role/dynamodb.application-autoscaling.amazonaws.com/
AWSServiceRoleForApplicationAutoScaling_DynamoDBTable",
                    "ScalingPolicies": [
                        {
                            "PolicyName": "DynamoDBWriteCapacityUtilization:table/
MusicCollection",
                            "TargetTrackingScalingPolicyConfiguration": {
                                "TargetValue": 80.0
                            }
                        }
                    ]
                },
                "ReplicaStatus": "ACTIVE"
            },
            {
                "RegionName": "us-east-2",
                "GlobalSecondaryIndexes": [],
                "ReplicaProvisionedReadCapacityAutoScalingSettings": {
                    "MinimumUnits": 5,
                    "MaximumUnits": 40000,
```

```
                    "AutoScalingRoleArn": "arn:aws:iam::123456789012:role/
aws-service-role/dynamodb.application-autoscaling.amazonaws.com/
AWSServiceRoleForApplicationAutoScaling_DynamoDBTable",
                    "ScalingPolicies": [
                        {
                            "PolicyName": "DynamoDBReadCapacityUtilization:table/
MusicCollection",
                            "TargetTrackingScalingPolicyConfiguration": {
                                "TargetValue": 70.0
                            }
                        }
                    ]
                },
                "ReplicaProvisionedWriteCapacityAutoScalingSettings": {
                    "MinimumUnits": 10,
                    "MaximumUnits": 100,
                    "AutoScalingRoleArn": "arn:aws:iam::123456789012:role/
aws-service-role/dynamodb.application-autoscaling.amazonaws.com/
AWSServiceRoleForApplicationAutoScaling_DynamoDBTable",
                    "ScalingPolicies": [
                        {
                            "PolicyName": "DynamoDBWriteCapacityUtilization:table/
MusicCollection",
                            "TargetTrackingScalingPolicyConfiguration": {
                                "TargetValue": 80.0
                            }
                        }
                    ]
                },
                "ReplicaStatus": "ACTIVE"
            }
        ]
    }
}
```

Per ulteriori informazioni, consulta Tabelle globali DynamoDB nella Guida per gli sviluppatori di Amazon DynamoDB.

- Per i dettagli sull'API, consulta UpdateTableReplicaAutoScaling AWS CLICommand Reference.

`update-table`

Il seguente esempio di codice mostra come utilizzare`update-table`.

AWS CLI

Esempio 1: come modificare la modalità di fatturazione di una tabella

L'esempio `update-table` seguente aumenta la capacità di lettura e scrittura allocata per la tabella `MusicCollection`.

```
aws dynamodb update-table \
    --table-name MusicCollection \
    --billing-mode PROVISIONED \
    --provisioned-throughput ReadCapacityUnits=15,WriteCapacityUnits=10
```

Output:

```
{
    "TableDescription": {
        "AttributeDefinitions": [
            {
                "AttributeName": "AlbumTitle",
                "AttributeType": "S"
            },
            {
                "AttributeName": "Artist",
                "AttributeType": "S"
            },
            {
                "AttributeName": "SongTitle",
                "AttributeType": "S"
            }
        ],
        "TableName": "MusicCollection",
        "KeySchema": [
            {
                "AttributeName": "Artist",
                "KeyType": "HASH"
            },
            {
                "AttributeName": "SongTitle",
                "KeyType": "RANGE"
            }
        ],
        "TableStatus": "UPDATING",
        "CreationDateTime": "2020-05-26T15:59:49.473000-07:00",
```

```
        "ProvisionedThroughput": {
            "LastIncreaseDateTime": "2020-07-28T13:18:18.921000-07:00",
            "NumberOfDecreasesToday": 0,
            "ReadCapacityUnits": 15,
            "WriteCapacityUnits": 10
        },
        "TableSizeBytes": 182,
        "ItemCount": 2,
        "TableArn": "arn:aws:dynamodb:us-west-2:123456789012:table/MusicCollection",
        "TableId": "abcd0123-01ab-23cd-0123-abcdef123456",
        "BillingModeSummary": {
            "BillingMode": "PROVISIONED",
            "LastUpdateToPayPerRequestDateTime": "2020-07-28T13:14:48.366000-07:00"
        }
    }
}
```

Per ulteriori informazioni, consulta Aggiornamento di una tabella nella Guida per gli sviluppatori di Amazon DynamoDB.

Esempio 2: come creare un indice secondario globale

L'esempio seguente aggiunge un indice secondario globale alla tabella `MusicCollection`.

```
aws dynamodb update-table \
    --table-name MusicCollection \
    --attribute-definitions AttributeName=AlbumTitle,AttributeType=S \
    --global-secondary-index-updates file://gsi-updates.json
```

Contenuto di `gsi-updates.json`:

```
[
    {
        "Create": {
            "IndexName": "AlbumTitle-index",
            "KeySchema": [
                {
                    "AttributeName": "AlbumTitle",
                    "KeyType": "HASH"
                }
            ],
            "ProvisionedThroughput": {
                "ReadCapacityUnits": 10,
```

```
                "WriteCapacityUnits": 10
            },
            "Projection": {
                "ProjectionType": "ALL"
            }
        }
    }
]
```

Output:

```
{
    "TableDescription": {
        "AttributeDefinitions": [
            {
                "AttributeName": "AlbumTitle",
                "AttributeType": "S"
            },
            {
                "AttributeName": "Artist",
                "AttributeType": "S"
            },
            {
                "AttributeName": "SongTitle",
                "AttributeType": "S"
            }
        ],
        "TableName": "MusicCollection",
        "KeySchema": [
            {
                "AttributeName": "Artist",
                "KeyType": "HASH"
            },
            {
                "AttributeName": "SongTitle",
                "KeyType": "RANGE"
            }
        ],
        "TableStatus": "UPDATING",
        "CreationDateTime": "2020-05-26T15:59:49.473000-07:00",
        "ProvisionedThroughput": {
            "LastIncreaseDateTime": "2020-07-28T12:59:17.537000-07:00",
            "NumberOfDecreasesToday": 0,
```

```
            "ReadCapacityUnits": 15,
            "WriteCapacityUnits": 10
        },
        "TableSizeBytes": 182,
        "ItemCount": 2,
        "TableArn": "arn:aws:dynamodb:us-west-2:123456789012:table/MusicCollection",
        "TableId": "abcd0123-01ab-23cd-0123-abcdef123456",
        "BillingModeSummary": {
            "BillingMode": "PROVISIONED",
            "LastUpdateToPayPerRequestDateTime": "2020-07-28T13:14:48.366000-07:00"
        },
        "GlobalSecondaryIndexes": [
            {
                "IndexName": "AlbumTitle-index",
                "KeySchema": [
                    {
                        "AttributeName": "AlbumTitle",
                        "KeyType": "HASH"
                    }
                ],
                "Projection": {
                    "ProjectionType": "ALL"
                },
                "IndexStatus": "CREATING",
                "Backfilling": false,
                "ProvisionedThroughput": {
                    "NumberOfDecreasesToday": 0,
                    "ReadCapacityUnits": 10,
                    "WriteCapacityUnits": 10
                },
                "IndexSizeBytes": 0,
                "ItemCount": 0,
                "IndexArn": "arn:aws:dynamodb:us-west-2:123456789012:table/
MusicCollection/index/AlbumTitle-index"
            }
        ]
    }
}
```

Per ulteriori informazioni, consulta Aggiornamento di una tabella nella Guida per gli sviluppatori di Amazon DynamoDB.

Esempio 3: come abilitare i flussi DynamoDB in una tabella

Il comando seguente abilita i flussi Amazon DynamoDB sulla tabella `MusicCollection`.

```
aws dynamodb update-table \
    --table-name MusicCollection \
    --stream-specification StreamEnabled=true,StreamViewType=NEW_IMAGE
```

Output:

```
{
    "TableDescription": {
        "AttributeDefinitions": [
            {
                "AttributeName": "AlbumTitle",
                "AttributeType": "S"
            },
            {
                "AttributeName": "Artist",
                "AttributeType": "S"
            },
            {
                "AttributeName": "SongTitle",
                "AttributeType": "S"
            }
        ],
        "TableName": "MusicCollection",
        "KeySchema": [
            {
                "AttributeName": "Artist",
                "KeyType": "HASH"
            },
            {
                "AttributeName": "SongTitle",
                "KeyType": "RANGE"
            }
        ],
        "TableStatus": "UPDATING",
        "CreationDateTime": "2020-05-26T15:59:49.473000-07:00",
        "ProvisionedThroughput": {
            "LastIncreaseDateTime": "2020-07-28T12:59:17.537000-07:00",
            "NumberOfDecreasesToday": 0,
            "ReadCapacityUnits": 15,
            "WriteCapacityUnits": 10
        },
```

```
        "TableSizeBytes": 182,
        "ItemCount": 2,
        "TableArn": "arn:aws:dynamodb:us-west-2:123456789012:table/MusicCollection",
        "TableId": "abcd0123-01ab-23cd-0123-abcdef123456",
        "BillingModeSummary": {
            "BillingMode": "PROVISIONED",
            "LastUpdateToPayPerRequestDateTime": "2020-07-28T13:14:48.366000-07:00"
        },
        "LocalSecondaryIndexes": [
            {
                "IndexName": "AlbumTitleIndex",
                "KeySchema": [
                    {
                        "AttributeName": "Artist",
                        "KeyType": "HASH"
                    },
                    {
                        "AttributeName": "AlbumTitle",
                        "KeyType": "RANGE"
                    }
                ],
                "Projection": {
                    "ProjectionType": "INCLUDE",
                    "NonKeyAttributes": [
                        "Year",
                        "Genre"
                    ]
                },
                "IndexSizeBytes": 139,
                "ItemCount": 2,
                "IndexArn": "arn:aws:dynamodb:us-west-2:123456789012:table/
MusicCollection/index/AlbumTitleIndex"
            }
        ],
        "GlobalSecondaryIndexes": [
            {
                "IndexName": "AlbumTitle-index",
                "KeySchema": [
                    {
                        "AttributeName": "AlbumTitle",
                        "KeyType": "HASH"
                    }
                ],
                "Projection": {
```

```
                    "ProjectionType": "ALL"
                },
                "IndexStatus": "ACTIVE",
                "ProvisionedThroughput": {
                    "NumberOfDecreasesToday": 0,
                    "ReadCapacityUnits": 10,
                    "WriteCapacityUnits": 10
                },
                "IndexSizeBytes": 0,
                "ItemCount": 0,
                "IndexArn": "arn:aws:dynamodb:us-west-2:123456789012:table/
MusicCollection/index/AlbumTitle-index"
            }
        ],
        "StreamSpecification": {
            "StreamEnabled": true,
            "StreamViewType": "NEW_IMAGE"
        },
        "LatestStreamLabel": "2020-07-28T21:53:39.112",
        "LatestStreamArn": "arn:aws:dynamodb:us-west-2:123456789012:table/
MusicCollection/stream/2020-07-28T21:53:39.112"
    }
}
```

Per ulteriori informazioni, consulta Aggiornamento di una tabella nella Guida per gli sviluppatori di Amazon DynamoDB.

Esempio 4: come abilitare la crittografia lato server

L'esempio seguente abilita la crittografia lato server sulla tabella `MusicCollection`.

```
aws dynamodb update-table \
    --table-name MusicCollection \
    --sse-specification Enabled=true,SSEType=KMS
```

Output:

```
{
    "TableDescription": {
        "AttributeDefinitions": [
            {
                "AttributeName": "AlbumTitle",
                "AttributeType": "S"
```

```
            },
            {
                "AttributeName": "Artist",
                "AttributeType": "S"
            },
            {
                "AttributeName": "SongTitle",
                "AttributeType": "S"
            }
        ],
        "TableName": "MusicCollection",
        "KeySchema": [
            {
                "AttributeName": "Artist",
                "KeyType": "HASH"
            },
            {
                "AttributeName": "SongTitle",
                "KeyType": "RANGE"
            }
        ],
        "TableStatus": "ACTIVE",
        "CreationDateTime": "2020-05-26T15:59:49.473000-07:00",
        "ProvisionedThroughput": {
            "LastIncreaseDateTime": "2020-07-28T12:59:17.537000-07:00",
            "NumberOfDecreasesToday": 0,
            "ReadCapacityUnits": 15,
            "WriteCapacityUnits": 10
        },
        "TableSizeBytes": 182,
        "ItemCount": 2,
        "TableArn": "arn:aws:dynamodb:us-west-2:123456789012:table/MusicCollection",
        "TableId": "abcd0123-01ab-23cd-0123-abcdef123456",
        "BillingModeSummary": {
            "BillingMode": "PROVISIONED",
            "LastUpdateToPayPerRequestDateTime": "2020-07-28T13:14:48.366000-07:00"
        },
        "LocalSecondaryIndexes": [
            {
                "IndexName": "AlbumTitleIndex",
                "KeySchema": [
                    {
                        "AttributeName": "Artist",
                        "KeyType": "HASH"
```

```
                    },
                    {
                        "AttributeName": "AlbumTitle",
                        "KeyType": "RANGE"
                    }
                ],
                "Projection": {
                    "ProjectionType": "INCLUDE",
                    "NonKeyAttributes": [
                        "Year",
                        "Genre"
                    ]
                },
                "IndexSizeBytes": 139,
                "ItemCount": 2,
                "IndexArn": "arn:aws:dynamodb:us-west-2:123456789012:table/
MusicCollection/index/AlbumTitleIndex"
            }
        ],
        "GlobalSecondaryIndexes": [
            {
                "IndexName": "AlbumTitle-index",
                "KeySchema": [
                    {
                        "AttributeName": "AlbumTitle",
                        "KeyType": "HASH"
                    }
                ],
                "Projection": {
                    "ProjectionType": "ALL"
                },
                "IndexStatus": "ACTIVE",
                "ProvisionedThroughput": {
                    "NumberOfDecreasesToday": 0,
                    "ReadCapacityUnits": 10,
                    "WriteCapacityUnits": 10
                },
                "IndexSizeBytes": 0,
                "ItemCount": 0,
                "IndexArn": "arn:aws:dynamodb:us-west-2:123456789012:table/
MusicCollection/index/AlbumTitle-index"
            }
        ],
        "StreamSpecification": {
```

```
            "StreamEnabled": true,
            "StreamViewType": "NEW_IMAGE"
        },
        "LatestStreamLabel": "2020-07-28T21:53:39.112",
        "LatestStreamArn": "arn:aws:dynamodb:us-west-2:123456789012:table/
MusicCollection/stream/2020-07-28T21:53:39.112",
        "SSEDescription": {
            "Status": "UPDATING"
        }
    }
}
```

Per ulteriori informazioni, consulta Aggiornamento di una tabella nella Guida per gli sviluppatori di Amazon DynamoDB.

- Per i dettagli sull'API, consulta UpdateTable AWS CLICommand Reference.

`update-time-to-live`

Il seguente esempio di codice mostra come utilizzare`update-time-to-live`.

AWS CLI

Come aggiornare le impostazioni Time to Live in una tabella

L'esempio `update-time-to-live` seguente abilita le impostazioni Time to Live nella tabella specificata.

```
aws dynamodb update-time-to-live \
    --table-name MusicCollection \
    --time-to-live-specification Enabled=true,AttributeName=ttl
```

Output:

```
{
    "TimeToLiveSpecification": {
        "Enabled": true,
        "AttributeName": "ttl"
    }
}
```

Per ulteriori informazioni, consulta Time to Live nella Guida per gli sviluppatori di Amazon DynamoDB.

- Per i dettagli sull'API, consulta UpdateTimeToLive AWS CLICommand Reference.

Esempi di DynamoDB Streams utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface con DynamoDB Streams.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`describe-stream`

Il seguente esempio di codice mostra come utilizzare. `describe-stream`

AWS CLI

Come ottenere informazioni su un flusso DynamoDB

Il comando `describe-stream` seguente visualizza informazioni sul flusso DynamoDB specifico.

```
aws dynamodbstreams describe-stream \
    --stream-arn arn:aws:dynamodb:us-west-1:123456789012:table/Music/
stream/2019-10-22T18:02:01.576
```

Output:

```
{
    "StreamDescription": {
```

```
        "StreamArn": "arn:aws:dynamodb:us-west-1:123456789012:table/Music/
stream/2019-10-22T18:02:01.576",
        "StreamLabel": "2019-10-22T18:02:01.576",
        "StreamStatus": "ENABLED",
        "StreamViewType": "NEW_AND_OLD_IMAGES",
        "CreationRequestDateTime": 1571767321.571,
        "TableName": "Music",
        "KeySchema": [
            {
                "AttributeName": "Artist",
                "KeyType": "HASH"
            },
            {
                "AttributeName": "SongTitle",
                "KeyType": "RANGE"
            }
        ],
        "Shards": [
            {
                "ShardId": "shardId-00000001571767321804-697ce3d2",
                "SequenceNumberRange": {
                    "StartingSequenceNumber": "400000000000642977831",
                    "EndingSequenceNumber": "400000000000642977831"
                }
            },
            {
                "ShardId": "shardId-00000001571780995058-40810d86",
                "SequenceNumberRange": {
                    "StartingSequenceNumber": "757400000000005655171150"
                },
                "ParentShardId": "shardId-00000001571767321804-697ce3d2"
            }
        ]
    }
}
```

Per ulteriori informazioni, consulta Acquisizione dell'attività della tabella con flussi DynamoDB nella Guida per sviluppatori di Amazon DynamoDB.

- Per i dettagli sull'API, consulta DescribeStream AWS CLICommand Reference.

`get-records`

Il seguente esempio di codice mostra come utilizzare`get-records`.

AWS CLI

Come ottenere dei record da un flusso Dynamodb

Il comando `get-records` seguente recupera i record utilizzando l'iteratore di shard Amazon DynamoDB specificato.

```
aws dynamodbstreams get-records \
    --shard-iterator "arn:aws:dynamodb:us-west-1:123456789012:table/Music/
stream/2019-10-22T18:02:01.576|1|
AAAAAAAAAAGgM3YZ89vLZZxjmoQeo33r9M4x3+zmmTLsiL86MfrF4+B4EbsByi52InVmiONmy6xVW4IRcIIbs1zO7MNI
+CjNPlqQjnyRSAnfOwWmKhL1/KNParWSfz2odf780oOObIDIWRRMkt7+Hyzh9SD
+hFxFAWR5C7QIlOXPc8mRBfNIazfrVCjJK8/jsjCzsqNyXKzJbhh+GXCoxYN
+Kpmg4nyj1EAsYhbGL35muvHFoHjcyuynbsczbWaXNfThDwRAyvoTmc8XhHKtAWUbJiaVd8ZPtQwDsThCrmDRPIdmTRG
+w/lEGS05ha1qNP+Vl4+tuhz2TRnhnJo/pny9GI/yGpce97mWvSPr5KPwy+Dtcm5BHayBs
+PVYHITaTliInFlT
+LCwvaz1QH3MY3b8A05Z800wjpktm60iQqtMeDwN4NX6FrcxR34JoFKGsgR8XkHVJzz2xr1xqSJ12ycpNTyHnndusw==
```

Output:

```
{
    "Records": [
        {
            "eventID": "c3b5d798eef6215d42f8137b19a88e50",
            "eventName": "INSERT",
            "eventVersion": "1.1",
            "eventSource": "aws:dynamodb",
            "awsRegion": "us-west-1",
            "dynamodb": {
                "ApproximateCreationDateTime": 1571849028.0,
                "Keys": {
                    "Artist": {
                        "S": "No One You Know"
                    },
                    "SongTitle": {
                        "S": "Call Me Today"
                    }
                },
                "NewImage": {
```

```
                    "AlbumTitle": {
                        "S": "Somewhat Famous"
                    },
                    "Artist": {
                        "S": "No One You Know"
                    },
                    "Awards": {
                        "N": "1"
                    },
                    "SongTitle": {
                        "S": "Call Me Today"
                    }
                },
                "SequenceNumber": "700000000013256296913",
                "SizeBytes": 119,
                "StreamViewType": "NEW_AND_OLD_IMAGES"
            }
        },
        {
            "eventID": "878960a6967867e2da16b27380a27328",
            "eventName": "INSERT",
            "eventVersion": "1.1",
            "eventSource": "aws:dynamodb",
            "awsRegion": "us-west-1",
            "dynamodb": {
                "ApproximateCreationDateTime": 1571849029.0,
                "Keys": {
                    "Artist": {
                        "S": "Acme Band"
                    },
                    "SongTitle": {
                        "S": "Happy Day"
                    }
                },
                "NewImage": {
                    "AlbumTitle": {
                        "S": "Songs About Life"
                    },
                    "Artist": {
                        "S": "Acme Band"
                    },
                    "Awards": {
                        "N": "10"
                    },
```

```
                    "SongTitle": {
                        "S": "Happy Day"
                    }
                },
                "SequenceNumber": "800000000013256297217",
                "SizeBytes": 100,
                "StreamViewType": "NEW_AND_OLD_IMAGES"
            }
        },
        {
            "eventID": "520fabde080e159fc3710b15ee1d4daa",
            "eventName": "MODIFY",
            "eventVersion": "1.1",
            "eventSource": "aws:dynamodb",
            "awsRegion": "us-west-1",
            "dynamodb": {
                "ApproximateCreationDateTime": 1571849734.0,
                "Keys": {
                    "Artist": {
                        "S": "Acme Band"
                    },
                    "SongTitle": {
                        "S": "Happy Day"
                    }
                },
                "NewImage": {
                    "AlbumTitle": {
                        "S": "Updated Album Title"
                    },
                    "Artist": {
                        "S": "Acme Band"
                    },
                    "Awards": {
                        "N": "10"
                    },
                    "SongTitle": {
                        "S": "Happy Day"
                    }
                },
                "OldImage": {
                    "AlbumTitle": {
                        "S": "Songs About Life"
                    },
                    "Artist": {
```

```
                        "S": "Acme Band"
                    },
                    "Awards": {
                        "N": "10"
                    },
                    "SongTitle": {
                        "S": "Happy Day"
                    }
                },
                "SequenceNumber": "900000000013256687845",
                "SizeBytes": 170,
                "StreamViewType": "NEW_AND_OLD_IMAGES"
            }
        }
    ],
    "NextShardIterator": "arn:aws:dynamodb:us-west-1:123456789012:table/
Music/stream/2019-10-23T16:41:08.740|1|AAAAAAAAAAEhEI04jkFLW
+LKOwivjT8d/IHEh3iExV2xK00aTxEzVy1C1C7Kbb5+ZOW6bT9VQ2n1/
mrs7+PRiaOZCHJu7JHJVW7zlsqOi/ges3fw8GYEymyL+piEk35cx67rQqwKKyq
+Q6w9JyjreIOj4F2lWLV26lBwRTrIYC4IB7C3BZZK4715QwYdDxNdVHiSBRZX8UqoS6WOt0F87xZLNB9F/
NhYBLXi/wcGvAcBcC0TNIOH+N0NqwtoB/
FGCkNrf8YZ0xRoNN6RgGuVWHF3pxOhxEJeFZoSoJTIKeG9YcYxzi5Ci/
mhdtm7tBXnbw5c6xmsGsBqTirNjlDyJLcWl8Cl0UOLX63Ufo/5QliztcjEbKsQe28x8LM8o7VH1Is0fF/
ITt8awSA4igyJS0P87GN8Qri8kj8iaE35805jBHWF2wvwT6Iy2xGrR2r2HzYps9dwGOarVdEITaJfWzNoL4HajMhmREZ
+V04i1YIeHMXJfcwetNRuIbdQXfJht2NQZa4PVV6iknY6d19MrdbSTMKoqAuvp6g3Q2jH4t7GKCLWgodcPAn8g5+43Da
}
```


Per ulteriori informazioni, consulta Acquisizione dell'attività della tabella con flussi DynamoDB nella Guida per sviluppatori di Amazon DynamoDB.

- Per i dettagli sull'API, consulta GetRecords AWS CLICommand Reference.

`get-shard-iterator`

Il seguente esempio di codice mostra come utilizzare`get-shard-iterator`.

AWS CLI

Come ottenere un iteratore shard

Il comando `get-shard-iterator` seguente recupera un iteratore di shard per lo shard specificato.

```
aws dynamodbstreams get-shard-iterator \
    --stream-arn arn:aws:dynamodb:us-west-1:12356789012:table/Music/
stream/2019-10-22T18:02:01.576 \
    --shard-id shardId-00000001571780995058-40810d86 \
    --shard-iterator-type LATEST
```

Output:

```
{
    "ShardIterator": "arn:aws:dynamodb:us-west-1:123456789012:table/Music/
stream/2019-10-22T18:02:01.576|1|
AAAAAAAAAAGgM3YZ89vLZZxjmoQeo33r9M4x3+zmmTLsiL86MfrF4+B4EbsByi52InVmiONmy6xVW4IRcIIbs1zO7MNI
+CjNPlqQjnyRSAnfOwWmKhL1/KNParWSfz2odf780oOObIDIWRRMkt7+Hyzh9SD
+hFxFAWR5C7QIlOXPc8mRBfNIazfrVCjJK8/jsjCzsqNyXKzJbhh+GXCoxYN
+Kpmg4nyj1EAsYhbGL35muvHFoHjcyuynbsczbWaXNfThDwRAyvoTmc8XhHKtAWUbJiaVd8ZPtQwDsThCrmDRPIdmTRG
+w/lEGS05ha1qNP+Vl4+tuhz2TRnhnJo/pny9GI/yGpce97mWvSPr5KPwy+Dtcm5BHayBs
+PVYHITaTliInFlT
+LCwvaz1QH3MY3b8A05Z800wjpktm60iQqtMeDwN4NX6FrcxR34JoFKGsgR8XkHVJzz2xr1xqSJ12ycpNTyHnndusw==
}
```

Per ulteriori informazioni, consulta Acquisizione dell'attività della tabella con flussi DynamoDB nella Guida per sviluppatori di Amazon DynamoDB.

- Per i dettagli sull'API, consulta GetShardIterator AWS CLICommand Reference.

`list-streams`

Il seguente esempio di codice mostra come utilizzare`list-streams`.

AWS CLI

Come elencare i flussi DynamoDB

Il `list-streams` comando seguente elenca tutti i flussi Amazon DynamoDB esistenti all'interno della regione predefinita. AWS

```
aws dynamodbstreams list-streams
```

Output:

```
{
    "Streams": [
```

```
        {
            "StreamArn": "arn:aws:dynamodb:us-west-1:123456789012:table/Music/
stream/2019-10-22T18:02:01.576",
            "TableName": "Music",
            "StreamLabel": "2019-10-22T18:02:01.576"
        }
    ]
}
```

Per ulteriori informazioni, consulta Acquisizione dell'attività della tabella con flussi DynamoDB nella Guida per sviluppatori di Amazon DynamoDB.

- Per i dettagli sull'API, consulta Command ListStreamsReference AWS CLI .

EC2 Esempi di utilizzo di Amazon AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando il AWS Command Line Interface con Amazon EC2.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`accept-address-transfer`

Il seguente esempio di codice mostra come usare`accept-address-transfer`.

AWS CLI

Come accettare il trasferimento di un indirizzo IP elastico nell'account

L'esempio `accept-address-transfer` seguente accetta il trasferimento dell'indirizzo IP elastico specificato nell'account.

```
aws ec2 accept-address-transfer \
    --address 100.21.184.216
```

Output:

```
{
    "AddressTransfer": {
        "PublicIp": "100.21.184.216",
        "AllocationId": "eipalloc-09ad461b0d03f6aaf",
        "TransferAccountId": "123456789012",
        "TransferOfferExpirationTimestamp": "2023-02-22T20:51:10.000Z",
        "TransferOfferAcceptedTimestamp": "2023-02-22T22:52:54.000Z",
        "AddressTransferStatus": "accepted"
    }
}
```

Per ulteriori informazioni, consulta Trasferimento degli indirizzi IP elastici nella Guida per l'utente di Amazon VPC.

- Per i dettagli sull'API, consulta AcceptAddressTransfer AWS CLICommand Reference.

`accept-reserved-instances-exchange-quote`

Il seguente esempio di codice mostra come utilizzare`accept-reserved-instances-exchange-quote`.

AWS CLI

Come eseguire uno scambio di istanze riservate modificabili

Questo esempio esegue uno scambio delle istanze riservate modificabili specificate.

Comando:

```
aws ec2 accept-reserved-instances-exchange-quote --reserved-
instance-ids 7b8750c3-397e-4da4-bbcb-a45ebexample --target-
configurations OfferingId=b747b472-423c-48f3-8cee-679bcexample
```

Output:

```
{
```

```
  "ExchangeId": "riex-e68ed3c1-8bc8-4c17-af77-811afexample"
}
```

- Per i dettagli sull'API, consulta AcceptReservedInstancesExchangeQuote AWS CLICommand Reference.

`accept-transit-gateway-peering-attachment`

Il seguente esempio di codice mostra come utilizzare`accept-transit-gateway-peering-attachment`.

AWS CLI

Come accettare un accessorio di peering del gateway di transito

L'esempio `accept-transit-gateway-peering-attachment` seguente accetta l'accessorio di peering del gateway di transito specificato. Il parametro `--region` specifica la Regione in cui si trova il gateway di transito accettante.

```
aws ec2 accept-transit-gateway-peering-attachment \
    --transit-gateway-attachment-id tgw-attach-4455667788aabbccd \
    --region us-east-2
```

Output:

```
{
    "TransitGatewayPeeringAttachment": {
        "TransitGatewayAttachmentId": "tgw-attach-4455667788aabbccd",
        "RequesterTgwInfo": {
            "TransitGatewayId": "tgw-123abc05e04123abc",
            "OwnerId": "123456789012",
            "Region": "us-west-2"
        },
        "AccepterTgwInfo": {
            "TransitGatewayId": "tgw-11223344aabbcc112",
            "OwnerId": "123456789012",
            "Region": "us-east-2"
        },
        "State": "pending",
        "CreationTime": "2019-12-09T11:38:31.000Z"
    }
```

```
}
```

Per ulteriori informazioni, consulta Accessori di peering del gateway di transito nella Guida di Transit Gateway.

- Per i dettagli sull'API, consulta AcceptTransitGatewayPeeringAttachment AWS CLICommand Reference.

`accept-transit-gateway-vpc-attachment`

Il seguente esempio di codice mostra come utilizzare`accept-transit-gateway-vpc-attachment`.

AWS CLI

Come accettare una richiesta per collegare un VPC a un gateway di transito.

L'esempio `accept-transit-gateway-vpc-attachment` seguente accetta il collegamento specificato forte della richiesta.

```
aws ec2 accept-transit-gateway-vpc-attachment \
    --transit-gateway-attachment-id tgw-attach-0a34fe6b4fEXAMPLE
```

Output:

```
{
    "TransitGatewayVpcAttachment": {
        "TransitGatewayAttachmentId": "tgw-attach-0a34fe6b4fEXAMPLE",
        "TransitGatewayId": "tgw-0262a0e521EXAMPLE",
        "VpcId": "vpc-07e8ffd50fEXAMPLE",
        "VpcOwnerId": "123456789012",
        "State": "pending",
        "SubnetIds": [
            "subnet-0752213d59EXAMPLE"
        ],
        "CreationTime": "2019-07-10T17:33:46.000Z",
        "Options": {
            "DnsSupport": "enable",
            "Ipv6Support": "disable"
        }
    }
}
```

Per ulteriori informazioni, consulta Collegamenti del gateway di transito alla VPN a un VPC nella Guida di Transit Gateway.

- Per i dettagli sull'API, consulta AcceptTransitGatewayVpcAttachment AWS CLICommand Reference.

`accept-vpc-endpoint-connections`

Il seguente esempio di codice mostra come utilizzare`accept-vpc-endpoint-connections`.

AWS CLI

Come accettare le richieste di connessione endpoint dell'interfaccia

Questo esempio accetta la richiesta di connessione endpoint specificata per il servizio endpoint specificato.

Comando:

```
aws ec2 accept-vpc-endpoint-connections --service-id vpce-svc-03d5ebb7d9579a2b3 --
vpc-endpoint-ids vpce-0c1308d7312217abc
```

Output:

```
{
  "Unsuccessful": []
}
```

- Per i dettagli sull'API, consulta AcceptVpcEndpointConnections AWS CLICommand Reference.

`accept-vpc-peering-connection`

Il seguente esempio di codice mostra come utilizzare`accept-vpc-peering-connection`.

AWS CLI

Come accettare una connessione peering VPC

Questo esempio accetta la richiesta di connessione peering VPC specificata.

Comando:

```
aws ec2 accept-vpc-peering-connection --vpc-peering-connection-id pcx-1a2b3c4d
```

Output:

```
{
  "VpcPeeringConnection": {
    "Status": {
      "Message": "Provisioning",
      "Code": "provisioning"
    },
    "Tags": [],
    "AccepterVpcInfo": {
      "OwnerId": "444455556666",
      "VpcId": "vpc-44455566",
      "CidrBlock": "10.0.1.0/28"
    },
    "VpcPeeringConnectionId": "pcx-1a2b3c4d",
    "RequesterVpcInfo": {
      "OwnerId": "444455556666",
      "VpcId": "vpc-111abc45",
      "CidrBlock": "10.0.0.0/28"
    }
  }
}
```

- Per i dettagli sull'API, consulta AcceptVpcPeeringConnection AWS CLICommand Reference.

`advertise-byoip-cidr`

Il seguente esempio di codice mostra come utilizzare`advertise-byoip-cidr`.

AWS CLI

Come pubblicizzare un intervallo di indirizzi

L'`advertise-byoip-cidr`esempio seguente pubblicizza l'intervallo di IPv4 indirizzi pubblici specificato.

```
aws ec2 advertise-byoip-cidr \
    --cidr 203.0.113.25/24
```

Output:

```
{
    "ByoipCidr": {
        "Cidr": "203.0.113.25/24",
        "StatusMessage": "ipv4pool-ec2-1234567890abcdef0",
        "State": "provisioned"
    }
}
```

- Per i dettagli sull'API, vedere AdvertiseByoipCidrin AWS CLI Command Reference.

`allocate-address`

Il seguente esempio di codice mostra come utilizzare`allocate-address`.

AWS CLI

Esempio 1: come allocare un indirizzo IP elastico dal pool di indirizzi Amazon

Nell'esempio di `allocate-address` seguente viene allocato un indirizzo IP elastico. Amazon EC2 seleziona l'indirizzo dal pool di indirizzi di Amazon.

```
aws ec2 allocate-address
```

Output:

```
{
    "PublicIp": "70.224.234.241",
    "AllocationId": "eipalloc-01435ba59eEXAMPLE",
    "PublicIpv4Pool": "amazon",
    "NetworkBorderGroup": "us-west-2",
    "Domain": "vpc"
}
```

Per ulteriori informazioni, consulta gli indirizzi IP elastici nella Amazon EC2 User Guide.

Esempio 2: come allocare un indirizzo IP elastico e associarlo a un gruppo di confine di rete

Nell'esempio di `allocate-address` seguente viene allocato un indirizzo IP elastico e viene associato al gruppo di confine di rete specificato.

```
aws ec2 allocate-address \
    --network-border-group us-west-2-lax-1
```

Output:

```
{
    "PublicIp": "70.224.234.241",
    "AllocationId": "eipalloc-e03dd489ceEXAMPLE",
    "PublicIpv4Pool": "amazon",
    "NetworkBorderGroup": "us-west-2-lax-1",
    "Domain": "vpc"
}
```

Per ulteriori informazioni, consulta gli indirizzi IP elastici nella Amazon EC2 User Guide.

Esempio 3: come allocare un indirizzo IP elastico da un pool di indirizzi proprietario

Nell'esempio di `allocate-address` seguente viene allocato un indirizzo IP elastico da un pool di indirizzi trasferito sull'account Amazon Web Services. Amazon EC2 seleziona l'indirizzo dal pool di indirizzi.

```
aws ec2 allocate-address \
    --public-ipv4-pool ipv4pool-ec2-1234567890abcdef0
```

Output:

```
{
    "AllocationId": "eipalloc-02463d08ceEXAMPLE",
    "NetworkBorderGroup": "us-west-2",
    "CustomerOwnedIp": "18.218.95.81",
    "CustomerOwnedIpv4Pool": "ipv4pool-ec2-1234567890abcdef0",
    "Domain": "vpc"
    "NetworkBorderGroup": "us-west-2",
}
```

Per ulteriori informazioni, consulta gli indirizzi IP elastici nella Amazon EC2 User Guide.

Esempio 4: come allocare un indirizzo IP elastico dal pool IPAM

L'esempio `allocate-address` seguente alloca un indirizzo IP elastico /32 specifico a un pool Gestione indirizzi IP (IPAM) di Amazon VPC.

```
aws ec2 allocate-address \
    --region us-east-1 \
    --ipam-pool-id ipam-pool-1234567890abcdef0 \
    --address 192.0.2.0
```

Output:

```
{
    "PublicIp": "192.0.2.0",
    "AllocationId": "eipalloc-abcdef01234567890",
    "PublicIpv4Pool": "ipam-pool-1234567890abcdef0",
    "NetworkBorderGroup": "us-east-1",
    "Domain": "vpc"
}
```

Per ulteriori informazioni, consulta Assegnare indirizzi IP elastici sequenziali da un pool IPAM nella Guida per l'utente di Amazon VPC IPAM.

- Per i dettagli sull'API, consulta AllocateAddress AWS CLICommand Reference.

`allocate-hosts`

Il seguente esempio di codice mostra come utilizzare`allocate-hosts`.

AWS CLI

Esempio 1: come allocare un host dedicato

L'esempio `allocate-hosts` seguente alloca un unico host dedicato nella zona di disponibilità `eu-west-1a`, dove è possibile avviare le istanze `m5.large`. Per impostazione predefinita, l'host dedicato accetta solo l'avvio di istanze di destinazione e non supporta il ripristino dell'host.

```
aws ec2 allocate-hosts \
    --instance-type m5.large \
    --availability-zone eu-west-1a \
    --quantity 1
```

Output:

```
{
```

```
    "HostIds": [
        "h-07879acf49EXAMPLE"
    ]
}
```

Esempio 2: come allocare un host dedicato con posizionamento automatico e ripristino dell'host abilitati

L'esempio `allocate-hosts` seguente alloca un unico host dedicato nella zona di disponibilità `eu-west-1a` con posizionamento automatico e ripristino dell'host abilitati.

```
aws ec2 allocate-hosts \
    --instance-type m5.large \
    --availability-zone eu-west-1a \
    --auto-placement on \
    --host-recovery on \
    --quantity 1
```

Output:

```
{
     "HostIds": [
         "h-07879acf49EXAMPLE"
     ]
}
```

Esempio 3: come allocare un host dedicato con tag

L'esempio `allocate-hosts` seguente alloca un unico host dedicato e applica un tag con una chiave denominata `purpose` e il valore `production`.

```
aws ec2 allocate-hosts \
    --instance-type m5.large \
    --availability-zone eu-west-1a \
    --quantity 1 \
    --tag-specifications 'ResourceType=dedicated-
host,Tags={Key=purpose,Value=production}'
```

Output:

```
{
```

```
    "HostIds": [
        "h-07879acf49EXAMPLE"
    ]
}
```

Per ulteriori informazioni, consulta Allocate a Dedicated Host nella Amazon EC2 User Guide.

- Per i dettagli sull'API, consulta AllocateHosts AWS CLICommand Reference.

`allocate-ipam-pool-cidr`

Il seguente esempio di codice mostra come utilizzare`allocate-ipam-pool-cidr`.

AWS CLI

Come allocare un CIDR da un pool IPAM

L'esempio `allocate-ipam-pool-cidr` seguente alloca un CIDR da un pool IPAM.

Linux:

```
aws ec2 allocate-ipam-pool-cidr \
    --ipam-pool-id ipam-pool-0533048da7d823723 \
    --netmask-length 24
```

Windows:

```
aws ec2 allocate-ipam-pool-cidr ^
   --ipam-pool-id ipam-pool-0533048da7d823723 ^
   --netmask-length 24
```

Output:

```
{
    "IpamPoolAllocation": {
        "Cidr": "10.0.0.0/24",
        "IpamPoolAllocationId": "ipam-pool-alloc-018ecc28043b54ba38e2cd99943cebfbd",
        "ResourceType": "custom",
        "ResourceOwner": "123456789012"
    }
}
```

Per ulteriori informazioni, consulta Allocare manualmente un CIDR a un pool per riservare lo spazio degli indirizzi IP nella Guida per l'utente IPAM di Amazon VPC.

- Per i dettagli sull'API, consulta AllocateIpamPoolCidr AWS CLICommand Reference.

`apply-security-groups-to-client-vpn-target-network`

Il seguente esempio di codice mostra come utilizzare`apply-security-groups-to-client-vpn-target-network`.

AWS CLI

Come applicare gruppi di sicurezza a una rete di destinazione per un endpoint VPN Client

L'esempio `apply-security-groups-to-client-vpn-target-network` seguente applica il gruppo di sicurezza `sg-01f6e627a89f4db32` all'associazione tra la rete di destinazione specificata e l'endpoint VPN Client.

```
aws ec2 apply-security-groups-to-client-vpn-target-network \
    --security-group-ids sg-01f6e627a89f4db32 \
    --vpc-id vpc-0e2110c2f324332e0 \
    --client-vpn-endpoint-id cvpn-endpoint-123456789123abcde
```

Output:

```
{
    "SecurityGroupIds": [
        "sg-01f6e627a89f4db32"
    ]
}
```

Per ulteriori informazioni, consulta Reti target nella Guida per l'amministratore di VPN Client di AWS .

- Per i dettagli sull'API, consulta ApplySecurityGroupsToClientVpnTargetNetwork AWS CLICommand Reference.

`assign-ipv6-addresses`

Il seguente esempio di codice mostra come utilizzare`assign-ipv6-addresses`.

AWS CLI

Per assegnare IPv6 indirizzi specifici a un'interfaccia di rete

Questo esempio assegna IPv6 gli indirizzi specificati all'interfaccia di rete specificata.

Comando:

```
aws ec2 assign-ipv6-addresses --network-interface-id eni-38664473 --ipv6-
addresses 2001:db8:1234:1a00:3304:8879:34cf:4071 2001:db8:1234:1a00:9691:9503:25ad:1761
```

Output:

```
{
  "AssignedIpv6Addresses": [
      "2001:db8:1234:1a00:3304:8879:34cf:4071",
      "2001:db8:1234:1a00:9691:9503:25ad:1761"
  ],
  "NetworkInterfaceId": "eni-38664473"
}
```

Per assegnare IPv6 gli indirizzi selezionati da Amazon a un'interfaccia di rete

Questo esempio assegna due IPv6 indirizzi all'interfaccia di rete specificata. Amazon assegna automaticamente questi IPv6 indirizzi dagli IPv6 indirizzi disponibili nell'intervallo di blocchi IPv6 CIDR della sottorete.

Comando:

```
aws ec2 assign-ipv6-addresses --network-interface-id eni-38664473 --ipv6-address-
count 2
```

Output:

```
{
  "AssignedIpv6Addresses": [
      "2001:db8:1234:1a00:3304:8879:34cf:4071",
      "2001:db8:1234:1a00:9691:9503:25ad:1761"
  ],
  "NetworkInterfaceId": "eni-38664473"
```

```
}
```

- Per i dettagli sull'API, consulta AssignIpv6Addresses in Command Reference.AWS CLI

`assign-private-ip-addresses`

Il seguente esempio di codice mostra come utilizzare`assign-private-ip-addresses`.

AWS CLI

Come assegnare un indirizzo IP privato secondario specifico a un'interfaccia di rete

Questo esempio assegna l'indirizzo IP privato secondario specificato all'interfaccia di rete specificata. Se il comando va a buon fine, non viene restituito alcun output.

Comando:

```
aws ec2 assign-private-ip-addresses --network-interface-id eni-e5aa89a3 --private-ip-addresses 10.0.0.82
```

Per assegnare indirizzi IP privati secondari EC2 selezionati da Amazon a un'interfaccia di rete

Questo esempio assegna due indirizzi IP privati secondari all'interfaccia di rete specificata. Amazon assegna EC2 automaticamente questi indirizzi IP dagli indirizzi IP disponibili nell'intervallo di blocchi CIDR della sottorete a cui è associata l'interfaccia di rete. Se il comando va a buon fine, non viene restituito alcun output.

Comando:

```
aws ec2 assign-private-ip-addresses --network-interface-id eni-e5aa89a3 --secondary-private-ip-address-count 2
```

- Per i dettagli sull'API, consulta Command AssignPrivateIpAddressesReference AWS CLI .

`assign-private-nat-gateway-address`

Il seguente esempio di codice mostra come utilizzare`assign-private-nat-gateway-address`.

AWS CLI

Come assegnare indirizzi IP privati al gateway NAT privato

L'esempio `assign-private-nat-gateway-address` seguente assegna due indirizzi IP privati al gateway NAT privato specificato.

```
aws ec2 assign-private-nat-gateway-address \
    --nat-gateway-id nat-1234567890abcdef0 \
    --private-ip-address-count 2
```

Output:

```
{
    "NatGatewayId": "nat-1234567890abcdef0",
    "NatGatewayAddresses": [
        {
            "NetworkInterfaceId": "eni-0065a61b324d1897a",
            "IsPrimary": false,
            "Status": "assigning"
        },
        {
            "NetworkInterfaceId": "eni-0065a61b324d1897a",
            "IsPrimary": false,
            "Status": "assigning"
        }
    ]
}
```

Per ulteriori informazioni, consulta Gateway NAT nella Guida per l'utente di Amazon VPC.

- Per i dettagli sull'API, consulta AssignPrivateNatGatewayAddress AWS CLICommand Reference.

`associate-address`

Il seguente esempio di codice mostra come utilizzare`associate-address`.

AWS CLI

Esempio 1: come associare un indirizzo IP elastico a un'istanza

L'`associate-address`esempio seguente associa un indirizzo IP elastico all' EC2 istanza specificata.

```
aws ec2 associate-address \
```

```
    --instance-id i-0b263919b6498b123 \
    --allocation-id eipalloc-64d5890a
```

Output:

```
{
    "AssociationId": "eipassoc-2bebb745"
}
```

Esempio 2: come associare un indirizzo IP elastico a un'interfaccia di rete

L'esempio `associate-address` seguente associa l'indirizzo IP elastico specificato all'interfaccia di rete specificata.

```
aws ec2 associate-address
    --allocation-id eipalloc-64d5890a \
    --network-interface-id eni-1a2b3c4d
```

Output:

```
{
    "AssociationId": "eipassoc-2bebb745"
}
```

Esempio 3: come associare un indirizzo IP elastico a un indirizzo IP privato

L'esempio `associate-address` seguente associa l'indirizzo IP elastico specificato all'indirizzo IP privato specificato nell'interfaccia di rete specificata.

```
aws ec2 associate-address \
    --allocation-id eipalloc-64d5890a \
    --network-interface-id eni-1a2b3c4d \
    --private-ip-address 10.0.0.85
```

Output:

```
{
    "AssociationId": "eipassoc-2bebb745"
}
```

Per ulteriori informazioni, consulta gli indirizzi IP elastici nella Amazon EC2 User Guide.

- Per i dettagli sull'API, consulta AssociateAddress AWS CLICommand Reference.

`associate-client-vpn-target-network`

Il seguente esempio di codice mostra come utilizzare`associate-client-vpn-target-network`.

AWS CLI

Come associare una rete di destinazione a un endpoint VPN Client

L'esempio `associate-client-vpn-target-network` seguente associa una sottorete a un endpoint VPN Client.

```
aws ec2 associate-client-vpn-target-network \
    --subnet-id subnet-0123456789abcabca \
    --client-vpn-endpoint-id cvpn-endpoint-123456789123abcde
```

Output:

```
{
    "AssociationId": "cvpn-assoc-12312312312312312",
    "Status": {
        "Code": "associating"
    }
}
```

Per ulteriori informazioni, consulta Reti target nella Guida per l'amministratore di VPN Client di AWS .

- Per i dettagli sull'API, consulta AssociateClientVpnTargetNetwork AWS CLICommand Reference.

`associate-dhcp-options`

Il seguente esempio di codice mostra come utilizzare`associate-dhcp-options`.

AWS CLI

Come associare un set di opzioni DHCP al VPC in uso

Questo esempio associa il set di opzioni DHCP specificato al VPC specificato. Se il comando va a buon fine, non viene restituito alcun output.

Comando:

```
aws ec2 associate-dhcp-options --dhcp-options-id dopt-d9070ebb --vpc-id vpc-a01106c2
```

Come associare il set di opzioni DHCP predefinito al VPC in uso

Nell'esempio seguente viene associato il set di opzioni DHCP predefinite al VPC specificato. Se il comando va a buon fine, non viene restituito alcun output.

Comando:

```
aws ec2 associate-dhcp-options --dhcp-options-id default --vpc-id vpc-a01106c2
```

- Per i dettagli sull'API, consulta AssociateDhcpOptions AWS CLICommand Reference.

`associate-iam-instance-profile`

Il seguente esempio di codice mostra come utilizzare`associate-iam-instance-profile`.

AWS CLI

Come associare un profilo dell'istanza IAM a un'istanza

Questo esempio associa un profilo dell'istanza IAM denominata `admin-role` all'istanza `i-123456789abcde123`.

Comando:

```
aws ec2 associate-iam-instance-profile --instance-id i-123456789abcde123 --iam-instance-profile Name=admin-role
```

Output:

```
{
  "IamInstanceProfileAssociation": {
      "InstanceId": "i-123456789abcde123",
      "State": "associating",
      "AssociationId": "iip-assoc-0e7736511a163c209",
```

```
        "IamInstanceProfile": {
            "Id": "AIPAJBLK7RKJKWDXVHIEC",
            "Arn": "arn:aws:iam::123456789012:instance-profile/admin-role"
        }
    }
}
```

- Per i dettagli sull'API, consulta AssociateIamInstanceProfile AWS CLICommand Reference.

`associate-instance-event-window`

Il seguente esempio di codice mostra come utilizzare`associate-instance-event-window`.

AWS CLI

Esempio 1: come associare una o più istanze a una finestra di eventi

L'esempio `associate-instance-event-window` seguente associa una o più istanze a una finestra di eventi.

```
aws ec2 associate-instance-event-window \
    --region us-east-1 \
    --instance-event-window-id iew-0abcdef1234567890 \
    --association-target "InstanceIds=i-1234567890abcdef0,i-0598c7d356eba48d7"
```

Output:

```
{
    "InstanceEventWindow": {
        "InstanceEventWindowId": "iew-0abcdef1234567890",
        "Name": "myEventWindowName",
        "CronExpression": "* 21-23 * * 2,3",
        "AssociationTarget": {
            "InstanceIds": [
                "i-1234567890abcdef0",
                "i-0598c7d356eba48d7"
            ],
            "Tags": [],
            "DedicatedHostIds": []
        },
        "State": "creating"
    }
```

```
}
```

Per i vincoli relativi alla finestra degli eventi, consulta Considerazioni nella sezione Eventi pianificati della Amazon EC2 User Guide.

Esempio 2: come associare i tag di istanza a una finestra di eventi

L'esempio `associate-instance-event-window` seguente associa i tag di istanza a una finestra di eventi. Immettere un parametro `instance-event-window-id` per specificare la finestra di eventi. Per associare i tag di istanza, specifica il parametro `association-target` e specifica uno o più tag per il valore del parametro.

```
aws ec2 associate-instance-event-window \
    --region us-east-1 \
    --instance-event-window-id iew-0abcdef1234567890 \
    --association-target "InstanceTags=[{Key=k2,Value=v2},{Key=k1,Value=v1}]"
```

Output:

```
{
    "InstanceEventWindow": {
        "InstanceEventWindowId": "iew-0abcdef1234567890",
        "Name": "myEventWindowName",
        "CronExpression": "* 21-23 * * 2,3",
        "AssociationTarget": {
            "InstanceIds": [],
            "Tags": [
                {
                    "Key": "k2",
                    "Value": "v2"
                },
                {
                    "Key": "k1",
                    "Value": "v1"
                }
            ],
            "DedicatedHostIds": []
        },
        "State": "creating"
    }
}
```

Per i vincoli relativi alla finestra degli eventi, consulta Considerazioni nella sezione Eventi pianificati della Amazon EC2 User Guide.

Esempio 3: come associare un host dedicato a una finestra di eventi

L'esempio `associate-instance-event-window` seguente associa un host dedicato a una finestra di eventi. Immettere un parametro `instance-event-window-id` per specificare la finestra di eventi. Per associare un host dedicato, specifica il `--association-target` parametro e, per i valori dei parametri, specifica uno o più host dedicati. IDs

```
aws ec2 associate-instance-event-window \
    --region us-east-1 \
    --instance-event-window-id iew-0abcdef1234567890 \
    --association-target "DedicatedHostIds=h-029fa35a02b99801d"
```

Output:

```
{
    "InstanceEventWindow": {
        "InstanceEventWindowId": "iew-0abcdef1234567890",
        "Name": "myEventWindowName",
        "CronExpression": "* 21-23 * * 2,3",
        "AssociationTarget": {
            "InstanceIds": [],
            "Tags": [],
            "DedicatedHostIds": [
                "h-029fa35a02b99801d"
            ]
        },
        "State": "creating"
    }
}
```

Per i vincoli relativi alla finestra degli eventi, consulta Considerazioni nella sezione Eventi pianificati della Amazon EC2 User Guide.

- Per i dettagli sull'API, consulta Command AssociateInstanceEventWindowReference AWS CLI .

`associate-ipam-resource-discovery`

Il seguente esempio di codice mostra come utilizzare`associate-ipam-resource-discovery`.

AWS CLI

Come associare un rilevamento delle risorse a un'IPAM

In questo esempio, sei un amministratore delegato IPAM e un rilevamento delle risorse è stato creato e condiviso con te da un altro AWS account in modo da poter utilizzare IPAM per gestire e monitorare le risorse di CIDRs proprietà dell'altro account.

Nota

Per completare questa richiesta, avrai bisogno dell'ID di scoperta della risorsa che puoi utilizzare describe-ipam-resource-discoveriese dell'ID IPAM che puoi ottenere con describe-ipams. L'individuazione delle risorse che stai associando deve essere stata prima condivisa con il tuo account utilizzando la AWS RAM. L'`--region`ID che inserisci deve corrispondere alla regione di origine dell'IPAM a cui la stai associando.

L'esempio `associate-ipam-resource-discovery` seguente associa un rilevamento di risorse a un'IPAM.

```
aws ec2 associate-ipam-resource-discovery \
    --ipam-id ipam-005f921c17ebd5107 \
    --ipam-resource-discovery-id ipam-res-disco-03e0406de76a044ee \
    --tag-specifications 'ResourceType=ipam-resource-discovery,Tags=[{Key=cost-
center,Value=cc123}]' \
    --region us-east-1
```

Output:

```
{
    {
        "IpamResourceDiscoveryAssociation": {
            "OwnerId": "320805250157",
            "IpamResourceDiscoveryAssociationId": "ipam-res-disco-
assoc-04382a6346357cf82",
            "IpamResourceDiscoveryAssociationArn": "arn:aws:ec2::320805250157:ipam-
resource-discovery-association/ipam-res-disco-assoc-04382a6346357cf82",
            "IpamResourceDiscoveryId": "ipam-res-disco-0365d2977fc1672fe",
            "IpamId": "ipam-005f921c17ebd5107",
            "IpamArn": "arn:aws:ec2::320805250157:ipam/ipam-005f921c17ebd5107",
            "IpamRegion": "us-east-1",
            "IsDefault": false,
            "ResourceDiscoveryStatus": "active",
```

```
            "State": "associate-in-progress",
            "Tags": []
        }
    }
}
```

Una volta associata l'individuazione di una risorsa, è possibile monitorare e and/or gestire gli indirizzi IP delle risorse create dagli altri account. Per ulteriori informazioni, consulta Come integrare IPAM con account esterni alla tua organizzazione nella Guida per l'utente IPAM di Amazon VPC.

- Per i dettagli sull'API, consulta AssociateIpamResourceDiscovery AWS CLICommand Reference.

`associate-nat-gateway-address`

Il seguente esempio di codice mostra come utilizzare`associate-nat-gateway-address`.

AWS CLI

Come associare un indirizzo IP elastico a un gateway NAT pubblico

L'`associate-nat-gateway-address`esempio seguente associa l'indirizzo IP elastico specificato al gateway NAT pubblico specificato. AWS assegna automaticamente un indirizzo privato secondario. IPv4

```
aws ec2 associate-nat-gateway-address \
    --nat-gateway-id nat-1234567890abcdef0 \
    --allocation-ids eipalloc-0be6ecac95EXAMPLE
```

Output:

```
{
    "NatGatewayId": "nat-1234567890abcdef0",
    "NatGatewayAddresses": [
        {
            "AllocationId": "eipalloc-0be6ecac95EXAMPLE",
            "NetworkInterfaceId": "eni-09cc4b2558794f7f9",
            "IsPrimary": false,
            "Status": "associating"
        }
```

```
    ]
}
```

Per ulteriori informazioni, consulta Gateway NAT nella Guida per l'utente di Amazon VPC.

- Per i dettagli sull'API, consulta AssociateNatGatewayAddress AWS CLICommand Reference.

`associate-route-table`

Il seguente esempio di codice mostra come utilizzare`associate-route-table`.

AWS CLI

Come associare una tabella di routing a una sottorete

Questo esempio associa la tabella di routing specificata alla sottorete specificata.

Comando:

```
aws ec2 associate-route-table --route-table-id rtb-22574640 --subnet-
id subnet-9d4a7b6c
```

Output:

```
{
    "AssociationId": "rtbassoc-781d0d1a"
}
```

- Per i dettagli sull'API, consulta AssociateRouteTable AWS CLICommand Reference.

`associate-security-group-vpc`

Il seguente esempio di codice mostra come utilizzare`associate-security-group-vpc`.

AWS CLI

Come associare un gruppo di sicurezza a un altro VPC

L'esempio `associate-security-group-vpc` seguente associa il gruppo di sicurezza specificato al VPC specificato.

```
aws ec2 associate-security-group-vpc \
```

```
    --group-id sg-04dbb43907d3f8a78 \
    --vpc-id vpc-0bf4c2739bc05a694
```

Output:

```
{
    "State": "associating"
}
```

Per ulteriori informazioni, consulta Associare gruppi di sicurezza a più gruppi VPCs nella Amazon VPC User Guide.

- Per i dettagli sull'API, consulta AssociateSecurityGroupVpc AWS CLICommand Reference.

`associate-subnet-cidr-block`

Il seguente esempio di codice mostra come utilizzare`associate-subnet-cidr-block`.

AWS CLI

Per associare un blocco IPv6 CIDR a una sottorete

Questo esempio associa un blocco IPv6 CIDR alla sottorete specificata.

Comando:

```
aws ec2 associate-subnet-cidr-block --subnet-id subnet-5f46ec3b --ipv6-cidr-
block 2001:db8:1234:1a00::/64
```

Output:

```
{
  "SubnetId": "subnet-5f46ec3b",
  "Ipv6CidrBlockAssociation": {
      "Ipv6CidrBlock": "2001:db8:1234:1a00::/64",
      "AssociationId": "subnet-cidr-assoc-3aa54053",
      "Ipv6CidrBlockState": {
          "State": "associating"
      }
  }
}
```

- Per i dettagli sull'API, vedere AssociateSubnetCidrBlockin AWS CLI Command Reference.

`associate-transit-gateway-multicast-domain`

Il seguente esempio di codice mostra come utilizzare`associate-transit-gateway-multicast-domain`.

AWS CLI

Come associare un gateway di transito a un dominio multicast (trasmissione uno a molti)

L'esempio `associate-transit-gateway-multicast-domain` seguente associa la sottorete e l'allegato specificati al dominio multicast (trasmissione uno a molti) specificato.

```
aws ec2 associate-transit-gateway-multicast-domain \
    --transit-gateway-multicast-domain-id tgw-mcast-domain-0c4905cef79d6e597 \
    --transit-gateway-attachment-id tgw-attach-028c1dd0f8f5cbe8e \
    --subnet-ids subnet-000de86e3b49c932a \
    --transit-gateway-multicast-domain-id tgw-mcast-domain-0c4905cef7EXAMPLE
```

Output:

```
{
    "Associations": {
        "TransitGatewayMulticastDomainId": "tgw-mcast-domain-0c4905cef79d6e597",
        "TransitGatewayAttachmentId": "tgw-attach-028c1dd0f8f5cbe8e",
        "ResourceId": "vpc-01128d2c240c09bd5",
        "ResourceType": "vpc",
        "Subnets": [
            {
                "SubnetId": "subnet-000de86e3b49c932a",
                "State": "associating"
            }
        ]
    }
}
```

Per ulteriori informazioni, consulta Domini multicast (trasmissione uno a molti) nella Guida di Transit Gateway.

- Per i dettagli sull'API, consulta AssociateTransitGatewayMulticastDomain AWS CLICommand Reference.

`associate-transit-gateway-route-table`

Il seguente esempio di codice mostra come utilizzare`associate-transit-gateway-route-table`.

AWS CLI

Come associare una tabella di routing del gateway di transito a un collegamento del gateway di transito alla VPN

L'esempio seguente associa la tabella di routing del gateway di transito specificata al collegamento VPC specificato.

```
aws ec2 associate-transit-gateway-route-table \
    --transit-gateway-route-table-id tgw-rtb-002573ed1eEXAMPLE \
    --transit-gateway-attachment-id tgw-attach-0b5968d3b6EXAMPLE
```

Output:

```
{
    "Association": {
        "TransitGatewayRouteTableId": "tgw-rtb-002573ed1eEXAMPLE",
        "TransitGatewayAttachmentId": "tgw-attach-0b5968d3b6EXAMPLE",
        "ResourceId": "vpc-0065acced4EXAMPLE",
        "ResourceType": "vpc",
        "State": "associating"
    }
}
```

Per ulteriori informazioni, consulta Associare una tabella di routing di un gateway di transito nella Guida di AWS Transit Gateway.

- Per i dettagli sull'API, consulta AssociateTransitGatewayRouteTable AWS CLICommand Reference.

`associate-vpc-cidr-block`

Il seguente esempio di codice mostra come utilizzare`associate-vpc-cidr-block`.

AWS CLI

Esempio 1: associare un blocco IPv6 CIDR fornito da Amazon a un VPC

L'`associate-vpc-cidr-block`esempio seguente associa un blocco IPv6 CIDR al VPC specificato. :

```
aws ec2 associate-vpc-cidr-block \
    --amazon-provided-ipv6-cidr-block \
    --ipv6-cidr-block-network-border-group us-west-2-lax-1  \
    --vpc-id vpc-8EXAMPLE
```

Output:

```
{
    "Ipv6CidrBlockAssociation": {
        "AssociationId": "vpc-cidr-assoc-0838ce7d9dEXAMPLE",
        "Ipv6CidrBlockState": {
            "State": "associating"
        },
        "NetworkBorderGroup": "us-west-2-lax-1"
    },
    "VpcId": "vpc-8EXAMPLE"
}
```

Esempio 2:Per associare un blocco IPv4 CIDR aggiuntivo a un VPC

L'`associate-vpc-cidr-block`esempio seguente associa il blocco IPv4 CIDR al `10.2.0.0/16` VPC specificato.

```
aws ec2 associate-vpc-cidr-block \
    --vpc-id vpc-1EXAMPLE \
    --cidr-block 10.2.0.0/16
```

Output:

```
{
    "CidrBlockAssociation": {
        "AssociationId": "vpc-cidr-assoc-2EXAMPLE",
        "CidrBlock": "10.2.0.0/16",
        "CidrBlockState": {
            "State": "associating"
        }
    },
    "VpcId": "vpc-1EXAMPLE"
```

```
}
```

- Per i dettagli sull'API, vedere AssociateVpcCidrBlockin AWS CLI Command Reference.

`attach-classic-link-vpc`

Il seguente esempio di codice mostra come utilizzare`attach-classic-link-vpc`.

AWS CLI

Per collegare (allegare) un'istanza EC2 -Classic a un VPC

Questo esempio collega l'istanza i-1234567890abcdef0 al VPC vpc-88888888 tramite il gruppo di sicurezza VPC sg-12312312.

Comando:

```
aws ec2 attach-classic-link-vpc --instance-id  i-1234567890abcdef0 --vpc-
id vpc-88888888 --groups sg-12312312
```

Output:

```
{
  "Return": true
}
```

- Per i dettagli sull'API, consulta AWS CLI Command AttachClassicLinkVpcReference.

`attach-internet-gateway`

Il seguente esempio di codice mostra come utilizzare`attach-internet-gateway`.

AWS CLI

Come collegare un gateway internet al VPC

L'esempio `attach-internet-gateway` seguente collega il gateway Internet specificato al VPC specificato.

```
aws ec2 attach-internet-gateway \
    --internet-gateway-id igw-0d0fb496b3EXAMPLE \
```

```
    --vpc-id vpc-0a60eb65b4EXAMPLE
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Gateway Internet nella Guida per l'utente di Amazon VPC.

- Per i dettagli sull'API, consulta AttachInternetGateway AWS CLICommand Reference.

`attach-network-interface`

Il seguente esempio di codice mostra come utilizzare`attach-network-interface`.

AWS CLI

Esempio 1: come collegare un'interfaccia di rete a un'istanza

L'esempio `attach-network-interface` seguente collega l'interfaccia di rete specificata all'istanza specificata.

```
aws ec2 attach-network-interface \
    --network-interface-id eni-0dc56a8d4640ad10a \
    --instance-id i-1234567890abcdef0 \
    --device-index 1
```

Output:

```
{
    "AttachmentId": "eni-attach-01a8fc87363f07cf9"
}
```

Per ulteriori informazioni, consulta Interfacce di rete elastiche nella Amazon EC2 User Guide.

Esempio 2: come collegare un'interfaccia di rete a un'istanza con più schede di rete

L'esempio `attach-network-interface` seguente collega l'interfaccia di rete specificata all'istanza e alla scheda di specificate.

```
aws ec2 attach-network-interface \
    --network-interface-id eni-07483b1897541ad83 \
    --instance-id i-01234567890abcdef \
    --network-card-index 1 \
    --device-index 1
```

Output:

```
{
    "AttachmentId": "eni-attach-0fbd7ee87a88cd06c"
}
```

Per ulteriori informazioni, consulta Interfacce di rete elastiche nella Amazon EC2 User Guide.

- Per i dettagli sull'API, consulta AttachNetworkInterface AWS CLICommand Reference.

`attach-verified-access-trust-provider`

Il seguente esempio di codice mostra come utilizzare`attach-verified-access-trust-provider`.

AWS CLI

Come collegare un provider di attendibilità a un'istanza

L'esempio `attach-verified-access-trust-provider` seguente collega il provider di attendibilità di accesso verificato specificato all'istanza di accesso verificato specificata.

```
aws ec2 attach-verified-access-trust-provider \
    --verified-access-instance-id vai-0ce000c0b7643abea \
    --verified-access-trust-provider-id vatp-0bb32de759a3e19e7
```

Output:

```
{
    "VerifiedAccessTrustProvider": {
        "VerifiedAccessTrustProviderId": "vatp-0bb32de759a3e19e7",
        "Description": "",
        "TrustProviderType": "user",
        "UserTrustProviderType": "iam-identity-center",
        "PolicyReferenceName": "idc",
        "CreationTime": "2023-08-25T19:00:38",
        "LastUpdatedTime": "2023-08-25T19:00:38"
    },
    "VerifiedAccessInstance": {
        "VerifiedAccessInstanceId": "vai-0ce000c0b7643abea",
        "Description": "",
        "VerifiedAccessTrustProviders": [
```

```
            {
                "VerifiedAccessTrustProviderId": "vatp-0bb32de759a3e19e7",
                "TrustProviderType": "user",
                "UserTrustProviderType": "iam-identity-center"
            }
        ],
        "CreationTime": "2023-08-25T18:27:56",
        "LastUpdatedTime": "2023-08-25T18:27:56"
    }
}
```

Per ulteriori informazioni, consulta Istanze di Accesso verificato nella Guida per l'utente di Accesso verificato da AWS .

- Per i dettagli sull'API, consulta AttachVerifiedAccessTrustProvider AWS CLICommand Reference.

`attach-volume`

Il seguente esempio di codice mostra come utilizzare`attach-volume`.

AWS CLI

Come collegare un volume a un'istanza

Questo esempio collega un volume (`vol-1234567890abcdef0`) a un'istanza (`i-01474ef662b89480`) come `/dev/sdf`.

Comando:

```
aws ec2 attach-volume --volume-id vol-1234567890abcdef0 --instance-
id i-01474ef662b89480 --device /dev/sdf
```

Output:

```
{
    "AttachTime": "YYYY-MM-DDTHH:MM:SS.000Z",
    "InstanceId": "i-01474ef662b89480",
    "VolumeId": "vol-1234567890abcdef0",
    "State": "attaching",
    "Device": "/dev/sdf"
}
```

- Per i dettagli sull'API, consulta AttachVolume AWS CLICommand Reference.

`attach-vpn-gateway`

Il seguente esempio di codice mostra come utilizzare`attach-vpn-gateway`.

AWS CLI

Come collegare un gateway privato virtuale a un VPC

L'esempio `attach-vpn-gateway` seguente collega il gateway privato virtuale specificato al VPC specificato.

```
aws ec2 attach-vpn-gateway \
    --vpn-gateway-id vgw-9a4cacf3 \
    --vpc-id vpc-a01106c2
```

Output:

```
{
    "VpcAttachment": {
        "State": "attaching",
        "VpcId": "vpc-a01106c2"
    }
}
```

- Per i dettagli sull'API, consulta AttachVpnGateway AWS CLICommand Reference.

`authorize-client-vpn-ingress`

Il seguente esempio di codice mostra come utilizzare`authorize-client-vpn-ingress`.

AWS CLI

Come aggiungere una regola di autorizzazione a un endpoint VPN Client

L'esempio `authorize-client-vpn-ingress` seguente aggiunge una regola di autorizzazione che concede a tutti i client l'accesso a Internet (`0.0.0.0/0`).

```
aws ec2 authorize-client-vpn-ingress \
```

```
    --client-vpn-endpoint-id cvpn-endpoint-123456789123abcde \
    --target-network-cidr 0.0.0.0/0 \
    --authorize-all-groups
```

Output:

```
{
    "Status": {
        "Code": "authorizing"
    }
}
```

Per ulteriori informazioni, consulta Regole di autorizzazione nella Guida per l'amministratore di VPN Client di AWS .

- Per i dettagli sull'API, consulta AuthorizeClientVpnIngress AWS CLICommand Reference.

`authorize-security-group-egress`

Il seguente esempio di codice mostra come utilizzare`authorize-security-group-egress`.

AWS CLI

Esempio 1: come aggiungere una regola che consenta il traffico in uscita verso un intervallo di indirizzi specifico

L'esempio `authorize-security-group-egress` seguente aggiunge una regola che concede l'accesso agli intervalli di indirizzi specificati sulla porta TCP 80.

```
aws ec2 authorize-security-group-egress \
    --group-id sg-1234567890abcdef0 \
    --ip-permissions
 'IpProtocol=tcp,FromPort=80,ToPort=80,IpRanges=[{CidrIp=10.0.0.0/16}]'
```

Output:

```
{
    "Return": true,
    "SecurityGroupRules": [
        {
            "SecurityGroupRuleId": "sgr-0b15794cdb17bf29c",
```

```
            "GroupId": "sg-1234567890abcdef0",
            "GroupOwnerId": "123456789012",
            "IsEgress": true,
            "IpProtocol": "tcp",
            "FromPort": 80,
            "ToPort": 80,
            "CidrIpv4": "10.0.0.0/16"
        }
    ]
}
```

Esempio 2: come aggiungere una regola che consenta il traffico in uscita verso un gruppo di sicurezza specifico

L'esempio `authorize-security-group-egress` seguente aggiunge una regola che concede l'accesso al gruppo di sicurezza specificato sulla porta TCP 80.

```
aws ec2 authorize-security-group-egress \
    --group-id sg-1234567890abcdef0 \
    --ip-permissions
 'IpProtocol=tcp,FromPort=80,ToPort=80,UserIdGroupPairs=[{GroupId=sg-0aad1c26bbeec5c22}]'
```

Output:

```
{
    "Return": true,
    "SecurityGroupRules": [
        {
            "SecurityGroupRuleId": "sgr-0b5dd815afcea9cc3",
            "GroupId": "sg-1234567890abcdef0",
            "GroupOwnerId": "123456789012",
            "IsEgress": true,
            "IpProtocol": "tcp",
            "FromPort": 80,
            "ToPort": 80,
            "ReferencedGroupInfo": {
                "GroupId": "sg-0aad1c26bbeec5c22",
                "UserId": "123456789012"
            }
        }
    ]
}
```

Per ulteriori informazioni, consulta Gruppi di sicurezza nella Guida per l'utente di Amazon VPC.

- Per i dettagli sull'API, consulta AuthorizeSecurityGroupEgress AWS CLICommand Reference.

`authorize-security-group-ingress`

Il seguente esempio di codice mostra come utilizzare`authorize-security-group-ingress`.

AWS CLI

Esempio 1: come aggiungere una regola che consenta il traffico SSH in entrata

Nell'esempio di `authorize-security-group-ingress` seguente viene aggiunta una regola che consente il traffico in entrata nella porta TCP 22 (SSH).

```
aws ec2 authorize-security-group-ingress \
    --group-id sg-1234567890abcdef0 \
    --protocol tcp \
    --port 22 \
    --cidr 203.0.113.0/24
```

Output:

```
{
    "Return": true,
    "SecurityGroupRules": [
        {
            "SecurityGroupRuleId": "sgr-01afa97ef3e1bedfc",
            "GroupId": "sg-1234567890abcdef0",
            "GroupOwnerId": "123456789012",
            "IsEgress": false,
            "IpProtocol": "tcp",
            "FromPort": 22,
            "ToPort": 22,
            "CidrIpv4": "203.0.113.0/24"
        }
    ]
}
```

Esempio 2: come aggiungere una regola che consenta il traffico HTTP in entrata da un altro gruppo di sicurezza

Nell'esempio di `authorize-security-group-ingress` seguente viene aggiunta una regola che consente l'accesso in entrata alla porta TCP 80 dal gruppo di sicurezza di origine `sg-1a2b3c4d`. Il gruppo di origine deve trovarsi nello stesso VPC o in un VPC peer (richiede una connessione peering VPC). Il traffico in entrata è autorizzato in base agli indirizzi IP privati delle istanze associate al gruppo di sicurezza di origine (e non in base all'indirizzo IP elastico o pubblico).

```
aws ec2 authorize-security-group-ingress \
    --group-id sg-1234567890abcdef0 \
    --protocol tcp \
    --port 80 \
    --source-group sg-1a2b3c4d
```

Output:

```
{
    "Return": true,
    "SecurityGroupRules": [
        {
            "SecurityGroupRuleId": "sgr-01f4be99110f638a7",
            "GroupId": "sg-1234567890abcdef0",
            "GroupOwnerId": "123456789012",
            "IsEgress": false,
            "IpProtocol": "tcp",
            "FromPort": 80,
            "ToPort": 80,
            "ReferencedGroupInfo": {
                "GroupId": "sg-1a2b3c4d",
                "UserId": "123456789012"
            }
        }
    ]
}
```

Esempio 3: come aggiungere più regole nella stessa chiamata

Nell'esempio di `authorize-security-group-ingress` seguente viene utilizzato il parametro `ip-permissions` per aggiungere due regole in entrata, una che consenta l'accesso in entrata sulla porta TCP 3389 (RDP) e l'altra che consenta ping/ICMP.

```
aws ec2 authorize-security-group-ingress \
```

```
    --group-id sg-1234567890abcdef0 \
    --ip-permissions
 'IpProtocol=tcp,FromPort=3389,ToPort=3389,IpRanges=[{CidrIp=172.31.0.0/16}]'
 'IpProtocol=icmp,FromPort=-1,ToPort=-1,IpRanges=[{CidrIp=172.31.0.0/16}]'
```

Output:

```
{
    "Return": true,
    "SecurityGroupRules": [
        {
            "SecurityGroupRuleId": "sgr-00e06e5d3690f29f3",
            "GroupId": "sg-1234567890abcdef0",
            "GroupOwnerId": "123456789012",
            "IsEgress": false,
            "IpProtocol": "tcp",
            "FromPort": 3389,
            "ToPort": 3389,
            "CidrIpv4": "172.31.0.0/16"
        },
        {
            "SecurityGroupRuleId": "sgr-0a133dd4493944b87",
            "GroupId": "sg-1234567890abcdef0",
            "GroupOwnerId": "123456789012",
            "IsEgress": false,
            "IpProtocol": "tcp",
            "FromPort": -1,
            "ToPort": -1,
            "CidrIpv4": "172.31.0.0/16"
        }
    ]
}
```

Esempio 4: come aggiungere una regola per il traffico ICMP

L'esempio `authorize-security-group-ingress` seguente utilizza il parametro `ip-permissions` per aggiungere una regola in entrata che consenta il messaggio ICMP `Destination Unreachable: Fragmentation Needed and Don't Fragment was Set` (tipo 3, codice 4) da qualsiasi luogo.

```
aws ec2 authorize-security-group-ingress \
    --group-id sg-1234567890abcdef0 \
```

```
    --ip-permissions
 'IpProtocol=icmp,FromPort=3,ToPort=4,IpRanges=[{CidrIp=0.0.0.0/0}]'
```

Output:

```
{
    "Return": true,
    "SecurityGroupRules": [
        {
            "SecurityGroupRuleId": "sgr-0de3811019069b787",
            "GroupId": "sg-1234567890abcdef0",
            "GroupOwnerId": "123456789012",
            "IsEgress": false,
            "IpProtocol": "icmp",
            "FromPort": 3,
            "ToPort": 4,
            "CidrIpv4": "0.0.0.0/0"
        }
    ]
}
```

Esempio 5: aggiungere una regola per il IPv6 traffico

L'`authorize-security-group-ingress`esempio seguente utilizza il `ip-permissions` parametro per aggiungere una regola in entrata che consente l'accesso SSH (porta 22) dall' IPv6 intervallo. `2001:db8:1234:1a00::/64`

```
aws ec2 authorize-security-group-ingress \
    --group-id sg-1234567890abcdef0 \
    --ip-permissions
 'IpProtocol=tcp,FromPort=22,ToPort=22,Ipv6Ranges=[{CidrIpv6=2001:db8:1234:1a00::/64}]'
```

Output:

```
{
    "Return": true,
    "SecurityGroupRules": [
        {
            "SecurityGroupRuleId": "sgr-0455bc68b60805563",
            "GroupId": "sg-1234567890abcdef0",
            "GroupOwnerId": "123456789012",
```

```
            "IsEgress": false,
            "IpProtocol": "tcp",
            "FromPort": 22,
            "ToPort": 22,
            "CidrIpv6": "2001:db8:1234:1a00::/64"
        }
    ]
}
```

Esempio 6: per aggiungere una regola per il traffico ICMPv6

L'`authorize-security-group-ingress`esempio seguente utilizza il `ip-permissions` parametro per aggiungere una regola in entrata che consente il ICMPv6 traffico da qualsiasi luogo.

```
aws ec2 authorize-security-group-ingress \
    --group-id sg-1234567890abcdef0 \
    --ip-permissions 'IpProtocol=icmpv6,Ipv6Ranges=[{CidrIpv6=::/0}]'
```

Output:

```
{
    "Return": true,
    "SecurityGroupRules": [
        {
            "SecurityGroupRuleId": "sgr-04b612d9363ab6327",
            "GroupId": "sg-1234567890abcdef0",
            "GroupOwnerId": "123456789012",
            "IsEgress": false,
            "IpProtocol": "icmpv6",
            "FromPort": -1,
            "ToPort": -1,
            "CidrIpv6": "::/0"
        }
    ]
}
```

Esempio 7: aggiungere una regola con una descrizione

L'`authorize-security-group-ingress`esempio seguente utilizza il `ip-permissions` parametro per aggiungere una regola in entrata che consente il traffico RDP dall'intervallo di

indirizzi specificato IPv4 . La regola include una descrizione per consentirne l'identificazione in un secondo momento.

```
aws ec2 authorize-security-group-ingress \
    --group-id sg-1234567890abcdef0 \
    --ip-permissions
 'IpProtocol=tcp,FromPort=3389,ToPort=3389,IpRanges=[{CidrIp=203.0.113.0/24,Description='RDP
 office'}]'
```

Output:

```
{
    "Return": true,
    "SecurityGroupRules": [
        {
            "SecurityGroupRuleId": "sgr-0397bbcc01e974db3",
            "GroupId": "sg-1234567890abcdef0",
            "GroupOwnerId": "123456789012",
            "IsEgress": false,
            "IpProtocol": "tcp",
            "FromPort": 3389,
            "ToPort": 3389,
            "CidrIpv4": "203.0.113.0/24",
            "Description": "RDP access from NY office"
        }
    ]
}
```

Esempio 8: come aggiungere una regola in entrata che utilizza un elenco di prefissi

L'esempio `authorize-security-group-ingress` seguente utilizza il parametro `ip-permissions` per aggiungere una regola in entrata che consenta tutto il traffico per gli intervalli CIDR nell'elenco di prefissi specificato.

```
aws ec2 authorize-security-group-ingress \
    --group-id sg-04a351bfe432d4e71 \
    --ip-permissions
 'IpProtocol=all,PrefixListIds=[{PrefixListId=pl-002dc3ec097de1514}]'
```

Output:

```
{
```

```
    "Return": true,
    "SecurityGroupRules": [
        {
            "SecurityGroupRuleId": "sgr-09c74b32f677c6c7c",
            "GroupId": "sg-1234567890abcdef0",
            "GroupOwnerId": "123456789012",
            "IsEgress": false,
            "IpProtocol": "-1",
            "FromPort": -1,
            "ToPort": -1,
            "PrefixListId": "pl-0721453c7ac4ec009"
        }
    ]
}
```

Per ulteriori informazioni, consulta Gruppi di sicurezza nella Guida per l'utente di Amazon VPC.

- Per i dettagli sull'API, consulta AWS CLI Command AuthorizeSecurityGroupIngressReference.

`bundle-instance`

Il seguente esempio di codice mostra come utilizzare`bundle-instance`.

AWS CLI

Come creare il bundle di un'istanza

Questo esempio crea il bundle dell'istanza `i-1234567890abcdef0` in un bucket denominato `bundletasks`. Prima di specificare i valori per la chiave di accesso IDs, consulta e segui le indicazioni contenute nelle migliori pratiche per la gestione delle chiavi di AWS accesso.

Comando:

```
aws ec2 bundle-instance --instance-id i-1234567890abcdef0 --bucket bundletasks --
prefix winami --owner-akid AK12AJEXAMPLE --owner-sak example123example
```

Output:

```
{
  "BundleTask": {
    "UpdateTime": "2015-09-15T13:30:35.000Z",
```

```
    "InstanceId": "i-1234567890abcdef0",
    "Storage": {
      "S3": {
        "Prefix": "winami",
        "Bucket": "bundletasks"
      }
    },
    "State": "pending",
    "StartTime": "2015-09-15T13:30:35.000Z",
    "BundleId": "bun-294e041f"
  }
}
```

- Per i dettagli sull'API, consulta BundleInstance AWS CLICommand Reference.

`cancel-bundle-task`

Il seguente esempio di codice mostra come utilizzare`cancel-bundle-task`.

AWS CLI

Come annullare un'attività bundle

Questo esempio annulla l'attività bundle `bun-2a4e041c`.

Comando:

```
aws ec2 cancel-bundle-task --bundle-id bun-2a4e041c
```

Output:

```
{
  "BundleTask": {
    "UpdateTime": "2015-09-15T13:27:40.000Z",
    "InstanceId": "i-1234567890abcdef0",
    "Storage": {
      "S3": {
        "Prefix": "winami",
        "Bucket": "bundletasks"
      }
    },
    "State": "cancelling",
```

```
    "StartTime": "2015-09-15T13:24:35.000Z",
    "BundleId": "bun-2a4e041c"
  }
}
```

- Per i dettagli sull'API, consulta CancelBundleTask AWS CLICommand Reference.

`cancel-capacity-reservation-fleets`

Il seguente esempio di codice mostra come utilizzare`cancel-capacity-reservation-fleets`.

AWS CLI

Come annullare un parco di prenotazione della capacità

L'esempio `cancel-capacity-reservation-fleets` seguente annulla il parco di prenotazione della capacità specificata e la capacità che viene riservata. Quando annulli un parco istanze, il suo stato cambia in `cancelled` e non può più creare nuove prenotazioni di capacità. Inoltre, tutte le singole prenotazioni di capacità nel parco istanze vengono annullate e le istanze precedentemente in esecuzione nella capacità riservata continuano a funzionare normalmente in capacità condivisa.

```
aws ec2 cancel-capacity-reservation-fleets \
    --capacity-reservation-fleet-ids crf-abcdef01234567890
```

Output:

```
{
    "SuccessfulFleetCancellations": [
        {
            "CurrentFleetState": "cancelling",
            "PreviousFleetState": "active",
            "CapacityReservationFleetId": "crf-abcdef01234567890"
        }
    ],
    "FailedFleetCancellations": []
}
```

Per ulteriori informazioni sulle flotte di prenotazione della capacità, consulta Flotte di prenotazione della capacità nella Guida per gli EC2 utenti di Amazon.

- Per i dettagli sull'API, consulta AWS CLI Command CancelCapacityReservationFleetsReference.

`cancel-capacity-reservation`

Il seguente esempio di codice mostra come utilizzare`cancel-capacity-reservation`.

AWS CLI

Come annullare una prenotazione della capacità

L'esempio `cancel-capacity-reservation` seguente annulla la prenotazione della capacità specificata.

```
aws ec2 cancel-capacity-reservation \
    --capacity-reservation-id cr-1234abcd56EXAMPLE
```

Output:

```
{
    "Return": true
}
```

Per ulteriori informazioni, consulta Annullare una prenotazione di capacità nella Amazon EC2 User Guide.

- Per i dettagli sull'API, consulta CancelCapacityReservation AWS CLICommand Reference.

`cancel-conversion-task`

Il seguente esempio di codice mostra come utilizzare`cancel-conversion-task`.

AWS CLI

Come annullare una conversione attiva di un'istanza o di un volume

Questo esempio annulla il caricamento associato all'ID dell'attività import-i-fh 95npoc. Se il comando va a buon fine, non viene restituito alcun output.

Comando:

```
aws ec2 cancel-conversion-task --conversion-task-id import-i-fh95npoc
```

- Per i dettagli sull'API, consulta CancelConversionTaskCommand Reference.AWS CLI

`cancel-export-task`

Il seguente esempio di codice mostra come utilizzare`cancel-export-task`.

AWS CLI

Come annullare un'attività di esportazione attiva

Questo esempio annulla un'attività di esportazione attiva con l'ID attività export-i-fgelt 0i7. Se il comando va a buon fine, non viene restituito alcun output.

Comando:

```
aws ec2 cancel-export-task --export-task-id export-i-fgelt0i7
```

- Per i dettagli sull'API, consulta Command CancelExportTaskReference AWS CLI .

`cancel-image-launch-permission`

Il seguente esempio di codice mostra come utilizzare`cancel-image-launch-permission`.

AWS CLI

Come annullare un'AMI condivisa con l'account Amazon Web Services

L'esempio `cancel-image-launch-permission` seguente rimuove l'account dalle autorizzazione di avvio dell'AMI specificata:

```
aws ec2 cancel-image-launch-permission \
    --image-id ami-0123456789example \
    --region us-east-1
```

Output:

```
{
    "Return": true
```

```
}
```

Per ulteriori informazioni, consulta Annullare la condivisione di un'AMI con il tuo account Amazon Web Services nella Amazon EC2 User Guide.

- Per i dettagli sull'API, consulta CancelImageLaunchPermission AWS CLICommand Reference.

`cancel-import-task`

Il seguente esempio di codice mostra come utilizzare`cancel-import-task`.

AWS CLI

Come annullare un'attività di importazione

L'esempio `cancel-import-task` seguente annulla l'attività di importazione di immagini specificata.

```
aws ec2 cancel-import-task \
    --import-task-id import-ami-1234567890abcdef0
```

Output:

```
{
    "ImportTaskId": "import-ami-1234567890abcdef0",
    "PreviousState": "active",
    "State": "deleting"
}
```

- Per i dettagli sull'API, consulta CancelImportTask AWS CLICommand Reference.

`cancel-reserved-instances-listing`

Il seguente esempio di codice mostra come utilizzare`cancel-reserved-instances-listing`.

AWS CLI

Come annullare un elenco di istanze riservate

L'esempio `cancel-reserved-instances-listing` seguente annulla l'elenco di istanze riservate specificato.

```
aws ec2 cancel-reserved-instances-listing \
    --reserved-instances-listing-id 5ec28771-05ff-4b9b-aa31-9e57dexample
```

- Per i dettagli sull'API, consulta CancelReservedInstancesListing AWS CLICommand Reference.

`cancel-spot-fleet-requests`

Il seguente esempio di codice mostra come utilizzare`cancel-spot-fleet-requests`.

AWS CLI

Esempio 1: come annullare una richiesta del parco istanze spot e terminare le istanze associate

L'esempio `cancel-spot-fleet-requests` seguente annulla una richiesta del parco istanze spot e termina le istanze on demand e le istanze spot associate.

```
aws ec2 cancel-spot-fleet-requests \
    --spot-fleet-request-ids sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE \
    --terminate-instances
```

Output:

```
{
    "SuccessfulFleetRequests": [
        {
            "SpotFleetRequestId": "sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE",
            "CurrentSpotFleetRequestState": "cancelled_terminating",
            "PreviousSpotFleetRequestState": "active"
        }
    ],
    "UnsuccessfulFleetRequests": []
}
```

Esempio 2: come annullare una richiesta del parco istanze spot senza terminare le istanze associate

L'esempio `cancel-spot-fleet-requests` seguente annulla una richiesta del parco istanze spot senza terminare le istanze on demand e le istanze spot associate.

```
aws ec2 cancel-spot-fleet-requests \
```

```
    --spot-fleet-request-ids sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE \
    --no-terminate-instances
```

Output:

```
{
    "SuccessfulFleetRequests": [
        {
            "SpotFleetRequestId": "sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE",
            "CurrentSpotFleetRequestState": "cancelled_running",
            "PreviousSpotFleetRequestState": "active"
        }
    ],
    "UnsuccessfulFleetRequests": []
}
```

Per ulteriori informazioni, consulta Annullare una richiesta Spot Fleet nella Amazon EC2 User Guide.

- Per i dettagli sull'API, consulta CancelSpotFleetRequests AWS CLICommand Reference.

`cancel-spot-instance-requests`

Il seguente esempio di codice mostra come utilizzare`cancel-spot-instance-requests`.

AWS CLI

Come annullare le richieste di istanze spot

Questo comando di esempio annulla una richiesta di istanza spot.

Comando:

```
aws ec2 cancel-spot-instance-requests --spot-instance-request-ids sir-08b93456
```

Output:

```
{
    "CancelledSpotInstanceRequests": [
        {
            "State": "cancelled",
```

```
            "SpotInstanceRequestId": "sir-08b93456"
        }
    ]
}
```

- Per i dettagli sull'API, consulta CancelSpotInstanceRequests AWS CLICommand Reference.

`confirm-product-instance`

Il seguente esempio di codice mostra come utilizzare`confirm-product-instance`.

AWS CLI

Come confermare l'istanza del prodotto

Questo esempio determina se il codice prodotto specificato è associato all'istanza specificata.

Comando:

```
aws ec2 confirm-product-instance --product-code 774F4FF8 --instance-
id i-1234567890abcdef0
```

Output:

```
{
  "OwnerId": "123456789012"
}
```

- Per i dettagli sull'API, consulta ConfirmProductInstance AWS CLICommand Reference.

`copy-fpga-image`

Il seguente esempio di codice mostra come utilizzare`copy-fpga-image`.

AWS CLI

Come copiare un'immagine FPGA di Amazon (AFI)

Questo esempio copia l'AFI specificata dalla regione `us-east-1` alla regione corrente (`eu-west-1`).

Comando:

```
aws ec2 copy-fpga-image --name copy-afi --source-fpga-image-id afi-0d123e123bfc85abc
 --source-region us-east-1 --region eu-west-1
```

Output:

```
{
  "FpgaImageId": "afi-06b12350a123fbabc"
}
```

- Per i dettagli sull'API, consulta CopyFpgaImage AWS CLICommand Reference.

`copy-image`

Il seguente esempio di codice mostra come utilizzare`copy-image`.

AWS CLI

Esempio 1: come copiare un'AMI in un'altra Regione

L'esempio `copy-image` seguente copia l'AMI specificata dalla Regione `us-west-2` alla Regione `us-east-1` e aggiunge una breve descrizione.

```
aws ec2 copy-image \
    --region us-east-1 \
    --name ami-name \
    --source-region us-west-2 \
    --source-image-id ami-066877671789bd71b \
    --description "This is my copied image."
```

Output:

```
{
    "ImageId": "ami-0123456789abcdefg"
}
```

Per ulteriori informazioni, consulta Copiare un'AMI nella Amazon EC2 User Guide.

Esempio 2: come copiare un'AMI in un'altra Regione e crittografare lo snapshot di supporto

Il comando `copy-image` seguente copia l'AMI specificata dalla Regione `us-west-2` nella Regione corrente ed esegue la crittografia dello snapshot di supporto utilizzando la chiave KMS specificata.

```
aws ec2 copy-image \
    --source-region us-west-2 \
    --name ami-name \
    --source-image-id ami-066877671789bd71b \
    --encrypted \
    --kms-key-id alias/my-kms-key
```

Output:

```
{
    "ImageId": "ami-0123456789abcdefg"
}
```

Per ulteriori informazioni, consulta Copiare un'AMI nella Amazon EC2 User Guide.

Esempio 3: come includere i tag AMI definiti dall'utente durante la copia di un'AMI

Il comando `copy-image` seguente utilizza il parametro `--copy-image-tags` per copiare i tag AMI definiti dall'utente durante la copia dell'AMI.

```
aws ec2 copy-image \
    --region us-east-1 \
    --name ami-name \
    --source-region us-west-2 \
    --source-image-id ami-066877671789bd71b \
    --description "This is my copied image."
    --copy-image-tags
```

Output:

```
{
    "ImageId": "ami-0123456789abcdefg"
}
```

Per ulteriori informazioni, consulta Copiare un'AMI nella Amazon EC2 User Guide.

- Per i dettagli sull'API, consulta CopyImage AWS CLICommand Reference.

`copy-snapshot`

Il seguente esempio di codice mostra come utilizzare`copy-snapshot`.

AWS CLI

Esempio 1: come copiare uno snapshot in un'altra Regione

L'esempio `copy-snapshot` seguente copia lo snapshot specificato dalla Regione `us-west-2` nella Regione `us-east-1` e aggiunge una breve descrizione.

```
aws ec2 copy-snapshot \
    --region us-east-1 \
    --source-region us-west-2 \
    --source-snapshot-id snap-066877671789bd71b \
    --description 'This is my copied snapshot.'
```

Output:

```
{
    "SnapshotId": "snap-066877671789bd71b"
}
```

Esempio 2: copiare uno snapshot non crittografato e crittografare il nuovo snapshot

Il comando `copy-snapshot` seguente copia lo snapshot non crittografato specificato dalla Regione `us-west-2` nella Regione corrente ed esegue la crittografia del nuovo snapshot utilizzando la chiave KMS specificata.

```
aws ec2 copy-snapshot \
    --source-region us-west-2 \
    --source-snapshot-id snap-066877671789bd71b \
    --encrypted \
    --kms-key-id alias/my-kms-key
```

Output:

```
{
    "SnapshotId": "snap-066877671789bd71b"
```

```
}
```

Per ulteriori informazioni, consulta Copia di uno snapshot Amazon EBS nella Guida per l'utente di Amazon EBS.

- Per i dettagli sull'API, consulta CopySnapshot AWS CLICommand Reference.

`create-capacity-reservation-fleet`

Il seguente esempio di codice mostra come utilizzare`create-capacity-reservation-fleet`.

AWS CLI

Come creare un parco di prenotazione della capacità

L'esempio `create-capacity-reservation-fleet` seguente crea un parco di prenotazione della capacità per il tipo di istanza specificato nella richiesta, fino alla capacità di destinazione totale specificata. Il numero di istanze per le quali il parco istanze di prenotazione della capacità prenota quest'ultima dipende dalla capacità target totale e dai pesi del tipo di istanza specificati nella richiesta. Specifica i tipi di istanza da utilizzare e una priorità per ciascuno dei tipi di istanza designati.

```
aws ec2 create-capacity-reservation-fleet \
--total-target-capacity 24 \
--allocation-strategy prioritized \
--instance-match-criteria open \
--tenancy default \
--end-date 2022-12-31T23:59:59.000Z \
--instance-type-specifications file://instanceTypeSpecification.json
```

Contenuto di `instanceTypeSpecification.json`:

```
[
    {
        "InstanceType": "m5.xlarge",
        "InstancePlatform": "Linux/UNIX",
        "Weight": 3.0,
        "AvailabilityZone":"us-east-1a",
        "EbsOptimized": true,
        "Priority" : 1
    }
```

```
]
```

Output:

```
{
    "Status": "submitted",
    "TotalFulfilledCapacity": 0.0,
    "CapacityReservationFleetId": "crf-abcdef01234567890",
    "TotalTargetCapacity": 24
}
```

Per ulteriori informazioni sulle flotte di prenotazione della capacità, consulta Flotte di prenotazione della capacità nella Guida per gli EC2 utenti di Amazon.

Per ulteriori informazioni sul peso del tipo di istanza e sulla capacità target totale, consulta Peso del tipo di istanza e Capacità target totale nella Amazon EC2 User Guide.

Per ulteriori informazioni sulla designazione della priorità per tipi di istanze specifici, consulta Strategia di allocazione e Priorità del tipo di istanza nella Amazon EC2 User Guide.

- Per i dettagli sull'API, consulta AWS CLI Command CreateCapacityReservationFleetReference.

`create-capacity-reservation`

Il seguente esempio di codice mostra come utilizzare`create-capacity-reservation`.

AWS CLI

Esempio 1: come creare una prenotazione della capacità

L'`create-capacity-reservation`esempio seguente crea una prenotazione di capacità nella zona di `eu-west-1a` disponibilità, nella quale è possibile avviare tre `t2.medium` istanze che eseguono un sistema Linux/Unix operativo. Per impostazione predefinita, la prenotazione della capacità viene creata con criteri di corrispondenza delle istanze aperte e senza il supporto dell'archiviazione temporanea e rimane attiva finché non viene annullata manualmente.

```
aws ec2 create-capacity-reservation \
    --availability-zone eu-west-1a \
    --instance-type t2.medium \
    --instance-platform Linux/UNIX \
    --instance-count 3
```

Output:

```
{
    "CapacityReservation": {
        "CapacityReservationId": "cr-1234abcd56EXAMPLE ",
        "EndDateType": "unlimited",
        "AvailabilityZone": "eu-west-1a",
        "InstanceMatchCriteria": "open",
        "EphemeralStorage": false,
        "CreateDate": "2019-08-16T09:27:35.000Z",
        "AvailableInstanceCount": 3,
        "InstancePlatform": "Linux/UNIX",
        "TotalInstanceCount": 3,
        "State": "active",
        "Tenancy": "default",
        "EbsOptimized": false,
        "InstanceType": "t2.medium"
    }
}
```

Esempio 2: come creare una prenotazione della capacità che termini automaticamente a una data/ora specificata

L'`create-capacity-reservation`esempio seguente crea una prenotazione di capacità nella zona di `eu-west-1a` disponibilità, nella quale è possibile avviare tre `m5.large` istanze che eseguono un Linux/Unix sistema operativo. Questa prenotazione della capacità termina automaticamente il 31/08/2019 alle 23:59:59.

```
aws ec2 create-capacity-reservation \
    --availability-zone eu-west-1a \
    --instance-type m5.large \
    --instance-platform Linux/UNIX \
    --instance-count 3 \
    --end-date-type limited \
    --end-date 2019-08-31T23:59:59Z
```

Output:

```
{
    "CapacityReservation": {
        "CapacityReservationId": "cr-1234abcd56EXAMPLE ",
```

```
        "EndDateType": "limited",
        "AvailabilityZone": "eu-west-1a",
        "EndDate": "2019-08-31T23:59:59.000Z",
        "InstanceMatchCriteria": "open",
        "EphemeralStorage": false,
        "CreateDate": "2019-08-16T10:15:53.000Z",
        "AvailableInstanceCount": 3,
        "InstancePlatform": "Linux/UNIX",
        "TotalInstanceCount": 3,
        "State": "active",
        "Tenancy": "default",
        "EbsOptimized": false,
        "InstanceType": "m5.large"
    }
}
```

Esempio 3: come creare una prenotazione della capacità che accetta solo avvii di istanze mirate

L'esempio `create-capacity-reservation` seguente crea una prenotazione della capacità che accetta solo avvii di istanze di destinazione.

```
aws ec2 create-capacity-reservation \
    --availability-zone eu-west-1a \
    --instance-type m5.large \
    --instance-platform Linux/UNIX \
    --instance-count 3 \
    --instance-match-criteria targeted
```

Output:

```
{
    "CapacityReservation": {
        "CapacityReservationId": "cr-1234abcd56EXAMPLE ",
        "EndDateType": "unlimited",
        "AvailabilityZone": "eu-west-1a",
        "InstanceMatchCriteria": "targeted",
        "EphemeralStorage": false,
        "CreateDate": "2019-08-16T10:21:57.000Z",
        "AvailableInstanceCount": 3,
        "InstancePlatform": "Linux/UNIX",
        "TotalInstanceCount": 3,
        "State": "active",
        "Tenancy": "default",
```

```
        "EbsOptimized": false,
        "InstanceType": "m5.large"
    }
}
```

Per ulteriori informazioni, consulta Creare una prenotazione di capacità nella Amazon EC2 User Guide.

- Per i dettagli sull'API, consulta CreateCapacityReservation AWS CLICommand Reference.

`create-carrier-gateway`

Il seguente esempio di codice mostra come utilizzare`create-carrier-gateway`.

AWS CLI

Come creare un gateway carrier

L'esempio `create-carrier-gateway` seguente crea un gateway carrier per il VPC specificato.

```
aws ec2 create-carrier-gateway \
    --vpc-id vpc-0c529aEXAMPLE1111
```

Output:

```
{
    "CarrierGateway": {
        "CarrierGatewayId": "cagw-0465cdEXAMPLE1111",
        "VpcId": "vpc-0c529aEXAMPLE1111",
        "State": "pending",
        "OwnerId": "123456789012"
    }
}
```

Per ulteriori informazioni, consulta Gateway carrier nella Guida per l'utente di AWS Wavelength.

- Per i dettagli sull'API, consulta CreateCarrierGateway AWS CLICommand Reference.

`create-client-vpn-endpoint`

Il seguente esempio di codice mostra come utilizzare`create-client-vpn-endpoint`.

AWS CLI

Come creare un endpoint Client VPN

L'esempio `create-client-vpn-endpoint` seguente crea un endpoint VPN Client che utilizza l'autenticazione reciproca e specifica un valore per l'intervallo CIDR del client.

```
aws ec2 create-client-vpn-endpoint \
    --client-cidr-block "172.31.0.0/16" \
    --server-certificate-arn arn:aws:acm:ap-south-1:123456789012:certificate/
a1b2c3d4-5678-90ab-cdef-11111EXAMPLE \
    --authentication-options Type=certificate-
authentication,MutualAuthentication={ClientRootCertificateChainArn=arn:aws:acm:ap-
south-1:123456789012:certificate/a1b2c3d4-5678-90ab-cdef-22222EXAMPLE} \
    --connection-log-options Enabled=false
```

Output:

```
{
    "ClientVpnEndpointId": "cvpn-endpoint-123456789123abcde",
    "Status": {
        "Code": "pending-associate"
    },
    "DnsName": "cvpn-endpoint-123456789123abcde.prod.clientvpn.ap-
south-1.amazonaws.com"
}
```

Per ulteriori informazioni, consulta Endpoint Client VPN nella Guida per l'amministratore di VPN Client di AWS .

- Per i dettagli sull'API, consulta CreateClientVpnEndpoint AWS CLICommand Reference.

`create-client-vpn-route`

Il seguente esempio di codice mostra come utilizzare`create-client-vpn-route`.

AWS CLI

Come creare una route per l'endpoint VPN Client

L'esempio `create-client-vpn-route` seguente aggiunge una route a internet (`0.0.0.0/0`) per la sottorete specificata dell'endpoint VPN Client.

```
aws ec2 create-client-vpn-route \
    --client-vpn-endpoint-id cvpn-endpoint-123456789123abcde \
    --destination-cidr-block 0.0.0.0/0  \
    --target-vpc-subnet-id subnet-0123456789abcabca
```

Output:

```
{
    "Status": {
        "Code": "creating"
    }
}
```

Per ulteriori informazioni, consulta Route nella Guida per l'amministratore di VPN Client di AWS .

- Per i dettagli sull'API, consulta CreateClientVpnRoute AWS CLICommand Reference.

`create-coip-cidr`

Il seguente esempio di codice mostra come utilizzare`create-coip-cidr`.

AWS CLI

Come creare un intervallo di indirizzi IP di proprietà del cliente (CoIP)

L'esempio `create-coip-cidr` seguente crea l'intervallo specificato di indirizzi CoIP nel pool CoIP specificato.

```
aws ec2 create-coip-cidr \
    --cidr 15.0.0.0/24 \
    --coip-pool-id ipv4pool-coip-1234567890abcdefg
```

Output:

```
{
    "CoipCidr": {
        "Cidr": "15.0.0.0/24",
        "CoipPoolId": "ipv4pool-coip-1234567890abcdefg",
        "LocalGatewayRouteTableId": "lgw-rtb-abcdefg1234567890"
    }
```

```
}
```

Per ulteriori informazioni, consulta Indirizzi IP di proprietà del cliente nella Guida per l'utente di AWS Outposts.

- Per i dettagli sull'API, consulta CreateCoipCidr AWS CLICommand Reference.

`create-coip-pool`

Il seguente esempio di codice mostra come utilizzare`create-coip-pool`.

AWS CLI

Come creare un pool di indirizzi IP di proprietà del cliente (CoIP)

L'esempio `create-coip-pool` seguente crea un pool CoIP per gli indirizzi CoIP nella tabella di routing del gateway locale specificata.

```
aws ec2 create-coip-pool \
    --local-gateway-route-table-id lgw-rtb-abcdefg1234567890
```

Output:

```
{
    "CoipPool": {
        "PoolId": "ipv4pool-coip-1234567890abcdefg",
        "LocalGatewayRouteTableId": "lgw-rtb-abcdefg1234567890",
        "PoolArn": "arn:aws:ec2:us-west-2:123456789012:coip-pool/ipv4pool-
coip-1234567890abcdefg"
    }
}
```

Per ulteriori informazioni, consulta Indirizzi IP di proprietà del cliente nella Guida per l'utente di AWS Outposts.

- Per i dettagli sull'API, consulta CreateCoipPool AWS CLICommand Reference.

`create-customer-gateway`

Il seguente esempio di codice mostra come utilizzare`create-customer-gateway`.

AWS CLI

Come creare un gateway del cliente

Questo esempio crea un gateway del cliente con l'indirizzo IP specificato per la relativa interfaccia esterna.

Comando:

```
aws ec2 create-customer-gateway --type ipsec.1 --public-ip 12.1.2.3 --bgp-asn 65534
```

Output:

```
{
    "CustomerGateway": {
        "CustomerGatewayId": "cgw-0e11f167",
        "IpAddress": "12.1.2.3",
        "State": "available",
        "Type": "ipsec.1",
        "BgpAsn": "65534"
    }
}
```

- Per i dettagli sull'API, consulta CreateCustomerGateway AWS CLICommand Reference.

`create-default-subnet`

Il seguente esempio di codice mostra come utilizzare`create-default-subnet`.

AWS CLI

Come creare una sottorete predefinita

Questo esempio crea una sottorete predefinita nella zona di disponibilità `us-east-2a`.

Comando:

```
 aws ec2 create-default-subnet --availability-zone us-east-2a

{
   "Subnet": {
```

```
        "AvailabilityZone": "us-east-2a",
        "Tags": [],
        "AvailableIpAddressCount": 4091,
        "DefaultForAz": true,
        "Ipv6CidrBlockAssociationSet": [],
        "VpcId": "vpc-1a2b3c4d",
        "State": "available",
        "MapPublicIpOnLaunch": true,
        "SubnetId": "subnet-1122aabb",
        "CidrBlock": "172.31.32.0/20",
        "AssignIpv6AddressOnCreation": false
    }
}
```

- Per i dettagli sull'API, consulta CreateDefaultSubnet AWS CLICommand Reference.

`create-default-vpc`

Il seguente esempio di codice mostra come utilizzare`create-default-vpc`.

AWS CLI

Come creare un VPC predefinito

Questo esempio crea un VPC predefinito.

Comando:

```
aws ec2 create-default-vpc
```

Output:

```
{
    "Vpc": {
        "VpcId": "vpc-8eaae5ea",
        "InstanceTenancy": "default",
        "Tags": [],
        "Ipv6CidrBlockAssociationSet": [],
        "State": "pending",
        "DhcpOptionsId": "dopt-af0c32c6",
        "CidrBlock": "172.31.0.0/16",
        "IsDefault": true
```

```
    }
}
```

- Per i dettagli sull'API, consulta CreateDefaultVpc AWS CLICommand Reference.

`create-dhcp-options`

Il seguente esempio di codice mostra come utilizzare`create-dhcp-options`.

AWS CLI

Come creare un set di opzioni DHCP

L'esempio `create-dhcp-options` seguente crea un set di opzioni DHCP che specifica il nome di dominio, i server dei nomi di dominio e il tipo di nodo NetBIOS.

```
aws ec2 create-dhcp-options \
    --dhcp-configuration \
        "Key=domain-name-servers,Values=10.2.5.1,10.2.5.2" \
        "Key=domain-name,Values=example.com" \
        "Key=netbios-node-type,Values=2"
```

Output:

```
{
    "DhcpOptions": {
        "DhcpConfigurations": [
            {
                "Key": "domain-name",
                "Values": [
                    {
                        "Value": "example.com"
                    }
                ]
            },
            {
                "Key": "domain-name-servers",
                "Values": [
                    {
                        "Value": "10.2.5.1"
                    },
                    {
```

```
                        "Value": "10.2.5.2"
                    }
                ]
            },
            {
                "Key": "netbios-node-type",
                "Values": [
                    {
                        "Value": "2"
                    }
                ]
            }
        ],
        "DhcpOptionsId": "dopt-06d52773eff4c55f3"
    }
}
```

- Per i dettagli sull'API, consulta CreateDhcpOptions AWS CLICommand Reference.

`create-egress-only-internet-gateway`

Il seguente esempio di codice mostra come utilizzare`create-egress-only-internet-gateway`.

AWS CLI

Come creare un gateway Internet di sola uscita

Questo esempio crea un gateway Internet di sola uscita per il VPC specificato.

Comando:

```
aws ec2 create-egress-only-internet-gateway --vpc-id vpc-0c62a468
```

Output:

```
{
  "EgressOnlyInternetGateway": {
      "EgressOnlyInternetGatewayId": "eigw-015e0e244e24dfe8a",
      "Attachments": [
          {
              "State": "attached",
              "VpcId": "vpc-0c62a468"
```

```
            }
        ]
    }
}
```

- Per i dettagli sull'API, consulta CreateEgressOnlyInternetGateway AWS CLICommand Reference.

`create-fleet`

Il seguente esempio di codice mostra come utilizzare`create-fleet`.

AWS CLI

Esempio 1: creare una EC2 flotta che avvii le istanze Spot come modello di acquisto predefinito

L'`create-fleet`esempio seguente crea una EC2 flotta utilizzando i parametri minimi richiesti per lanciare una flotta: un modello di lancio, una capacità target e un modello di acquisto predefinito. Il modello di avvio viene identificato dall'ID e dal numero di versione del modello di avvio. La capacità di destinazione per il parco è di 2 istanze e il modello d'acquisto predefinito è `spot`; ne consegue che il parco avvia 2 istanze spot.

Quando crei una EC2 flotta, utilizza un file JSON per specificare le informazioni sulle istanze da avviare.

```
aws ec2 create-fleet \
    --cli-input-json file://file_name.json
```

Contenuto del file_name.json:

```
{
    "LaunchTemplateConfigs": [
    {
        "LaunchTemplateSpecification": {
        "LaunchTemplateId": "lt-0e8c754449b27161c",
        "Version": "1"
        }
    }
    ],
    "TargetCapacitySpecification": {
```

```
        "TotalTargetCapacity": 2,
        "DefaultTargetCapacityType": "spot"
    }
}
```

Output:

```
{
    "FleetId": "fleet-12a34b55-67cd-8ef9-ba9b-9208dEXAMPLE"
}
```

Esempio 2: creare una EC2 flotta che avvii istanze On-Demand come modello di acquisto predefinito

L'`create-fleet`esempio seguente crea una EC2 flotta utilizzando i parametri minimi richiesti per lanciare una flotta: un modello di lancio, una capacità target e un modello di acquisto predefinito. Il modello di avvio viene identificato dall'ID e dal numero di versione del modello di avvio. La capacità di destinazione per il parco è di 2 istanze e l'opzione d'acquisto predefinita è `on-demand`; ne consegue che il parco avvia 2 istanze on demand.

Quando crei una EC2 flotta, utilizza un file JSON per specificare le informazioni sulle istanze da avviare.

```
aws ec2 create-fleet \
    --cli-input-json file://file_name.json
```

Contenuto del file_name.json:

```
{
    "LaunchTemplateConfigs": [
    {
        "LaunchTemplateSpecification": {
        "LaunchTemplateId": "lt-0e8c754449b27161c",
        "Version": "1"
        }
    }
    ],
    "TargetCapacitySpecification": {
    "TotalTargetCapacity": 2,
    "DefaultTargetCapacityType": "on-demand"
```

```
    }
}
```

Output:

```
{
    "FleetId": "fleet-12a34b55-67cd-8ef9-ba9b-9208dEXAMPLE"
}
```

Esempio 3: creare un EC2 parco istanze che lanci istanze on demand come capacità principale

L'`create-fleet`esempio seguente crea una EC2 flotta che specifica la capacità target totale di 2 istanze per il parco istanze e una capacità target di 1 istanza On-Demand. Il modello di acquisto predefinito è `spot`. Il parco avvia 1 istanza on demand come indicato, ma deve avviare un'altra istanza per soddisfare la capacità di destinazione totale. L'opzione di acquisto per la differenza viene calcolata come `TotalTargetCapacity` – `OnDemandTargetCapacity` = `DefaultTargetCapacityType`, ne consegue che il parco avvia 1 istanza spot.

Quando crei un EC2 parco istanze, utilizza un file JSON per specificare le informazioni sulle istanze da avviare.

```
aws ec2 create-fleet \
    --cli-input-json file://file_name.json
```

Contenuto del file_name.json:

```
{
    "LaunchTemplateConfigs": [
    {
        "LaunchTemplateSpecification": {
        "LaunchTemplateId": "lt-0e8c754449b27161c",
        "Version": "1"
        }
    }
    ],
    "TargetCapacitySpecification": {
        "TotalTargetCapacity": 2,
        "OnDemandTargetCapacity":1,
        "DefaultTargetCapacityType": "spot"
    }
```

```
}
```

Output:

```
{
    "FleetId": "fleet-12a34b55-67cd-8ef9-ba9b-9208dEXAMPLE"
}
```

Esempio 4: creare una EC2 flotta che avvii istanze Spot utilizzando la strategia di allocazione al prezzo più basso

Se non è indicata la strategia di allocazione per le istanze spot, viene utilizzata la strategia di allocazione predefinita, ovvero `lowest-price`. L'`create-fleet`esempio seguente crea una flotta utilizzando la strategia di allocazione. EC2 `lowest-price` Le tre specifiche di avvio, che sostituiscono il modello di avvio, hanno tipi di istanza diversi ma capacità ponderata e sottorete uguali. La capacità di destinazione totale è di 2 istanze e il modello d'acquisto predefinito è `spot`. The EC2 Fleet lancia 2 istanze Spot utilizzando il tipo di istanza della specifica di lancio al prezzo più basso.

Quando crei un EC2 parco istanze, utilizza un file JSON per specificare le informazioni sulle istanze da avviare.

```
aws ec2 create-fleet \
    --cli-input-json file://file_name.jsonContents of file_name.json::

{
    "LaunchTemplateConfigs": [
    {
        "LaunchTemplateSpecification": {
        "LaunchTemplateId": "lt-0e8c754449b27161c",
        "Version": "1"
        },
        "Overrides": [
            {
                "InstanceType": "c4.large",
                "WeightedCapacity": 1,
                "SubnetId": "subnet-a4f6c5d3"
            },
            {
                "InstanceType": "c3.large",
                "WeightedCapacity": 1,
```

```
                "SubnetId": "subnet-a4f6c5d3"
            },
            {
                "InstanceType": "c5.large",
                "WeightedCapacity": 1,
                "SubnetId": "subnet-a4f6c5d3"
            }
        ]
    }
    ],
    "TargetCapacitySpecification": {
        "TotalTargetCapacity": 2,
        "DefaultTargetCapacityType": "spot"

  }
}
```

Output:

```
{
    "FleetId": "fleet-12a34b55-67cd-8ef9-ba9b-9208dEXAMPLE"
}
```

- Per i dettagli sull'API, consulta AWS CLI Command CreateFleetReference.

`create-flow-logs`

Il seguente esempio di codice mostra come utilizzare`create-flow-logs`.

AWS CLI

Esempio 1: come creare un log di flusso

L'esempio `create-flow-logs` seguente crea un log di flusso che acquisisce tutto il traffico rifiutato per l'interfaccia di rete specificata. I log di flusso vengono consegnati a un gruppo di log in CloudWatch Logs utilizzando le autorizzazioni nel ruolo IAM specificato.

```
aws ec2 create-flow-logs \
    --resource-type NetworkInterface \
    --resource-ids eni-11223344556677889 \
    --traffic-type REJECT \
```

```
    --log-group-name my-flow-logs \
    --deliver-logs-permission-arn arn:aws:iam::123456789101:role/publishFlowLogs
```

Output:

```
{
    "ClientToken": "so0eNA2uSHUNlHI0S2cJ305GuIX1CezaRdGtexample",
    "FlowLogIds": [
        "fl-12345678901234567"
    ],
    "Unsuccessful": []
}
```

Per ulteriori informazioni, consulta Log di flusso VPC nella Guida per l'utente di Amazon VPC.

Esempio 2: come creare un log di flusso con un formato personalizzato

L'esempio `create-flow-logs` seguente crea un log di flusso che acquisisce tutto il traffico per il VPC specificato e consegna i log di flusso a un bucket Amazon S3. Il parametro `--log-format` specifica un formato personalizzato per i record di log di flusso. Per eseguire questo comando in Windows, modifica le virgolette singole (‘’) in virgolette doppie (“”).

```
aws ec2 create-flow-logs \
    --resource-type VPC \
    --resource-ids vpc-00112233344556677 \
    --traffic-type ALL \
    --log-destination-type s3 \
    --log-destination arn:aws:s3:::flow-log-bucket/my-custom-flow-logs/ \
    --log-format '${version} ${vpc-id} ${subnet-id} ${instance-id} ${srcaddr}
 ${dstaddr} ${srcport} ${dstport} ${protocol} ${tcp-flags} ${type} ${pkt-srcaddr}
 ${pkt-dstaddr}'
```

Per ulteriori informazioni, consulta Log di flusso VPC nella Guida per l'utente di Amazon VPC.

Esempio 3: come creare un log di flusso con un intervallo di aggregazione massimo di un minuto

L'esempio `create-flow-logs` seguente crea un log di flusso che acquisisce tutto il traffico per il VPC specificato e consegna i log di flusso a un bucket Amazon S3. Il parametro `--max-aggregation-interval` specifica un intervallo di aggregazione massimo di 60 secondi (1 minuto).

```
aws ec2 create-flow-logs \
    --resource-type VPC \
    --resource-ids vpc-00112233344556677 \
    --traffic-type ALL \
    --log-destination-type s3 \
    --log-destination arn:aws:s3:::flow-log-bucket/my-custom-flow-logs/ \
    --max-aggregation-interval 60
```

Per ulteriori informazioni, consulta Log di flusso VPC nella Guida per l'utente di Amazon VPC.

- Per i dettagli sull'API, consulta Command CreateFlowLogsReference AWS CLI .

`create-fpga-image`

Il seguente esempio di codice mostra come utilizzare`create-fpga-image`.

AWS CLI

Come creare un'immagine FPGA di Amazon (AFI)

Questo esempio crea un'AFI dal tarball specificato nel bucket specificato.

Comando:

```
aws ec2 create-fpga-image --name my-afi --description test-afi --input-storage-
location Bucket=my-fpga-bucket,Key=dcp/17_12_22-103226.Developer_CL.tar --logs-
storage-location Bucket=my-fpga-bucket,Key=logs
```

Output:

```
{
  "FpgaImageId": "afi-0d123e123bfc85abc",
  "FpgaImageGlobalId": "agfi-123cb27b5e84a0abc"
}
```

- Per i dettagli sull'API, consulta CreateFpgaImage AWS CLICommand Reference.

`create-image`

Il seguente esempio di codice mostra come utilizzare`create-image`.

AWS CLI

Esempio 1: come creare un'AMI da un'istanza supportata da Amazon EBS

L'esempio `create-image` seguente crea un'AMI dall'istanza specificata.

```
aws ec2 create-image \
    --instance-id i-1234567890abcdef0 \
    --name "My server" \
    --description "An AMI for my server"
```

Output:

```
{
    "ImageId": "ami-abcdef01234567890"
}
```

Per ulteriori informazioni su come specificare una mappatura dei dispositivi a blocchi per la tua AMI, consulta Specificare una mappatura dei dispositivi a blocchi per un'AMI nella Amazon User Guide. EC2

Esempio 2: come creare un'AMI da un'istanza supportata da Amazon EBS senza riavvio

L'esempio `create-image` seguente crea un'AMI e imposta il parametro --no-reboot in modo che l'istanza non venga riavviata prima della creazione dell'immagine.

```
aws ec2 create-image \
    --instance-id i-1234567890abcdef0 \
    --name "My server" \
    --no-reboot
```

Output:

```
{
    "ImageId": "ami-abcdef01234567890"
}
```

Per ulteriori informazioni su come specificare una mappatura dei dispositivi a blocchi per la tua AMI, consulta Specificare una mappatura dei dispositivi a blocchi per un'AMI nella Amazon User Guide. EC2

Esempio 3: come assegnare tag a un'AMI e agli snapshot durante la creazione

L'esempio `create-image` seguente crea un'AMI e assegna all'AMI e agli snapshot lo stesso tag `cost-center=cc123`.

```
aws ec2 create-image \
    --instance-id i-1234567890abcdef0 \
    --name "My server" \
    --tag-specifications "ResourceType=image,Tags=[{Key=cost-
center,Value=cc123}]" "ResourceType=snapshot,Tags=[{Key=cost-center,Value=cc123}]"
```

Output:

```
{
    "ImageId": "ami-abcdef01234567890"
}
```

Per ulteriori informazioni sull'etichettatura delle risorse al momento della creazione, consulta Aggiungere tag alla creazione di risorse nella Amazon EC2 User Guide.

- Per i dettagli sull'API, consulta CreateImage AWS CLICommand Reference.

`create-instance-connect-endpoint`

Il seguente esempio di codice mostra come utilizzare`create-instance-connect-endpoint`.

AWS CLI

Per creare un endpoint EC2 Instance Connect

L'`create-instance-connect-endpoint`esempio seguente crea un endpoint EC2 Instance Connect nella sottorete specificata.

```
aws ec2 create-instance-connect-endpoint \
    --region us-east-1 \
    --subnet-id subnet-0123456789example
```

Output:

```
{
```

```
    "VpcId": "vpc-0123abcd",
    "InstanceConnectEndpointArn": "arn:aws:ec2:us-east-1:111111111111:instance-
connect-endpoint/eice-0123456789example",
    "AvailabilityZone": "us-east-1a",
    "NetworkInterfaceIds": [
        "eni-0123abcd"
    ],
    "PreserveClientIp": true,
    "Tags": [],
    "FipsDnsName": "eice-0123456789example.0123abcd.fips.ec2-instance-connect-
endpoint.us-east-1.amazonaws.com",
    "StateMessage": "",
    "State": "create-complete",
    "DnsName": "eice-0123456789example.0123abcd.ec2-instance-connect-endpoint.us-
east-1.amazonaws.com",
    "SubnetId": "subnet-0123abcd",
    "OwnerId": "111111111111",
    "SecurityGroupIds": [
        "sg-0123abcd"
    ],
    "InstanceConnectEndpointId": "eice-0123456789example",
    "CreatedAt": "2023-04-07T15:43:53.000Z"
}
```

Per ulteriori informazioni, consulta Create an EC2 Instance Connect Endpoint nella Amazon EC2 User Guide.

- Per i dettagli sull'API, consulta CreateInstanceConnectEndpoint AWS CLICommand Reference.

`create-instance-event-window`

Il seguente esempio di codice mostra come utilizzare`create-instance-event-window`.

AWS CLI

Esempio 1: come creare una finestra di eventi con un intervallo temporale

L'esempio `create-instance-event-window` seguente crea una finestra di eventi con un intervallo temporale. Non è possibile specificare anche il parametro `cron-expression`.

```
aws ec2 create-instance-event-window \
    --region us-east-1 \
```

```
    --time-range StartWeekDay=monday,StartHour=2,EndWeekDay=wednesday,EndHour=8 \
    --tag-specifications "ResourceType=instance-event-
window,Tags=[{Key=K1,Value=V1}]" \
    --name myEventWindowName
```

Output:

```
{
    "InstanceEventWindow": {
        "InstanceEventWindowId": "iew-0abcdef1234567890",
        "TimeRanges": [
            {
                "StartWeekDay": "monday",
                "StartHour": 2,
                "EndWeekDay": "wednesday",
                "EndHour": 8
            }
        ],
        "Name": "myEventWindowName",
        "State": "creating",
        "Tags": [
            {
                "Key": "K1",
                "Value": "V1"
            }
        ]
    }
}
```

Per i vincoli relativi alla finestra degli eventi, consulta Considerazioni nella sezione Eventi pianificati della Amazon EC2 User Guide.

Esempio 2: come creare una finestra di eventi con un'espressione Cron

L'esempio `create-instance-event-window` seguente crea una finestra di eventi con un'espressione Cron. Non è possibile specificare anche il parametro `time-range`.

```
aws ec2 create-instance-event-window \
    --region us-east-1 \
    --cron-expression "* 21-23 * * 2,3" \
    --tag-specifications "ResourceType=instance-event-
window,Tags=[{Key=K1,Value=V1}]" \
```

```
    --name myEventWindowName
```

Output:

```
{
    "InstanceEventWindow": {
        "InstanceEventWindowId": "iew-0abcdef1234567890",
        "Name": "myEventWindowName",
        "CronExpression": "* 21-23 * * 2,3",
        "State": "creating",
        "Tags": [
            {
                "Key": "K1",
                "Value": "V1"
            }
        ]
    }
}
```

Per i vincoli relativi alla finestra degli eventi, consulta Considerazioni nella sezione Eventi pianificati della Amazon EC2 User Guide.

- Per i dettagli sull'API, consulta Command CreateInstanceEventWindowReference AWS CLI .

`create-instance-export-task`

Il seguente esempio di codice mostra come utilizzare`create-instance-export-task`.

AWS CLI

Come esportare un'istanza

Questo comando di esempio crea un'attività per esportare l'istanza i-1234567890abcdef0 nel bucket Amazon S3 myexportbucket.

Comando:

```
aws ec2 create-instance-export-task --description "RHEL5 instance" --
instance-id i-1234567890abcdef0 --target-environment vmware --export-to-s3-
task DiskImageFormat=vmdk,ContainerFormat=ova,S3Bucket=myexportbucket,S3Prefix=RHEL5
```

Output:

```
{
    "ExportTask": {
        "State": "active",
        "InstanceExportDetails": {
            "InstanceId": "i-1234567890abcdef0",
            "TargetEnvironment": "vmware"
        },
        "ExportToS3Task": {
            "S3Bucket": "myexportbucket",
            "S3Key": "RHEL5export-i-fh8sjjsq.ova",
            "DiskImageFormat": "vmdk",
            "ContainerFormat": "ova"
        },
        "Description": "RHEL5 instance",
        "ExportTaskId": "export-i-fh8sjjsq"
    }
}
```

- Per i dettagli sull'API, consulta CreateInstanceExportTask AWS CLICommand Reference.

`create-internet-gateway`

Il seguente esempio di codice mostra come utilizzare`create-internet-gateway`.

AWS CLI

Come creare un gateway Internet

L'esempio `create-internet-gateway` seguente crea un gateway Internet con il tag `Name=my-igw`.

```
aws ec2 create-internet-gateway \
    --tag-specifications ResourceType=internet-gateway,Tags=[{Key=Name,Value=my-igw}]
```

Output:

```
{
    "InternetGateway": {
        "Attachments": [],
        "InternetGatewayId": "igw-0d0fb496b3994d755",
```

```
        "OwnerId": "123456789012",
        "Tags": [
            {
                "Key": "Name",
                "Value": "my-igw"
            }
        ]
    }
}
```

Per ulteriori informazioni, consulta Gateway Internet nella Guida per l'utente di Amazon VPC.

- Per i dettagli sull'API, consulta CreateInternetGateway AWS CLICommand Reference.

`create-ipam-pool`

Il seguente esempio di codice mostra come utilizzare`create-ipam-pool`.

AWS CLI

Come creare un pool IPAM

L'esempio `create-ipam-pool` seguente crea un pool IPAM.

Linux:

```
aws ec2 create-ipam-pool \
    --ipam-scope-id ipam-scope-02fc38cd4c48e7d38 \
    --address-family ipv4 \
    --auto-import \
    --allocation-min-netmask-length 16 \
    --allocation-max-netmask-length 26 \
    --allocation-default-netmask-length 24 \
    --allocation-resource-tags "Key=Environment,Value=Preprod" \
    --tag-specifications 'ResourceType=ipam-pool,Tags=[{Key=Name,Value="Preprod
 pool"}]'
```

Windows:

```
aws ec2 create-ipam-pool ^
    --ipam-scope-id ipam-scope-02fc38cd4c48e7d38 ^
    --address-family ipv4 ^
```

```
    --auto-import ^
    --allocation-min-netmask-length 16 ^
    --allocation-max-netmask-length 26 ^
    --allocation-default-netmask-length 24 ^
    --allocation-resource-tags "Key=Environment,Value=Preprod" ^
    --tag-specifications ResourceType=ipam-pool,Tags=[{Key=Name,Value="Preprod
 pool"}]
```

Output:

```
{
    "IpamPool": {
        "OwnerId": "123456789012",
        "IpamPoolId": "ipam-pool-0533048da7d823723",
        "IpamPoolArn": "arn:aws:ec2::123456789012:ipam-pool/ipam-
pool-0533048da7d823723",
        "IpamScopeArn": "arn:aws:ec2::123456789012:ipam-scope/ipam-
scope-02fc38cd4c48e7d38",
        "IpamScopeType": "private",
        "IpamArn": "arn:aws:ec2::123456789012:ipam/ipam-08440e7a3acde3908",
        "IpamRegion": "us-east-1",
        "Locale": "None",
        "PoolDepth": 1,
        "State": "create-in-progress",
        "AutoImport": true,
        "AddressFamily": "ipv4",
        "AllocationMinNetmaskLength": 16,
        "AllocationMaxNetmaskLength": 26,
        "AllocationDefaultNetmaskLength": 24,
        "AllocationResourceTags": [
            {
                "Key": "Environment",
                "Value": "Preprod"
            }
        ],
        "Tags": [
            {
                "Key": "Name",
                "Value": "Preprod pool"
            }
        ]
    }
}
```

Per ulteriori informazioni, consulta Pianificare il provisioning degli indirizzi IP nella Guida per l'utente IPAM di Amazon VPC.

- Per i dettagli sull'API, consulta CreateIpamPool AWS CLICommand Reference.

`create-ipam-resource-discovery`

Il seguente esempio di codice mostra come utilizzare`create-ipam-resource-discovery`.

AWS CLI

Come creare un'individuazione delle risorse

In questo esempio, sei un amministratore IPAM delegato che desidera creare e condividere un'individuazione delle risorse con l'amministratore IPAM di un'altra AWS organizzazione in modo che l'amministratore dell'altra organizzazione possa gestire e monitorare gli indirizzi IP delle risorse dell'organizzazione.

Importante

Questo esempio include entrambe le opzioni `--region` e `--operating-regions` perché, sebbene siano facoltative, devono essere configurate in un modo particolare per integrare correttamente un rilevamento di risorse con un'IPAM. * `--operating-regions` deve corrispondere alle Regioni in cui sono presenti risorse che devono essere rilevate dall'IPAM. Se ci sono Regioni in cui l'IPAM non deve gestire gli indirizzi IP (ad esempio per motivi di conformità), non includerle. * `--region` deve corrispondere alla Regione di origine dell'IPAM a cui si desidera associarlo. È necessario creare il rilevamento di risorse nella stessa Regione in cui è stato creato l'IPAM. Ad esempio, se l'IPAM a cui ci si sta associando è stato creato in us-east-1, includere `--region us-east-1` nella richiesta. Entrambe le opzioni `--region` e `--operating-regions` sono predefinite nella Regione in cui si sta eseguendo il comando se non vengono specificate.

In questo esempio, le Regioni operative IPAM con cui si sta effettuando l'integrazione includono `us-west-1`, `us-west-2` e `ap-south-1`. Quando si crea il rilevamento di risorse, l'IPAM deve rilevare gli indirizzi IP delle risorse in `us-west-1` e `us-west-2` ma non in `ap-south-1`. Pertanto, nella richiesta viene incluso solo `--operating-regions RegionName='us-west-1' RegionName='us-west-2'`.

L'esempio `create-ipam-resource-discovery` seguente crea un rilevamento di risorse IPAM.

```
aws ec2 create-ipam-resource-discovery \
    --description 'Example-resource-discovery' \
    --tag-specifications 'ResourceType=ipam-resource-discovery,Tags=[{Key=cost-
center,Value=cc123}]' \
    --operating-regions RegionName='us-west-1' RegionName='us-west-2' \
    --region us-east-1
```

Output:

```
{
    "IpamResourceDiscovery":{
        "OwnerId": "149977607591",
        "IpamResourceDiscoveryId": "ipam-res-disco-0257046d8aa78b8bc",
        "IpamResourceDiscoveryArn": "arn:aws:ec2::149977607591:ipam-resource-
discovery/ipam-res-disco-0257046d8aa78b8bc",
        "IpamResourceDiscoveryRegion": "us-east-1",
        "Description": "'Example-resource-discovery'",
        "OperatingRegions":[
            {"RegionName": "us-west-1"},
            {"RegionName": "us-west-2"},
            {"RegionName": "us-east-1"}
        ],
        "IsDefault": false,
        "State": "create-in-progress",
        "Tags": [
            {
                "Key": "cost-center",
                "Value": "cc123"
            }
        ]
}
```

Dopo aver creato un rilevamento delle risorse, potresti volerlo condividere con un altro amministratore delegato IPAM, cosa che puoi fare. create-resource-share Per ulteriori informazioni, consulta Come integrare IPAM con account esterni alla tua organizzazione nella Guida per l'utente IPAM di Amazon VPC.

- Per i dettagli sull'API, consulta AWS CLI Command CreateIpamResourceDiscoveryReference.

`create-ipam-scope`

Il seguente esempio di codice mostra come utilizzare`create-ipam-scope`.

AWS CLI

Come creare un ambito IPAM

L'esempio `create-ipam-scope` seguente crea un ambito IPAM.

Linux:

```
aws ec2 create-ipam-scope \
    --ipam-id ipam-08440e7a3acde3908 \
    --description "Example description" \
    --tag-specifications 'ResourceType=ipam-scope,Tags=[{Key=Name,Value="Example
 name value"}]'
```

Windows:

```
aws ec2 create-ipam-scope ^
    --ipam-id ipam-08440e7a3acde3908 ^
    --description "Example description" ^
    --tag-specifications ResourceType=ipam-scope,Tags=[{Key=Name,Value="Example name
 value"}]
```

Output:

```
{
    "IpamScope": {
        "OwnerId": "123456789012",
        "IpamScopeId": "ipam-scope-01c1ebab2b63bd7e4",
        "IpamScopeArn": "arn:aws:ec2::123456789012:ipam-scope/ipam-
scope-01c1ebab2b63bd7e4",
        "IpamArn": "arn:aws:ec2::123456789012:ipam/ipam-08440e7a3acde3908",
        "IpamRegion": "us-east-1",
        "IpamScopeType": "private",
        "IsDefault": false,
        "Description": "Example description",
        "PoolCount": 0,
        "State": "create-in-progress",
        "Tags": [
            {
                "Key": "Name",
                "Value": "Example name value"
            }
```

```
        ]
    }
}
```

Per ulteriori informazioni, consulta Creazione di ambiti aggiuntivi nella Guida per l'utente IPAM di Amazon VPC.

- Per i dettagli sull'API, consulta CreateIpamScope AWS CLICommand Reference.

`create-ipam`

Il seguente esempio di codice mostra come utilizzare`create-ipam`.

AWS CLI

Come creare un'IPAM

L'esempio `create-ipam` seguente crea un'IPAM.

Linux:

```
aws ec2 create-ipam \
    --description "Example description" \
    --operating-regions "RegionName=us-east-2" "RegionName=us-west-1" \
    --tag-specifications 'ResourceType=ipam,Tags=[{Key=Name,Value=ExampleIPAM}]'
```

Windows:

```
aws ec2 create-ipam ^
    --description "Example description" ^
    --operating-regions "RegionName=us-east-2" "RegionName=us-west-1" ^
    --tag-specifications ResourceType=ipam,Tags=[{Key=Name,Value=ExampleIPAM}]
```

Output:

```
{
    "Ipam": {
        "OwnerId": "123456789012",
        "IpamId": "ipam-036486dfa6af58ee0",
        "IpamArn": "arn:aws:ec2::123456789012:ipam/ipam-036486dfa6af58ee0",
```

```
        "IpamRegion": "us-east-1",
        "PublicDefaultScopeId": "ipam-scope-071b8042b0195c183",
        "PrivateDefaultScopeId": "ipam-scope-0807405dece705a30",
        "ScopeCount": 2,
        "OperatingRegions": [
            {
                "RegionName": "us-east-2"
            },
            {
                "RegionName": "us-west-1"
            },
            {
                "RegionName": "us-east-1"
            }
        ],
        "State": "create-in-progress",
        "Tags": [
            {
                "Key": "Name",
                "Value": "ExampleIPAM"
            }
        ]
    }
}
```

Per ulteriori informazioni, consulta Creare un IPAM nella Guida per l'utente IPAM di Amazon VPC.

- Per i dettagli sull'API, consulta CreateIpam AWS CLICommand Reference.

`create-key-pair`

Il seguente esempio di codice mostra come utilizzare`create-key-pair`.

AWS CLI

Come creare una coppia di chiavi

Nell'esempio seguente viene creata una coppia di chiavi denominata `MyKeyPair`.

Comando:

```
aws ec2 create-key-pair --key-name MyKeyPair
```

L'output è una versione ASCII della chiave privata e l'impronta della chiave. È necessario salvare la chiave in un file.

Per ulteriori informazioni, consulta Utilizzo delle coppie di chiavi nella Guida per l'utente dell'Interfaccia a riga di comando AWS .

- Per i dettagli sull'API, consulta CreateKeyPair AWS CLICommand Reference.

`create-launch-template-version`

Il seguente esempio di codice mostra come utilizzare`create-launch-template-version`.

AWS CLI

Come creare una versione del modello di avvio

Questo esempio crea una nuova versione del modello di avvio basata sulla versione 1 del modello di avvio e specifica un ID AMI diverso.

Comando:

```
aws ec2 create-launch-template-version --launch-template-id lt-0abcd290751193123
 --version-description WebVersion2 --source-version 1 --launch-template-data
 '{"ImageId":"ami-c998b6b2"}'
```

Output:

```
{
  "LaunchTemplateVersion": {
      "VersionDescription": "WebVersion2",
      "LaunchTemplateId": "lt-0abcd290751193123",
      "LaunchTemplateName": "WebServers",
      "VersionNumber": 2,
      "CreatedBy": "arn:aws:iam::123456789012:root",
      "LaunchTemplateData": {
          "ImageId": "ami-c998b6b2",
          "InstanceType": "t2.micro",
          "NetworkInterfaces": [
              {
                  "Ipv6Addresses": [
                      {
                          "Ipv6Address": "2001:db8:1234:1a00::123"
                      }
```

```
                ],
                "DeviceIndex": 0,
                "SubnetId": "subnet-7b16de0c",
                "AssociatePublicIpAddress": true
            }
        ]
    },
    "DefaultVersion": false,
    "CreateTime": "2017-12-01T13:35:46.000Z"
  }
}
```

- Per i dettagli sull'API, consulta CreateLaunchTemplateVersion AWS CLICommand Reference.

`create-launch-template`

Il seguente esempio di codice mostra come utilizzare`create-launch-template`.

AWS CLI

Esempio 1: come creare un modello di avvio

L'`create-launch-template`esempio seguente crea un modello di avvio che specifica la sottorete in cui avviare l'istanza, assegna un indirizzo IP pubblico e un IPv6 indirizzo all'istanza e crea un tag per l'istanza.

```
aws ec2 create-launch-template \
    --launch-template-name TemplateForWebServer \
    --version-description WebVersion1 \
    --launch-template-data '{"NetworkInterfaces":
[{"AssociatePublicIpAddress":true,"DeviceIndex":0,"Ipv6AddressCount":1,"SubnetId":"subnet-7b
[{"ResourceType":"instance","Tags":[{"Key":"purpose","Value":"webserver"}]}]}'
```

Output:

```
{
    "LaunchTemplate": {
        "LatestVersionNumber": 1,
        "LaunchTemplateId": "lt-01238c059e3466abc",
        "LaunchTemplateName": "TemplateForWebServer",
        "DefaultVersionNumber": 1,
        "CreatedBy": "arn:aws:iam::123456789012:user/Bob",
```

```
        "CreateTime": "2019-01-27T09:13:24.000Z"
    }
}
```

Per ulteriori informazioni, consulta Avvio di istanze da un modello di avvio nella Guida per l'utente di Amazon Elastic Compute Cloud. Per informazioni sulla citazione di parametri in formato JSON, consulta Virgolette con stringhe nella Guida per l'utente dell'Interfaccia a riga di comando AWS .

Esempio 2: creare un modello di lancio per Amazon EC2 Auto Scaling

Nell'esempio di `create-launch-template` seguente viene creato un modello di avvio con tag multipli e mappatura dei dispositivi a blocchi per specificare un volume EBS aggiuntivo quando viene avviata un'istanza. Specificare un valore per `Groups` che corrisponde ai gruppi di sicurezza per il VPC nel quale il gruppo Auto Scaling avvierà le istanze. Specificare il VPC e le sottoreti come proprietà del gruppo Auto Scaling.

```
aws ec2 create-launch-template \
    --launch-template-name TemplateForAutoScaling \
    --version-description AutoScalingVersion1 \
    --launch-template-data '{"NetworkInterfaces":
[{"DeviceIndex":0,"AssociatePublicIpAddress":true,"Groups":
["sg-7c227019,sg-903004f8"],"DeleteOnTermination":true}],"ImageId":"ami-
b42209de","InstanceType":"m4.large","TagSpecifications":
[{"ResourceType":"instance","Tags":[{"Key":"environment","Value":"production"},
{"Key":"purpose","Value":"webserver"}]},{"ResourceType":"volume","Tags":
[{"Key":"environment","Value":"production"},{"Key":"cost-
center","Value":"cc123"}]}],"BlockDeviceMappings":[{"DeviceName":"/dev/sda1","Ebs":
{"VolumeSize":100}}]}' --region us-east-1
```

Output:

```
{
    "LaunchTemplate": {
        "LatestVersionNumber": 1,
        "LaunchTemplateId": "lt-0123c79c33a54e0abc",
        "LaunchTemplateName": "TemplateForAutoScaling",
        "DefaultVersionNumber": 1,
        "CreatedBy": "arn:aws:iam::123456789012:user/Bob",
        "CreateTime": "2019-04-30T18:16:06.000Z"
    }
}
```

Per ulteriori informazioni, consulta Creating a Launch Template for an Auto Scaling Group nella Amazon Auto EC2 Scaling User Guide. Per informazioni sulla citazione di parametri in formato JSON, consulta Inserimento di stringhe tra virgolette nella Guida per l'utente dell'Interfaccia a riga di comando AWS .

Esempio 3: come creare un modello di avvio che specifica la crittografia dei volumi EBS

Nell'esempio di `create-launch-template` seguente viene creato un modello di avvio che include volumi EBS crittografati creati da uno snapshot non crittografato. Inoltre, vengono applicati i tag ai volumi durante la creazione. Se la crittografia predefinita è disabilitata, è necessario specificare l'opzione `"Encrypted"` come mostrato nel seguente esempio. Se si utilizza l'opzione `"KmsKeyId"` per specificare una CMK gestita dal cliente, è necessario specificare l'opzione `"Encrypted"` anche se la crittografia predefinita è abilitata.

```
aws ec2 create-launch-template \
  --launch-template-name TemplateForEncryption \
  --launch-template-data file://config.json
```

Contenuto di `config.json`:

```
{
    "BlockDeviceMappings":[
        {
            "DeviceName":"/dev/sda1",
            "Ebs":{
                "VolumeType":"gp2",
                "DeleteOnTermination":true,
                "SnapshotId":"snap-066877671789bd71b",
                "Encrypted":true,
                "KmsKeyId":"arn:aws:kms:us-east-1:012345678910:key/abcd1234-
a123-456a-a12b-a123b4cd56ef"
            }
        }
    ],
    "ImageId":"ami-00068cd7555f543d5",
    "InstanceType":"c5.large",
    "TagSpecifications":[
        {
            "ResourceType":"volume",
            "Tags":[
                {
                    "Key":"encrypted",
```

```
                    "Value":"yes"
                }
            ]
        }
    ]
}
```

Output:

```
{
    "LaunchTemplate": {
        "LatestVersionNumber": 1,
        "LaunchTemplateId": "lt-0d5bd51bcf8530abc",
        "LaunchTemplateName": "TemplateForEncryption",
        "DefaultVersionNumber": 1,
        "CreatedBy": "arn:aws:iam::123456789012:user/Bob",
        "CreateTime": "2020-01-07T19:08:36.000Z"
    }
}
```

Per ulteriori informazioni, consulta Ripristino di un volume EBS da uno snapshot e Crittografia per impostazione predefinita nella Guida per l'utente di Amazon Elastic Compute Cloud.

- Per i dettagli sull'API, consulta AWS CLI Command CreateLaunchTemplateReference.

`create-local-gateway-route-table-virtual-interface-group-association`

Il seguente esempio di codice mostra come utilizzare`create-local-gateway-route-table-virtual-interface-group-association`.

AWS CLI

Per associare una tabella di routing del gateway locale a un gruppo di interfacce virtuali (VIFs)

L'esempio `create-local-gateway-route-table-virtual-interface-group-association` seguente crea un'associazione tra la tabella di routing del gateway locale specificata e il gruppo VIF.

```
aws ec2 create-local-gateway-route-table-virtual-interface-group-association \
    --local-gateway-route-table-id lgw-rtb-exampleidabcd1234 \
    --local-gateway-virtual-interface-group-id lgw-vif-grp-exampleid0123abcd
```

Output:

```
{
    "LocalGatewayRouteTableVirtualInterfaceGroupAssociation": {
        "LocalGatewayRouteTableVirtualInterfaceGroupAssociationId": "lgw-vif-grp-
assoc-exampleid12345678",
        "LocalGatewayVirtualInterfaceGroupId": "lgw-vif-grp-exampleid0123abcd",
        "LocalGatewayId": "lgw-exampleid11223344",
        "LocalGatewayRouteTableId": "lgw-rtb-exampleidabcd1234",
        "LocalGatewayRouteTableArn": "arn:aws:ec2:us-west-2:111122223333:local-
gateway-route-table/lgw-rtb-exampleidabcd1234",
        "OwnerId": "111122223333",
        "State": "pending",
        "Tags": []
    }
}
```

Per ulteriori informazioni, consulta Associazioni di gruppi VIF nella Guida per l'utente di AWS Outposts.

- Per i dettagli sull'API, vedere CreateLocalGatewayRouteTableVirtualInterfaceGroupAssociationin AWS CLI Command Reference.

`create-local-gateway-route-table-vpc-association`

Il seguente esempio di codice mostra come utilizzare`create-local-gateway-route-table-vpc-association`.

AWS CLI

Come associare un VPC a una tabella di routing

L'esempio `create-local-gateway-route-table-vpc-association` seguente associa il VPC specificato alla tabella di routing del gateway locale specificata.

```
aws ec2 create-local-gateway-route-table-vpc-association \
    --local-gateway-route-table-id lgw-rtb-059615ef7dEXAMPLE \
    --vpc-id vpc-07ef66ac71EXAMPLE
```

Output:

```
{
    "LocalGatewayRouteTableVpcAssociation": {
        "LocalGatewayRouteTableVpcAssociationId": "lgw-vpc-assoc-0ee765bcc8EXAMPLE",
        "LocalGatewayRouteTableId": "lgw-rtb-059615ef7dEXAMPLE",
        "LocalGatewayId": "lgw-09b493aa7cEXAMPLE",
        "VpcId": "vpc-07ef66ac71EXAMPLE",
        "State": "associated"
    }
}
```

- Per i dettagli sull'API, consulta CreateLocalGatewayRouteTableVpcAssociation AWS CLICommand Reference.

`create-local-gateway-route-table`

Il seguente esempio di codice mostra come utilizzare`create-local-gateway-route-table`.

AWS CLI

Come creare la tabella di routing del gateway locale

L'esempio `create-local-gateway-route-table` seguente crea una tabella di routing del gateway locale con la modalità di routing VPC diretta.

```
aws ec2 create-local-gateway-route-table \
    --local-gateway-id lgw-1a2b3c4d5e6f7g8h9 \
    --mode direct-vpc-routing
```

Output:

```
{
    "LocalGatewayRouteTable": {
        "LocalGatewayRouteTableId": "lgw-rtb-abcdefg1234567890",
        "LocalGatewayRouteTableArn": "arn:aws:ec2:us-west-2:111122223333:local-
gateway-route-table/lgw-rtb-abcdefg1234567890",
        "LocalGatewayId": "lgw-1a2b3c4d5e6f7g8h9",
        "OutpostArn": "arn:aws:outposts:us-west-2:111122223333:outpost/
op-021345abcdef67890",
        "OwnerId": "111122223333",
        "State": "pending",
        "Tags": [],
```

```
        "Mode": "direct-vpc-routing"
    }
}
```

Per ulteriori informazioni, consulta Tabelle di routing del gateway locale nella Guida per l'utente di AWS Outposts.

- Per i dettagli sull'API, consulta CreateLocalGatewayRouteTable AWS CLICommand Reference.

`create-local-gateway-route`

Il seguente esempio di codice mostra come utilizzare`create-local-gateway-route`.

AWS CLI

Come creare una route statica per una tabella di routing del gateway locale

L'esempio `create-local-gateway-route` seguente crea la route specificata nella tabella di routing del gateway locale specificata.

```
aws ec2 create-local-gateway-route \
    --destination-cidr-block 0.0.0.0/0 \
    --local-gateway-route-table-id lgw-rtb-059615ef7dEXAMPLE
```

Output:

```
{
    "Route": {
        "DestinationCidrBlock": "0.0.0.0/0",
        "LocalGatewayVirtualInterfaceGroupId": "lgw-vif-grp-07145b276bEXAMPLE",
        "Type": "static",
        "State": "deleted",
        "LocalGatewayRouteTableId": "lgw-rtb-059615ef7dEXAMPLE"
    }
}
```

- Per i dettagli sull'API, consulta CreateLocalGatewayRoute AWS CLICommand Reference.

`create-managed-prefix-list`

Il seguente esempio di codice mostra come utilizzare`create-managed-prefix-list`.

AWS CLI

Come creare un elenco di prefissi

L'`create-managed-prefix-list`esempio seguente crea un elenco di IPv4 prefissi con un massimo di 10 voci e crea 2 voci nell'elenco dei prefissi.

```
aws ec2 create-managed-prefix-list \
    --address-family IPv4 \
    --max-entries 10 \
    --entries Cidr=10.0.0.0/16,Description=vpc-a Cidr=10.2.0.0/16,Description=vpc-b
 \
    --prefix-list-name vpc-cidrs
```

Output:

```
{
    "PrefixList": {
        "PrefixListId": "pl-0123456abcabcabc1",
        "AddressFamily": "IPv4",
        "State": "create-in-progress",
        "PrefixListArn": "arn:aws:ec2:us-west-2:123456789012:prefix-list/
pl-0123456abcabcabc1",
        "PrefixListName": "vpc-cidrs",
        "MaxEntries": 10,
        "Version": 1,
        "Tags": [],
        "OwnerId": "123456789012"
    }
}
```

Per ulteriori informazioni, consulta Elenchi di prefissi gestiti nella Guida per l'utente di Amazon VPC.

- Per i dettagli sull'API, vedere CreateManagedPrefixListin AWS CLI Command Reference.

`create-nat-gateway`

Il seguente esempio di codice mostra come utilizzare`create-nat-gateway`.

AWS CLI

Esempio 1: come creare un gateway NAT pubblico

L'esempio `create-nat-gateway` seguente crea un gateway NAT pubblico nella sottorete specificata e associa l'indirizzo IP elastico all'ID di allocazione specificato. Quando si crea un gateway NAT pubblico, occorre associare un indirizzo IP elastico.

```
aws ec2 create-nat-gateway \
    --subnet-id subnet-0250c25a1fEXAMPLE \
    --allocation-id eipalloc-09ad461b0dEXAMPLE
```

Output:

```
{
    "NatGateway": {
        "CreateTime": "2021-12-01T22:22:38.000Z",
        "NatGatewayAddresses": [
            {
                "AllocationId": "eipalloc-09ad461b0dEXAMPLE"
            }
        ],
        "NatGatewayId": "nat-0c61bf8a12EXAMPLE",
        "State": "pending",
        "SubnetId": "subnet-0250c25a1fEXAMPLE",
        "VpcId": "vpc-0a60eb65b4EXAMPLE",
        "ConnectivityType": "public"
    }
}
```

Per ulteriori informazioni, consulta Gateway NAT nella Guida per l'utente di Amazon VPC.

Esempio 2: come creare un gateway NAT privato

L'esempio `create-nat-gateway` seguente crea un gateway NAT privato nella sottorete specificata. Un gateway NAT privato non ha un indirizzo IP elastico associato.

```
aws ec2 create-nat-gateway \
    --subnet-id subnet-0250c25a1fEXAMPLE \
    --connectivity-type private
```

Output:

```
{
    "NatGateway": {
        "CreateTime": "2021-12-01T22:26:00.000Z",
```

```
        "NatGatewayAddresses": [
            {}
        ],
        "NatGatewayId": "nat-011b568379EXAMPLE",
        "State": "pending",
        "SubnetId": "subnet-0250c25a1fEXAMPLE",
        "VpcId": "vpc-0a60eb65b4EXAMPLE",
        "ConnectivityType": "private"
    }
}
```

Per ulteriori informazioni, consulta Gateway NAT nella Guida per l'utente di Amazon VPC.

- Per i dettagli sull'API, consulta CreateNatGateway AWS CLICommand Reference.

`create-network-acl-entry`

Il seguente esempio di codice mostra come utilizzare`create-network-acl-entry`.

AWS CLI

Come creare una voce della lista di controllo degli accessi alla rete (ACL)

Questo esempio crea una voce per la lista di controllo degli accessi alla rete (ACL) specificata. La regola consente il traffico in ingresso da qualsiasi IPv4 indirizzo (0.0.0.0/0) sulla porta UDP 53 (DNS) in qualsiasi sottorete associata. Se il comando va a buon fine, non viene restituito alcun output.

Comando:

```
aws ec2 create-network-acl-entry --network-acl-id acl-5fb85d36 --ingress --rule-number 100 --protocol udp --port-range From=53,To=53 --cidr-block 0.0.0.0/0 --rule-action allow
```

Questo esempio crea una regola per l'ACL di rete specificato che consente il traffico in ingresso da qualsiasi IPv6 indirizzo (:: /0) sulla porta TCP 80 (HTTP).

Comando:

```
aws ec2 create-network-acl-entry --network-acl-id acl-5fb85d36 --ingress --rule-number 120 --protocol tcp --port-range From=80,To=80 --ipv6-cidr-block ::/0 --rule-action allow
```

- Per i dettagli sull'API, vedere CreateNetworkAclEntryin Command Reference.AWS CLI

`create-network-acl`

Il seguente esempio di codice mostra come utilizzare`create-network-acl`.

AWS CLI

Come creare una lista di controllo degli accessi alla rete

Questo esempio crea una lista di controllo degli accessi alla rete per il VPC specificato.

Comando:

```
aws ec2 create-network-acl --vpc-id vpc-a01106c2
```

Output:

```
{
    "NetworkAcl": {
        "Associations": [],
        "NetworkAclId": "acl-5fb85d36",
        "VpcId": "vpc-a01106c2",
        "Tags": [],
        "Entries": [
            {
                "CidrBlock": "0.0.0.0/0",
                "RuleNumber": 32767,
                "Protocol": "-1",
                "Egress": true,
                "RuleAction": "deny"
            },
            {
                "CidrBlock": "0.0.0.0/0",
                "RuleNumber": 32767,
                "Protocol": "-1",
                "Egress": false,
                "RuleAction": "deny"
            }
        ],
        "IsDefault": false
    }
```

```
}
```

- Per i dettagli sull'API, consulta CreateNetworkAcl AWS CLICommand Reference.

`create-network-insights-access-scope`

Il seguente esempio di codice mostra come utilizzare`create-network-insights-access-scope`.

AWS CLI

Come creare un ambito di accesso alla rete

L'esempio `create-network-insights-access-scope` seguente crea un ambito di accesso alla rete.

```
aws ec2 create-network-insights-access-scope \
    --cli-input-json file://access-scope-file.json
```

Contenuto di `access-scope-file.json`:

```
{
    "MatchPaths": [
        {
            "Source": {
                "ResourceStatement": {
                    "Resources": [
                        "vpc-abcd12e3"
                    ]
                }
            }
         }
    ],
    "ExcludePaths": [
        {
            "Source": {
                "ResourceStatement": {
                    "ResourceTypes": [
                        "AWS::EC2::InternetGateway"
                    ]
                }
            }
```

```
        }
    ]
}
```

Output:

```
{
    "NetworkInsightsAccessScope": {
        "NetworkInsightsAccessScopeId": "nis-123456789abc01234",
        "NetworkInsightsAccessScopeArn": "arn:aws:ec2:us-
east-1:123456789012:network-insights-access-scope/nis-123456789abc01234",
        "CreatedDate": "2022-01-25T19:20:28.796000+00:00",
        "UpdatedDate": "2022-01-25T19:20:28.797000+00:00"
    },
    "NetworkInsightsAccessScopeContent": {
        "NetworkInsightsAccessScopeId": "nis-123456789abc01234",
        "MatchPaths": [
            {
                "Source": {
                    "ResourceStatement": {
                        "Resources": [
                            "vpc-abcd12e3"
                        ]
                    }
                }
            }
        ],
        "ExcludePaths": [
            {
                "Source": {
                    "ResourceStatement": {
                        "ResourceTypes": [
                            "AWS::EC2::InternetGateway"
                        ]
                    }
                }
            }
        ]
    }
}
```

Per ulteriori informazioni, consulta Guida introduttiva a Network Access Analyzer utilizzando la AWS CLI nella Guida Network Access Analyzer.

- Per i dettagli sull'API, consulta Command CreateNetworkInsightsAccessScopeReference AWS CLI .

`create-network-insights-path`

Il seguente esempio di codice mostra come utilizzare`create-network-insights-path`.

AWS CLI

Come creare un percorso

L'esempio `create-network-insights-path` seguente crea un percorso. L'origine è il gateway Internet specificato e la destinazione è l' EC2 istanza specificata. Per determinare se la destinazione è raggiungibile utilizzando il protocollo e la porta specificati, analizza il percorso utilizzando il comando `start-network-insights-analysis`.

```
aws ec2 create-network-insights-path \
    --source igw-0797cccdc9d73b0e5 \
    --destination i-0495d385ad28331c7 \
    --destination-port 22 \
    --protocol TCP
```

Output:

```
{
    "NetworkInsightsPaths": {
        "NetworkInsightsPathId": "nip-0b26f224f1d131fa8",
        "NetworkInsightsPathArn": "arn:aws:ec2:us-east-1:123456789012:network-
insights-path/nip-0b26f224f1d131fa8",
        "CreatedDate": "2021-01-20T22:43:46.933Z",
        "Source": "igw-0797cccdc9d73b0e5",
        "Destination": "i-0495d385ad28331c7",
        "Protocol": "tcp"
    }
}
```

Per ulteriori informazioni, consulta Guida introduttiva all'uso della AWS CLI nella Reachability Analyzer Guide.

- Per i dettagli sull'API, consulta Command CreateNetworkInsightsPathReference AWS CLI .

`create-network-interface-permission`

Il seguente esempio di codice mostra come utilizzare`create-network-interface-permission`.

AWS CLI

Come creare l'autorizzazione di un'interfaccia di rete

Questo esempio concede l'autorizzazione all'account `123456789012` per collegare l'interfaccia di rete `eni-1a2b3c4d` a un'istanza.

Comando:

```
aws ec2 create-network-interface-permission --network-interface-id eni-1a2b3c4d --
aws-account-id 123456789012 --permission INSTANCE-ATTACH
```

Output:

```
{
  "InterfacePermission": {
      "PermissionState": {
          "State": "GRANTED"
      },
      "NetworkInterfacePermissionId": "eni-perm-06fd19020ede149ea",
      "NetworkInterfaceId": "eni-1a2b3c4d",
      "Permission": "INSTANCE-ATTACH",
      "AwsAccountId": "123456789012"
  }
}
```

- Per i dettagli sull'API, consulta CreateNetworkInterfacePermission AWS CLICommand Reference.

`create-network-interface`

Il seguente esempio di codice mostra come utilizzare`create-network-interface`.

AWS CLI

Esempio 1: specificare un IPv4 indirizzo per un'interfaccia di rete

L'`create-network-interface`esempio seguente crea un'interfaccia di rete per la sottorete specificata con l' IPv4 indirizzo primario specificato.

```
aws ec2 create-network-interface \
    --subnet-id subnet-00a24d0d67acf6333 \
    --description "my network interface" \
    --groups sg-09dfba7ed20cda78b \
    --private-ip-address 10.0.8.17
```

Output:

```
{
    "NetworkInterface": {
        "AvailabilityZone": "us-west-2a",
        "Description": "my network interface",
        "Groups": [
            {
                "GroupName": "my-security-group",
                "GroupId": "sg-09dfba7ed20cda78b"
            }
        ],
        "InterfaceType": "interface",
        "Ipv6Addresses": [],
        "MacAddress": "06:6a:0f:9a:49:37",
        "NetworkInterfaceId": "eni-0492b355f0cf3b3f8",
        "OwnerId": "123456789012",
        "PrivateDnsName": "ip-10-0-8-18.us-west-2.compute.internal",
        "PrivateIpAddress": "10.0.8.17",
        "PrivateIpAddresses": [
            {
                "Primary": true,
                "PrivateDnsName": "ip-10-0-8-17.us-west-2.compute.internal",
                "PrivateIpAddress": "10.0.8.17"
            }
        ],
        "RequesterId": "AIDA4Z3Y7GSXTMEXAMPLE",
        "RequesterManaged": false,
        "SourceDestCheck": true,
        "Status": "pending",
        "SubnetId": "subnet-00a24d0d67acf6333",
        "TagSet": [],
        "VpcId": "vpc-02723a0feeeb9d57b"
    }
```

```
}
```

Esempio 2: creare un'interfaccia di rete con un IPv4 indirizzo e un IPv6 indirizzo

L'`create-network-interface`esempio seguente crea un'interfaccia di rete per la sottorete specificata con un IPv4 indirizzo e un IPv6 indirizzo selezionati da Amazon EC2.

```
aws ec2 create-network-interface \
    --subnet-id subnet-00a24d0d67acf6333 \
    --description "my dual stack network interface" \
    --ipv6-address-count 1 \
    --groups sg-09dfba7ed20cda78b
```

Output:

```
{
    "NetworkInterface": {
        "AvailabilityZone": "us-west-2a",
        "Description": "my dual stack network interface",
        "Groups": [
            {
                "GroupName": "my-security-group",
                "GroupId": "sg-09dfba7ed20cda78b"
            }
        ],
        "InterfaceType": "interface",
        "Ipv6Addresses": [
            {
                "Ipv6Address": "2600:1f13:cfe:3650:a1dc:237c:393a:4ba7",
                "IsPrimaryIpv6": false
            }
        ],
        "MacAddress": "06:b8:68:d2:b2:2d",
        "NetworkInterfaceId": "eni-05da417453f9a84bf",
        "OwnerId": "123456789012",
        "PrivateDnsName": "ip-10-0-8-18.us-west-2.compute.internal",
        "PrivateIpAddress": "10.0.8.18",
        "PrivateIpAddresses": [
            {
                "Primary": true,
                "PrivateDnsName": "ip-10-0-8-18.us-west-2.compute.internal",
                "PrivateIpAddress": "10.0.8.18"
```

```
            }
        ],
        "RequesterId": "AIDA4Z3Y7GSXTMEXAMPLE",
        "RequesterManaged": false,
        "SourceDestCheck": true,
        "Status": "pending",
        "SubnetId": "subnet-00a24d0d67acf6333",
        "TagSet": [],
        "VpcId": "vpc-02723a0feeeb9d57b",
        "Ipv6Address": "2600:1f13:cfe:3650:a1dc:237c:393a:4ba7"
    }
}
```

Esempio 3: come creare un'interfaccia di rete con opzioni di configurazione per il monitoraggio delle connessioni

L'esempio `create-network-interface` seguente crea un'interfaccia di rete e configura i timeout di tracciamento delle connessioni inattive.

```
aws ec2 create-network-interface \
    --subnet-id subnet-00a24d0d67acf6333 \
    --groups sg-02e57dbcfe0331c1b \
    --connection-tracking-specification TcpEstablishedTimeout=86400,UdpTimeout=60
```

Output:

```
{
    "NetworkInterface": {
        "AvailabilityZone": "us-west-2a",
        "ConnectionTrackingConfiguration": {
            "TcpEstablishedTimeout": 86400,
            "UdpTimeout": 60
        },
        "Description": "",
        "Groups": [
            {
                "GroupName": "my-security-group",
                "GroupId": "sg-02e57dbcfe0331c1b"
            }
        ],
        "InterfaceType": "interface",
        "Ipv6Addresses": [],
```

```
        "MacAddress": "06:4c:53:de:6d:91",
        "NetworkInterfaceId": "eni-0c133586e08903d0b",
        "OwnerId": "123456789012",
        "PrivateDnsName": "ip-10-0-8-94.us-west-2.compute.internal",
        "PrivateIpAddress": "10.0.8.94",
        "PrivateIpAddresses": [
            {
                "Primary": true,
                "PrivateDnsName": "ip-10-0-8-94.us-west-2.compute.internal",
                "PrivateIpAddress": "10.0.8.94"
            }
        ],
        "RequesterId": "AIDA4Z3Y7GSXTMEXAMPLE",
        "RequesterManaged": false,
        "SourceDestCheck": true,
        "Status": "pending",
        "SubnetId": "subnet-00a24d0d67acf6333",
        "TagSet": [],
        "VpcId": "vpc-02723a0feeeb9d57b"
    }
}
```

Esempio 4: come creare un Elastic Fabric Adapter

L'esempio `create-network-interface` seguente crea un EFA.

```
aws ec2 create-network-interface \
    --interface-type efa \
    --subnet-id subnet-00a24d0d67acf6333 \
    --description "my efa" \
    --groups sg-02e57dbcfe0331c1b
```

Output:

```
{
    "NetworkInterface": {
        "AvailabilityZone": "us-west-2a",
        "Description": "my efa",
        "Groups": [
            {
                "GroupName": "my-efa-sg",
                "GroupId": "sg-02e57dbcfe0331c1b"
            }
```

```
        ],
        "InterfaceType": "efa",
        "Ipv6Addresses": [],
        "MacAddress": "06:d7:a4:f7:4d:57",
        "NetworkInterfaceId": "eni-034acc2885e862b65",
        "OwnerId": "123456789012",
        "PrivateDnsName": "ip-10-0-8-180.us-west-2.compute.internal",
        "PrivateIpAddress": "10.0.8.180",
        "PrivateIpAddresses": [
            {
                "Primary": true,
                "PrivateDnsName": "ip-10-0-8-180.us-west-2.compute.internal",
                "PrivateIpAddress": "10.0.8.180"
            }
        ],
        "RequesterId": "AIDA4Z3Y7GSXTMEXAMPLE",
        "RequesterManaged": false,
        "SourceDestCheck": true,
        "Status": "pending",
        "SubnetId": "subnet-00a24d0d67acf6333",
        "TagSet": [],
        "VpcId": "vpc-02723a0feeeb9d57b"
    }
}
```

Per ulteriori informazioni, consulta Interfacce di rete elastiche nella Amazon EC2 User Guide.

- Per i dettagli sull'API, consulta CreateNetworkInterface AWS CLICommand Reference.

`create-placement-group`

Il seguente esempio di codice mostra come utilizzare`create-placement-group`.

AWS CLI

Come creare un gruppo di posizionamento

Questo comando di esempio crea un gruppo di posizionamento con il nome specificato.

Comando:

```
aws ec2 create-placement-group --group-name my-cluster --strategy cluster
```

Come creare un gruppo di posizionamento delle partizioni

Questo comando di esempio crea un gruppo di posizionamento delle partizioni denominato `HDFS-Group-A` con cinque partizioni.

Comando:

```
aws ec2 create-placement-group --group-name HDFS-Group-A --strategy partition --
partition-count 5
```

- Per i dettagli sull'API, consulta CreatePlacementGroup AWS CLICommand Reference.

`create-replace-root-volume-task`

Il seguente esempio di codice mostra come utilizzare`create-replace-root-volume-task`.

AWS CLI

Esempio 1: come ripristinare un volume root allo stato di avvio iniziale

L'esempio `create-replace-root-volume-task` seguente ripristina il volume root dell'istanza i-0123456789abcdefa allo stato di avvio iniziale.

```
aws ec2 create-replace-root-volume-task \
    --instance-id i-0123456789abcdefa
```

Output:

```
{
    "ReplaceRootVolumeTask":
    {
        "InstanceId": "i-0123456789abcdefa",
            "ReplaceRootVolumeTaskId": "replacevol-0111122223333abcd",
            "TaskState": "pending",
            "StartTime": "2022-03-14T15:06:38Z",
            "Tags": []
    }
}
```

Esempio 2: come ripristinare un volume root in uno snapshot specifico

L'esempio `create-replace-root-volume-task` seguente ripristina il volume root dell'istanza i-0123456789abcdefa allo snapshot snap-0abcdef1234567890.

```
aws ec2 create-replace-root-volume-task \
    --instance-id i-0123456789abcdefa \
    --snapshot-id  snap-0abcdef1234567890
```

Output:

```
{
    "ReplaceRootVolumeTask":
    {
        "InstanceId": "i-0123456789abcdefa",
        "ReplaceRootVolumeTaskId": "replacevol-0555566667777abcd",
        "TaskState": "pending",
        "StartTime": "2022-03-14T15:16:28Z",
        "Tags": []
    }
}
```

Per ulteriori informazioni, consulta Replace a root volume nella Amazon EC2 User Guide.

- Per i dettagli sull'API, consulta CreateReplaceRootVolumeTask AWS CLICommand Reference.

`create-reserved-instances-listing`

Il seguente esempio di codice mostra come utilizzare`create-reserved-instances-listing`.

AWS CLI

Come elencare un'istanza riservata nel Marketplace di istanze riservate

L'esempio `create-reserved-instances-listing` seguente crea un elenco per l'istanza riservata specificata nel Marketplace di istanze riservate.

```
aws ec2 create-reserved-instances-listing \
    --reserved-instances-id 5ec28771-05ff-4b9b-aa31-9e57dexample \
    --instance-count 3 \
    --price-schedules CurrencyCode=USD,Price=25.50 \
    --client-token 550e8400-e29b-41d4-a716-446655440000
```

- Per i dettagli sull'API, consulta CreateReservedInstancesListing AWS CLICommand Reference.

`create-restore-image-task`

Il seguente esempio di codice mostra come utilizzare`create-restore-image-task`.

AWS CLI

Come ripristinare un'AMI da un bucket S3

L'esempio `create-restore-image-task` seguente ripristina un'AMI da un bucket S3. Utilizza i valori per `S3ObjectKey `` and ``Bucket` dall'output `describe-store-image-tasks`, specifica la chiave oggetto dell'AMI e il nome del bucket S3 in cui è stata copiata l'AMI e specifica il nome dell'AMI ripristinata. Il nome di questo account deve essere univoco nella regione. AMIs L'AMI ripristinata riceverà un nuovo ID AMI.

```
aws ec2 create-restore-image-task \
    --object-key ami-1234567890abcdef0.bin \
    --bucket my-ami-bucket \
    --name 'New AMI Name'
```

Output:

```
{
    "ImageId": "ami-0eab20fe36f83e1a8"
}
```

Per ulteriori informazioni, consulta Archiviare e ripristinare un'AMI utilizzando S3 nella Amazon EC2 User Guide.

- Per i dettagli sull'API, consulta CreateRestoreImageTask AWS CLICommand Reference.

`create-route-table`

Il seguente esempio di codice mostra come utilizzare`create-route-table`.

AWS CLI

Come creare una tabella di routing

Nell'esempio seguente viene creata una tabella di routing per il VPC specificato.

Comando:

```
aws ec2 create-route-table --vpc-id vpc-a01106c2
```

Output:

```
{
    "RouteTable": {
        "Associations": [],
        "RouteTableId": "rtb-22574640",
        "VpcId": "vpc-a01106c2",
        "PropagatingVgws": [],
        "Tags": [],
        "Routes": [
            {
                "GatewayId": "local",
                "DestinationCidrBlock": "10.0.0.0/16",
                "State": "active"
            }
        ]
    }
}
```

- Per i dettagli sull'API, consulta CreateRouteTable AWS CLICommand Reference.

`create-route`

Il seguente esempio di codice mostra come utilizzare`create-route`.

AWS CLI

Come creare una route

Questo esempio crea una route per la tabella di routing specificata. Il percorso corrisponde a tutto IPv4 il traffico (`0.0.0.0/0`) e lo indirizza al gateway Internet specificato. Se il comando va a buon fine, non viene restituito alcun output.

Comando:

```
aws ec2 create-route --route-table-id rtb-22574640 --destination-cidr-
block 0.0.0.0/0 --gateway-id igw-c0a643a9
```

Questo comando di esempio crea una route nella tabella di routing rtb-g8ff4ea2. Il percorso corrisponde al traffico per il blocco IPv4 CIDR 10.0.0.0/16 e lo indirizza alla connessione peering VPC, pcx-111aaa22. Questa route consente di indirizzare il traffico al VPC peer nella connessione peering VPC. Se il comando va a buon fine, non viene restituito alcun output.

Comando:

```
aws ec2 create-route --route-table-id rtb-g8ff4ea2 --destination-cidr-
block 10.0.0.0/16 --vpc-peering-connection-id pcx-1a2b3c4d
```

Questo esempio crea una route nella tabella di route specificata che corrisponde a tutto IPv6 il traffico () e la indirizza verso il gateway Internet di sola uscita specificato. `::/0`

Comando:

```
aws ec2 create-route --route-table-id rtb-dce620b8 --destination-ipv6-cidr-
block ::/0 --egress-only-internet-gateway-id eigw-01eadbd45ecd7943f
```

- Per i dettagli sull'API, vedere CreateRoutein AWS CLI Command Reference.

`create-security-group`

Il seguente esempio di codice mostra come utilizzare`create-security-group`.

AWS CLI

Per creare un gruppo di sicurezza per EC2 -Classic

Nell'esempio seguente viene creato un gruppo di sicurezza denominato `MySecurityGroup`.

Comando:

```
aws ec2 create-security-group --group-name MySecurityGroup --description "My
 security group"
```

Output:

```
{
    "GroupId": "sg-903004f8"
```

```
}
```

Per creare un gruppo di sicurezza per EC2 -VPC

Nell'esempio seguente viene creato un gruppo di sicurezza denominato `MySecurityGroup` per il VPC specificato.

Comando:

```
aws ec2 create-security-group --group-name MySecurityGroup --description "My
 security group" --vpc-id vpc-1a2b3c4d
```

Output:

```
{
    "GroupId": "sg-903004f8"
}
```

Per ulteriori informazioni, consulta Utilizzo dei gruppi di sicurezza nella Guida per l'utente dell'Interfaccia a riga di comando AWS .

- Per i dettagli sull'API, consulta AWS CLI Command CreateSecurityGroupReference.

`create-snapshot`

Il seguente esempio di codice mostra come utilizzare`create-snapshot`.

AWS CLI

Come creare uno snapshot

Questo comando di esempio crea uno snapshot del volume con l'ID volume `vol-1234567890abcdef0` e una breve descrizione per identificare lo snapshot.

Comando:

```
aws ec2 create-snapshot --volume-id vol-1234567890abcdef0 --description "This is my
 root volume snapshot"
```

Output:

```
{
    "Description": "This is my root volume snapshot",
    "Tags": [],
    "Encrypted": false,
    "VolumeId": "vol-1234567890abcdef0",
    "State": "pending",
    "VolumeSize": 8,
    "StartTime": "2018-02-28T21:06:01.000Z",
    "Progress": "",
    "OwnerId": "012345678910",
    "SnapshotId": "snap-066877671789bd71b"
}
```

Come creare uno snapshot con tag

Questo comando di esempio crea uno snapshot e applica due tag, ovvero purpose=prod e costcenter=123.

Comando:

```
aws ec2 create-snapshot --volume-id vol-1234567890abcdef0 --description 'Prod
 backup' --tag-specifications 'ResourceType=snapshot,Tags=[{Key=purpose,Value=prod},
{Key=costcenter,Value=123}]'
```

Output:

```
{
    "Description": "Prod backup",
    "Tags": [
        {
            "Value": "prod",
            "Key": "purpose"
        },
        {
            "Value": "123",
            "Key": "costcenter"
        }
     ],
     "Encrypted": false,
     "VolumeId": "vol-1234567890abcdef0",
     "State": "pending",
     "VolumeSize": 8,
```

```
    "StartTime": "2018-02-28T21:06:06.000Z",
    "Progress": "",
    "OwnerId": "012345678910",
    "SnapshotId": "snap-09ed24a70bc19bbe4"
}
```

- Per i dettagli sull'API, consulta CreateSnapshot AWS CLICommand Reference.

`create-snapshots`

Il seguente esempio di codice mostra come utilizzare`create-snapshots`.

AWS CLI

Esempio 1: come creare uno snapshot a più volumi

L'esempio `create-snapshots` seguente crea gli snapshot di tutti i volumi collegati all'istanza specificata.

```
aws ec2 create-snapshots \
    --instance-specification InstanceId=i-1234567890abcdef0 \
    --description "This is snapshot of a volume from my-instance"
```

Output:

```
{
    "Snapshots": [
        {
            "Description": "This is a snapshot of a volume from my-instance",
            "Tags": [],
            "Encrypted": false,
            "VolumeId": "vol-0a01d2d5a34697479",
            "State": "pending",
            "VolumeSize": 16,
            "StartTime": "2019-08-05T16:58:19.000Z",
            "Progress": "",
            "OwnerId": "123456789012",
            "SnapshotId": "snap-07f30e3909aa0045e"
        },
        {
            "Description": "This is a snapshot of a volume from my-instance",
            "Tags": [],
```

```
            "Encrypted": false,
            "VolumeId": "vol-02d0d4947008cb1a2",
            "State": "pending",
            "VolumeSize": 20,
            "StartTime": "2019-08-05T16:58:19.000Z",
            "Progress": "",
            "OwnerId": "123456789012",
            "SnapshotId": "snap-0ec20b602264aad48"
        },
        ...
    ]
}
```

Esempio 2: come creare uno snapshot a più volumi con tag dal volume di origine

L'esempio `create-snapshots` seguente crea snapshot di tutti i volumi collegati all'istanza specificata e copia i tag da ciascun volume nello snapshot corrispondente.

```
aws ec2 create-snapshots \
    --instance-specification InstanceId=i-1234567890abcdef0 \
    --copy-tags-from-source volume \
    --description "This is snapshot of a volume from my-instance"
```

Output:

```
{
    "Snapshots": [
        {
            "Description": "This is a snapshot of a volume from my-instance",
            "Tags": [
                {
                    "Key": "Name",
                    "Value": "my-volume"
                }
            ],
            "Encrypted": false,
            "VolumeId": "vol-02d0d4947008cb1a2",
            "State": "pending",
            "VolumeSize": 20,
            "StartTime": "2019-08-05T16:53:04.000Z",
            "Progress": "",
            "OwnerId": "123456789012",
```

```
            "SnapshotId": "snap-053bfaeb821a458dd"
        }
        ...
    ]
}
```

Esempio 3: come creare uno snapshot a più volumi che non includa il volume root

L'esempio `create-snapshots` seguente crea uno snapshot di tutti i volumi collegati all'istanza specificata ad eccezione del volume root.

```
aws ec2 create-snapshots \
    --instance-specification InstanceId=i-1234567890abcdef0,ExcludeBootVolume=true
```

Vedi l'esempio 1 per un output di esempio.

Esempio 4: come creare uno snapshot a più volumi e aggiungere tag

L'esempio `create-snapshots` seguente crea gli snapshot di tutti i volumi collegati all'istanza specificata e aggiunge due tag a ciascun snapshot.

```
aws ec2 create-snapshots \
    --instance-specification InstanceId=i-1234567890abcdef0 \
    --tag-specifications 'ResourceType=snapshot,Tags=[{Key=Name,Value=backup},
{Key=costcenter,Value=123}]'
```

Vedi l'esempio 1 per un output di esempio.

- Per i dettagli sull'API, consulta CreateSnapshots AWS CLICommand Reference.

`create-spot-datafeed-subscription`

Il seguente esempio di codice mostra come utilizzare`create-spot-datafeed-subscription`.

AWS CLI

Come creare un feed di dati dell'istanza spot

L'esempio `create-spot-datafeed-subscription` seguente crea un feed di dati dell'istanza spot.

```
aws ec2 create-spot-datafeed-subscription \
    --bucket amzn-s3-demo-bucket \
    --prefix spot-data-feed
```

Output:

```
{
    "SpotDatafeedSubscription": {
        "Bucket": "amzn-s3-demo-bucket",
        "OwnerId": "123456789012",
        "Prefix": "spot-data-feed",
        "State": "Active"
    }
}
```

Il feed di dati viene archiviato nel bucket Amazon S3 specificato. I nomi di file di questo feed di dati hanno il formato seguente.

```
amzn-s3-demo-bucket.s3.amazonaws.com/spot-data-feed/123456789012.YYYY-MM-DD-
HH.n.abcd1234.gz
```

Per ulteriori informazioni, consulta il data feed di istanze Spot nella Amazon EC2 User Guide.

- Per i dettagli sull'API, consulta CreateSpotDatafeedSubscription AWS CLICommand Reference.

`create-store-image-task`

Il seguente esempio di codice mostra come utilizzare`create-store-image-task`.

AWS CLI

Come archiviare un'AMI in un bucket S3

L'esempio `create-store-image-task` seguente archivia un'AMI in un bucket S3. Specificare l'ID AMI e il nome del bucket S3 in cui archiviare l'AMI.

```
aws ec2 create-store-image-task \
  --image-id ami-1234567890abcdef0 \
  --bucket my-ami-bucket
```

Output:

```
{
    "ObjectKey": "ami-1234567890abcdef0.bin"
}
```

Per ulteriori informazioni, consulta Archiviare e ripristinare un'AMI utilizzando S3 nella Amazon EC2 User Guide.

- Per i dettagli sull'API, consulta CreateStoreImageTask AWS CLICommand Reference.

`create-subnet-cidr-reservation`

Il seguente esempio di codice mostra come utilizzare`create-subnet-cidr-reservation`.

AWS CLI

Come creare una prenotazione CIDR della sottorete

L'esempio `create-subnet-cidr-reservation` seguente crea una prenotazione CIDR della sottorete per la sottorete e l'intervallo CIDR specificati.

```
aws ec2 create-subnet-cidr-reservation \
    --subnet-id subnet-03c51e2eEXAMPLE \
    --reservation-type prefix \
    --cidr 10.1.0.20/26
```

Output:

```
{
    "SubnetCidrReservation": {
        "SubnetCidrReservationId": "scr-044f977c4eEXAMPLE",
        "SubnetId": "subnet-03c51e2e6cEXAMPLE",
        "Cidr": "10.1.0.16/28",
        "ReservationType": "prefix",
        "OwnerId": "123456789012"
    }
}
```

Per ulteriori informazioni, consulta Prenotazioni della CIDR per la sottorete nella Guida per l'utente di Amazon VPC.

- Per i dettagli sull'API, consulta CreateSubnetCidrReservation AWS CLICommand Reference.

`create-subnet`

Il seguente esempio di codice mostra come utilizzare`create-subnet`.

AWS CLI

Esempio 1: creare una sottorete con solo un blocco IPv4 CIDR

L'`create-subnet`esempio seguente crea una sottorete nel VPC specificato con il blocco CIDR IPv4 specificato.

```
aws ec2 create-subnet \
    --vpc-id vpc-081ec835f3EXAMPLE \
    --cidr-block 10.0.0.0/24 \
    --tag-specifications ResourceType=subnet,Tags=[{Key=Name,Value=my-ipv4-only-
subnet}]
```

Output:

```
{
    "Subnet": {
        "AvailabilityZone": "us-west-2a",
        "AvailabilityZoneId": "usw2-az2",
        "AvailableIpAddressCount": 251,
        "CidrBlock": "10.0.0.0/24",
        "DefaultForAz": false,
        "MapPublicIpOnLaunch": false,
        "State": "available",
        "SubnetId": "subnet-0e99b93155EXAMPLE",
        "VpcId": "vpc-081ec835f3EXAMPLE",
        "OwnerId": "123456789012",
        "AssignIpv6AddressOnCreation": false,
        "Ipv6CidrBlockAssociationSet": [],
        "Tags": [
            {
                "Key": "Name",
                "Value": "my-ipv4-only-subnet"
            }
        ],
        "SubnetArn": "arn:aws:ec2:us-west-2:123456789012:subnet/
subnet-0e99b93155EXAMPLE"
```

```
    }
}
```

Esempio 2: creare una sottorete con entrambi i blocchi CIDR IPv4 IPv6

L'`create-subnet`esempio seguente crea una sottorete nel VPC specificato con i blocchi IPv4 specificati IPv6 e CIDR.

```
aws ec2 create-subnet \
    --vpc-id vpc-081ec835f3EXAMPLE \
    --cidr-block 10.0.0.0/24 \
    --ipv6-cidr-block 2600:1f16:cfe:3660::/64 \
    --tag-specifications ResourceType=subnet,Tags=[{Key=Name,Value=my-ipv4-ipv6-
subnet}]
```

Output:

```
{
    "Subnet": {
        "AvailabilityZone": "us-west-2a",
        "AvailabilityZoneId": "usw2-az2",
        "AvailableIpAddressCount": 251,
        "CidrBlock": "10.0.0.0/24",
        "DefaultForAz": false,
        "MapPublicIpOnLaunch": false,
        "State": "available",
        "SubnetId": "subnet-0736441d38EXAMPLE",
        "VpcId": "vpc-081ec835f3EXAMPLE",
        "OwnerId": "123456789012",
        "AssignIpv6AddressOnCreation": false,
        "Ipv6CidrBlockAssociationSet": [
            {
                "AssociationId": "subnet-cidr-assoc-06c5f904499fcc623",
                "Ipv6CidrBlock": "2600:1f13:cfe:3660::/64",
                "Ipv6CidrBlockState": {
                    "State": "associating"
                }
            }
        ],
        "Tags": [
            {
                "Key": "Name",
                "Value": "my-ipv4-ipv6-subnet"
```

```
            }
        ],
        "SubnetArn": "arn:aws:ec2:us-west-2:123456789012:subnet/
subnet-0736441d38EXAMPLE"
    }
}
```

Esempio 3: creare una sottorete con solo un blocco CIDR IPv6

L'`create-subnet`esempio seguente crea una sottorete nel VPC specificato con il blocco CIDR IPv6 specificato.

```
aws ec2 create-subnet \
    --vpc-id vpc-081ec835f3EXAMPLE \
    --ipv6-native \
    --ipv6-cidr-block 2600:1f16:115:200::/64 \
    --tag-specifications ResourceType=subnet,Tags=[{Key=Name,Value=my-ipv6-only-
subnet}]
```

Output:

```
{
    "Subnet": {
        "AvailabilityZone": "us-west-2a",
        "AvailabilityZoneId": "usw2-az2",
        "AvailableIpAddressCount": 0,
        "DefaultForAz": false,
        "MapPublicIpOnLaunch": false,
        "State": "available",
        "SubnetId": "subnet-03f720e7deEXAMPLE",
        "VpcId": "vpc-081ec835f3EXAMPLE",
        "OwnerId": "123456789012",
        "AssignIpv6AddressOnCreation": true,
        "Ipv6CidrBlockAssociationSet": [
            {
                "AssociationId": "subnet-cidr-assoc-01ef639edde556709",
                "Ipv6CidrBlock": "2600:1f13:cfe:3660::/64",
                "Ipv6CidrBlockState": {
                    "State": "associating"
                }
            }
        ],
        "Tags": [
```

```
            {
                "Key": "Name",
                "Value": "my-ipv6-only-subnet"
            }
        ],
        "SubnetArn": "arn:aws:ec2:us-west-2:123456789012:subnet/
subnet-03f720e7deEXAMPLE"
    }
}
```

Per ulteriori informazioni, consulta VPCs e sottoreti nella Amazon VPC User Guide.

- Per i dettagli sull'API, consulta Command CreateSubnetReference AWS CLI .

`create-tags`

Il seguente esempio di codice mostra come utilizzare`create-tags`.

AWS CLI

Esempio 1: come aggiungere un tag a una risorsa

Nell'esempio di `create-tags` seguente viene aggiunto il tag `Stack=production` all'immagine specificata o sovrascritto un tag esistente per l'AMI in cui la chiave tag è `Stack`.

```
aws ec2 create-tags \
    --resources ami-1234567890abcdef0 \
    --tags Key=Stack,Value=production
```

Questo comando non produce alcun output.

Esempio 2: come aggiungere tag a più risorse

Nell'esempio di `create-tags` seguente vengono aggiunti (o sovrascritti) due tag per un'AMI e un'istanza. Uno dei tag ha una chiave (`webserver`), ma nessun valore (il valore è impostato su una stringa vuota). L'altro tag ha una chiave (`stack`) e un valore (`Production`).

```
aws ec2 create-tags \
    --resources ami-1a2b3c4d i-1234567890abcdef0 \
    --tags Key=webserver,Value=   Key=stack,Value=Production
```

Questo comando non produce alcun output.

Esempio 3: come aggiungere tag contenenti caratteri speciali

L'esempio `create-tags` seguente aggiunge il tag `[Group]=test` a un'istanza. Le parentesi quadre ([e]) sono caratteri speciali per i quali occorre eseguire l'escape. Negli esempi seguenti viene utilizzato anche il carattere di continuazione della riga adeguato a ogni ambiente.

Se si utilizza Windows, racchiudere l'elemento con caratteri speciali tra virgolette doppie (""), quindi anteporre ad ogni virgoletta doppia una barra rovesciata (\) come segue:

```
aws ec2 create-tags ^
    --resources i-1234567890abcdef0 ^
    --tags Key=\"[Group]\",Value=test
```

Se utilizzate Windows PowerShell, racchiudete l'elemento con caratteri speciali tra virgolette doppie («), fate precedere ogni virgoletta doppia da una barra rovesciata (\), quindi racchiudete l'intera struttura di chiavi e valori tra virgolette singole (') come segue.

```
aws ec2 create-tags `
    --resources i-1234567890abcdef0 `
    --tags 'Key=\"[Group]\",Value=test'
```

Se si utilizza Linux o OS X, racchiudere l'elemento contenente caratteri speciali con virgolette doppie (""), quindi racchiudere l'intera struttura di chiave e valore tra virgolette singole (''), come segue:

```
aws ec2 create-tags \
    --resources i-1234567890abcdef0 \
    --tags 'Key="[Group]",Value=test'
```

Per ulteriori informazioni, consulta Tagga le tue EC2 risorse Amazon nella Amazon EC2 User Guide.

- Per i dettagli sull'API, consulta CreateTags AWS CLICommand Reference.

`create-traffic-mirror-filter-rule`

Il seguente esempio di codice mostra come utilizzare`create-traffic-mirror-filter-rule`.

AWS CLI

Come creare una regola di filtro per il traffico TCP in entrata

L'esempio `create-traffic-mirror-filter-rule` seguente crea una regola che è possibile utilizzare per eseguire il mirroring di tutto il traffico TCP in entrata. Prima di eseguire questo comando, utilizza `create-traffic-mirror-filter` per creare il filtro di mirroring del traffico.

```
aws ec2 create-traffic-mirror-filter-rule \
    --description 'TCP Rule' \
    --destination-cidr-block 0.0.0.0/0  \
    --protocol 6 \
    --rule-action accept \
    --rule-number 1 \
    --source-cidr-block 0.0.0.0/0 \
    --traffic-direction ingress \
    --traffic-mirror-filter-id tmf-04812ff784b25ae67
```

Output:

```
{
    "TrafficMirrorFilterRule": {
        "DestinationCidrBlock": "0.0.0.0/0",
        "TrafficMirrorFilterId": "tmf-04812ff784b25ae67",
        "TrafficMirrorFilterRuleId": "tmfr-02d20d996673f3732",
        "SourceCidrBlock": "0.0.0.0/0",
        "TrafficDirection": "ingress",
        "Description": "TCP Rule",
        "RuleNumber": 1,
        "RuleAction": "accept",
        "Protocol": 6
    },
    "ClientToken": "4752b573-40a6-4eac-a8a4-a72058761219"
}
```

Per ulteriori informazioni, consulta Creazione di un filtro di mirroring del traffico nella Guida di Mirroring del traffico.

- Per i dettagli sull'API, consulta CreateTrafficMirrorFilterRule AWS CLICommand Reference.

`create-traffic-mirror-filter`

Il seguente esempio di codice mostra come utilizzare`create-traffic-mirror-filter`.

AWS CLI

Come creare un filtro di mirroring del traffico

L'esempio `create-traffic-mirror-filter` seguente crea un filtro di mirroring del traffico. Dopo aver creato il filtro, utilizza `create-traffic-mirror-filter-rule` per aggiungere regole.

```
aws ec2 create-traffic-mirror-filter \
    --description 'TCP Filter'
```

Output:

```
{
    "ClientToken": "28908518-100b-4987-8233-8c744EXAMPLE",
    "TrafficMirrorFilter": {
        "TrafficMirrorFilterId": "tmf-04812ff784EXAMPLE",
        "Description": "TCP Filter",
        "EgressFilterRules": [],
        "IngressFilterRules": [],
        "Tags": [],
        "NetworkServices": []
    }
}
```

Per ulteriori informazioni, consulta Creazione di un filtro di mirroring del traffico nella Guida di Mirroring del traffico.

- Per i dettagli sull'API, consulta CreateTrafficMirrorFilter AWS CLICommand Reference.

`create-traffic-mirror-session`

Il seguente esempio di codice mostra come utilizzare`create-traffic-mirror-session`.

AWS CLI

Come creare una sessione di mirroring del traffico

Il comando `create-traffic-mirror-session` seguente crea una sessione di mirroring del traffico per l'origine e la destinazione specificate per 25 byte del pacchetto.

```
aws ec2 create-traffic-mirror-session \
```

```
    --description 'example session' \
    --traffic-mirror-target-id tmt-07f75d8feeEXAMPLE \
    --network-interface-id eni-070203f901EXAMPLE \
    --session-number 1  \
    --packet-length 25 \
    --traffic-mirror-filter-id tmf-04812ff784EXAMPLE
```

Output:

```
{
    "TrafficMirrorSession": {
        "TrafficMirrorSessionId": "tms-08a33b1214EXAMPLE",
        "TrafficMirrorTargetId": "tmt-07f75d8feeEXAMPLE",
        "TrafficMirrorFilterId": "tmf-04812ff784EXAMPLE",
        "NetworkInterfaceId": "eni-070203f901EXAMPLE",
        "OwnerId": "111122223333",
        "PacketLength": 25,
        "SessionNumber": 1,
        "VirtualNetworkId": 7159709,
        "Description": "example session",
        "Tags": []
    },
    "ClientToken": "5236cffc-ee13-4a32-bb5b-388d9da09d96"
}
```

Per ulteriori informazioni, consulta Creazione di un filtro del mirroring del traffico nella Guida di Mirroring del traffico.

- Per i dettagli sull'API, consulta CreateTrafficMirrorSession AWS CLICommand Reference.

`create-traffic-mirror-target`

Il seguente esempio di codice mostra come utilizzare`create-traffic-mirror-target`.

AWS CLI

Come creare una destinazione di mirroring del traffico con un Network Load Balancer

L'esempio `create-traffic-mirror-target` seguente crea una destinazione di mirroring del traffico di Network Load Balancer.

```
aws ec2 create-traffic-mirror-target \
```

```
    --description 'Example Network Load Balancer Target' \
    --network-load-balancer-arn arn:aws:elasticloadbalancing:us-
east-1:111122223333:loadbalancer/net/NLB/7cdec873EXAMPLE
```

Output:

```
{
    "TrafficMirrorTarget": {
        "Type": "network-load-balancer",
        "Tags": [],
        "Description": "Example Network Load Balancer Target",
        "OwnerId": "111122223333",
        "NetworkLoadBalancerArn": "arn:aws:elasticloadbalancing:us-
east-1:724145273726:loadbalancer/net/NLB/7cdec873EXAMPLE",
        "TrafficMirrorTargetId": "tmt-0dabe9b0a6EXAMPLE"
    },
    "ClientToken": "d5c090f5-8a0f-49c7-8281-72c796a21f72"
}
```

Come creare una destinazione di mirroring del traffico di rete

L'esempio `create-traffic-mirror-target` seguente crea una destinazione di mirroring del traffico per l'interfaccia di rete.

```
aws ec2 create-traffic-mirror-target \
    --description 'Network interface target' \
    --network-interface-id eni-eni-01f6f631eEXAMPLE
```

Output:

```
{
    "ClientToken": "5289a345-0358-4e62-93d5-47ef3061d65e",
    "TrafficMirrorTarget": {
        "Description": "Network interface target",
        "NetworkInterfaceId": "eni-01f6f631eEXAMPLE",
        "TrafficMirrorTargetId": "tmt-02dcdbe2abEXAMPLE",
        "OwnerId": "111122223333",
        "Type": "network-interface",
        "Tags": []
    }
}
```

Per ulteriori informazioni, consulta Creazione di una destinazione di mirroring del traffico nella Guida di Mirroring del traffico.

- Per i dettagli sull'API, consulta CreateTrafficMirrorTarget AWS CLICommand Reference.

`create-transit-gateway-connect-peer`

Il seguente esempio di codice mostra come utilizzare`create-transit-gateway-connect-peer`.

AWS CLI

Come creare un peer di Transit Gateway Connect

L'esempio `create-transit-gateway-connect-peer` seguente crea un peer di Connect.

```
aws ec2 create-transit-gateway-connect-peer \
    --transit-gateway-attachment-id tgw-attach-0f0927767cEXAMPLE \
    --peer-address 172.31.1.11 \
    --inside-cidr-blocks 169.254.6.0/29
```

Output:

```
{
    "TransitGatewayConnectPeer": {
        "TransitGatewayAttachmentId": "tgw-attach-0f0927767cEXAMPLE",
        "TransitGatewayConnectPeerId": "tgw-connect-peer-0666adbac4EXAMPLE",
        "State": "pending",
        "CreationTime": "2021-10-13T03:35:17.000Z",
        "ConnectPeerConfiguration": {
            "TransitGatewayAddress": "10.0.0.234",
            "PeerAddress": "172.31.1.11",
            "InsideCidrBlocks": [
                "169.254.6.0/29"
            ],
            "Protocol": "gre",
            "BgpConfigurations": [
                {
                    "TransitGatewayAsn": 64512,
                    "PeerAsn": 64512,
                    "TransitGatewayAddress": "169.254.6.2",
                    "PeerAddress": "169.254.6.1",
                    "BgpStatus": "down"
                },
```

```
                {
                    "TransitGatewayAsn": 64512,
                    "PeerAsn": 64512,
                    "TransitGatewayAddress": "169.254.6.3",
                    "PeerAddress": "169.254.6.1",
                    "BgpStatus": "down"
                }
            ]
        }
    }
}
```

Per ulteriori informazioni, consulta Collegamenti Transit gateway Connect e peer Connect nella Guida di Transit Gateway.

- Per i dettagli sull'API, consulta CreateTransitGatewayConnectPeer AWS CLICommand Reference.

`create-transit-gateway-connect`

Il seguente esempio di codice mostra come utilizzare`create-transit-gateway-connect`.

AWS CLI

Come creare un collegamento per il Transit Gateway Connect

L'esempio `create-transit-gateway-connect` seguente crea un collegamento Connect, con il protocollo "gre" per il collegamento specificato.

```
aws ec2 create-transit-gateway-connect \
    --transport-transit-gateway-attachment-id tgw-attach-0a89069f57EXAMPLE \
    --options "Protocol=gre"
```

Output:

```
{
    "TransitGatewayConnect": {
        "TransitGatewayAttachmentId": "tgw-attach-037012e5dcEXAMPLE",
        "TransportTransitGatewayAttachmentId": "tgw-attach-0a89069f57EXAMPLE",
        "TransitGatewayId": "tgw-02f776b1a7EXAMPLE",
        "State": "pending",
        "CreationTime": "2021-03-09T19:59:17+00:00",
        "Options": {
```

```
            "Protocol": "gre"
        }
    }
}
```

Per ulteriori informazioni, consulta Collegamenti Transit gateway Connect e peer Connect nella Guida di Transit Gateway.

- Per i dettagli sull'API, consulta CreateTransitGatewayConnect AWS CLICommand Reference.

`create-transit-gateway-multicast-domain`

Il seguente esempio di codice mostra come utilizzare`create-transit-gateway-multicast-domain`.

AWS CLI

Esempio 1: come creare un dominio multicast (trasmissione uno a molti) IGMP

L'esempio `create-transit-gateway-multicast-domain` seguente crea un dominio multicast (trasmissione uno a molti) per il gateway di transito specificato. Con le fonti statiche disabilitate, tutte le istanze presenti nelle sottoreti associate al dominio multicast (trasmissione uno a molti) possono inviare traffico multicast (trasmissione uno a molti). Se almeno un membro utilizza il protocollo IGMP, è necessario abilitare il IGMPv2 supporto.

```
aws ec2 create-transit-gateway-multicast-domain \
    --transit-gateway-id tgw-0bf0bffefaEXAMPLE \
    --options StaticSourcesSupport=disable,Igmpv2Support=enable
```

Output:

```
{
    "TransitGatewayMulticastDomain": {
        "TransitGatewayMulticastDomainId": "tgw-mcast-domain-0c9e29e2a7EXAMPLE",
        "TransitGatewayId": "tgw-0bf0bffefaEXAMPLE",
        "TransitGatewayMulticastDomainArn": "arn:aws:ec2:us-
west-2:123456789012:transit-gateway-multicast-domain/tgw-mcast-
domain-0c9e29e2a7EXAMPLE",
        "OwnerId": "123456789012",
        "Options": {
            "Igmpv2Support": "enable",
            "StaticSourcesSupport": "disable",
```

```
            "AutoAcceptSharedAssociations": "disable"
        },
        "State": "pending",
        "CreationTime": "2021-09-29T22:17:13.000Z"
    }
}
```

Esempio 2: come creare un dominio multicast (trasmissione uno a molti) statico

L'esempio `create-transit-gateway-multicast-domain` seguente crea un dominio multicast (trasmissione uno a molti) per il gateway di transito specificato. Con le fonti statiche abilitate, è necessario aggiungere fonti staticamente.

```
aws ec2 create-transit-gateway-multicast-domain \
    --transit-gateway-id tgw-0bf0bffefaEXAMPLE \
    --options StaticSourcesSupport=enable,Igmpv2Support=disable
```

Output:

```
{
    "TransitGatewayMulticastDomain": {
        "TransitGatewayMulticastDomainId": "tgw-mcast-domain-000fb24d04EXAMPLE",
        "TransitGatewayId": "tgw-0bf0bffefaEXAMPLE",
        "TransitGatewayMulticastDomainArn": "arn:aws:ec2:us-
west-2:123456789012:transit-gateway-multicast-domain/tgw-mcast-
domain-000fb24d04EXAMPLE",
        "OwnerId": "123456789012",
        "Options": {
            "Igmpv2Support": "disable",
            "StaticSourcesSupport": "enable",
            "AutoAcceptSharedAssociations": "disable"
        },
        "State": "pending",
        "CreationTime": "2021-09-29T22:20:19.000Z"
    }
}
```

Per ulteriori informazioni, consulta Gestione dei domini multicast (trasmissione uno a molti) nella Guida di Transit Gateway.

- Per i dettagli sull'API, vedere CreateTransitGatewayMulticastDomainin AWS CLI Command Reference.

`create-transit-gateway-peering-attachment`

Il seguente esempio di codice mostra come utilizzare`create-transit-gateway-peering-attachment`.

AWS CLI

Come creare un collegamento peering del gateway di transito

L'esempio `create-transit-gateway-peering-attachment` seguente crea una richiesta di collegamento peering tra i due gateway di transito specificati.

```
aws ec2 create-transit-gateway-peering-attachment \
    --transit-gateway-id tgw-123abc05e04123abc \
    --peer-transit-gateway-id tgw-11223344aabbcc112 \
    --peer-account-id 123456789012 \
    --peer-region us-east-2
```

Output:

```
{
    "TransitGatewayPeeringAttachment": {
        "TransitGatewayAttachmentId": "tgw-attach-4455667788aabbccd",
        "RequesterTgwInfo": {
            "TransitGatewayId": "tgw-123abc05e04123abc",
            "OwnerId": "123456789012",
            "Region": "us-west-2"
        },
        "AccepterTgwInfo": {
            "TransitGatewayId": "tgw-11223344aabbcc112",
            "OwnerId": "123456789012",
            "Region": "us-east-2"
        },
        "State": "initiatingRequest",
        "CreationTime": "2019-12-09T11:38:05.000Z"
    }
}
```

Per ulteriori informazioni, consulta Accessori di peering del gateway di transito nella Guida di Transit Gateway.

- Per i dettagli sull'API, consulta CreateTransitGatewayPeeringAttachment AWS CLICommand Reference.

`create-transit-gateway-policy-table`

Il seguente esempio di codice mostra come utilizzare`create-transit-gateway-policy-table`.

AWS CLI

Come creare una tabella di policy del gateway di transito

L'esempio `create-transit-gateway-policy-table` seguente crea una tabella di policy del gateway di transito per il gateway di transito specificato.

```
aws ec2 create-transit-gateway-policy-table \
    --transit-gateway-id tgw-067f8505c18f0bd6e
```

Output:

```
{
    "TransitGatewayPolicyTable": {
        "TransitGatewayPolicyTableId": "tgw-ptb-0a16f134b78668a81",
        "TransitGatewayId": "tgw-067f8505c18f0bd6e",
        "State": "pending",
        "CreationTime": "2023-11-28T16:36:43+00:00"
    }
}
```

Per ulteriori informazioni, consulta Tabelle di policy del gateway di transito nella Guida per l'utente di Transit Gateway.

- Per i dettagli sull'API, consulta CreateTransitGatewayPolicyTable AWS CLICommand Reference.

`create-transit-gateway-prefix-list-reference`

Il seguente esempio di codice mostra come utilizzare`create-transit-gateway-prefix-list-reference`.

AWS CLI

Come creare un riferimento all'elenco di prefissi

L'esempio `create-transit-gateway-prefix-list-reference` seguente crea un riferimento all'elenco di prefissi specificato nella tabella di routing del gateway di transito specificato.

```
aws ec2 create-transit-gateway-prefix-list-reference \
    --transit-gateway-route-table-id tgw-rtb-0123456789abcd123 \
    --prefix-list-id pl-1111112222222333 \
    --transit-gateway-attachment-id tgw-attach-aaaaaabbbbbb11111
```

Output:

```
{
    "TransitGatewayPrefixListReference": {
        "TransitGatewayRouteTableId": "tgw-rtb-0123456789abcd123",
        "PrefixListId": "pl-11111122222222333",
        "PrefixListOwnerId": "123456789012",
        "State": "pending",
        "Blackhole": false,
        "TransitGatewayAttachment": {
            "TransitGatewayAttachmentId": "tgw-attach-aaaaaabbbbbb11111",
            "ResourceType": "vpc",
            "ResourceId": "vpc-112233445566aabbc"
        }
    }
}
```

Per ulteriori informazioni, consulta Creare un riferimento all'elenco dei prefissi nella Guida di Transit Gateway.

- Per i dettagli sull'API, consulta CreateTransitGatewayPrefixListReference AWS CLICommand Reference.

`create-transit-gateway-route-table`

Il seguente esempio di codice mostra come utilizzare`create-transit-gateway-route-table`.

AWS CLI

Come creare una tabella di routing di un gateway di transito

L'esempio `create-transit-gateway-route-table` seguente crea una tabella di routing per il gateway di transito specificato.

```
aws ec2 create-transit-gateway-route-table \
    --transit-gateway-id tgw-0262a0e521EXAMPLE
```

Output:

```
{
    "TransitGatewayRouteTable": {
        "TransitGatewayRouteTableId": "tgw-rtb-0960981be7EXAMPLE",
        "TransitGatewayId": "tgw-0262a0e521EXAMPLE",
        "State": "pending",
        "DefaultAssociationRouteTable": false,
        "DefaultPropagationRouteTable": false,
        "CreationTime": "2019-07-10T19:01:46.000Z"
    }
}
```

Per ulteriori informazioni, consulta Creare una tabella di routing del gateway di transito nella Guida di Transit Gateway.

- Per i dettagli sull'API, consulta CreateTransitGatewayRouteTable AWS CLICommand Reference.

`create-transit-gateway-route`

Il seguente esempio di codice mostra come utilizzare`create-transit-gateway-route`.

AWS CLI

Come creare una route del gateway di transito

L'esempio `create-transit-gateway-route` seguente crea una route, con la destinazione specificata, per la tabella di routing specificata.

```
aws ec2 create-transit-gateway-route \
    --destination-cidr-block 10.0.2.0/24 \
    --transit-gateway-route-table-id tgw-rtb-0b6f6aaa01EXAMPLE \
    --transit-gateway-attachment-id tgw-attach-0b5968d3b6EXAMPLE
```

Output:

```
{
    "Route": {
```

```
        "DestinationCidrBlock": "10.0.2.0/24",
        "TransitGatewayAttachments": [
            {
                "ResourceId": "vpc-0065acced4EXAMPLE",
                "TransitGatewayAttachmentId": "tgw-attach-0b5968d3b6EXAMPLE",
                "ResourceType": "vpc"
            }
        ],
        "Type": "static",
        "State": "active"
    }
}
```

Per ulteriori informazioni, consulta Tabelle di routing del gateway di transito nella Guida di Transit Gateway.

- Per i dettagli sull'API, consulta CreateTransitGatewayRoute AWS CLICommand Reference.

`create-transit-gateway-vpc-attachment`

Il seguente esempio di codice mostra come utilizzare`create-transit-gateway-vpc-attachment`.

AWS CLI

Esempio 1: come associare un gateway di transito a un VPC

L'esempio `create-transit-gateway-vpc-attachment` seguente crea un collegamento del gateway di transito alla VPN al VPC specificato.

```
aws ec2 create-transit-gateway-vpc-attachment \
    --transit-gateway-id tgw-0262a0e521EXAMPLE \
    --vpc-id vpc-07e8ffd50f49335df \
    --subnet-id subnet-0752213d59EXAMPLE
```

Output:

```
{
    "TransitGatewayVpcAttachment": {
        "TransitGatewayAttachmentId": "tgw-attach-0a34fe6b4fEXAMPLE",
        "TransitGatewayId": "tgw-0262a0e521EXAMPLE",
        "VpcId": "vpc-07e8ffd50fEXAMPLE",
        "VpcOwnerId": "111122223333",
```

```
        "State": "pending",
        "SubnetIds": [
            "subnet-0752213d59EXAMPLE"
        ],
        "CreationTime": "2019-07-10T17:33:46.000Z",
        "Options": {
            "DnsSupport": "enable",
            "Ipv6Support": "disable"
        }
    }
}
```

Per ulteriori informazioni, consulta Creare un collegamento del gateway di transito alla VPN a un VPC nella Guida di Transit Gateway.

Esempio 2: come associare un gateway di transito a più sottoreti in un VPC

L'esempio `create-transit-gateway-vpc-attachment` seguente crea un collegamento del gateway di transito alla VPN al VPC e alle sottoreti specificati.

```
aws ec2 create-transit-gateway-vpc-attachment \
    --transit-gateway-id tgw-02f776b1a7EXAMPLE  \
    --vpc-id vpc-3EXAMPLE \
    --subnet-ids "subnet-dEXAMPLE" "subnet-6EXAMPLE"
```

Output:

```
{
    "TransitGatewayVpcAttachment": {
        "TransitGatewayAttachmentId": "tgw-attach-0e141e0bebEXAMPLE",
        "TransitGatewayId": "tgw-02f776b1a7EXAMPLE",
        "VpcId": "vpc-3EXAMPLE",
        "VpcOwnerId": "111122223333",
        "State": "pending",
        "SubnetIds": [
            "subnet-6EXAMPLE",
            "subnet-dEXAMPLE"
        ],
        "CreationTime": "2019-12-17T20:07:52.000Z",
        "Options": {
            "DnsSupport": "enable",
            "Ipv6Support": "disable"
        }
```

```
    }
}
```

Per ulteriori informazioni, consulta Creare un collegamento del gateway di transito alla VPN a un VPC nella Guida di Transit Gateway.

- Per i dettagli sull'API, consulta CreateTransitGatewayVpcAttachment AWS CLICommand Reference.

`create-transit-gateway`

Il seguente esempio di codice mostra come utilizzare`create-transit-gateway`.

AWS CLI

Come creare un gateway di transito

L'esempio `create-transit-gateway` seguente crea un gateway di transito.

```
aws ec2 create-transit-gateway \
    --description MyTGW \
    --
options AmazonSideAsn=64516,AutoAcceptSharedAttachments=enable,DefaultRouteTableAssociation=
```

Output:

```
{
    "TransitGateway": {
        "TransitGatewayId": "tgw-0262a0e521EXAMPLE",
        "TransitGatewayArn": "arn:aws:ec2:us-east-2:111122223333:transit-gateway/
tgw-0262a0e521EXAMPLE",
        "State": "pending",
        "OwnerId": "111122223333",
        "Description": "MyTGW",
        "CreationTime": "2019-07-10T14:02:12.000Z",
        "Options": {
            "AmazonSideAsn": 64516,
            "AutoAcceptSharedAttachments": "enable",
            "DefaultRouteTableAssociation": "enable",
            "AssociationDefaultRouteTableId": "tgw-rtb-018774adf3EXAMPLE",
            "DefaultRouteTablePropagation": "enable",
            "PropagationDefaultRouteTableId": "tgw-rtb-018774adf3EXAMPLE",
            "VpnEcmpSupport": "enable",
```

```
            "DnsSupport": "enable"
        }
    }
}
```

Per ulteriori informazioni, consulta Creazione di un gateway di transito nella Guida di Transit Gateway.

- Per i dettagli sull'API, consulta CreateTransitGateway AWS CLICommand Reference.

`create-verified-access-endpoint`

Il seguente esempio di codice mostra come utilizzare`create-verified-access-endpoint`.

AWS CLI

Come creare un endpoint di accesso verificato

L'esempio `create-verified-access-endpoint` seguente crea un endpoint di accesso verificato per il gruppo di accesso verificato specificato. L'interfaccia di rete e il gruppo di sicurezza specificati devono appartenere allo stesso VPC.

```
aws ec2 create-verified-access-endpoint \
    --verified-access-group-id vagr-0dbe967baf14b7235 \
    --endpoint-type network-interface \
    --attachment-type vpc \
    --domain-certificate-arn arn:aws:acm:us-east-2:123456789012:certificate/
eb065ea0-26f9-4e75-a6ce-0a1a7EXAMPLE \
    --application-domain example.com \
    --endpoint-domain-prefix my-ava-app \
    --security-group-ids sg-004915970c4c8f13a \
    --network-interface-
options NetworkInterfaceId=eni-0aec70418c8d87a0f,Protocol=https,Port=443 \
    --tag-specifications ResourceType=verified-access-
endpoint,Tags=[{Key=Name,Value=my-va-endpoint}]
```

Output:

```
{
    "VerifiedAccessEndpoint": {
        "VerifiedAccessInstanceId": "vai-0ce000c0b7643abea",
        "VerifiedAccessGroupId": "vagr-0dbe967baf14b7235",
        "VerifiedAccessEndpointId": "vae-066fac616d4d546f2",
```

```
        "ApplicationDomain": "example.com",
        "EndpointType": "network-interface",
        "AttachmentType": "vpc",
        "DomainCertificateArn": "arn:aws:acm:us-east-2:123456789012:certificate/
eb065ea0-26f9-4e75-a6ce-0a1a7EXAMPLE",
        "EndpointDomain": "my-ava-
app.edge-00c3372d53b1540bb.vai-0ce000c0b7643abea.prod.verified-access.us-
east-2.amazonaws.com",
        "SecurityGroupIds": [
            "sg-004915970c4c8f13a"
        ],
        "NetworkInterfaceOptions": {
            "NetworkInterfaceId": "eni-0aec70418c8d87a0f",
            "Protocol": "https",
            "Port": 443
        },
        "Status": {
            "Code": "pending"
        },
        "Description": "",
        "CreationTime": "2023-08-25T20:54:43",
        "LastUpdatedTime": "2023-08-25T20:54:43",
        "Tags": [
            {
                "Key": "Name",
                "Value": "my-va-endpoint"
            }
        ]
    }
}
```

Per ulteriori informazioni, consulta Endpoint di Accesso verificato nella Guida per l'utente di Accesso verificato da AWS .

- Per i dettagli sull'API, consulta CreateVerifiedAccessEndpoint AWS CLICommand Reference.

`create-verified-access-group`

Il seguente esempio di codice mostra come utilizzare`create-verified-access-group`.

AWS CLI

Come creare un gruppo di accesso verificato

L'esempio `create-verified-access-group` seguente crea un gruppo di accesso verificato per l'istanza di accesso verificato specificata.

```
aws ec2 create-verified-access-group \
    --verified-access-instance-id vai-0ce000c0b7643abea \
    --tag-specifications ResourceType=verified-access-
group,Tags=[{Key=Name,Value=my-va-group}]
```

Output:

```
{
    "VerifiedAccessGroup": {
        "VerifiedAccessGroupId": "vagr-0dbe967baf14b7235",
        "VerifiedAccessInstanceId": "vai-0ce000c0b7643abea",
        "Description": "",
        "Owner": "123456789012",
        "VerifiedAccessGroupArn": "arn:aws:ec2:us-east-2:123456789012:verified-
access-group/vagr-0dbe967baf14b7235",
        "CreationTime": "2023-08-25T19:55:19",
        "LastUpdatedTime": "2023-08-25T19:55:19",
        "Tags": [
            {
                "Key": "Name",
                "Value": "my-va-group"
            }
        ]
    }
}
```

Per ulteriori informazioni, consulta Gruppi di Accesso verificato nella Guida per l'utente di Accesso verificato da AWS .

- Per i dettagli sull'API, consulta CreateVerifiedAccessGroup AWS CLICommand Reference.

`create-verified-access-instance`

Il seguente esempio di codice mostra come utilizzare`create-verified-access-instance`.

AWS CLI

Come creare un'istanza di accesso verificato

L'esempio `create-verified-access-instance` seguente crea un'istanza di accesso verificato con un tag del nome.

```
aws ec2 create-verified-access-instance \
    --tag-specifications ResourceType=verified-access-
instance,Tags=[{Key=Name,Value=my-va-instance}]
```

Output:

```
{
    "VerifiedAccessInstance": {
        "VerifiedAccessInstanceId": "vai-0ce000c0b7643abea",
        "Description": "",
        "VerifiedAccessTrustProviders": [],
        "CreationTime": "2023-08-25T18:27:56",
        "LastUpdatedTime": "2023-08-25T18:27:56",
        "Tags": [
            {
                "Key": "Name",
                "Value": "my-va-instance"
            }
        ]
    }
}
```

Per ulteriori informazioni, consulta Istanze di Accesso verificato nella Guida per l'utente di Accesso verificato da AWS .

- Per i dettagli sull'API, consulta CreateVerifiedAccessInstance AWS CLICommand Reference.

`create-verified-access-trust-provider`

Il seguente esempio di codice mostra come utilizzare`create-verified-access-trust-provider`.

AWS CLI

Come creare un provider di attendibilità di accesso verificato

L'`create-verified-access-trust-provider`esempio seguente configura un provider fiduciario ad accesso verificato utilizzando AWS Identity Center.

```
aws ec2 create-verified-access-trust-provider \
    --trust-provider-type user \
    --user-trust-provider-type iam-identity-center \
    --policy-reference-name idc \
    --tag-specifications ResourceType=verified-access-trust-
provider,Tags=[{Key=Name,Value=my-va-trust-provider}]
```

Output:

```
{
    "VerifiedAccessTrustProvider": {
        "VerifiedAccessTrustProviderId": "vatp-0bb32de759a3e19e7",
        "Description": "",
        "TrustProviderType": "user",
        "UserTrustProviderType": "iam-identity-center",
        "PolicyReferenceName": "idc",
        "CreationTime": "2023-08-25T18:40:36",
        "LastUpdatedTime": "2023-08-25T18:40:36",
        "Tags": [
            {
                "Key": "Name",
                "Value": "my-va-trust-provider"
            }
        ]
    }
}
```

Per ulteriori informazioni, consulta Provider di attendibilità per Accesso verificato nella Guida per l'utente di Accesso verificato da AWS .

- Per i dettagli sull'API, consulta CreateVerifiedAccessTrustProvider AWS CLICommand Reference.

`create-volume`

Il seguente esempio di codice mostra come utilizzare`create-volume`.

AWS CLI

Come creare un volume SSD per uso generico (gp2) vuoto

L'esempio `create-volume` seguente crea un volume SSD (gp2) per scopi generici da 80 GiB nella zona di disponibilità specificata. Si noti che la Regione corrente deve essere `us-east-1`, oppure è possibile aggiungere il parametro `--region` per specificare la Regione per il comando.

```
aws ec2 create-volume \
    --volume-type gp2 \
    --size 80 \
    --availability-zone us-east-1a
```

Output:

```
{
    "AvailabilityZone": "us-east-1a",
    "Tags": [],
    "Encrypted": false,
    "VolumeType": "gp2",
    "VolumeId": "vol-1234567890abcdef0",
    "State": "creating",
    "Iops": 240,
    "SnapshotId": "",
    "CreateTime": "YYYY-MM-DDTHH:MM:SS.000Z",
    "Size": 80
}
```

Se non si specifica un tipo di volume, il tipo di volume predefinito è gp2.

```
aws ec2 create-volume \
    --size 80 \
    --availability-zone us-east-1a
```

Esempio 2: come creare un volume SSD con IOPS con capacità di IOPS allocata (io1) da un'snapshot

L'esempio `create-volume` seguente crea un volume SSD con capacità di IOPS allocata (io1) con 1000 IOPS assegnati nella zona di disponibilità specificata utilizzando lo snapshot specificato.

```
aws ec2 create-volume \
    --volume-type io1 \
    --iops 1000 \
    --snapshot-id snap-066877671789bd71b \
```

```
    --availability-zone us-east-1a
```

Output:

```
{
    "AvailabilityZone": "us-east-1a",
    "Tags": [],
    "Encrypted": false,
    "VolumeType": "io1",
    "VolumeId": "vol-1234567890abcdef0",
    "State": "creating",
    "Iops": 1000,
    "SnapshotId": "snap-066877671789bd71b",
    "CreateTime": "YYYY-MM-DDTHH:MM:SS.000Z",
    "Size": 500
}
```

Esempio 3: come creare un volume crittografato

L'esempio `create-volume` seguente crea un volume crittografato utilizzando la chiave gestita dal cliente (CMK) predefinita per la crittografia EBS. Se la crittografia è disabilitata per impostazione predefinita, è necessario specificare il parametro `--encrypted` come segue.

```
aws ec2 create-volume \
    --size 80 \
    --encrypted \
    --availability-zone us-east-1a
```

Output:

```
{
    "AvailabilityZone": "us-east-1a",
    "Tags": [],
    "Encrypted": true,
    "VolumeType": "gp2",
    "VolumeId": "vol-1234567890abcdef0",
    "State": "creating",
    "Iops": 240,
    "SnapshotId": "",
    "CreateTime": "YYYY-MM-DDTHH:MM:SS.000Z",
    "Size": 80
```

```
}
```

Se la crittografia è abilitata per impostazione predefinita, il seguente comando di esempio crea un volume crittografato anche senza il parametro `--encrypted`.

```
aws ec2 create-volume \
    --size 80 \
    --availability-zone us-east-1a
```

Se si utilizza il parametro `--kms-key-id` per specificare una chiave gestita dal cliente (CMK), è necessario specificare il parametro `--encrypted` anche se la crittografia predefinita è abilitata.

```
aws ec2 create-volume \
    --volume-type gp2 \
    --size 80 \
    --encrypted \
    --kms-key-id 0ea3fef3-80a7-4778-9d8c-1c0c6EXAMPLE \
    --availability-zone us-east-1a
```

Esempio 4: come creare un volume con tag

L'esempio `create-volume` seguente crea un volume e aggiunge due tag.

```
aws ec2 create-volume \
    --availability-zone us-east-1a \
    --volume-type gp2 \
    --size 80 \
    --tag-specifications 'ResourceType=volume,Tags=[{Key=purpose,Value=production},
{Key=cost-center,Value=cc123}]'
```

- Per i dettagli sull'API, consulta CreateVolume AWS CLICommand Reference.

`create-vpc-endpoint-connection-notification`

Il seguente esempio di codice mostra come utilizzare`create-vpc-endpoint-connection-notification`.

AWS CLI

Come creare una notifica di connessione endpoint

Questo esempio crea una notifica per un servizio endpoint specifico che avvisa quando gli endpoint dell'interfaccia si sono connessi al servizio e quando gli endpoint sono stati accettati per il servizio.

Comando:

```
aws ec2 create-vpc-endpoint-connection-notification --connection-notification-
arn arn:aws:sns:us-east-2:123456789012:VpceNotification --connection-
events Connect Accept --service-id vpce-svc-1237881c0d25a3abc
```

Output:

```
{
   "ConnectionNotification": {
       "ConnectionNotificationState": "Enabled",
       "ConnectionNotificationType": "Topic",
       "ServiceId": "vpce-svc-1237881c0d25a3abc",
       "ConnectionEvents": [
           "Accept",
           "Connect"
       ],
       "ConnectionNotificationId": "vpce-nfn-008776de7e03f5abc",
       "ConnectionNotificationArn": "arn:aws:sns:us-
east-2:123456789012:VpceNotification"
   }
 }
```

- Per i dettagli sull'API, consulta CreateVpcEndpointConnectionNotification AWS CLICommand Reference.

`create-vpc-endpoint-service-configuration`

Il seguente esempio di codice mostra come utilizzare`create-vpc-endpoint-service-configuration`.

AWS CLI

Esempio 1: come creare una configurazione del servizio endpoint per un endpoint di interfaccia

L'esempio `create-vpc-endpoint-service-configuration` seguente crea una configurazione del servizio endpoint VPC utilizzando Network Load Balancer `nlb-vpce`. Questo

esempio specifica, inoltre, che le richieste di connessione al servizio tramite un endpoint di interfaccia devono essere accettate.

```
aws ec2 create-vpc-endpoint-service-configuration \
    --network-load-balancer-arns arn:aws:elasticloadbalancing:us-
east-1:123456789012:loadbalancer/net/nlb-vpce/e94221227f1ba532 \
    --acceptance-required
```

Output:

```
{
   "ServiceConfiguration": {
       "ServiceType": [
           {
               "ServiceType": "Interface"
           }
       ],
       "NetworkLoadBalancerArns": [
           "arn:aws:elasticloadbalancing:us-east-1:123456789012:loadbalancer/net/
nlb-vpce/e94221227f1ba532"
       ],
       "ServiceName": "com.amazonaws.vpce.us-east-1.vpce-svc-03d5ebb7d9579a2b3",
       "ServiceState": "Available",
       "ServiceId": "vpce-svc-03d5ebb7d9579a2b3",
       "AcceptanceRequired": true,
       "AvailabilityZones": [
           "us-east-1d"
       ],
       "BaseEndpointDnsNames": [
           "vpce-svc-03d5ebb7d9579a2b3.us-east-1.vpce.amazonaws.com"
       ]
   }
}
```

Per ulteriori informazioni, consulta Creare un servizio endpoint nella Guida per l'AWS PrivateLink utente.

Esempio 2: come creare una configurazione del servizio endpoint per gli endpoint Gateway Load Balancer

L'esempio `create-vpc-endpoint-service-configuration` seguente crea una configurazione del servizio endpoint VPC utilizzando Gateway Load Balancer `GWLBService`.

Le richieste di connessione al servizio tramite un endpoint Gateway Load Balancer vengono accettate automaticamente.

```
aws ec2 create-vpc-endpoint-service-configuration \
    --gateway-load-balancer-arns arn:aws:elasticloadbalancing:us-
east-1:123456789012:loadbalancer/gwy/GWLBService/123123123123abcc \
    --no-acceptance-required
```

Output:

```
{
    "ServiceConfiguration": {
        "ServiceType": [
            {
                "ServiceType": "GatewayLoadBalancer"
            }
        ],
        "ServiceId": "vpce-svc-123123a1c43abc123",
        "ServiceName": "com.amazonaws.vpce.us-east-1.vpce-svc-123123a1c43abc123",
        "ServiceState": "Available",
        "AvailabilityZones": [
            "us-east-1d"
        ],
        "AcceptanceRequired": false,
        "ManagesVpcEndpoints": false,
        "GatewayLoadBalancerArns": [
            "arn:aws:elasticloadbalancing:us-east-1:123456789012:loadbalancer/gwy/
GWLBService/123123123123abcc"
        ]
    }
}
```

Per ulteriori informazioni, consulta Creare un servizio endpoint Gateway Load Balancer nella Guida per l'AWS PrivateLink utente.

- Per i dettagli sull'API, consulta AWS CLI Command CreateVpcEndpointServiceConfigurationReference.

`create-vpc-endpoint`

Il seguente esempio di codice mostra come utilizzare`create-vpc-endpoint`.

AWS CLI

Esempio 1: come creare un endpoint del gateway

L'esempio `create-vpc-endpoint` seguente crea un endpoint VPC del gateway tra il VPC `vpc-1a2b3c4d` e Amazon S3 nella Regione `us-east-1` e associa la tabella di routing `rtb-11aa22bb` all'endpoint.

```
aws ec2 create-vpc-endpoint \
    --vpc-id vpc-1a2b3c4d \
    --service-name com.amazonaws.us-east-1.s3 \
    --route-table-ids rtb-11aa22bb
```

Output:

```
{
    "VpcEndpoint": {
        "PolicyDocument": "{\"Version\":\"2008-10-17\",\"Statement\":[{\"Sid\":\"\",
\"Effect\":\"Allow\",\"Principal\":\"*\",\"Action\":\"*\",\"Resource\":\"*\"}]}",
        "VpcId": "vpc-1a2b3c4d",
        "State": "available",
        "ServiceName": "com.amazonaws.us-east-1.s3",
        "RouteTableIds": [
            "rtb-11aa22bb"
        ],
        "VpcEndpointId": "vpc-1a2b3c4d",
        "CreationTimestamp": "2015-05-15T09:40:50Z"
    }
}
```

Per ulteriori informazioni, consulta Creare un endpoint gateway nella Guida per l'AWS PrivateLink utente.

Esempio 2: come creare un endpoint dell'interfaccia

L'esempio `create-vpc-endpoint` seguente crea un endpoint VPC dell'interfaccia tra il VPC `vpc-1a2b3c4d` e Amazon S3 nella Regione `us-east-1`. Il comando crea l'endpoint nella sottorete `subnet-1a2b3c4d`, lo associa al gruppo di sicurezza `sg-1a2b3c4d` e aggiunge un tag con la chiave "Service" e il valore di "S3".

```
aws ec2 create-vpc-endpoint \
```

```
    --vpc-id vpc-1a2b3c4d \
    --vpc-endpoint-type Interface \
    --service-name com.amazonaws.us-east-1.s3 \
    --subnet-ids subnet-7b16de0c \
    --security-group-id sg-1a2b3c4d \
    --tag-specifications ResourceType=vpc-endpoint,Tags=[{Key=service,Value=S3}]
```

Output:

```
{
    "VpcEndpoint": {
        "VpcEndpointId": "vpce-1a2b3c4d5e6f1a2b3",
        "VpcEndpointType": "Interface",
        "VpcId": "vpc-1a2b3c4d",
        "ServiceName": "com.amazonaws.us-east-1.s3",
        "State": "pending",
        "RouteTableIds": [],
        "SubnetIds": [
            "subnet-1a2b3c4d"
        ],
        "Groups": [
            {
                "GroupId": "sg-1a2b3c4d",
                "GroupName": "default"
            }
        ],
        "PrivateDnsEnabled": false,
        "RequesterManaged": false,
        "NetworkInterfaceIds": [
            "eni-0b16f0581c8ac6877"
        ],
        "DnsEntries": [
            {
                "DnsName": "*.vpce-1a2b3c4d5e6f1a2b3-9hnenorg.s3.us-
east-1.vpce.amazonaws.com",
                "HostedZoneId": "Z7HUB22UULQXV"
            },
            {
                "DnsName": "*.vpce-1a2b3c4d5e6f1a2b3-9hnenorg-us-east-1c.s3.us-
east-1.vpce.amazonaws.com",
                "HostedZoneId": "Z7HUB22UULQXV"
            }
        ],
```

```
        "CreationTimestamp": "2021-03-05T14:46:16.030000+00:00",
        "Tags": [
            {
                "Key": "service",
                "Value": "S3"
            }
        ],
        "OwnerId": "123456789012"
    }
}
```

Per ulteriori informazioni, consulta Creare un endpoint VPC di interfaccia nella Guida per l'AWS PrivateLink utente.

Esempio 3: come creare un endpoint Gateway Load Balancer

L'esempio `create-vpc-endpoint` seguente crea un endpoint Gateway Load Balancer tra il VPC `vpc-111122223333aabbc` e un servizio configurato tramite un Gateway Load Balancer.

```
aws ec2 create-vpc-endpoint \
    --service-name com.amazonaws.vpce.us-east-1.vpce-svc-123123a1c43abc123 \
    --vpc-endpoint-type GatewayLoadBalancer \
    --vpc-id vpc-111122223333aabbc \
    --subnet-ids subnet-0011aabbcc2233445
```

Output:

```
{
    "VpcEndpoint": {
        "VpcEndpointId": "vpce-aabbaabbaabbaabba",
        "VpcEndpointType": "GatewayLoadBalancer",
        "VpcId": "vpc-111122223333aabbc",
        "ServiceName": "com.amazonaws.vpce.us-east-1.vpce-svc-123123a1c43abc123",
        "State": "pending",
        "SubnetIds": [
            "subnet-0011aabbcc2233445"
        ],
        "RequesterManaged": false,
        "NetworkInterfaceIds": [
            "eni-01010120203030405"
        ],
        "CreationTimestamp": "2020-11-11T08:06:03.522Z",
```

```
        "OwnerId": "123456789012"
    }
}
```

Per ulteriori informazioni, consulta gli endpoint Gateway Load Balancer nella Guida per l'AWS PrivateLink utente.

Esempio 4: come creare un endpoint della risorsa

L'esempio `create-vpc-endpoint` seguente crea un endpoint delle risorse.

```
aws ec2 create-vpc-endpoint \
    --vpc-endpoint-type Resource \
    --vpc-id vpc-111122223333aabbc \
    --subnet-ids subnet-0011aabbcc2233445 \
    --resource-configuration-arn arn:aws:vpc-lattice-us-
east-1:123456789012:resourceconfiguration/rcfg-0123abcde98765432
```

Output:

```
{
    "VpcEndpoint": {
        "VpcEndpointId": "vpce-00939a7ed9EXAMPLE",
        "VpcEndpointType": "Resource",
        "VpcId": "vpc-111122223333aabbc",
        "State": "Pending",
        "SubnetIds": [
            "subnet-0011aabbcc2233445"
        ],
        "Groups": [
            {
                "GroupId": "sg-03e2f15fbfc09b000",
                "GroupName": "default"
            }
        ],
        "IpAddressType": "IPV4",
        "PrivateDnsEnabled": false,
        "CreationTimestamp": "2025-02-06T23:38:49.525000+00:00",
        "Tags": [],
        "OwnerId": "123456789012",
        "ResourceConfigurationArn": "arn:aws:vpc-lattice:us-
east-1:123456789012:resourceconfiguration/rcfg-0123abcde98765432"
    }
```

```
}
```

Per ulteriori informazioni, consulta Resource endpoints nella Guida per l'AWS PrivateLink utente.

Esempio 5: come creare un endpoint della rete di servizi

L'esempio `create-vpc-endpoint` seguente crea un endpoint della rete di servizi.

```
aws ec2 create-vpc-endpoint \
    --vpc-endpoint-type ServiceNetwork \
    --vpc-id vpc-111122223333aabbc \
    --subnet-ids subnet-0011aabbcc2233445 \
    --service-network-arn arn:aws:vpc-lattice:us-east-1:123456789012:servicenetwork/
sn-0101abcd5432abcd0 \
    --security-group-ids sg-0123456789012abcd
```

Output:

```
{
    "VpcEndpoint": {
        "VpcEndpointId": "vpce-0f00567fa8EXAMPLE",
        "VpcEndpointType": "ServiceNetwork",
        "VpcId": "vpc-111122223333aabbc",
        "State": "Pending",
        "SubnetIds": [
            "subnet-0011aabbcc2233445"
        ],
        "Groups": [
            {
                "GroupId": "sg-0123456789012abcd",
                "GroupName": "my-security-group"
            }
        ],
        "IpAddressType": "IPV4",
        "PrivateDnsEnabled": false,
        "CreationTimestamp": "2025-02-06T23:44:20.449000+00:00",
        "Tags": [],
        "OwnerId": "123456789012",
        "ServiceNetworkArn": "arn:aws:vpc-lattice:us-
east-1:123456789012:servicenetwork/sn-0101abcd5432abcd0"
    }
}
```

Per ulteriori informazioni, consulta Service Network Endpoints nella Guida per l'AWS PrivateLink utente.

- Per i dettagli sull'API, consulta CreateVpcEndpoint AWS CLICommand Reference.

`create-vpc-peering-connection`

Il seguente esempio di codice mostra come utilizzare`create-vpc-peering-connection`.

AWS CLI

Per creare una connessione peering VPC tra i tuoi VPCs

Questo esempio richiede una connessione peering tra vpc-1a2b3c4d e VPCs vpc-11122233.

Comando:

```
aws ec2 create-vpc-peering-connection --vpc-id vpc-1a2b3c4d --peer-vpc-
id vpc-11122233
```

Output:

```
{
    "VpcPeeringConnection": {
        "Status": {
            "Message": "Initiating Request to 444455556666",
            "Code": "initiating-request"
        },
        "Tags": [],
        "RequesterVpcInfo": {
            "OwnerId": "444455556666",
            "VpcId": "vpc-1a2b3c4d",
            "CidrBlock": "10.0.0.0/28"
        },
        "VpcPeeringConnectionId": "pcx-111aaa111",
        "ExpirationTime": "2014-04-02T16:13:36.000Z",
        "AccepterVpcInfo": {
            "OwnerId": "444455556666",
            "VpcId": "vpc-11122233"
        }
    }
}
```

Come creare una connessione peering VPC a un VPC in un altro account

Questo esempio richiede una connessione peering tra il tuo VPC (vpc-1a2b3c4d) e un VPC (vpc-11122233) che appartiene all'account 123456789012. AWS

Comando:

```
aws ec2 create-vpc-peering-connection --vpc-id vpc-1a2b3c4d --peer-vpc-
id vpc-11122233 --peer-owner-id 123456789012
```

Come creare una connessione peering VPC a un VPC in un'altra Regione

Questo esempio richiede una connessione peering tra il VPC nella Regione corrente (vpc-1a2b3c4d) e un VPC (vpc-11122233) nell'account nella Regione `us-west-2`.

Comando:

```
aws ec2 create-vpc-peering-connection --vpc-id vpc-1a2b3c4d --peer-vpc-
id vpc-11122233 --peer-region us-west-2
```

Questo esempio richiede una connessione peering tra il tuo VPC nella regione corrente (vpc-1a2b3c4d) e un VPC (vpc-11122233) che appartiene all'account 123456789012 che si trova nella regione. AWS `us-west-2`

Comando:

```
aws ec2 create-vpc-peering-connection --vpc-id vpc-1a2b3c4d --peer-vpc-
id vpc-11122233 --peer-owner-id 123456789012 --peer-region us-west-2
```

- Per CreateVpcPeeringConnectioni AWS CLI dettagli sull'API, consulta Command Reference.

`create-vpc`

Il seguente esempio di codice mostra come utilizzare`create-vpc`.

AWS CLI

Esempio 1: come creare un VPC

L'`create-vpc`esempio seguente crea un VPC con il blocco IPv4 CIDR specificato e un tag Name.

```
aws ec2 create-vpc \
    --cidr-block 10.0.0.0/16 \
    --tag-specifications ResourceType=vpc,Tags=[{Key=Name,Value=MyVpc}]
```

Output:

```
{
    "Vpc": {
        "CidrBlock": "10.0.0.0/16",
        "DhcpOptionsId": "dopt-5EXAMPLE",
        "State": "pending",
        "VpcId": "vpc-0a60eb65b4EXAMPLE",
        "OwnerId": "123456789012",
        "InstanceTenancy": "default",
        "Ipv6CidrBlockAssociationSet": [],
        "CidrBlockAssociationSet": [
            {
                "AssociationId": "vpc-cidr-assoc-07501b79ecEXAMPLE",
                "CidrBlock": "10.0.0.0/16",
                "CidrBlockState": {
                    "State": "associated"
                }
            }
        ],
        "IsDefault": false,
        "Tags": [
            {
                "Key": "Name",
                "Value": MyVpc"
            }
        ]
    }
}
```

Esempio 2: come creare un VPC con tenancy dedicata

L'`create-vpc`esempio seguente crea un VPC con il blocco IPv4 CIDR specificato e una tenancy dedicata.

```
aws ec2 create-vpc \
    --cidr-block 10.0.0.0/16 \
    --instance-tenancy dedicated
```

Output:

```
{
    "Vpc": {
        "CidrBlock": "10.0.0.0/16",
        "DhcpOptionsId": "dopt-19edf471",
        "State": "pending",
        "VpcId": "vpc-0a53287fa4EXAMPLE",
        "OwnerId": "111122223333",
        "InstanceTenancy": "dedicated",
        "Ipv6CidrBlockAssociationSet": [],
        "CidrBlockAssociationSet": [
            {
                "AssociationId": "vpc-cidr-assoc-00b24cc1c2EXAMPLE",
                "CidrBlock": "10.0.0.0/16",
                "CidrBlockState": {
                    "State": "associated"
                }
            }
        ],
        "IsDefault": false
    }
}
```

Esempio 3: creare un VPC con un IPv6 blocco CIDR

L'`create-vpc`esempio seguente crea un VPC con un blocco CIDR fornito da Amazon IPv6 .

```
aws ec2 create-vpc \
    --cidr-block 10.0.0.0/16 \
    --amazon-provided-ipv6-cidr-block
```

Output:

```
{
    "Vpc": {
        "CidrBlock": "10.0.0.0/16",
        "DhcpOptionsId": "dopt-dEXAMPLE",
        "State": "pending",
        "VpcId": "vpc-0fc5e3406bEXAMPLE",
        "OwnerId": "123456789012",
        "InstanceTenancy": "default",
```

```
        "Ipv6CidrBlockAssociationSet": [
            {
                "AssociationId": "vpc-cidr-assoc-068432c60bEXAMPLE",
                "Ipv6CidrBlock": "",
                "Ipv6CidrBlockState": {
                    "State": "associating"
                },
                "Ipv6Pool": "Amazon",
                "NetworkBorderGroup": "us-west-2"
            }
        ],
        "CidrBlockAssociationSet": [
            {
                "AssociationId": "vpc-cidr-assoc-0669f8f9f5EXAMPLE",
                "CidrBlock": "10.0.0.0/16",
                "CidrBlockState": {
                    "State": "associated"
                }
            }
        ],
        "IsDefault": false
    }
}
```

Esempio 4: come creare un VPC con un CIDR di un pool IPAM

Nell'esempio di `create-vpc` seguente viene creato un VPC con un CIDR di un pool di Gestione indirizzi IP (IPAM) di Amazon VPC.

Linux e macOS:

```
aws ec2 create-vpc \
    --ipv4-ipam-pool-id ipam-pool-0533048da7d823723 \
    --tag-specifications ResourceType=vpc,Tags='[{Key=Environment,Value="Preprod"},
{Key=Owner,Value="Build Team"}]'
```

Windows:

```
aws ec2 create-vpc ^
    --ipv4-ipam-pool-id ipam-pool-0533048da7d823723 ^
    --tag-specifications ResourceType=vpc,Tags=[{Key=Environment,Value="Preprod"},
{Key=Owner,Value="Build Team"}]
```

Output:

```
{
    "Vpc": {
        "CidrBlock": "10.0.1.0/24",
        "DhcpOptionsId": "dopt-2afccf50",
        "State": "pending",
        "VpcId": "vpc-010e1791024eb0af9",
        "OwnerId": "123456789012",
        "InstanceTenancy": "default",
        "Ipv6CidrBlockAssociationSet": [],
        "CidrBlockAssociationSet": [
            {
                "AssociationId": "vpc-cidr-assoc-0a77de1d803226d4b",
                "CidrBlock": "10.0.1.0/24",
                "CidrBlockState": {
                    "State": "associated"
                }
            }
        ],
        "IsDefault": false,
        "Tags": [
            {
                "Key": "Environment",
                "Value": "Preprod"
            },
            {
                "Key": "Owner",
                "Value": "Build Team"
            }
        ]
    }
}
```

Per ulteriori informazioni, consulta Creare un VPC che utilizza un CIDR del pool IPAM nella Guida per l'utente di Amazon VPC IPAM.

- Per i dettagli sull'API, consulta CreateVpcCommand Reference.AWS CLI

`create-vpn-connection-route`

Il seguente esempio di codice mostra come utilizzare`create-vpn-connection-route`.

AWS CLI

Come creare una route statica per una connessione VPN

Questo esempio crea una route statica per la connessione VPN specificata. Se il comando va a buon fine, non viene restituito alcun output.

Comando:

```
aws ec2 create-vpn-connection-route --vpn-connection-id vpn-40f41529 --destination-cidr-block 11.12.0.0/16
```

- Per i dettagli sull'API, consulta CreateVpnConnectionRoute AWS CLICommand Reference.

`create-vpn-connection`

Il seguente esempio di codice mostra come utilizzare`create-vpn-connection`.

AWS CLI

Esempio 1: come creare una connessione VPN con routing dinamico

L'esempio `create-vpn-connection` seguente crea una connessione VPN tra il gateway privato virtuale e il gateway del cliente specificati e applica i tag alla connessione VPN. L'output include le informazioni di configurazione per il dispositivo del gateway del cliente, in formato XML.

```
aws ec2 create-vpn-connection \
    --type ipsec.1 \
    --customer-gateway-id cgw-001122334455aabbc \
    --vpn-gateway-id vgw-1a1a1a1a1a1a2b2b2 \
    --tag-specification 'ResourceType=vpn-connection,Tags=[{Key=Name,Value=BGP-VPN}]'
```

Output:

```
{
    "VpnConnection": {
        "CustomerGatewayConfiguration": "...configuration information...",
        "CustomerGatewayId": "cgw-001122334455aabbc",
        "Category": "VPN",
        "State": "pending",
```

```
        "VpnConnectionId": "vpn-123123123123abcab",
        "VpnGatewayId": "vgw-1a1a1a1a1a1a2b2b2",
        "Options": {
            "EnableAcceleration": false,
            "StaticRoutesOnly": false,
            "LocalIpv4NetworkCidr": "0.0.0.0/0",
            "RemoteIpv4NetworkCidr": "0.0.0.0/0",
            "TunnelInsideIpVersion": "ipv4",
            "TunnelOptions": [
                {},
                {}
            ]
        },
        "Routes": [],
        "Tags": [
             {
                "Key": "Name",
                "Value": "BGP-VPN"
            }
        ]
    }
}
```

Per ulteriori informazioni, consulta Come funziona la AWS Site-to-Site VPN nella Guida per l'utente della AWS Site-to-Site VPN.

Esempio 2: come creare una connessione VPN con routing statico

L'esempio `create-vpn-connection` seguente crea una connessione VPN tra il gateway privato virtuale e il gateway del cliente specificati. Le opzioni specificano il routing statico. L'output include le informazioni di configurazione per il dispositivo del gateway del cliente, in formato XML.

```
aws ec2 create-vpn-connection \
    --type ipsec.1 \
    --customer-gateway-id cgw-001122334455aabbc \
    --vpn-gateway-id vgw-1a1a1a1a1a1a2b2b2 \
    --options "{\"StaticRoutesOnly\":true}"
```

Output:

```
{
    "VpnConnection": {
```

```
        "CustomerGatewayConfiguration": "..configuration information...",
        "CustomerGatewayId": "cgw-001122334455aabbc",
        "Category": "VPN",
        "State": "pending",
        "VpnConnectionId": "vpn-123123123123abcab",
        "VpnGatewayId": "vgw-1a1a1a1a1a1a2b2b2",
        "Options": {
            "EnableAcceleration": false,
            "StaticRoutesOnly": true,
            "LocalIpv4NetworkCidr": "0.0.0.0/0",
            "RemoteIpv4NetworkCidr": "0.0.0.0/0",
            "TunnelInsideIpVersion": "ipv4",
            "TunnelOptions": [
                {},
                {}
            ]
        },
        "Routes": [],
        "Tags": []
    }
}
```

Per ulteriori informazioni, consulta Come funziona la AWS Site-to-Site VPN nella Guida per l'utente della AWS Site-to-Site VPN.

Esempio 3: come creare una connessione VPN e specificare il CIDR interno e la chiave precondivisa

L'esempio `create-vpn-connection` seguente crea una connessione VPN e specifica l'intervallo CIDR interno degli indirizzi IP e una chiave pre-condivisa personalizzata per ogni tunnel. I valori specificati vengono restituiti nelle informazioni `CustomerGatewayConfiguration`.

```
aws ec2 create-vpn-connection \
    --type ipsec.1 \
    --customer-gateway-id cgw-001122334455aabbc \
    --vpn-gateway-id vgw-1a1a1a1a1a1a2b2b2 \
    --options
 TunnelOptions='[{TunnelInsideCidr=169.254.12.0/30,PreSharedKey=ExamplePreSharedKey1},
{TunnelInsideCidr=169.254.13.0/30,PreSharedKey=ExamplePreSharedKey2}]'
```

Output:

```
{
    "VpnConnection": {
        "CustomerGatewayConfiguration": "..configuration information...",
        "CustomerGatewayId": "cgw-001122334455aabbc",
        "Category": "VPN",
        "State": "pending",
        "VpnConnectionId": "vpn-123123123123abcab",
        "VpnGatewayId": "vgw-1a1a1a1a1a1a2b2b2",
        "Options": {
            "EnableAcceleration": false,
            "StaticRoutesOnly": false,
            "LocalIpv4NetworkCidr": "0.0.0.0/0",
            "RemoteIpv4NetworkCidr": "0.0.0.0/0",
            "TunnelInsideIpVersion": "ipv4",
            "TunnelOptions": [
                {
                    "OutsideIpAddress": "203.0.113.3",
                    "TunnelInsideCidr": "169.254.12.0/30",
                    "PreSharedKey": "ExamplePreSharedKey1"
                },
                {
                    "OutsideIpAddress": "203.0.113.5",
                    "TunnelInsideCidr": "169.254.13.0/30",
                    "PreSharedKey": "ExamplePreSharedKey2"
                }
            ]
        },
        "Routes": [],
        "Tags": []
    }
}
```

Per ulteriori informazioni, consulta Come funziona la AWS Site-to-Site VPN nella Guida per l'utente della AWS Site-to-Site VPN.

Esempio 4: creare una connessione VPN che supporti IPv6 il traffico

L'`create-vpn-connection`esempio seguente crea una connessione VPN che supporta il IPv6 traffico tra il gateway di transito specificato e il gateway del cliente specificato. Le opzioni di tunnel per entrambi i tunnel specificano che AWS deve avviare la negoziazione IKE.

```
aws ec2 create-vpn-connection \
```

```
    --type ipsec.1 \
    --transit-gateway-id tgw-12312312312312312 \
    --customer-gateway-id cgw-001122334455aabbc \
    --options TunnelInsideIpVersion=ipv6,TunnelOptions=[{StartupAction=start},
{StartupAction=start}]
```

Output:

```
{
    "VpnConnection": {
        "CustomerGatewayConfiguration": "..configuration information...",
        "CustomerGatewayId": "cgw-001122334455aabbc",
        "Category": "VPN",
        "State": "pending",
        "VpnConnectionId": "vpn-11111111122222222",
        "TransitGatewayId": "tgw-12312312312312312",
        "Options": {
            "EnableAcceleration": false,
            "StaticRoutesOnly": false,
            "LocalIpv6NetworkCidr": "::/0",
            "RemoteIpv6NetworkCidr": "::/0",
            "TunnelInsideIpVersion": "ipv6",
            "TunnelOptions": [
                {
                    "OutsideIpAddress": "203.0.113.3",
                    "StartupAction": "start"
                },
                {
                    "OutsideIpAddress": "203.0.113.5",
                    "StartupAction": "start"
                }
            ]
        },
        "Routes": [],
        "Tags": []
    }
}
```

Per ulteriori informazioni, consulta Come funziona la AWS Site-to-Site VPN nella Guida per l'utente della AWS Site-to-Site VPN.

- Per i dettagli sull'API, consulta CreateVpnConnection AWS CLICommand Reference.

`create-vpn-gateway`

Il seguente esempio di codice mostra come utilizzare`create-vpn-gateway`.

AWS CLI

Come creare un gateway privato virtuale

Questo esempio crea un gateway privato virtuale.

Comando:

```
aws ec2 create-vpn-gateway --type ipsec.1
```

Output:

```
{
    "VpnGateway": {
        "AmazonSideAsn": 64512,
        "State": "available",
        "Type": "ipsec.1",
        "VpnGatewayId": "vgw-9a4cacf3",
        "VpcAttachments": []
    }
}
```

Come creare un gateway privato virtuale con un ASN specifico per il lato Amazon

Questo esempio crea un gateway privato virtuale e specifica l'Autonomous System Number (ASN) per il lato Amazon della sessione BGP.

Comando:

```
aws ec2 create-vpn-gateway --type ipsec.1 --amazon-side-asn 65001
```

Output:

```
{
    "VpnGateway": {
        "AmazonSideAsn": 65001,
        "State": "available",
        "Type": "ipsec.1",
        "VpnGatewayId": "vgw-9a4cacf3",
```

```
        "VpcAttachments": []
    }
}
```

- Per i dettagli sull'API, consulta CreateVpnGateway AWS CLICommand Reference.

`delete-carrier-gateway`

Il seguente esempio di codice mostra come utilizzare`delete-carrier-gateway`.

AWS CLI

Come eliminare un gateway carrier

L'esempio `delete-carrier-gateway` seguente elimina il gateway carrier specificato.

```
aws ec2 delete-carrier-gateway \
    --carrier-gateway-id cagw-0465cdEXAMPLE1111
```

Output:

```
{
    "CarrierGateway": {
        "CarrierGatewayId": "cagw-0465cdEXAMPLE1111",
        "VpcId": "vpc-0c529aEXAMPLE1111",
        "State": "deleting",
        "OwnerId": "123456789012"
    }
}
```

Per ulteriori informazioni, consulta Gateway carrier nella Guida per l'utente di Amazon Virtual Private Cloud.

- Per i dettagli sull'API, consulta DeleteCarrierGateway AWS CLICommand Reference.

`delete-client-vpn-endpoint`

Il seguente esempio di codice mostra come utilizzare`delete-client-vpn-endpoint`.

AWS CLI

Come eliminare un endpoint VPN Client

L'esempio `delete-client-vpn-endpoint` seguente elimina l'endpoint VPN Client specificato.

```
aws ec2 delete-client-vpn-endpoint \
    --client-vpn-endpoint-id cvpn-endpoint-123456789123abcde
```

Output:

```
{
    "Status": {
        "Code": "deleting"
    }
}
```

Per ulteriori informazioni, consulta Endpoint Client VPN nella Guida per l'amministratore di VPN Client di AWS .

- Per i dettagli sull'API, consulta DeleteClientVpnEndpoint AWS CLICommand Reference.

`delete-client-vpn-route`

Il seguente esempio di codice mostra come utilizzare`delete-client-vpn-route`.

AWS CLI

Come eliminare una route di un endpoint VPN Client

L'esempio `delete-client-vpn-route` seguente elimina la route `0.0.0.0/0` per la sottorete specificata di un endpoint VPN Client.

```
aws ec2 delete-client-vpn-route \
    --client-vpn-endpoint-id cvpn-endpoint-123456789123abcde \
    --destination-cidr-block 0.0.0.0/0 \
    --target-vpc-subnet-id subnet-0123456789abcabca
```

Output:

```
{
    "Status": {
        "Code": "deleting"
    }
}
```

Per ulteriori informazioni, consulta Route nella Guida per l'amministratore di VPN Client di AWS .

- Per i dettagli sull'API, consulta DeleteClientVpnRoute AWS CLICommand Reference.

`delete-coip-cidr`

Il seguente esempio di codice mostra come utilizzare`delete-coip-cidr`.

AWS CLI

Come eliminare un intervallo di indirizzi IP di proprietà del cliente (CoIP)

L'esempio `delete-coip-cidr` seguente elimina l'intervallo specificato di indirizzi CoIP nel pool CoIP specificato.

```
aws ec2 delete-coip-cidr \
    --cidr 14.0.0.0/24 \
    --coip-pool-id ipv4pool-coip-1234567890abcdefg
```

Output:

```
{
    "CoipCidr": {
        "Cidr": "14.0.0.0/24",
        "CoipPoolId": "ipv4pool-coip-1234567890abcdefg",
        "LocalGatewayRouteTableId": "lgw-rtb-abcdefg1234567890"
    }
}
```

Per ulteriori informazioni, consulta Indirizzi IP di proprietà del cliente nella Guida per l'utente di AWS Outposts.

- Per i dettagli sull'API, consulta DeleteCoipCidr AWS CLICommand Reference.

`delete-coip-pool`

Il seguente esempio di codice mostra come utilizzare`delete-coip-pool`.

AWS CLI

Come eliminare un pool di indirizzi IP di proprietà del cliente (CoIP)

L'esempio `delete-coip-pool` seguente elimina un pool CoIP di indirizzi CoIP.

```
aws ec2 delete-coip-pool \
    --coip-pool-id ipv4pool-coip-1234567890abcdefg
```

Output:

```
{
    "CoipPool": {
        "PoolId": "ipv4pool-coip-1234567890abcdefg",
        "LocalGatewayRouteTableId": "lgw-rtb-abcdefg1234567890",
        "PoolArn": "arn:aws:ec2:us-west-2:123456789012:coip-pool/ipv4pool-
coip-1234567890abcdefg"
    }
}
```

Per ulteriori informazioni, consulta Indirizzi IP di proprietà del cliente nella Guida per l'utente di AWS Outposts.

- Per i dettagli sull'API, consulta DeleteCoipPool AWS CLICommand Reference.

`delete-customer-gateway`

Il seguente esempio di codice mostra come utilizzare`delete-customer-gateway`.

AWS CLI

Come eliminare un gateway del cliente

Questo esempio elimina il gateway del cliente specificato. Se il comando va a buon fine, non viene restituito alcun output.

Comando:

```
aws ec2 delete-customer-gateway --customer-gateway-id cgw-0e11f167
```

- Per i dettagli sull'API, consulta DeleteCustomerGateway AWS CLICommand Reference.

`delete-dhcp-options`

Il seguente esempio di codice mostra come utilizzare`delete-dhcp-options`.

AWS CLI

Come eliminare un set di opzioni DHCP

Questo esempio elimina il set di opzioni DHCP specificato. Se il comando va a buon fine, non viene restituito alcun output.

Comando:

```
aws ec2 delete-dhcp-options --dhcp-options-id dopt-d9070ebb
```

- Per i dettagli sull'API, consulta DeleteDhcpOptions AWS CLICommand Reference.

`delete-egress-only-internet-gateway`

Il seguente esempio di codice mostra come utilizzare`delete-egress-only-internet-gateway`.

AWS CLI

Come eliminare un gateway Internet di sola uscita

Questo esempio elimina il gateway Internet di sola uscita specificato.

Comando:

```
aws ec2 delete-egress-only-internet-gateway --egress-only-internet-gateway-
id eigw-01eadbd45ecd7943f
```

Output:

```
{
  "ReturnCode": true
}
```

- Per i dettagli sull'API, consulta DeleteEgressOnlyInternetGateway AWS CLICommand Reference.

`delete-fleets`

Il seguente esempio di codice mostra come utilizzare`delete-fleets`.

AWS CLI

Esempio 1: eliminare un EC2 parco istanze e terminare le istanze associate

L'`delete-fleets`esempio seguente elimina il EC2 parco istanze specificato e termina le istanze On-Demand e le istanze Spot associate.

```
aws ec2 delete-fleets \
    --fleet-ids fleet-12a34b55-67cd-8ef9-ba9b-9208dEXAMPLE \
    --terminate-instances
```

Output:

```
{
    "SuccessfulFleetDeletions": [
        {
            "CurrentFleetState": "deleted_terminating",
            "PreviousFleetState": "active",
            "FleetId": "fleet-12a34b55-67cd-8ef9-ba9b-9208dEXAMPLE"
        }
    ],
    "UnsuccessfulFleetDeletions": []
}
```

Per ulteriori informazioni, consulta Delete an EC2 Fleet nella Amazon Elastic Compute Cloud User Guide for Linux Instances.

Esempio 2: eliminare una EC2 flotta senza terminare le istanze associate

L'`delete-fleets`esempio seguente elimina il EC2 parco istanze specificato senza terminare le istanze On-Demand e le istanze Spot associate.

```
aws ec2 delete-fleets \
    --fleet-ids fleet-12a34b55-67cd-8ef9-ba9b-9208dEXAMPLE \
    --no-terminate-instances
```

Output:

```
{
    "SuccessfulFleetDeletions": [
```

```
        {
            "CurrentFleetState": "deleted_running",
            "PreviousFleetState": "active",
            "FleetId": "fleet-12a34b55-67cd-8ef9-ba9b-9208dEXAMPLE"
        }
    ],
    "UnsuccessfulFleetDeletions": []
}
```

Per ulteriori informazioni, consulta Delete an EC2 Fleet nella Amazon Elastic Compute Cloud User Guide for Linux Instances.

- Per i dettagli sull'API, consulta AWS CLI Command DeleteFleetsReference.

`delete-flow-logs`

Il seguente esempio di codice mostra come utilizzare`delete-flow-logs`.

AWS CLI

Come eliminare un log di flusso

L'esempio `delete-flow-logs` seguente elimina il log di flusso specificato.

```
aws ec2 delete-flow-logs --flow-log-id fl-11223344556677889
```

Output:

```
{
    "Unsuccessful": []
}
```

- Per i dettagli sull'API, consulta DeleteFlowLogs AWS CLICommand Reference.

`delete-fpga-image`

Il seguente esempio di codice mostra come utilizzare`delete-fpga-image`.

AWS CLI

Come eliminare un'immagine FPGA di Amazon (AFI)

Questo esempio elimina l'AFI specificata.

Comando:

```
aws ec2 delete-fpga-image --fpga-image-id afi-06b12350a123fbabc
```

Output:

```
{
  "Return": true
}
```

- Per i dettagli sull'API, consulta DeleteFpgaImage AWS CLICommand Reference.

`delete-instance-connect-endpoint`

Il seguente esempio di codice mostra come utilizzare`delete-instance-connect-endpoint`.

AWS CLI

Per eliminare un endpoint EC2 Instance Connect

L'`delete-instance-connect-endpoint`esempio seguente elimina l'endpoint EC2 Instance Connect specificato.

```
aws ec2 delete-instance-connect-endpoint \
    --instance-connect-endpoint-id eice-03f5e49b83924bbc7
```

Output:

```
{
    "InstanceConnectEndpoint": {
        "OwnerId": "111111111111",
        "InstanceConnectEndpointId": "eice-0123456789example",
        "InstanceConnectEndpointArn": "arn:aws:ec2:us-east-1:111111111111:instance-
connect-endpoint/eice-0123456789example",
        "State": "delete-in-progress",
        "StateMessage": "",
        "NetworkInterfaceIds": [],
        "VpcId": "vpc-0123abcd",
```

```
        "AvailabilityZone": "us-east-1d",
        "CreatedAt": "2023-02-07T12:05:37+00:00",
        "SubnetId": "subnet-0123abcd"
    }
}
```

Per ulteriori informazioni, consulta Remove EC2 Instance Connect Endpoint nella Amazon EC2 User Guide.

- Per i dettagli sull'API, consulta DeleteInstanceConnectEndpoint AWS CLICommand Reference.

`delete-instance-event-window`

Il seguente esempio di codice mostra come utilizzare`delete-instance-event-window`.

AWS CLI

Esempio 1: come eliminare una finestra di eventi

L'esempio `delete-instance-event-window` seguente elimina una finestra di eventi.

```
aws ec2 delete-instance-event-window \
    --region us-east-1 \
    --instance-event-window-id iew-0abcdef1234567890
```

Output:

```
{
    "InstanceEventWindowState": {
        "InstanceEventWindowId": "iew-0abcdef1234567890",
        "State": "deleting"
    }
}
```

Per i vincoli relativi alla finestra degli eventi, consulta Considerazioni nella sezione Eventi pianificati della Amazon EC2 User Guide.

Esempio 2: come forzare l'eliminazione di una finestra di eventi

L'esempio `delete-instance-event-window` seguente forza l'eliminazione di una finestra di eventi se la finestra di eventi è attualmente associata a destinazioni.

```
aws ec2 delete-instance-event-window \
    --region us-east-1 \
    --instance-event-window-id iew-0abcdef1234567890 \
    --force-delete
```

Output:

```
{
    "InstanceEventWindowState": {
        "InstanceEventWindowId": "iew-0abcdef1234567890",
        "State": "deleting"
    }
}
```

Per i vincoli relativi alla finestra degli eventi, consulta Considerazioni nella sezione Eventi pianificati della Amazon EC2 User Guide.

- Per i dettagli sull'API, consulta Command DeleteInstanceEventWindowReference AWS CLI .

`delete-internet-gateway`

Il seguente esempio di codice mostra come utilizzare`delete-internet-gateway`.

AWS CLI

Come eliminare un gateway Internet

L'esempio `delete-internet-gateway` seguente elimina il gateway Internet specificato.

```
aws ec2 delete-internet-gateway \
    --internet-gateway-id igw-0d0fb496b3EXAMPLE
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Gateway Internet nella Guida per l'utente di Amazon VPC.

- Per i dettagli sull'API, consulta DeleteInternetGateway AWS CLICommand Reference.

`delete-ipam-pool`

Il seguente esempio di codice mostra come utilizzare`delete-ipam-pool`.

AWS CLI

Come eliminare un pool IPAM

In questo esempio, l'utente è un amministratore delegato IPAM che desidera eliminare un pool IPAM che non è più necessario, ma al pool è stato assegnato un CIDR. Non è possibile eliminare un pool se ne è stato CIDRs fornito il provisioning a meno che non si utilizzi l'`--cascade`opzione, quindi la `--cascade` userete.

Per completare questa richiesta:

Avrai bisogno dell'ID del pool IPAM con cui puoi ottenere describe-ipam-pools. `--region` Deve essere la regione di origine IPAM.

L'`delete-ipam-pool`esempio seguente elimina un pool IPAM nel tuo account. AWS

```
aws ec2 delete-ipam-pool \
    --ipam-pool-id ipam-pool-050c886a3ca41cd5b \
    --cascade \
    --region us-east-1
```

Output:

```
{
    "IpamPool": {
        "OwnerId": "320805250157",
        "IpamPoolId": "ipam-pool-050c886a3ca41cd5b",
        "IpamPoolArn": "arn:aws:ec2::320805250157:ipam-pool/ipam-
pool-050c886a3ca41cd5b",
        "IpamScopeArn": "arn:aws:ec2::320805250157:ipam-scope/ipam-
scope-0a158dde35c51107b",
        "IpamScopeType": "private",
        "IpamArn": "arn:aws:ec2::320805250157:ipam/ipam-005f921c17ebd5107",
        "IpamRegion": "us-east-1",
        "Locale": "None",
        "PoolDepth": 1,
        "State": "delete-in-progress",
        "Description": "example",
        "AutoImport": false,
        "AddressFamily": "ipv4",
        "AllocationMinNetmaskLength": 0,
        "AllocationMaxNetmaskLength": 32
    }
```

```
}
```

Per ulteriori informazioni, consulta Eliminare un pool nella Guida per l'utente IPAM di Amazon VPC.

- Per i dettagli sull'API, consulta AWS CLI Command DeleteIpamPoolReference.

`delete-ipam-resource-discovery`

Il seguente esempio di codice mostra come utilizzare`delete-ipam-resource-discovery`.

AWS CLI

Come eliminare un'individuazione delle risorse

In questo esempio, un amministratore delegato IPAM desidera eliminare un rilevamento di risorse non predefinito creato per la condivisione con un altro amministratore IPAM durante il processo di integrazione IPAM con account esterni all'organizzazione.

Per completare questa richiesta:

`--region` deve essere la Regione in cui è stato creato il rilevamento di risorse. Non è possibile eliminare un rilevamento di risorse predefinito se `"IsDefault": true`. Un rilevamento di risorse predefinito viene creato automaticamente quando si crea un'IPAM. Per eliminare un rilevamento di risorse predefinito, è necessario eliminare l'IPAM.

L'esempio `delete-ipam-resource-discovery` seguente elimina un rilevamento di risorse.

```
aws ec2 delete-ipam-resource-discovery \
    --ipam-resource-discovery-id ipam-res-disco-0e39761475298ee0f \
    --region us-east-1
```

Output:

```
{
    "IpamResourceDiscovery": {
        "OwnerId": "149977607591",
        "IpamResourceDiscoveryId": "ipam-res-disco-0e39761475298ee0f",
        "IpamResourceDiscoveryArn": "arn:aws:ec2::149977607591:ipam-resource-
discovery/ipam-res-disco-0e39761475298ee0f",
        "IpamResourceDiscoveryRegion": "us-east-1",
        "OperatingRegions": [
```

```
            {
                "RegionName": "us-east-1"
            }
        ],
        "IsDefault": false,
        "State": "delete-in-progress"
    }
}
```

Per ulteriori informazioni sui rilevamenti delle risorse, consulta Lavorare con il rilevamento delle risorse nella Guida per l'utente IPAM di Amazon VPC.

- Per i dettagli sull'API, consulta DeleteIpamResourceDiscovery AWS CLICommand Reference.

`delete-ipam-scope`

Il seguente esempio di codice mostra come utilizzare`delete-ipam-scope`.

AWS CLI

Come eliminare un ambito IPAM

L'esempio `delete-ipam-scope` seguente elimina un'IPAM.

```
aws ec2 delete-ipam-scope \
    --ipam-scope-id ipam-scope-01c1ebab2b63bd7e4
```

Output:

```
{
    "IpamScope": {
        "OwnerId": "123456789012",
        "IpamScopeId": "ipam-scope-01c1ebab2b63bd7e4",
        "IpamScopeArn": "arn:aws:ec2::123456789012:ipam-scope/ipam-
scope-01c1ebab2b63bd7e4",
        "IpamArn": "arn:aws:ec2::123456789012:ipam/ipam-08440e7a3acde3908",
        "IpamRegion": "us-east-1",
        "IpamScopeType": "private",
        "IsDefault": false,
        "Description": "Example description",
        "PoolCount": 0,
        "State": "delete-in-progress"
    }
```

```
}
```

Per ulteriori informazioni, consulta Eliminare un ambito nella Guida per l'utente IPAM di Amazon VPC.

- Per i dettagli sull'API, consulta DeleteIpamScope AWS CLICommand Reference.

`delete-ipam`

Il seguente esempio di codice mostra come utilizzare`delete-ipam`.

AWS CLI

Come eliminare un IPAM

L'esempio `delete-ipam` seguente elimina un'IPAM.

```
aws ec2 delete-ipam \
    --ipam-id ipam-036486dfa6af58ee0
```

Output:

```
{
    "Ipam": {
        "OwnerId": "123456789012",
        "IpamId": "ipam-036486dfa6af58ee0",
        "IpamArn": "arn:aws:ec2::123456789012:ipam/ipam-036486dfa6af58ee0",
        "IpamRegion": "us-east-1",
        "PublicDefaultScopeId": "ipam-scope-071b8042b0195c183",
        "PrivateDefaultScopeId": "ipam-scope-0807405dece705a30",
        "ScopeCount": 2,
        "OperatingRegions": [
            {
                "RegionName": "us-east-1"
            },
            {
                "RegionName": "us-east-2"
            },
            {
                "RegionName": "us-west-1"
            }
        ],
        "State": "delete-in-progress"
```

```
    }
}
```

Per ulteriori informazioni, consulta Eliminare un IPAM nella Guida per l'utente IPAM di Amazon VPC.

- Per i dettagli sull'API, consulta DeleteIpam AWS CLICommand Reference.

`delete-key-pair`

Il seguente esempio di codice mostra come utilizzare`delete-key-pair`.

AWS CLI

Come eliminare una coppia di chiavi

L'esempio `delete-key-pair` seguente elimina la coppia di chiavi specificata.

```
aws ec2 delete-key-pair \
    --key-name my-key-pair
```

Output:

```
{
    "Return": true,
    "KeyPairId": "key-03c8d3aceb53b507"
}
```

Per ulteriori informazioni, consulta Create and delete key pairs nella Guida per l'utente dell'Interfaccia a riga di comando AWS .

- Per i dettagli sull'API, consulta DeleteKeyPair AWS CLICommand Reference.

`delete-launch-template-versions`

Il seguente esempio di codice mostra come utilizzare`delete-launch-template-versions`.

AWS CLI

Come eliminare una versione del modello di avvio

Questo esempio elimina la versione del modello di avvio specificato.

Comando:

```
aws ec2 delete-launch-template-versions --launch-template-id lt-0abcd290751193123 --
versions 1
```

Output:

```
{
  "UnsuccessfullyDeletedLaunchTemplateVersions": [],
  "SuccessfullyDeletedLaunchTemplateVersions": [
      {
          "LaunchTemplateName": "TestVersion",
          "VersionNumber": 1,
          "LaunchTemplateId": "lt-0abcd290751193123"
      }
  ]
}
```

- Per i dettagli sull'API, consulta DeleteLaunchTemplateVersions AWS CLICommand Reference.

`delete-launch-template`

Il seguente esempio di codice mostra come utilizzare`delete-launch-template`.

AWS CLI

Come eliminare un modello di avvio

In questo esempio viene eliminato il modello di avvio specificato.

Comando:

```
aws ec2 delete-launch-template --launch-template-id lt-0abcd290751193123
```

Output:

```
{
  "LaunchTemplate": {
      "LatestVersionNumber": 2,
      "LaunchTemplateId": "lt-0abcd290751193123",
      "LaunchTemplateName": "TestTemplate",
```

```
      "DefaultVersionNumber": 2,
      "CreatedBy": "arn:aws:iam::123456789012:root",
      "CreateTime": "2017-11-23T16:46:25.000Z"
  }
}
```

- Per i dettagli sull'API, consulta DeleteLaunchTemplate AWS CLICommand Reference.

`delete-local-gateway-route-table-virtual-interface-group-association`

Il seguente esempio di codice mostra come utilizzare`delete-local-gateway-route-table-virtual-interface-group-association`.

AWS CLI

Per dissociare una tabella di routing del gateway locale da un gruppo di interfacce virtuali () VIFs

L'esempio `delete-local-gateway-route-table-virtual-interface-group-association` seguente elimina l'associazione tra la tabella di routing del gateway locale specificata e il gruppo VIF.

```
aws ec2 delete-local-gateway-route-table-virtual-interface-group-association \
    --local-gateway-route-table-virtual-interface-group-association-id lgw-vif-grp-assoc-exampleid12345678
```

Output:

```
{
    "LocalGatewayRouteTableVirtualInterfaceGroupAssociation": {
        "LocalGatewayRouteTableVirtualInterfaceGroupAssociationId": "lgw-vif-grp-assoc-exampleid12345678",
        "LocalGatewayVirtualInterfaceGroupId": "lgw-vif-grp-exampleid0123abcd",
        "LocalGatewayId": "lgw-exampleid11223344",
        "LocalGatewayRouteTableId": "lgw-rtb-exampleidabcd1234",
        "LocalGatewayRouteTableArn": "arn:aws:ec2:us-west-2:111122223333:local-gateway-route-table/lgw-rtb-exampleidabcd1234",
        "OwnerId": "111122223333",
        "State": "disassociating",
        "Tags": []
    }
}
```

Per ulteriori informazioni, consulta Associazioni di gruppi VIF nella Guida per l'utente di AWS Outposts.

- Per i dettagli sull'API, vedere DeleteLocalGatewayRouteTableVirtualInterfaceGroupAssociationin AWS CLI Command Reference.

`delete-local-gateway-route-table-vpc-association`

Il seguente esempio di codice mostra come utilizzare`delete-local-gateway-route-table-vpc-association`.

AWS CLI

Come dissociare una tabella di routing del gateway locale da un VPC

L'esempio `delete-local-gateway-route-table-vpc-association` seguente elimina l'associazione tra la tabella di routing del gateway locale e il VPC.

```
aws ec2 delete-local-gateway-route-table-vpc-association \
    --local-gateway-route-table-vpc-association-id vpc-example0123456789
```

Output:

```
{
    "LocalGatewayRouteTableVpcAssociation": {
        "LocalGatewayRouteTableVpcAssociationId": "lgw-vpc-assoc-abcd1234wxyz56789",
        "LocalGatewayRouteTableId": "lgw-rtb-abcdefg1234567890",
        "LocalGatewayRouteTableArn": "arn:aws:ec2:us-west-2:555555555555:local-
gateway-route-table/lgw-rtb-abcdefg1234567890",
        "LocalGatewayId": "lgw-exampleid01234567",
        "VpcId": "vpc-example0123456789",
        "OwnerId": "555555555555",
        "State": "disassociating"
    }
}
```

Per ulteriori informazioni, consulta Associazioni VPC nella Guida per l'utente di AWS Outposts.

- Per i dettagli sull'API, consulta DeleteLocalGatewayRouteTableVpcAssociation AWS CLICommand Reference.

`delete-local-gateway-route-table`

Il seguente esempio di codice mostra come utilizzare`delete-local-gateway-route-table`.

AWS CLI

Come eliminare una tabella di routing del gateway locale

L'esempio `delete-local-gateway-route-table` seguente crea una tabella di routing del gateway locale con la modalità di routing VPC diretta.

```
aws ec2 delete-local-gateway-route-table \
    --local-gateway-route-table-id lgw-rtb-abcdefg1234567890
```

Output:

```
{
    "LocalGatewayRouteTable": {
        "LocalGatewayRouteTableId": "lgw-rtb-abcdefg1234567890",
        "LocalGatewayRouteTableArn": "arn:aws:ec2:us-west-2:111122223333:local-
gateway-route-table/lgw-rtb-abcdefg1234567890",
        "LocalGatewayId": "lgw-1a2b3c4d5e6f7g8h9",
        "OutpostArn": "arn:aws:outposts:us-west-2:111122223333:outpost/
op-021345abcdef67890",
        "OwnerId": "111122223333",
        "State": "deleting",
        "Tags": [],
        "Mode": "direct-vpc-routing"
    }
}
```

Per ulteriori informazioni, consulta Tabelle di routing del gateway locale nella Guida per l'utente di AWS Outposts.

- Per i dettagli sull'API, consulta DeleteLocalGatewayRouteTable AWS CLICommand Reference.

`delete-local-gateway-route`

Il seguente esempio di codice mostra come utilizzare`delete-local-gateway-route`.

AWS CLI

Come eliminare una route da una tabella di routing del gateway locale

L'esempio `delete-local-gateway-route` seguente elimina la route specificata dalla tabella di routing del gateway locale specificata.

```
aws ec2 delete-local-gateway-route \
    --destination-cidr-block 0.0.0.0/0 \
    --local-gateway-route-table-id lgw-rtb-059615ef7dEXAMPLE
```

Output:

```
{
    "Route": {
        "DestinationCidrBlock": "0.0.0.0/0",
        "LocalGatewayVirtualInterfaceGroupId": "lgw-vif-grp-07145b276bEXAMPLE",
        "Type": "static",
        "State": "deleted",
        "LocalGatewayRouteTableId": "lgw-rtb-059615ef7EXAMPLE"
    }
}
```

- Per i dettagli sull'API, consulta DeleteLocalGatewayRoute AWS CLICommand Reference.

`delete-managed-prefix-list`

Il seguente esempio di codice mostra come utilizzare`delete-managed-prefix-list`.

AWS CLI

Come eliminare un elenco di prefissi

L'esempio `delete-managed-prefix-list` seguente elimina l'elenco di prefissi specificato.

```
aws ec2 delete-managed-prefix-list \
    --prefix-list-id pl-0123456abcabcabc1
```

Output:

```
{
    "PrefixList": {
        "PrefixListId": "pl-0123456abcabcabc1",
        "AddressFamily": "IPv4",
```

```
        "State": "delete-in-progress",
        "PrefixListArn": "arn:aws:ec2:us-west-2:123456789012:prefix-list/
pl-0123456abcabcabc1",
        "PrefixListName": "test",
        "MaxEntries": 10,
        "Version": 1,
        "OwnerId": "123456789012"
    }
}
```

Per ulteriori informazioni, consulta Elenchi di prefissi gestiti nella Guida per l'utente di Amazon VPC.

- Per i dettagli sull'API, consulta DeleteManagedPrefixList AWS CLICommand Reference.

`delete-nat-gateway`

Il seguente esempio di codice mostra come utilizzare`delete-nat-gateway`.

AWS CLI

Come eliminare un gateway NAT

Questo esempio elimina il gateway NAT `nat-04ae55e711cec5680`.

Comando:

```
aws ec2 delete-nat-gateway --nat-gateway-id nat-04ae55e711cec5680
```

Output:

```
{
   "NatGatewayId": "nat-04ae55e711cec5680"
}
```

- Per i dettagli sull'API, consulta DeleteNatGateway AWS CLICommand Reference.

`delete-network-acl-entry`

Il seguente esempio di codice mostra come utilizzare`delete-network-acl-entry`.

AWS CLI

Come eliminare una voce della lista di controllo degli accessi alla rete (ACL)

Questo esempio elimina la regola di ingresso numero 100 dalla lista di controllo degli accessi alla rete (ACL) specificata. Se il comando va a buon fine, non viene restituito alcun output.

Comando:

```
aws ec2 delete-network-acl-entry --network-acl-id acl-5fb85d36 --ingress --rule-number 100
```

- Per i dettagli sull'API, consulta DeleteNetworkAclEntry AWS CLICommand Reference.

`delete-network-acl`

Il seguente esempio di codice mostra come utilizzare`delete-network-acl`.

AWS CLI

Come eliminare una lista di controllo degli accessi alla rete (ACL)

Questo esempio elimina la lista di controllo degli accessi alla rete (ACL) specificata. Se il comando va a buon fine, non viene restituito alcun output.

Comando:

```
aws ec2 delete-network-acl --network-acl-id acl-5fb85d36
```

- Per i dettagli sull'API, consulta DeleteNetworkAcl AWS CLICommand Reference.

`delete-network-insights-access-scope-analysis`

Il seguente esempio di codice mostra come utilizzare`delete-network-insights-access-scope-analysis`.

AWS CLI

Come eliminare un'analisi di ambito di accesso alla rete

L'esempio `delete-network-insights-access-scope-analysis` seguente elimina l'analisi di ambito di accesso alla rete specificata.

```
aws ec2 delete-network-insights-access-scope-analysis \
    --network-insights-access-scope-analysis-id nisa-01234567891abcdef
```

Output:

```
{
    "NetworkInsightsAccessScopeAnalysisId": "nisa-01234567891abcdef
}
```

Per ulteriori informazioni, consulta Guida introduttiva a Network Access Analyzer utilizzando la AWS CLI nella Guida Network Access Analyzer.

- Per i dettagli sull'API, consulta Command DeleteNetworkInsightsAccessScopeAnalysisReference AWS CLI .

`delete-network-insights-access-scope`

Il seguente esempio di codice mostra come utilizzare`delete-network-insights-access-scope`.

AWS CLI

Come eliminare un ambito di accesso alla rete

L'esempio `delete-network-insights-access-scope` seguente elimina l'analisi di ambito di accesso alla rete specificata.

```
aws ec2 delete-network-insights-access-scope \
    --network-insights-access-scope-id nis-123456789abc01234
```

Output:

```
{
    "NetworkInsightsAccessScopeId": "nis-123456789abc01234"
}
```

Per ulteriori informazioni, consulta Guida introduttiva a Network Access Analyzer utilizzando la AWS CLI nella Guida Network Access Analyzer.

- Per i dettagli sull'API, consulta Command DeleteNetworkInsightsAccessScopeReference AWS CLI .

`delete-network-insights-analysis`

Il seguente esempio di codice mostra come utilizzare`delete-network-insights-analysis`.

AWS CLI

Come eliminare un'analisi del percorso

L'esempio `delete-network-insights-analysis` seguente elimina l'analisi specificata.

```
aws ec2 delete-network-insights-analysis \
    --network-insights-analysis-id nia-02207aa13eb480c7a
```

Output:

```
{
    "NetworkInsightsAnalysisId": "nia-02207aa13eb480c7a"
}
```

Per ulteriori informazioni, consulta Guida introduttiva all'uso della AWS CLI nella Reachability Analyzer Guide.

- Per i dettagli sull'API, consulta Command DeleteNetworkInsightsAnalysisReference AWS CLI .

`delete-network-insights-path`

Il seguente esempio di codice mostra come utilizzare`delete-network-insights-path`.

AWS CLI

Come eliminare un percorso

L'esempio `delete-network-insights-path` seguente elimina il percorso specificato. Prima di poter eliminare un percorso, è necessario eliminare tutte le relative analisi utilizzando il comando `delete-network-insights-analysis`.

```
aws ec2 delete-network-insights-path \
    --network-insights-path-id nip-0b26f224f1d131fa8
```

Output:

```
{
```

```
    "NetworkInsightsPathId": "nip-0b26f224f1d131fa8"
}
```

Per ulteriori informazioni, consulta Guida introduttiva all'uso della AWS CLI nella Reachability Analyzer Guide.

- Per i dettagli sull'API, consulta Command DeleteNetworkInsightsPathReference AWS CLI .

`delete-network-interface-permission`

Il seguente esempio di codice mostra come utilizzare`delete-network-interface-permission`.

AWS CLI

Come eliminare l'autorizzazione di un'interfaccia di rete

Questo esempio elimina l'autorizzazione dell'interfaccia di rete specificata.

Comando:

```
aws ec2 delete-network-interface-permission --network-interface-permission-id eni-perm-06fd19020ede149ea
```

Output:

```
{
  "Return": true
}
```

- Per i dettagli sull'API, consulta DeleteNetworkInterfacePermission AWS CLICommand Reference.

`delete-network-interface`

Il seguente esempio di codice mostra come utilizzare`delete-network-interface`.

AWS CLI

Come eliminare un'interfaccia di rete

Questo esempio elimina l'interfaccia di rete specificata. Se il comando va a buon fine, non viene restituito alcun output.

Comando:

```
aws ec2 delete-network-interface --network-interface-id eni-e5aa89a3
```

- Per i dettagli sull'API, consulta DeleteNetworkInterface AWS CLICommand Reference.

`delete-placement-group`

Il seguente esempio di codice mostra come utilizzare`delete-placement-group`.

AWS CLI

Come eliminare un gruppo di posizionamento

Questo comando di esempio elimina il gruppo di posizionamento specificato.

Comando:

```
aws ec2 delete-placement-group --group-name my-cluster
```

- Per i dettagli sull'API, consulta DeletePlacementGroup AWS CLICommand Reference.

`delete-queued-reserved-instances`

Il seguente esempio di codice mostra come utilizzare`delete-queued-reserved-instances`.

AWS CLI

Come eliminare un acquisto in coda

L'esempio `delete-queued-reserved-instances` seguente elimina l'istanza riservata specificata, che era in coda per l'acquisto.

```
aws ec2 delete-queued-reserved-instances \
    --reserved-instances-ids af9f760e-6f91-4559-85f7-4980eexample
```

Output:

```
{
    "SuccessfulQueuedPurchaseDeletions": [
        {
```

```
            "ReservedInstancesId": "af9f760e-6f91-4559-85f7-4980eexample"
        }
    ],
    "FailedQueuedPurchaseDeletions": []
}
```

- Per i dettagli sull'API, consulta DeleteQueuedReservedInstances AWS CLICommand Reference.

`delete-route-table`

Il seguente esempio di codice mostra come utilizzare`delete-route-table`.

AWS CLI

Come eliminare una tabella di routing

Questo esempio elimina la tabella di routing specificata. Se il comando va a buon fine, non viene restituito alcun output.

Comando:

```
aws ec2 delete-route-table --route-table-id rtb-22574640
```

- Per i dettagli sull'API, consulta DeleteRouteTable AWS CLICommand Reference.

`delete-route`

Il seguente esempio di codice mostra come utilizzare`delete-route`.

AWS CLI

Come eliminare una route

Questo esempio elimina la route specificata dalla tabella di routing specificata. Se il comando va a buon fine, non viene restituito alcun output.

Comando:

```
aws ec2 delete-route --route-table-id rtb-22574640 --destination-cidr-
block 0.0.0.0/0
```

- Per i dettagli sull'API, consulta DeleteRoute AWS CLICommand Reference.

`delete-security-group`

Il seguente esempio di codice mostra come utilizzare`delete-security-group`.

AWS CLI

[EC2-Classic] Per eliminare un gruppo di sicurezza

In questo esempio viene eliminato il gruppo di sicurezza denominato `MySecurityGroup`. Se il comando va a buon fine, non viene restituito alcun output.

Comando:

```
aws ec2 delete-security-group --group-name MySecurityGroup
```

[EC2-VPC] Per eliminare un gruppo di sicurezza

In questo esempio viene eliminato il gruppo di sicurezza con ID `sg-903004f8`. Nota che non puoi fare riferimento a un gruppo di sicurezza per EC2 -VPC per nome. Se il comando va a buon fine, non viene restituito alcun output.

Comando:

```
aws ec2 delete-security-group --group-id sg-903004f8
```

Per ulteriori informazioni, consulta Utilizzo dei gruppi di sicurezza nella Guida per l'utente dell'Interfaccia a riga di comando AWS .

- Per i dettagli sull'API, consulta AWS CLI Command DeleteSecurityGroupReference.

`delete-snapshot`

Il seguente esempio di codice mostra come utilizzare`delete-snapshot`.

AWS CLI

Come eliminare uno snapshot

Questo comando di esempio elimina uno snapshot con ID `snap-1234567890abcdef0`. Se il comando va a buon fine, non viene restituito alcun output.

Comando:

```
aws ec2 delete-snapshot --snapshot-id snap-1234567890abcdef0
```

- Per i dettagli sull'API, consulta DeleteSnapshot AWS CLICommand Reference.

`delete-spot-datafeed-subscription`

Il seguente esempio di codice mostra come utilizzare`delete-spot-datafeed-subscription`.

AWS CLI

Come annullare una sottoscrizione al feed di dati di un'istanza spot

Questo comando di esempio elimina una sottoscrizione al feed di dati spot per l'account. Se il comando va a buon fine, non viene restituito alcun output.

Comando:

```
aws ec2 delete-spot-datafeed-subscription
```

- Per i dettagli sull'API, consulta DeleteSpotDatafeedSubscription AWS CLICommand Reference.

`delete-subnet-cidr-reservation`

Il seguente esempio di codice mostra come utilizzare`delete-subnet-cidr-reservation`.

AWS CLI

Come eliminare una prenotazione CIDR della sottorete

L'esempio `delete-subnet-cidr-reservation` seguente elimina la prenotazione CIDR della sottorete specificata.

```
aws ec2 delete-subnet-cidr-reservation \
    --subnet-cidr-reservation-id scr-044f977c4eEXAMPLE
```

Output:

```
{
    "DeletedSubnetCidrReservation": {
        "SubnetCidrReservationId": "scr-044f977c4eEXAMPLE",
        "SubnetId": "subnet-03c51e2e6cEXAMPLE",
        "Cidr": "10.1.0.16/28",
        "ReservationType": "prefix",
        "OwnerId": "123456789012"
    }
}
```

Per ulteriori informazioni, consulta Prenotazioni della CIDR per la sottorete nella Guida per l'utente di Amazon VPC.

- Per i dettagli sull'API, consulta DeleteSubnetCidrReservation AWS CLICommand Reference.

`delete-subnet`

Il seguente esempio di codice mostra come utilizzare`delete-subnet`.

AWS CLI

Come eliminare una sottorete

Questo esempio elimina la sottorete specificata. Se il comando va a buon fine, non viene restituito alcun output.

Comando:

```
aws ec2 delete-subnet --subnet-id subnet-9d4a7b6c
```

- Per i dettagli sull'API, consulta DeleteSubnet AWS CLICommand Reference.

`delete-tags`

Il seguente esempio di codice mostra come utilizzare`delete-tags`.

AWS CLI

Esempio 1: come eliminare un tag da una risorsa

L'esempio `delete-tags` seguente elimina il tag `Stack=Test` dall'immagine specificata. Quando vengono specificati sia un valore che un nome di chiave, il tag viene eliminato solo se il valore del tag corrisponde al valore specificato.

```
aws ec2 delete-tags \
    --resources ami-1234567890abcdef0 \
    --tags Key=Stack,Value=Test
```

È facoltativo specificare il valore per un tag. L'esempio `delete-tags` seguente elimina il tag con il nome della chiave `purpose` dall'istanza specificata, indipendentemente dal valore del tag.

```
aws ec2 delete-tags \
    --resources i-1234567890abcdef0 \
    --tags Key=purpose
```

Se come valore del tag viene specificata una stringa vuota, il tag viene eliminato solo se il relativo valore corrisponde alla stringa vuota. L'esempio `delete-tags` seguente specifica la stringa vuota come valore del tag da eliminare.

```
aws ec2 delete-tags \
    --resources i-1234567890abcdef0 \
    --tags Key=Name,Value=
```

Esempio 2: come eliminare un tag da più risorse

L'esempio `delete-tags` seguente elimina il tag ``Purpose=test`` sia da un'istanza che da un'AMI. Come mostrato nell'esempio precedente, è possibile omettere il valore del tag dal comando.

```
aws ec2 delete-tags \
    --resources i-1234567890abcdef0 ami-1234567890abcdef0 \
    --tags Key=Purpose
```

- Per i dettagli sull'API, consulta DeleteTags AWS CLICommand Reference.

`delete-traffic-mirror-filter-rule`

Il seguente esempio di codice mostra come utilizzare`delete-traffic-mirror-filter-rule`.

AWS CLI

Come eliminare una regola di filtro di mirroring del traffico

L'esempio `delete-traffic-mirror-filter-rule` seguente elimina la regola di filtro di mirroring del traffico specificata.

```
aws ec2 delete-traffic-mirror-filter-rule \
    --traffic-mirror-filter-rule-id tmfr-081f71283bEXAMPLE
```

Output:

```
{
    "TrafficMirrorFilterRuleId": "tmfr-081f71283bEXAMPLE"
}
```

Per ulteriori informazioni, consulta Modificare le regole di filtro del mirroring del traffico nella Guida di Mirroring del traffico AWS .

- Per i dettagli sull'API, consulta DeleteTrafficMirrorFilterRule AWS CLICommand Reference.

`delete-traffic-mirror-filter`

Il seguente esempio di codice mostra come utilizzare`delete-traffic-mirror-filter`.

AWS CLI

Come eliminare un filtro del mirroring del traffico

L'esempio `delete-traffic-mirror-filter` seguente elimina il filtro di mirroring del traffico specificato.

```
aws ec2 delete-traffic-mirror-filter \
    --traffic-mirror-filter-id tmf-0be0b25fcdEXAMPLE
```

Output:

```
{
    "TrafficMirrorFilterId": "tmf-0be0b25fcdEXAMPLE"
}
```

Per ulteriori informazioni, consulta Eliminare un filtro del mirroring del traffico nella Guida di Mirroring del traffico AWS .

- Per i dettagli sull'API, consulta DeleteTrafficMirrorFilter AWS CLICommand Reference.

`delete-traffic-mirror-session`

Il seguente esempio di codice mostra come utilizzare`delete-traffic-mirror-session`.

AWS CLI

Come eliminare una sessione di mirroring del traffico

L'esempio `delete-traffic-mirror-session` seguente elimina la sessione di mirroring del traffico specificata.

```
aws ec2 delete-traffic-mirror-session \
    --traffic-mirror-session-id tms-0af3141ce5EXAMPLE
```

Output:

```
{
    "TrafficMirrorSessionId": "tms-0af3141ce5EXAMPLE"
}
```

Per ulteriori informazioni, consulta Eliminare una sessione di mirroring del traffico nella Guida di Mirroring del traffico AWS .

- Per i dettagli sull'API, consulta DeleteTrafficMirrorSession AWS CLICommand Reference.

`delete-traffic-mirror-target`

Il seguente esempio di codice mostra come utilizzare`delete-traffic-mirror-target`.

AWS CLI

Come eliminare una destinazione di mirroring del traffico

L'esempio `delete-traffic-mirror-target` seguente elimina la destinazione di mirroring del traffico specificata.

```
aws ec2 delete-traffic-mirror-target \
```

```
    --traffic-mirror-target-id tmt-060f48ce9EXAMPLE
```

Output:

```
{
    "TrafficMirrorTargetId": "tmt-060f48ce9EXAMPLE"
}
```

Per ulteriori informazioni, consulta Eliminare una destinazione del mirroring del traffico nella Guida di Mirroring del traffico AWS .

- Per i dettagli sull'API, consulta DeleteTrafficMirrorTarget AWS CLICommand Reference.

`delete-transit-gateway-connect-peer`

Il seguente esempio di codice mostra come utilizzare`delete-transit-gateway-connect-peer`.

AWS CLI

Come eliminare un peer di Transit Gateway Connect

L'esempio `delete-transit-gateway-connect-peer` seguente elimina il peer Connect specificato.

```
aws ec2 delete-transit-gateway-connect-peer \
    --transit-gateway-connect-peer-id tgw-connect-peer-0666adbac4EXAMPLE
```

Output:

```
{
    "TransitGatewayConnectPeer": {
        "TransitGatewayAttachmentId": "tgw-attach-0f0927767cEXAMPLE",
        "TransitGatewayConnectPeerId": "tgw-connect-peer-0666adbac4EXAMPLE",
        "State": "deleting",
        "CreationTime": "2021-10-13T03:35:17.000Z",
        "ConnectPeerConfiguration": {
            "TransitGatewayAddress": "10.0.0.234",
            "PeerAddress": "172.31.1.11",
            "InsideCidrBlocks": [
                "169.254.6.0/29"
            ],
```

```
            "Protocol": "gre",
            "BgpConfigurations": [
                {
                    "TransitGatewayAsn": 64512,
                    "PeerAsn": 64512,
                    "TransitGatewayAddress": "169.254.6.2",
                    "PeerAddress": "169.254.6.1",
                    "BgpStatus": "down"
                },
                {
                    "TransitGatewayAsn": 64512,
                    "PeerAsn": 64512,
                    "TransitGatewayAddress": "169.254.6.3",
                    "PeerAddress": "169.254.6.1",
                    "BgpStatus": "down"
                }
            ]
        }
    }
}
```

Per ulteriori informazioni, consulta Collegamenti Transit gateway Connect e peer Connect nella Guida di Transit Gateway.

- Per i dettagli sull'API, consulta DeleteTransitGatewayConnectPeer AWS CLICommand Reference.

`delete-transit-gateway-connect`

Il seguente esempio di codice mostra come utilizzare`delete-transit-gateway-connect`.

AWS CLI

Come eliminare un collegamento per il Transit Gateway Connect

L'esempio `delete-transit-gateway-connect` seguente elimina il collegamento Connect specificato.

```
aws ec2 delete-transit-gateway-connect \
    --transit-gateway-attachment-id tgw-attach-037012e5dcEXAMPLE
```

Output:

```
{
    "TransitGatewayConnect": {
        "TransitGatewayAttachmentId": "tgw-attach-037012e5dcEXAMPLE",
        "TransportTransitGatewayAttachmentId": "tgw-attach-0a89069f57EXAMPLE",
        "TransitGatewayId": "tgw-02f776b1a7EXAMPLE",
        "State": "deleting",
        "CreationTime": "2021-03-09T19:59:17+00:00",
        "Options": {
            "Protocol": "gre"
        }
    }
}
```

Per ulteriori informazioni, consulta Collegamenti Transit gateway Connect e peer Connect nella Guida di Transit Gateway.

- Per i dettagli sull'API, consulta DeleteTransitGatewayConnect AWS CLICommand Reference.

`delete-transit-gateway-multicast-domain`

Il seguente esempio di codice mostra come utilizzare`delete-transit-gateway-multicast-domain`.

AWS CLI

Come eliminare un dominio multicast (trasmissione uno a molti) del gateway di transito

L'esempio `delete-transit-gateway-multicast-domain` seguente elimina il dominio multicast (trasmissione uno a molti) specificato.

```
aws ec2 delete-transit-gateway-multicast-domain \
    --transit-gateway-multicast-domain-id tgw-mcast-domain-0c4905cef7EXAMPLE
```

Output:

```
{
    "TransitGatewayMulticastDomain": {
        "TransitGatewayMulticastDomainId": "tgw-mcast-domain-02bb79002bEXAMPLE",
        "TransitGatewayId": "tgw-0d88d2d0d5EXAMPLE",
        "State": "deleting",
        "CreationTime": "2019-11-20T22:02:03.000Z"
```

```
    }
}
```

Per ulteriori informazioni, consulta Gestione dei domini multicast (trasmissione uno a molti) nella Guida di Transit Gateway.

- Per i dettagli sull'API, consulta DeleteTransitGatewayMulticastDomain AWS CLICommand Reference.

`delete-transit-gateway-peering-attachment`

Il seguente esempio di codice mostra come utilizzare`delete-transit-gateway-peering-attachment`.

AWS CLI

Come eliminare un collegamento peering del gateway di transito

L'esempio `delete-transit-gateway-peering-attachment` seguente elimina il collegamento peering del gateway di transito specificato.

```
aws ec2 delete-transit-gateway-peering-attachment \
    --transit-gateway-attachment-id tgw-attach-4455667788aabbccd
```

Output:

```
{
    "TransitGatewayPeeringAttachment": {
        "TransitGatewayAttachmentId": "tgw-attach-4455667788aabbccd",
        "RequesterTgwInfo": {
            "TransitGatewayId": "tgw-123abc05e04123abc",
            "OwnerId": "123456789012",
            "Region": "us-west-2"
        },
        "AccepterTgwInfo": {
            "TransitGatewayId": "tgw-11223344aabbcc112",
            "OwnerId": "123456789012",
            "Region": "us-east-2"
        },
        "State": "deleting",
        "CreationTime": "2019-12-09T11:38:31.000Z"
    }
```

```
}
```

Per ulteriori informazioni, consulta Accessori di peering del gateway di transito nella Guida di Transit Gateway.

- Per i dettagli sull'API, consulta DeleteTransitGatewayPeeringAttachment AWS CLICommand Reference.

`delete-transit-gateway-policy-table`

Il seguente esempio di codice mostra come utilizzare`delete-transit-gateway-policy-table`.

AWS CLI

Come eliminare una tabella di policy del gateway di transito

L'esempio `delete-transit-gateway-policy-table` seguente elimina la tabella di policy del gateway di transito specificata.

```
aws ec2 delete-transit-gateway-policy-table \
    --transit-gateway-policy-table-id tgw-ptb-0a16f134b78668a81
```

Output:

```
{
    "TransitGatewayPolicyTables": [
        {
            "TransitGatewayPolicyTableId": "tgw-ptb-0a16f134b78668a81",
            "TransitGatewayId": "tgw-067f8505c18f0bd6e",
            "State": "deleting",
            "CreationTime": "2023-11-28T16:36:43+00:00",
            "Tags": []
        }
    ]
}
```

Per ulteriori informazioni, consulta Tabelle di policy del gateway di transito nella Guida per l'utente di Transit Gateway.

- Per i dettagli sull'API, consulta DeleteTransitGatewayPolicyTable AWS CLICommand Reference.

`delete-transit-gateway-prefix-list-reference`

Il seguente esempio di codice mostra come utilizzare`delete-transit-gateway-prefix-list-reference`.

AWS CLI

Come eliminare un riferimento all'elenco di prefissi

L'esempio `delete-transit-gateway-prefix-list-reference` seguente elimina il riferimento all'elenco di prefissi specificato.

```
aws ec2 delete-transit-gateway-prefix-list-reference \
    --transit-gateway-route-table-id tgw-rtb-0123456789abcd123 \
    --prefix-list-id pl-1111112222222333
```

Output:

```
{
    "TransitGatewayPrefixListReference": {
        "TransitGatewayRouteTableId": "tgw-rtb-0123456789abcd123",
        "PrefixListId": "pl-11111122222222333",
        "PrefixListOwnerId": "123456789012",
        "State": "deleting",
        "Blackhole": false,
        "TransitGatewayAttachment": {
            "TransitGatewayAttachmentId": "tgw-attach-aabbccddaabbccaab",
            "ResourceType": "vpc",
            "ResourceId": "vpc-112233445566aabbc"
        }
    }
}
```

Per ulteriori informazioni, consulta Riferimenti elenco dei prefissi nella Guida di Transit Gateway.

- Per i dettagli sull'API, consulta DeleteTransitGatewayPrefixListReference AWS CLICommand Reference.

`delete-transit-gateway-route-table`

Il seguente esempio di codice mostra come utilizzare`delete-transit-gateway-route-table`.

AWS CLI

Come eliminare la tabella di routing di un gateway di transito

L'esempio `delete-transit-gateway-route-table` seguente elimina la tabella di routing del gateway di transito specificata.

```
aws ec2  delete-transit-gateway-route-table \
    --transit-gateway-route-table-id tgw-rtb-0b6f6aaa01EXAMPLE
```

Output:

```
{
    "TransitGatewayRouteTable": {
        "TransitGatewayRouteTableId": "tgw-rtb-0b6f6aaa01EXAMPLE",
        "TransitGatewayId": "tgw-02f776b1a7EXAMPLE",
        "State": "deleting",
        "DefaultAssociationRouteTable": false,
        "DefaultPropagationRouteTable": false,
        "CreationTime": "2019-07-17T20:27:26.000Z"
    }
}
```

Per ulteriori informazioni, consulta Eliminare la tabella di routing di un gateway di transito nella Guida di Transit Gateway.

- Per i dettagli sull'API, consulta DeleteTransitGatewayRouteTable AWS CLICommand Reference.

`delete-transit-gateway-route`

Il seguente esempio di codice mostra come utilizzare`delete-transit-gateway-route`.

AWS CLI

Come eliminare un intervallo CIDR da una tabella di routing

L'esempio `delete-transit-gateway-route` seguente elimina l'intervallo CIDR dalla tabella di routing del gateway di transito specificata.

```
aws ec2 delete-transit-gateway-route \
    --transit-gateway-route-table-id tgw-rtb-0b6f6aaa01EXAMPLE \
```

```
    --destination-cidr-block 10.0.2.0/24
```

Output:

```
{
    "Route": {
        "DestinationCidrBlock": "10.0.2.0/24",
        "TransitGatewayAttachments": [
            {
                "ResourceId": "vpc-0065acced4EXAMPLE",
                "TransitGatewayAttachmentId": "tgw-attach-0b5968d3b6EXAMPLE",
                "ResourceType": "vpc"
            }
        ],
        "Type": "static",
        "State": "deleted"
    }
}
```

Per ulteriori informazioni, consulta Eliminare una route statica nella Guida di Transit Gateway.

- Per i dettagli sull'API, consulta DeleteTransitGatewayRoute AWS CLICommand Reference.

`delete-transit-gateway-vpc-attachment`

Il seguente esempio di codice mostra come utilizzare`delete-transit-gateway-vpc-attachment`.

AWS CLI

Come eliminare un collegamento VPC del gateway di transito

L'esempio `delete-transit-gateway-vpc-attachment` seguente elimina il collegamento VPC specificato.

```
aws ec2 delete-transit-gateway-vpc-attachment \
    --transit-gateway-attachment-id tgw-attach-0d2c54bdbEXAMPLE
```

Output:

```
{
    "TransitGatewayVpcAttachment": {
```

```
        "TransitGatewayAttachmentId": "tgw-attach-0d2c54bdb3EXAMPLE",
        "TransitGatewayId": "tgw-02f776b1a7EXAMPLE",
        "VpcId": "vpc-0065acced4f61c651",
        "VpcOwnerId": "111122223333",
        "State": "deleting",
        "CreationTime": "2019-07-17T16:04:27.000Z"
    }
}
```

Per ulteriori informazioni, consulta Eliminare un collegamento a un VPC nella Guida di Transit Gateway.

- Per i dettagli sull'API, consulta DeleteTransitGatewayVpcAttachment AWS CLICommand Reference.

`delete-transit-gateway`

Il seguente esempio di codice mostra come utilizzare`delete-transit-gateway`.

AWS CLI

Come eliminare un gateway di transito

L'esempio `delete-transit-gateway` seguente elimina il gateway di transito specificato.

```
aws ec2  delete-transit-gateway \
    --transit-gateway-id tgw-01f04542b2EXAMPLE
```

Output:

```
{
    "TransitGateway": {
        "TransitGatewayId": "tgw-01f04542b2EXAMPLE",
        "State": "deleting",
        "OwnerId": "123456789012",
        "Description": "Example Transit Gateway",
        "CreationTime": "2019-08-27T15:04:35.000Z",
        "Options": {
            "AmazonSideAsn": 64515,
            "AutoAcceptSharedAttachments": "disable",
            "DefaultRouteTableAssociation": "enable",
            "AssociationDefaultRouteTableId": "tgw-rtb-0ce7a6948fEXAMPLE",
```

```
            "DefaultRouteTablePropagation": "enable",
            "PropagationDefaultRouteTableId": "tgw-rtb-0ce7a6948fEXAMPLE",
            "VpnEcmpSupport": "enable",
            "DnsSupport": "enable"
        }
    }
}
```

Per ulteriori informazioni, consulta Eliminare un gateway di transito nella Guida di Transit Gateway.

- Per i dettagli sull'API, consulta DeleteTransitGateway AWS CLICommand Reference.

`delete-verified-access-endpoint`

Il seguente esempio di codice mostra come utilizzare`delete-verified-access-endpoint`.

AWS CLI

Come eliminare un endpoint di accesso verificato

L'esempio `delete-verified-access-endpoint` seguente elimina l'endpoint di accesso verificato specificato.

```
aws ec2 delete-verified-access-endpoint \
    --verified-access-endpoint-id vae-066fac616d4d546f2
```

Output:

```
{
    "VerifiedAccessEndpoint": {
        "VerifiedAccessInstanceId": "vai-0ce000c0b7643abea",
        "VerifiedAccessGroupId": "vagr-0dbe967baf14b7235",
        "VerifiedAccessEndpointId": "vae-066fac616d4d546f2",
        "ApplicationDomain": "example.com",
        "EndpointType": "network-interface",
        "AttachmentType": "vpc",
        "DomainCertificateArn": "arn:aws:acm:us-east-2:123456789012:certificate/
eb065ea0-26f9-4e75-a6ce-0a1a7EXAMPLE",
        "EndpointDomain": "my-ava-
app.edge-00c3372d53b1540bb.vai-0ce000c0b7643abea.prod.verified-access.us-
east-2.amazonaws.com",
        "SecurityGroupIds": [
```

```
            "sg-004915970c4c8f13a"
        ],
        "NetworkInterfaceOptions": {
            "NetworkInterfaceId": "eni-0aec70418c8d87a0f",
            "Protocol": "https",
            "Port": 443
        },
        "Status": {
            "Code": "deleting"
        },
        "Description": "Testing Verified Access",
        "CreationTime": "2023-08-25T20:54:43",
        "LastUpdatedTime": "2023-08-25T22:46:32"
    }
}
```

Per ulteriori informazioni, consulta Endpoint di Accesso verificato nella Guida per l'utente di Accesso verificato da AWS .

- Per i dettagli sull'API, consulta DeleteVerifiedAccessEndpoint AWS CLICommand Reference.

`delete-verified-access-group`

Il seguente esempio di codice mostra come utilizzare`delete-verified-access-group`.

AWS CLI

Come eliminare un gruppo di accesso verificato

L'esempio `delete-verified-access-group` seguente elimina il gruppo di accesso verificato specificato.

```
aws ec2 delete-verified-access-group \
    --verified-access-group-id vagr-0dbe967baf14b7235
```

Output:

```
{
    "VerifiedAccessGroup": {
        "VerifiedAccessGroupId": "vagr-0dbe967baf14b7235",
        "VerifiedAccessInstanceId": "vai-0ce000c0b7643abea",
        "Description": "Testing Verified Access",
        "Owner": "123456789012",
```

```
        "VerifiedAccessGroupArn": "arn:aws:ec2:us-east-2:123456789012:verified-
access-group/vagr-0dbe967baf14b7235",
        "CreationTime": "2023-08-25T19:55:19",
        "LastUpdatedTime": "2023-08-25T22:49:03",
        "DeletionTime": "2023-08-26T00:58:31"
    }
}
```

Per ulteriori informazioni, consulta Gruppi di Accesso verificato nella Guida per l'utente di Accesso verificato da AWS .

- Per i dettagli sull'API, consulta DeleteVerifiedAccessGroup AWS CLICommand Reference.

`delete-verified-access-instance`

Il seguente esempio di codice mostra come utilizzare`delete-verified-access-instance`.

AWS CLI

Come eliminare un'istanza di accesso verificato

L'esempio `delete-verified-access-instance` seguente elimina l'istanza di accesso verificato specificata.

```
aws ec2 delete-verified-access-instance \
    --verified-access-instance-id vai-0ce000c0b7643abea
```

Output:

```
{
    "VerifiedAccessInstance": {
        "VerifiedAccessInstanceId": "vai-0ce000c0b7643abea",
        "Description": "Testing Verified Access",
        "VerifiedAccessTrustProviders": [],
        "CreationTime": "2023-08-25T18:27:56",
        "LastUpdatedTime": "2023-08-26T01:00:18"
    }
}
```

Per ulteriori informazioni, consulta Istanze di Accesso verificato nella Guida per l'utente di Accesso verificato da AWS .

- Per i dettagli sull'API, consulta DeleteVerifiedAccessInstance AWS CLICommand Reference.

`delete-verified-access-trust-provider`

Il seguente esempio di codice mostra come utilizzare`delete-verified-access-trust-provider`.

AWS CLI

Come eliminare un provider di attendibilità di accesso verificato

L'esempio `delete-verified-access-trust-provider` seguente elimina il provider di attendibilità di accesso verificato specificato.

```
aws ec2 delete-verified-access-trust-provider \
    --verified-access-trust-provider-id vatp-0bb32de759a3e19e7
```

Output:

```
{
    "VerifiedAccessTrustProvider": {
        "VerifiedAccessTrustProviderId": "vatp-0bb32de759a3e19e7",
        "Description": "Testing Verified Access",
        "TrustProviderType": "user",
        "UserTrustProviderType": "iam-identity-center",
        "PolicyReferenceName": "idc",
        "CreationTime": "2023-08-25T18:40:36",
        "LastUpdatedTime": "2023-08-25T18:40:36"
    }
}
```

Per ulteriori informazioni, consulta Provider di attendibilità per Accesso verificato nella Guida per l'utente di Accesso verificato da AWS .

- Per i dettagli sull'API, consulta DeleteVerifiedAccessTrustProvider AWS CLICommand Reference.

`delete-volume`

Il seguente esempio di codice mostra come utilizzare`delete-volume`.

AWS CLI

Come eliminare un volume

Questo comando di esempio elimina un volume disponibile con l'ID volume di `vol-049df61146c4d7901`. Se il comando va a buon fine, non viene restituito alcun output.

Comando:

```
aws ec2 delete-volume --volume-id vol-049df61146c4d7901
```

- Per i dettagli sull'API, consulta DeleteVolume AWS CLICommand Reference.

`delete-vpc-endpoint-connection-notifications`

Il seguente esempio di codice mostra come utilizzare`delete-vpc-endpoint-connection-notifications`.

AWS CLI

Come eliminare una notifica della connessione endpoint

Questo esempio elimina la notifica della connessione endpoint specificata.

Comando:

```
aws ec2 delete-vpc-endpoint-connection-notifications --connection-notification-
ids vpce-nfn-008776de7e03f5abc
```

Output:

```
{
  "Unsuccessful": []
}
```

- Per i dettagli sull'API, consulta DeleteVpcEndpointConnectionNotifications AWS CLICommand Reference.

`delete-vpc-endpoint-service-configurations`

Il seguente esempio di codice mostra come utilizzare`delete-vpc-endpoint-service-configurations`.

AWS CLI

Come eliminare una configurazione di servizio endpoint

Questo esempio elimina la configurazione di servizio endpoint specificata.

Comando:

```
aws ec2 delete-vpc-endpoint-service-configurations --service-ids vpce-svc-03d5ebb7d9579a2b3
```

Output:

```
{
   "Unsuccessful": []
}
```

- Per i dettagli sull'API, consulta DeleteVpcEndpointServiceConfigurations AWS CLICommand Reference.

`delete-vpc-endpoints`

Il seguente esempio di codice mostra come utilizzare`delete-vpc-endpoints`.

AWS CLI

Come eliminare un endpoint

Questo esempio elimina gli endpoint vpce-aa22bb33 e vpce-1a2b3c4d. Se il comando ha esito parzialmente positivo o negativo, viene restituito un elenco di elementi non riusciti. Se il comando ha esito positivo, l'elenco restituito è vuoto.

Comando:

```
aws ec2 delete-vpc-endpoints --vpc-endpoint-ids vpce-aa22bb33 vpce-1a2b3c4d
```

Output:

```
{
  "Unsuccessful": []
}
```

- Per i dettagli sull'API, consulta DeleteVpcEndpoints AWS CLICommand Reference.

`delete-vpc-peering-connection`

Il seguente esempio di codice mostra come utilizzare`delete-vpc-peering-connection`.

AWS CLI

Come eliminare una connessione peering VPC

Questo esempio elimina la connessione peering VPC specificata.

Comando:

```
aws ec2 delete-vpc-peering-connection --vpc-peering-connection-id pcx-1a2b3c4d
```

Output:

```
{
    "Return": true
}
```

- Per i dettagli sull'API, consulta DeleteVpcPeeringConnection AWS CLICommand Reference.

`delete-vpc`

Il seguente esempio di codice mostra come utilizzare`delete-vpc`.

AWS CLI

Come eliminare un VPC

Questo esempio elimina il VPC specificato. Se il comando va a buon fine, non viene restituito alcun output.

Comando:

```
aws ec2 delete-vpc --vpc-id vpc-a01106c2
```

- Per i dettagli sull'API, consulta DeleteVpc AWS CLICommand Reference.

`delete-vpn-connection-route`

Il seguente esempio di codice mostra come utilizzare`delete-vpn-connection-route`.

AWS CLI

Come eliminare un routing statico da una connessione VPN

Questo esempio elimina la route statica specificata dalla connessione VPN specificata. Se il comando va a buon fine, non viene restituito alcun output.

Comando:

```
aws ec2 delete-vpn-connection-route --vpn-connection-id vpn-40f41529 --destination-cidr-block 11.12.0.0/16
```

- Per i dettagli sull'API, consulta DeleteVpnConnectionRoute AWS CLICommand Reference.

`delete-vpn-connection`

Il seguente esempio di codice mostra come utilizzare`delete-vpn-connection`.

AWS CLI

Come eliminare una connessione VPN

Questo esempio elimina la connessione VPN specificata. Se il comando va a buon fine, non viene restituito alcun output.

Comando:

```
aws ec2 delete-vpn-connection --vpn-connection-id vpn-40f41529
```

- Per i dettagli sull'API, consulta DeleteVpnConnection AWS CLICommand Reference.

`delete-vpn-gateway`

Il seguente esempio di codice mostra come utilizzare`delete-vpn-gateway`.

AWS CLI

Come eliminare un gateway privato virtuale

Questo esempio elimina il gateway privato virtuale specificato. Se il comando va a buon fine, non viene restituito alcun output.

Comando:

```
aws ec2 delete-vpn-gateway --vpn-gateway-id vgw-9a4cacf3
```

- Per i dettagli sull'API, consulta DeleteVpnGateway AWS CLICommand Reference.

`deprovision-byoip-cidr`

Il seguente esempio di codice mostra come utilizzare`deprovision-byoip-cidr`.

AWS CLI

Come rimuovere un intervallo di indirizzi IP dall'uso

L'esempio seguente rimuove l'intervallo di indirizzi specificato dall'uso con AWS.

```
aws ec2 deprovision-byoip-cidr \
    --cidr 203.0.113.25/24
```

Output:

```
{
    "ByoipCidr": {
        "Cidr": "203.0.113.25/24",
        "State": "pending-deprovision"
    }
}
```

- Per i dettagli sull'API, vedere DeprovisionByoipCidrin AWS CLI Command Reference.

`deprovision-ipam-pool-cidr`

Il seguente esempio di codice mostra come utilizzare`deprovision-ipam-pool-cidr`.

AWS CLI

Come revocare il provisioning di un CIDR del pool IPAM

L'esempio `deprovision-ipam-pool-cidr` seguente revoca il provisioning di un CIDR assegnato a un pool IPAM.

Linux:

```
aws ec2 deprovision-ipam-pool-cidr \
    --ipam-pool-id ipam-pool-02ec043a19bbe5d08 \
    --cidr 11.0.0.0/16
```

Windows:

```
aws ec2 deprovision-ipam-pool-cidr ^
    --ipam-pool-id ipam-pool-02ec043a19bbe5d08 ^
    --cidr 11.0.0.0/16
```

Output:

```
{
    "IpamPoolCidr": {
        "Cidr": "11.0.0.0/16",
        "State": "pending-deprovision"
    }
}
```

Per ulteriori informazioni, consulta Deprovisioning pool CIDRs nella Amazon VPC IPAM User Guide.

- Per i dettagli sull'API, consulta Command DeprovisionIpamPoolCidrReference AWS CLI .

`deregister-image`

Il seguente esempio di codice mostra come utilizzare`deregister-image`.

AWS CLI

Come annullare la registrazione di un'AMI

Questo esempio annulla la registrazione dell'AMI specificata. Se il comando va a buon fine, non viene restituito alcun output.

Comando:

```
aws ec2 deregister-image --image-id ami-4fa54026
```

- Per i dettagli sull'API, consulta DeregisterImage AWS CLICommand Reference.

`deregister-instance-event-notification-attributes`

Il seguente esempio di codice mostra come utilizzare`deregister-instance-event-notification-attributes`.

AWS CLI

Esempio 1: come rimuovere tutti i tag dalle notifiche eventi

L'esempio `deregister-instance-event-notification-attributes` seguente rimuove `IncludeAllTagsOfInstance=true`, che ha l'effetto di impostare `IncludeAllTagsOfInstance` su `false`.

```
aws ec2 deregister-instance-event-notification-attributes \
    --instance-tag-attribute IncludeAllTagsOfInstance=true
```

Output:

```
{
    "InstanceTagAttribute": {
        "InstanceTagKeys": [],
        "IncludeAllTagsOfInstance": true
    }
}
```

Per ulteriori informazioni, consulta Eventi pianificati per le istanze nella Guida per l'utente di Amazon Elastic Compute Cloud per le istanze Linux.

Esempio 2: come rimuovere tag specifici dalle notifiche eventi

L'esempio `deregister-instance-event-notification-attributes` seguente rimuove il tag specificato dai tag inclusi nelle notifiche eventi. Per descrivere i tag rimanenti inclusi nelle notifiche eventi, utilizza `describe-instance-event-notification-attributes`.

```
aws ec2 deregister-instance-event-notification-attributes \
```

```
    --instance-tag-attribute InstanceTagKeys="tag-key2"
```

Output:

```
{
    "InstanceTagAttribute": {
        "InstanceTagKeys": [
            "tag-key2"
        ],
        "IncludeAllTagsOfInstance": false
    }
}
```

Per ulteriori informazioni, consulta Eventi pianificati per le istanze nella Guida per l'utente di Amazon Elastic Compute Cloud per le istanze Linux.

- Per i dettagli sull'API, consulta DeregisterInstanceEventNotificationAttributes AWS CLICommand Reference.

`deregister-transit-gateway-multicast-group-members`

Il seguente esempio di codice mostra come utilizzare`deregister-transit-gateway-multicast-group-members`.

AWS CLI

Come annullare la registrazione di membri da un gruppo multicast (trasmissione uno a molti)

Questo esempio annulla la registrazione del membro del gruppo di interfaccia di rete specificato per il gruppo multicast (trasmissione uno a molti) del gateway di transito.

```
aws ec2 deregister-transit-gateway-multicast-group-members \
    --transit-gateway-multicast-domain-id tgw-mcast-domain-0c4905cef7EXAMPLE \
    --group-ip-address 224.0.1.0 \
    --network-interface-ids eni-0e246d3269EXAMPLE
```

Output:

```
{
    "DeregisteredMulticastGroupMembers": {
```

```
        "TransitGatewayMulticastDomainId": "tgw-mcast-domain-0c4905cef7EXAMPLE",
        "RegisteredNetworkInterfaceIds": [
            "eni-0e246d3269EXAMPLE"
        ],
        "GroupIpAddress": "224.0.1.0"
    }
}
```

Per ulteriori informazioni, consulta Annullare la registrazione di membri da un gruppo multicast (trasmissione uno a molti) nella Guida per l'utente di AWS Transit Gateway.

- Per i dettagli sull'API, consulta DeregisterTransitGatewayMulticastGroupMembers AWS CLICommand Reference.

`deregister-transit-gateway-multicast-group-source`

Il seguente esempio di codice mostra come utilizzare`deregister-transit-gateway-multicast-group-source`.

AWS CLI

Come annullare la registrazione di un'origine dal gruppo multicast (trasmissione uno a molti) del gateway di transito

Questo esempio annulla la registrazione dell'origine del gruppo di interfacce di rete specificato dal gruppo multicast (trasmissione uno a molti).

```
aws ec2 register-transit-gateway-multicast-group-sources \
    --transit-gateway-multicast-domain-id tgw-mcast-domain-0c4905cef79d6e597 \
    --group-ip-address 224.0.1.0 \
    --network-interface-ids eni-07f290fc3c090cbae
```

Output:

```
{
    "DeregisteredMulticastGroupSources": {
        "TransitGatewayMulticastDomainId": "tgw-mcast-domain-0c4905cef79d6e597",
        "DeregisteredNetworkInterfaceIds": [
            "eni-07f290fc3c090cbae"
        ],
        "GroupIpAddress": "224.0.1.0"
```

```
    }
}
```

Per ulteriori informazioni, consulta Annullamento della registrazione delle origini da un gruppo multicast (trasmissione uno a molti) nella Guida per l'utente di AWS Transit Gateway.

- Per i dettagli sull'API, consulta DeregisterTransitGatewayMulticastGroupSource AWS CLICommand Reference.

`describe-account-attributes`

Il seguente esempio di codice mostra come utilizzare`describe-account-attributes`.

AWS CLI

Per descrivere tutti gli attributi del tuo AWS account

Questo esempio descrive gli attributi del tuo AWS account.

Comando:

```
aws ec2 describe-account-attributes
```

Output:

```
{
    "AccountAttributes": [
        {
            "AttributeName": "vpc-max-security-groups-per-interface",
            "AttributeValues": [
                {
                    "AttributeValue": "5"
                }
            ]
        },
        {
            "AttributeName": "max-instances",
            "AttributeValues": [
                {
                    "AttributeValue": "20"
                }
            ]
```

```
        },
        {
            "AttributeName": "supported-platforms",
            "AttributeValues": [
                {
                    "AttributeValue": "EC2"
                },
                {
                    "AttributeValue": "VPC"
                }
            ]
        },
        {
            "AttributeName": "default-vpc",
            "AttributeValues": [
                {
                    "AttributeValue": "none"
                }
            ]
        },
        {
            "AttributeName": "max-elastic-ips",
            "AttributeValues": [
                {
                    "AttributeValue": "5"
                }
            ]
        },
        {
            "AttributeName": "vpc-max-elastic-ips",
            "AttributeValues": [
                {
                    "AttributeValue": "5"
                }
            ]
        }
    ]
}
```

Per descrivere un singolo attributo per il tuo AWS account

Questo esempio descrive l'`supported-platforms`attributo del tuo AWS account.

Comando:

```
aws ec2 describe-account-attributes --attribute-names supported-platforms
```

Output:

```
{
    "AccountAttributes": [
        {
            "AttributeName": "supported-platforms",
            "AttributeValues": [
                {
                    "AttributeValue": "EC2"
                },
                {
                    "AttributeValue": "VPC"
                }
            ]
        }
    ]
}
```

- Per i dettagli sull'API, consulta DescribeAccountAttributes AWS CLICommand Reference.

`describe-address-transfers`

Il seguente esempio di codice mostra come utilizzare`describe-address-transfers`.

AWS CLI

Come descrivere il trasferimento di un indirizzo IP elastico

L'esempio `describe-address-transfers` seguente descrive il trasferimento dell'indirizzo IP elastico per l'indirizzo IP elastico specificato.

```
aws ec2 describe-address-transfers \
    --allocation-ids eipalloc-09ad461b0d03f6aaf
```

Output:

```
{
    "AddressTransfers": [
        {
```

```
            "PublicIp": "100.21.184.216",
            "AllocationId": "eipalloc-09ad461b0d03f6aaf",
            "TransferAccountId": "123456789012",
            "TransferOfferExpirationTimestamp": "2023-02-22T22:51:01.000Z",
            "AddressTransferStatus": "pending"
        }
    ]
}
```

Per ulteriori informazioni, consulta Trasferimento degli indirizzi IP elastici nella Guida per l'utente di Amazon VPC.

- Per i dettagli sull'API, consulta DescribeAddressTransfers AWS CLICommand Reference.

`describe-addresses-attribute`

Il seguente esempio di codice mostra come utilizzare`describe-addresses-attribute`.

AWS CLI

Come visualizzare gli attributi del nome di dominio associato a un indirizzo IP elastico

Gli esempi `describe-addresses-attribute` seguenti restituiscono gli attributi del nome di dominio associato all'indirizzo IP elastico.

Linux:

```
aws ec2 describe-addresses-attribute \
    --allocation-ids eipalloc-abcdef01234567890 \
    --attribute domain-name
```

Windows:

```
aws ec2 describe-addresses-attribute ^
    --allocation-ids eipalloc-abcdef01234567890 ^
    --attribute domain-name
```

Output:

```
{
    "Addresses": [
```

```
        {
            "PublicIp": "192.0.2.0",
            "AllocationId": "eipalloc-abcdef01234567890",
            "PtrRecord": "example.com."
        }
    ]
}
```

Per visualizzare gli attributi di un indirizzo IP elastico, è necessario aver prima associato un nome di dominio all'indirizzo IP elastico. Per ulteriori informazioni, consulta Use reverse DNS per le applicazioni di posta elettronica nella Amazon EC2 User Guide o modify-address-attributenella AWS CLI Command Reference.

- Per i dettagli sull'API, consulta AWS CLI Command DescribeAddressesAttributeReference.

`describe-addresses`

Il seguente esempio di codice mostra come utilizzare`describe-addresses`.

AWS CLI

Esempio 1: come recuperare i dettagli di tutti gli indirizzi IP elastici

Nell'esempio di `describe addresses` seguente vengono visualizzati tutti i dettagli relativi agli indirizzi IP elastici.

```
aws ec2 describe-addresses
```

Output:

```
{
    "Addresses": [
        {
            "InstanceId": "i-1234567890abcdef0",
            "PublicIp": "198.51.100.0",
            "PublicIpv4Pool": "amazon",
            "Domain": "standard"
        },
        {
            "Domain": "vpc",
            "PublicIpv4Pool": "amazon",
            "InstanceId": "i-1234567890abcdef0",
```

```
            "NetworkInterfaceId": "eni-12345678",
            "AssociationId": "eipassoc-12345678",
            "NetworkInterfaceOwnerId": "123456789012",
            "PublicIp": "203.0.113.0",
            "AllocationId": "eipalloc-12345678",
            "PrivateIpAddress": "10.0.1.241"
        }
    ]
}
```

Esempio 2: per recuperare i dettagli degli indirizzi IP elastici per EC2 -VPC

Nell'esempio di `describe-addresses` seguente vengono visualizzati i dettagli relativi agli indirizzi IP elastici utilizzati per le istanze in un VPC.

```
aws ec2 describe-addresses \
    --filters "Name=domain,Values=vpc"
```

Output:

```
{
    "Addresses": [
        {
            "Domain": "vpc",
            "PublicIpv4Pool": "amazon",
            "InstanceId": "i-1234567890abcdef0",
            "NetworkInterfaceId": "eni-12345678",
            "AssociationId": "eipassoc-12345678",
            "NetworkInterfaceOwnerId": "123456789012",
            "PublicIp": "203.0.113.0",
            "AllocationId": "eipalloc-12345678",
            "PrivateIpAddress": "10.0.1.241"
        }
    ]
}
```

Esempio 3: come recuperare i dettagli di un indirizzo IP elastico specificato dall'ID di allocazione

L'`describe-addresses`esempio seguente visualizza i dettagli sull'indirizzo IP elastico con l'ID di allocazione specificato, associato a un'istanza in EC2 -VPC.

```
aws ec2 describe-addresses \
```

```
    --allocation-ids eipalloc-282d9641
```

Output:

```
{
    "Addresses": [
        {
            "Domain": "vpc",
            "PublicIpv4Pool": "amazon",
            "InstanceId": "i-1234567890abcdef0",
            "NetworkInterfaceId": "eni-1a2b3c4d",
            "AssociationId": "eipassoc-123abc12",
            "NetworkInterfaceOwnerId": "1234567891012",
            "PublicIp": "203.0.113.25",
            "AllocationId": "eipalloc-282d9641",
            "PrivateIpAddress": "10.251.50.12"
        }
    ]
}
```

Esempio 4: come recuperare i dettagli di un indirizzo IP elastico specificato dall'indirizzo IP privato del VPC

L'`describe-addresses`esempio seguente mostra i dettagli sull'indirizzo IP elastico associato a un particolare indirizzo IP privato in EC2 -VPC.

```
aws ec2 describe-addresses \
    --filters "Name=private-ip-address,Values=10.251.50.12"
```

Esempio 5: per recuperare i dettagli sugli indirizzi IP elastici in -Classic EC2

TThe l'`describe-addresses`esempio seguente mostra i dettagli sugli indirizzi IP elastici da utilizzare in EC2 -Classic.

```
aws ec2 describe-addresses \
    --filters "Name=domain,Values=standard"
```

Output:

```
{
```

```
    "Addresses": [
        {
            "InstanceId": "i-1234567890abcdef0",
            "PublicIp": "203.0.110.25",
            "PublicIpv4Pool": "amazon",
            "Domain": "standard"
        }
    ]
}
```

Esempio 6: come recuperare i dettagli di un indirizzo IP elastico specificato dall'indirizzo IP pubblico

L'`describe-addresses`esempio seguente visualizza i dettagli sull'indirizzo IP elastico con il valore`203.0.110.25`, che è associato a un'istanza in EC2 -Classic.

```
aws ec2 describe-addresses \
    --public-ips 203.0.110.25
```

Output:

```
{
    "Addresses": [
        {
            "InstanceId": "i-1234567890abcdef0",
            "PublicIp": "203.0.110.25",
            "PublicIpv4Pool": "amazon",
            "Domain": "standard"
        }
    ]
}
```

- Per i dettagli sull'API, consulta DescribeAddresses AWS CLICommand Reference.

`describe-aggregate-id-format`

Il seguente esempio di codice mostra come utilizzare`describe-aggregate-id-format`.

AWS CLI

Come descrivere le impostazioni del formato ID più lungo per tutti i tipi di risorse in una Regione

L'esempio `describe-aggregate-id-format` seguente descrive lo stato complessivo del formato ID lungo per la Regione corrente. Il valore `Deadline` indica che le scadenze per il passaggio permanente di queste risorse dal formato ID breve al formato ID lungo sono scadute. Il valore `UseLongIdsAggregated` indica che tutti gli utenti e i ruoli IAM sono configurati per utilizzare il formato ID lungo per tutti i tipi di risorse.

```
aws ec2 describe-aggregate-id-format
```

Output:

```
{
    "UseLongIdsAggregated": true,
    "Statuses": [
        {
            "Deadline": "2018-08-13T02:00:00.000Z",
            "Resource": "network-interface-attachment",
            "UseLongIds": true
        },
        {
            "Deadline": "2016-12-13T02:00:00.000Z",
            "Resource": "instance",
            "UseLongIds": true
        },
        {
            "Deadline": "2018-08-13T02:00:00.000Z",
            "Resource": "elastic-ip-association",
            "UseLongIds": true
        },
        ...
    ]
}
```

- Per i dettagli sull'API, consulta DescribeAggregateIdFormat AWS CLICommand Reference.

`describe-availability-zones`

Il seguente esempio di codice mostra come utilizzare`describe-availability-zones`.

AWS CLI

Come descrivere le zone di disponibilità

In questo esempio di `describe-availability-zones` vengono mostrati i dettagli della zona di disponibilità disponibili a te. La risposta include le zone di disponibilità solo per la regione attuale. In questo esempio si utilizza la regione predefinita del profilo `us-west-2` (Oregon).

```
aws ec2 describe-availability-zones
```

Output:

```
{
    "AvailabilityZones": [
        {
            "State": "available",
            "OptInStatus": "opt-in-not-required",
            "Messages": [],
            "RegionName": "us-west-2",
            "ZoneName": "us-west-2a",
            "ZoneId": "usw2-az1",
            "GroupName": "us-west-2",
            "NetworkBorderGroup": "us-west-2"
        },
        {
            "State": "available",
            "OptInStatus": "opt-in-not-required",
            "Messages": [],
            "RegionName": "us-west-2",
            "ZoneName": "us-west-2b",
            "ZoneId": "usw2-az2",
            "GroupName": "us-west-2",
            "NetworkBorderGroup": "us-west-2"
        },
        {
            "State": "available",
            "OptInStatus": "opt-in-not-required",
            "Messages": [],
            "RegionName": "us-west-2",
            "ZoneName": "us-west-2c",
            "ZoneId": "usw2-az3",
            "GroupName": "us-west-2",
            "NetworkBorderGroup": "us-west-2"
        },
        {
            "State": "available",
            "OptInStatus": "opt-in-not-required",
```

```
            "Messages": [],
            "RegionName": "us-west-2",
            "ZoneName": "us-west-2d",
            "ZoneId": "usw2-az4",
            "GroupName": "us-west-2",
            "NetworkBorderGroup": "us-west-2"
        },
        {
            "State": "available",
            "OptInStatus": "opted-in",
            "Messages": [],
            "RegionName": "us-west-2",
            "ZoneName": "us-west-2-lax-1a",
            "ZoneId": "usw2-lax1-az1",
            "GroupName": "us-west-2-lax-1",
            "NetworkBorderGroup": "us-west-2-lax-1"
        }
    ]
}
```

- Per i dettagli sull'API, consulta DescribeAvailabilityZones AWS CLICommand Reference.

`describe-aws-network-performance-metric-subscription`

Il seguente esempio di codice mostra come utilizzare`describe-aws-network-performance-metric-subscription`.

AWS CLI

Come descrivere le sottoscrizioni di parametri di prestazioni

L'esempio `describe-aws-network-performance-metric-subscriptions` seguente descrive le sottoscrizioni di parametri di prestazioni.

```
aws ec2 describe-aws-network-performance-metric-subscriptions
```

Output:

```
{
    "Subscriptions": [
        {
```

```
            "Source": "us-east-1",
            "Destination": "eu-west-1",
            "Metric": "aggregate-latency",
            "Statistic": "p50",
            "Period": "five-minutes"
        }
    ]
}
```

Per ulteriori informazioni, consulta Gestione delle sottoscrizioni nella Guida per l'utente di Infrastructure Performance.

- Per i dettagli sull'API, consulta DescribeAwsNetworkPerformanceMetricSubscription AWS CLICommand Reference.

`describe-aws-network-performance-metric-subscriptions`

Il seguente esempio di codice mostra come utilizzare`describe-aws-network-performance-metric-subscriptions`.

AWS CLI

Come descrivere le sottoscrizioni di parametri di prestazioni

L'esempio `describe-aws-network-performance-metric-subscriptions` seguente descrive le sottoscrizioni di parametri di prestazioni.

```
aws ec2 describe-aws-network-performance-metric-subscriptions
```

Output:

```
{
    "Subscriptions": [
        {
            "Source": "us-east-1",
            "Destination": "eu-west-1",
            "Metric": "aggregate-latency",
            "Statistic": "p50",
            "Period": "five-minutes"
        }
    ]
```

```
}
```

Per ulteriori informazioni, consulta Gestione delle sottoscrizioni nella Guida per l'utente di Infrastructure Performance.

- Per i dettagli sull'API, consulta DescribeAwsNetworkPerformanceMetricSubscriptions AWS CLICommand Reference.

`describe-bundle-tasks`

Il seguente esempio di codice mostra come utilizzare`describe-bundle-tasks`.

AWS CLI

Come descrivere le attività del bundle

Questo esempio descrive tutte le attività del bundle.

Comando:

```
aws ec2 describe-bundle-tasks
```

Output:

```
{
  "BundleTasks": [
    {
      "UpdateTime": "2015-09-15T13:26:54.000Z",
      "InstanceId": "i-1234567890abcdef0",
      "Storage": {
        "S3": {
            "Prefix": "winami",
            "Bucket": "bundletasks"
        }
      },
      "State": "bundling",
      "StartTime": "2015-09-15T13:24:35.000Z",
      "Progress": "3%",
      "BundleId": "bun-2a4e041c"
    }
  ]
```

```
}
```

- Per i dettagli sull'API, consulta DescribeBundleTasks AWS CLICommand Reference.

`describe-byoip-cidrs`

Il seguente esempio di codice mostra come utilizzare`describe-byoip-cidrs`.

AWS CLI

Come descrivere gli intervalli di indirizzi allocati

L'`describe-byoip-cidrs`esempio seguente visualizza i dettagli sugli intervalli di IPv4 indirizzi pubblici da AWS cui è stato assegnato il provisioning.

```
aws ec2 describe-byoip-cidrs
```

Output:

```
{
    "ByoipCidrs": [
        {
            "Cidr": "203.0.113.25/24",
            "StatusMessage": "ipv4pool-ec2-1234567890abcdef0",
            "State": "provisioned"
        }
    ]
}
```

- Per i dettagli sull'API, consulta DescribeByoipCidrs AWS CLICommand Reference.

`describe-capacity-reservation-fleets`

Il seguente esempio di codice mostra come utilizzare`describe-capacity-reservation-fleets`.

AWS CLI

Come visualizzare un parco di prenotazione della capacità

L'esempio `describe-capacity-reservation-fleets` seguente elenca le informazioni di configurazione e le capacità per il parco di prenotazione della capacità specificato. Elenca inoltre i dettagli relativi alle singole prenotazioni della capacità presenti nel parco.

```
aws ec2 describe-capacity-reservation-fleets \
    --capacity-reservation-fleet-ids crf-abcdef01234567890
```

Output:

```
{
    "CapacityReservationFleets": [
        {
            "State": "active",
            "EndDate": "2022-12-31T23:59:59.000Z",
            "InstanceMatchCriteria": "open",
            "Tags": [],
            "CapacityReservationFleetId": "crf-abcdef01234567890",
            "Tenancy": "default",
            "InstanceTypeSpecifications": [
                {
                    "CapacityReservationId": "cr-1234567890abcdef0",
                    "AvailabilityZone": "us-east-1a",
                    "FulfilledCapacity": 5.0,
                    "Weight": 1.0,
                    "CreateDate": "2022-07-02T08:34:33.398Z",
                    "InstancePlatform": "Linux/UNIX",
                    "TotalInstanceCount": 5,
                    "Priority": 1,
                    "EbsOptimized": true,
                    "InstanceType": "m5.xlarge"
                }
            ],
            "TotalTargetCapacity": 5,
            "TotalFulfilledCapacity": 5.0,
            "CreateTime": "2022-07-02T08:34:33.397Z",
            "AllocationStrategy": "prioritized"
        }
    ]
}
```

Per ulteriori informazioni sulle flotte di prenotazione della capacità, consulta Flotte di prenotazione della capacità nella Guida per gli EC2 utenti di Amazon.

- Per i dettagli sull'API, consulta AWS CLI Command DescribeCapacityReservationFleetsReference.

`describe-capacity-reservations`

Il seguente esempio di codice mostra come utilizzare`describe-capacity-reservations`.

AWS CLI

Esempio 1: come descrivere una o più prenotazioni della capacità

L'`describe-capacity-reservations`esempio seguente mostra i dettagli su tutte le prenotazioni di capacità nella AWS regione corrente.

```
aws ec2 describe-capacity-reservations
```

Output:

```
{
    "CapacityReservations": [
        {
            "CapacityReservationId": "cr-1234abcd56EXAMPLE ",
            "OwnerId": "123456789111",
            "CapacityReservationArn": "arn:aws:ec2:us-east-1:123456789111:capacity-
reservation/cr-1234abcd56EXAMPLE",
            "AvailabilityZoneId": "use1-az2",
            "InstanceType": "c5.large",
            "InstancePlatform": "Linux/UNIX",
            "AvailabilityZone": "us-east-1a",
            "Tenancy": "default",
            "TotalInstanceCount": 1,
            "AvailableInstanceCount": 1,
            "EbsOptimized": true,
            "EphemeralStorage": false,
            "State": "active",
            "StartDate": "2024-10-23T15:00:24+00:00",
            "EndDateType": "unlimited",
            "InstanceMatchCriteria": "open",
            "CreateDate": "2024-10-23T15:00:24+00:00",
            "Tags": [],
            "CapacityAllocations": []
        },
```

```
        {
            "CapacityReservationId": "cr-abcdEXAMPLE9876ef ",
            "OwnerId": "123456789111",
            "CapacityReservationArn": "arn:aws:ec2:us-east-1:123456789111:capacity-
reservation/cr-abcdEXAMPLE9876ef",
            "AvailabilityZoneId": "use1-az2",
            "InstanceType": "c4.large",
            "InstancePlatform": "Linux/UNIX",
            "AvailabilityZone": "us-east-1a",
            "Tenancy": "default",
            "TotalInstanceCount": 1,
            "AvailableInstanceCount": 1,
            "EbsOptimized": true,
            "EphemeralStorage": false,
            "State": "cancelled",
            "StartDate": "2024-10-23T15:01:03+00:00",
            "EndDateType": "unlimited",
            "InstanceMatchCriteria": "open",
            "CreateDate": "2024-10-23T15:01:02+00:00",
            "Tags": [],
            "CapacityAllocations": []
        }
    ]
}
```

Esempio 2: come descrivere una o più prenotazioni della capacità

L'esempio `describe-capacity-reservations` seguente visualizza i dettagli sulla prenotazione della capacità specificata.

```
aws ec2 describe-capacity-reservations \
    --capacity-reservation-ids cr-1234abcd56EXAMPLE
```

Output:

```
{
    "CapacityReservations": [
        {
            "CapacityReservationId": "cr-abcdEXAMPLE9876ef ",
            "OwnerId": "123456789111",
            "CapacityReservationArn": "arn:aws:ec2:us-east-1:123456789111:capacity-
reservation/cr-abcdEXAMPLE9876ef",
```

```
            "AvailabilityZoneId": "use1-az2",
            "InstanceType": "c4.large",
            "InstancePlatform": "Linux/UNIX",
            "AvailabilityZone": "us-east-1a",
            "Tenancy": "default",
            "TotalInstanceCount": 1,
            "AvailableInstanceCount": 1,
            "EbsOptimized": true,
            "EphemeralStorage": false,
            "State": "active",
            "StartDate": "2024-10-23T15:01:03+00:00",
            "EndDateType": "unlimited",
            "InstanceMatchCriteria": "open",
            "CreateDate": "2024-10-23T15:01:02+00:00",
            "Tags": [],
            "CapacityAllocations": []
        }
    ]
}
```


Per ulteriori informazioni, consulta Visualizzazione di una prenotazione della capacità nella Guida per l'utente di Amazon Elastic Compute Cloud per le istanze Linux.

- Per i dettagli sull'API, consulta DescribeCapacityReservations AWS CLICommand Reference.

`describe-carrier-gateways`

Il seguente esempio di codice mostra come utilizzare`describe-carrier-gateways`.

AWS CLI

Come descrivere tutti i gateway carrier

L'esempio `describe-carrier-gateways` seguente elenca tutti i gateway carrier in uso.

```
aws ec2 describe-carrier-gateways
```

Output:


```
{
    "CarrierGateways": [
        {
```

```
            "CarrierGatewayId": "cagw-0465cdEXAMPLE1111",
            "VpcId": "vpc-0c529aEXAMPLE",
            "State": "available",
            "OwnerId": "123456789012",
            "Tags": [
                {

                    "Key": "example",
                    "Value": "tag"
                }
            ]
        }
    ]
}
```

Per ulteriori informazioni, consulta Carrier gateways< https://docs.aws.amazon.com/vpc/ latest/ userguide/Carrier _gateway.html> nella Amazon Virtual Private Cloud User Guide.

- Per i dettagli sull'API, consulta Command Reference. DescribeCarrierGatewaysAWS CLI

`describe-classic-link-instances`

Il seguente esempio di codice mostra come utilizzare`describe-classic-link-instances`.

AWS CLI

Per descrivere le istanze EC2 -Classic collegate

Questo esempio elenca tutte le istanze EC2 -Classic collegate.

Comando:

```
aws ec2 describe-classic-link-instances
```

Output:

```
{
        "Instances": [
                {
                        "InstanceId": "i-1234567890abcdef0",
                        "VpcId": "vpc-88888888",
                        "Groups": [
```

```
                                {
                                        "GroupId": "sg-11122233"
                                }
                        ],
                        "Tags": [
                                {
                                        "Value": "ClassicInstance",
                                        "Key": "Name"
                                }
                        ]
                },
                {
                        "InstanceId": "i-0598c7d356eba48d7",
                        "VpcId": "vpc-12312312",
                        "Groups": [
                                {
                                        "GroupId": "sg-aabbccdd"
                                }
                        ],
                        "Tags": [
                                {
                                        "Value": "ClassicInstance2",
                                        "Key": "Name"
                                }
                        ]
                }
        ]
}
```

Questo esempio elenca tutte le istanze EC2 -Classic collegate e filtra la risposta per includere solo le istanze collegate a VPC vpc-88888888.

Comando:

```
aws ec2 describe-classic-link-instances --filter "Name=vpc-id,Values=vpc-88888888"
```

Output:

```
{
        "Instances": [
                {
                        "InstanceId": "i-1234567890abcdef0",
                        "VpcId": "vpc-88888888",
```

```
                        "Groups": [
                                {
                                        "GroupId": "sg-11122233"
                                }
                        ],
                        "Tags": [
                                {
                                        "Value": "ClassicInstance",
                                        "Key": "Name"
                                }
                        ]
                }
        ]
}
```

- Per i dettagli sull'API, consulta Command Reference. DescribeClassicLinkInstancesAWS CLI

`describe-client-vpn-authorization-rules`

Il seguente esempio di codice mostra come utilizzare`describe-client-vpn-authorization-rules`.

AWS CLI

Come descrivere le regole di autorizzazione per un endpoint VPN Client

L'esempio `describe-client-vpn-authorization-rules` seguente visualizza i dettagli relativi alle regole di autorizzazione per l'endpoint VPN Client specificato.

```
aws ec2 describe-client-vpn-authorization-rules \
    --client-vpn-endpoint-id cvpn-endpoint-123456789123abcde
```

Output:

```
{
    "AuthorizationRules": [
        {
            "ClientVpnEndpointId": "cvpn-endpoint-123456789123abcde",
            "GroupId": "",
            "AccessAll": true,
            "DestinationCidr": "0.0.0.0/0",
```

```
            "Status": {
                "Code": "active"
            }
        }
    ]
}
```

Per ulteriori informazioni, consulta Regole di autorizzazione nella Guida per l'amministratore di VPN Client di AWS .

- Per i dettagli sull'API, consulta DescribeClientVpnAuthorizationRules AWS CLICommand Reference.

`describe-client-vpn-connections`

Il seguente esempio di codice mostra come utilizzare`describe-client-vpn-connections`.

AWS CLI

Come descrivere le connessioni a un endpoint VPN Client

L'esempio `describe-client-vpn-connections` seguente visualizza i dettagli relativi alle connessioni client all'endpoint VPN Client specificato.

```
aws ec2 describe-client-vpn-connections \
    --client-vpn-endpoint-id cvpn-endpoint-123456789123abcde
```

Output:

```
{
    "Connections": [
        {
            "ClientVpnEndpointId": "cvpn-endpoint-123456789123abcde",
            "Timestamp": "2019-08-12 07:58:34",
            "ConnectionId": "cvpn-connection-0e03eb24267165acd",
            "ConnectionEstablishedTime": "2019-08-12 07:57:14",
            "IngressBytes": "32302",
            "EgressBytes": "5696",
            "IngressPackets": "332",
            "EgressPackets": "67",
            "ClientIp": "172.31.0.225",
```

```
            "CommonName": "client1.domain.tld",
            "Status": {
                "Code": "terminated"
            },
            "ConnectionEndTime": "2019-08-12 07:58:34"
        },
        {
            "ClientVpnEndpointId": "cvpn-endpoint-123456789123abcde",
            "Timestamp": "2019-08-12 08:02:54",
            "ConnectionId": "cvpn-connection-00668867a40f18253",
            "ConnectionEstablishedTime": "2019-08-12 08:02:53",
            "IngressBytes": "2951",
            "EgressBytes": "2611",
            "IngressPackets": "9",
            "EgressPackets": "6",
            "ClientIp": "172.31.0.226",
            "CommonName": "client1.domain.tld",
            "Status": {
                "Code": "active"
            },
            "ConnectionEndTime": "-"
        }
    ]
}
```

Per ulteriori informazioni, consulta Connessioni client nella Guida per l'amministratore di VPN Client di AWS .

- Per i dettagli sull'API, consulta DescribeClientVpnConnections AWS CLICommand Reference.

`describe-client-vpn-endpoints`

Il seguente esempio di codice mostra come utilizzare`describe-client-vpn-endpoints`.

AWS CLI

Come descrivere gli endpoint VPN Client

L'esempio `describe-client-vpn-endpoints` seguente mostra i dettagli di tutti gli endpoint VPN Client.

```
aws ec2 describe-client-vpn-endpoints
```

Output:


```
{
    "ClientVpnEndpoints": [
        {
            "ClientVpnEndpointId": "cvpn-endpoint-123456789123abcde",
            "Description": "Endpoint for Admin access",
            "Status": {
                "Code": "available"
            },
            "CreationTime": "2020-11-13T11:37:27",
            "DnsName": "*.cvpn-endpoint-123456789123abcde.prod.clientvpn.ap-
south-1.amazonaws.com",
            "ClientCidrBlock": "172.31.0.0/16",
            "DnsServers": [
                "8.8.8.8"
            ],
            "SplitTunnel": false,
            "VpnProtocol": "openvpn",
            "TransportProtocol": "udp",
            "VpnPort": 443,
            "ServerCertificateArn": "arn:aws:acm:ap-
south-1:123456789012:certificate/a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
            "AuthenticationOptions": [
                {
                    "Type": "certificate-authentication",
                    "MutualAuthentication": {
                        "ClientRootCertificateChain": "arn:aws:acm:ap-
south-1:123456789012:certificate/a1b2c3d4-5678-90ab-cdef-22222EXAMPLE"
                    }
                }
            ],
            "ConnectionLogOptions": {
                "Enabled": true,
                "CloudwatchLogGroup": "Client-vpn-connection-logs",
                "CloudwatchLogStream": "cvpn-endpoint-123456789123abcde-ap-
south-1-2020/11/13-FCD8HEMVaCcw"
            },
            "Tags": [
                {
                    "Key": "Name",
                    "Value": "Client VPN"
                }
            ],
```

```
            "SecurityGroupIds": [
                "sg-aabbcc11223344567"
            ],
            "VpcId": "vpc-a87f92c1",
            "SelfServicePortalUrl": "https://self-service.clientvpn.amazonaws.com/
endpoints/cvpn-endpoint-123456789123abcde",
            "ClientConnectOptions": {
                 "Enabled": false
            }
        }
    ]
}
```

Per ulteriori informazioni, consulta Endpoint Client VPN nella Guida per l'amministratore di VPN Client di AWS .

- Per i dettagli sull'API, consulta DescribeClientVpnEndpoints AWS CLICommand Reference.

`describe-client-vpn-routes`

Il seguente esempio di codice mostra come utilizzare`describe-client-vpn-routes`.

AWS CLI

Come descrivere le route per un endpoint VPN Client

L'esempio `describe-client-vpn-routes` seguente visualizza i dettagli delle route per l'endpoint VPN Client specificato.

```
aws ec2 describe-client-vpn-routes \
    --client-vpn-endpoint-id cvpn-endpoint-123456789123abcde
```

Output:

```
{
    "Routes": [
        {
            "ClientVpnEndpointId": "cvpn-endpoint-123456789123abcde",
            "DestinationCidr": "10.0.0.0/16",
            "TargetSubnet": "subnet-0123456789abcabca",
            "Type": "Nat",
            "Origin": "associate",
```

```
            "Status": {
                "Code": "active"
            },
            "Description": "Default Route"
        },
        {
            "ClientVpnEndpointId": "cvpn-endpoint-123456789123abcde",
            "DestinationCidr": "0.0.0.0/0",
            "TargetSubnet": "subnet-0123456789abcabca",
            "Type": "Nat",
            "Origin": "add-route",
            "Status": {
                "Code": "active"
            }
        }
    ]
}
```

Per ulteriori informazioni, consulta Route nella Guida per l'amministratore di VPN Client di AWS .

- Per i dettagli sull'API, consulta DescribeClientVpnRoutes AWS CLICommand Reference.

`describe-client-vpn-target-networks`

Il seguente esempio di codice mostra come utilizzare`describe-client-vpn-target-networks`.

AWS CLI

Come descrivere le reti di destinazione per un endpoint VPN Client

L'esempio `describe-client-vpn-target-networks` seguente mostra i dettagli relativi alle reti di destinazione per l'endpoint VPN Client specificato.

```
aws ec2 describe-client-vpn-target-networks \
    --client-vpn-endpoint-id cvpn-endpoint-123456789123abcde
```

Output:

```
{
    "ClientVpnTargetNetworks": [
        {
            "AssociationId": "cvpn-assoc-012e837060753dc3d",
```

```
            "VpcId": "vpc-11111222222333333",
            "TargetNetworkId": "subnet-0123456789abcabca",
            "ClientVpnEndpointId": "cvpn-endpoint-123456789123abcde",
            "Status": {
                "Code": "associating"
            },
            "SecurityGroups": [
                "sg-012345678910abcab"
            ]
        }
    ]
}
```

Per ulteriori informazioni, consulta Reti target nella Guida per l'amministratore di VPN Client di AWS .

- Per i dettagli sull'API, consulta DescribeClientVpnTargetNetworks AWS CLICommand Reference.

`describe-coip-pools`

Il seguente esempio di codice mostra come utilizzare`describe-coip-pools`.

AWS CLI

Come descrivere i pool di indirizzi IP di proprietà del cliente

L'`describe-coip-pools`esempio seguente descrive i pool di indirizzi IP di proprietà del cliente presenti nell'account AWS .

```
aws ec2 describe-coip-pools
```

Output:

```
{
    "CoipPools": [
        {
            "PoolId": "ipv4pool-coip-123a45678bEXAMPLE",
            "PoolCidrs": [
                "0.0.0.0/0"
            ],
            "LocalGatewayRouteTableId": "lgw-rtb-059615ef7dEXAMPLE",
```

```
            "PoolArn": "arn:aws:ec2:us-west-2:123456789012:coip-pool/ipv4pool-
coip-123a45678bEXAMPLE"
        }
    ]
}
```

Per ulteriori informazioni, consulta Indirizzi IP di proprietà del cliente nella Guida per l'utente di AWS Outposts.

- Per i dettagli sull'API, consulta AWS CLI Command DescribeCoipPoolsReference.

`describe-conversion-tasks`

Il seguente esempio di codice mostra come utilizzare`describe-conversion-tasks`.

AWS CLI

Come visualizzare lo stato di un'attività di conversione

Questo esempio restituisce lo stato di un'attività di conversione con l'ID import-i-ffvko 9js.

Comando:

```
aws ec2 describe-conversion-tasks --conversion-task-ids import-i-ffvko9js
```

Output:

```
{
    "ConversionTasks": [
        {
            "ConversionTaskId": "import-i-ffvko9js",
            "ImportInstance": {
                "InstanceId": "i-1234567890abcdef0",
                "Volumes": [
                    {
                        "Volume": {
                            "Id": "vol-049df61146c4d7901",
                            "Size": 16
                        },
                        "Status": "completed",
                        "Image": {
                            "Size": 1300687360,
```

```
                            "ImportManifestUrl": "https://s3.amazonaws.com/
myimportbucket/411443cd-d620-4f1c-9d66-13144EXAMPLE/RHEL5.vmdkmanifest.xml?
AWSAccessKeyId=AKIAIOSFODNN7EXAMPLE&Expires=140EXAMPLE&Signature=XYNhznHNgCqsjDxL9wRL
%2FJvEXAMPLE",
                            "Format": "VMDK"
                        },
                        "BytesConverted": 1300682960,
                        "AvailabilityZone": "us-east-1d"
                    }
                ]
            },
            "ExpirationTime": "2014-05-14T22:06:23Z",
            "State": "completed"
        }
    ]
}
```

- Per i dettagli sull'API, consulta DescribeConversionTasks AWS CLICommand Reference.

`describe-customer-gateways`

Il seguente esempio di codice mostra come utilizzare`describe-customer-gateways`.

AWS CLI

Come descrivere i gateway del cliente

Questo esempio descrive i gateway del cliente.

Comando:

```
aws ec2 describe-customer-gateways
```

Output:

```
{
    "CustomerGateways": [
        {
            "CustomerGatewayId": "cgw-b4dc3961",
            "IpAddress": "203.0.113.12",
            "State": "available",
            "Type": "ipsec.1",
            "BgpAsn": "65000"
```

```
        },
        {
            "CustomerGatewayId": "cgw-0e11f167",
            "IpAddress": "12.1.2.3",
            "State": "available",
            "Type": "ipsec.1",
            "BgpAsn": "65534"
        }
    ]
}
```

Come descrivere un gateway del cliente specifico

Questo esempio descrive il gateway del cliente specificato.

Comando:

```
aws ec2 describe-customer-gateways --customer-gateway-ids cgw-0e11f167
```

Output:

```
{
    "CustomerGateways": [
        {
            "CustomerGatewayId": "cgw-0e11f167",
            "IpAddress": "12.1.2.3",
            "State": "available",
            "Type": "ipsec.1",
            "BgpAsn": "65534"
        }
    ]
}
```

- Per i dettagli sull'API, consulta DescribeCustomerGateways AWS CLICommand Reference.

`describe-dhcp-options`

Il seguente esempio di codice mostra come utilizzare`describe-dhcp-options`.

AWS CLI

Esempio 1: come descrivere le opzioni DHCP

L'esempio `describe-dhcp-options` seguente recupera i dettagli delle opzioni DHCP in uso.

```
aws ec2 describe-dhcp-options
```

Output:

```
{
    "DhcpOptions": [
        {
            "DhcpConfigurations": [
                {
                    "Key": "domain-name",
                    "Values": [
                        {
                            "Value": "us-east-2.compute.internal"
                        }
                    ]
                },
                {
                    "Key": "domain-name-servers",
                    "Values": [
                        {
                            "Value": "AmazonProvidedDNS"
                        }
                    ]
                }
            ],
            "DhcpOptionsId": "dopt-19edf471",
            "OwnerId": "111122223333"
        },
        {
            "DhcpConfigurations": [
                {
                    "Key": "domain-name",
                    "Values": [
                        {
                            "Value": "us-east-2.compute.internal"
                        }
                    ]
                },
                {
                    "Key": "domain-name-servers",
                    "Values": [
```

```
                        {
                            "Value": "AmazonProvidedDNS"
                        }
                    ]
                }
            ],
            "DhcpOptionsId": "dopt-fEXAMPLE",
            "OwnerId": "111122223333"
        }
    ]
}
```

Per ulteriori informazioni, consulta Utilizzo dei set di opzioni DHCP nella Guida per l'utente di AWS VPC.

Esempio 2: come descrivere le opzioni DHCP e filtrare l'output

L'esempio `describe-dhcp-options` seguente descrive le opzioni DHCP e utilizza un filtro per restituire solo le opzioni DHCP che hanno `example.com` come server dei nomi di dominio. L'esempio utilizza il parametro `--query` per visualizzare solo le informazioni di configurazione e l'ID nell'output.

```
aws ec2 describe-dhcp-options \
    --filters Name=key,Values=domain-name-servers Name=value,Values=example.com \
    --query "DhcpOptions[*].[DhcpConfigurations,DhcpOptionsId]"
```

Output:

```
[
    [
        [
            {
                "Key": "domain-name",
                "Values": [
                    {
                        "Value": "example.com"
                    }
                ]
            },
            {
                "Key": "domain-name-servers",
                "Values": [
```

```
                    {
                        "Value": "172.16.16.16"
                    }
                ]
            }
        ],
        "dopt-00112334455667ab"
    ]
]
```

Per ulteriori informazioni, consulta Utilizzo dei set di opzioni DHCP nella Guida per l'utente di AWS VPC.

- Per i dettagli sull'API, consulta DescribeDhcpOptions AWS CLICommand Reference.

`describe-egress-only-internet-gateways`

Il seguente esempio di codice mostra come utilizzare`describe-egress-only-internet-gateways`.

AWS CLI

Come descrivere gateway Internet di sola uscita

Questo esempio descrive i gateway Internet di sola uscita.

Comando:

```
aws ec2 describe-egress-only-internet-gateways
```

Output:

```
{
  "EgressOnlyInternetGateways": [
      {
          "EgressOnlyInternetGatewayId": "eigw-015e0e244e24dfe8a",
          "Attachments": [
              {
                  "State": "attached",
                  "VpcId": "vpc-0c62a468"
              }
          ]
      }
```

```
  ]
}
```

- Per i dettagli sull'API, consulta DescribeEgressOnlyInternetGateways AWS CLICommand Reference.

`describe-elastic-gpus`

Il seguente esempio di codice mostra come utilizzare`describe-elastic-gpus`.

AWS CLI

Come descrivere una GPU elastica

Comando:

```
aws ec2 describe-elastic-gpus --elastic-gpu-
ids egpu-12345678901234567890abcdefghijkl
```

- Per i dettagli sull'API, consulta DescribeElasticGpus AWS CLICommand Reference.

`describe-export-image-tasks`

Il seguente esempio di codice mostra come utilizzare`describe-export-image-tasks`.

AWS CLI

Come monitorare un'attività di esportazione di immagini

L'esempio `describe-export-image-tasks` seguente controlla lo stato dell'attività di esportazione di immagini specificata. Il file di immagine risultante in Amazon S3 è `my-export-bucket/exports/export-ami-1234567890abcdef0.vmdk`.

```
aws ec2 describe-export-image-tasks \
    --export-image-task-ids export-ami-1234567890abcdef0
```

Output di un'attività di esportazione di immagini in corso.

```
{
    "ExportImageTasks": [
        {
```

```
            "ExportImageTaskId": "export-ami-1234567890abcdef0"
            "Progress": "21",
            "S3ExportLocation": {
                "S3Bucket": "my-export-bucket",
                "S3Prefix": "exports/"
            },
            "Status": "active",
            "StatusMessage": "updating"
        }
    ]
}
```

Output di un'attività di esportazione di immagini completata.

```
{
    "ExportImageTasks": [
        {
            "ExportImageTaskId": "export-ami-1234567890abcdef0"
            "S3ExportLocation": {
                "S3Bucket": "my-export-bucket",
                "S3Prefix": "exports/"
            },
            "Status": "completed"
        }
    ]
}
```

Per ulteriori informazioni, consulta Esportazione di una macchina virtuale da un'AMI nella Guida per l' Import/Export utente della macchina virtuale.

- Per i dettagli sull'API, vedere DescribeExportImageTasksin AWS CLI Command Reference.

`describe-export-tasks`

Il seguente esempio di codice mostra come utilizzare`describe-export-tasks`.

AWS CLI

Come elencare i dettagli di un'attività di esportazione di istanze

Questo esempio descrive l'attività di esportazione con ID export-i-fh 8sjjsq.

Comando:

```
aws ec2 describe-export-tasks --export-task-ids export-i-fh8sjjsq
```

Output:

```
{
    "ExportTasks": [
        {
            "State": "active",
            "InstanceExportDetails": {
                "InstanceId": "i-1234567890abcdef0",
                "TargetEnvironment": "vmware"
            },
            "ExportToS3Task": {
                "S3Bucket": "myexportbucket",
                "S3Key": "RHEL5export-i-fh8sjjsq.ova",
                "DiskImageFormat": "vmdk",
                "ContainerFormat": "ova"
            },
            "Description": "RHEL5 instance",
            "ExportTaskId": "export-i-fh8sjjsq"
        }
    ]
}
```

- Per i dettagli sull'API, consulta DescribeExportTasksCommand Reference.AWS CLI

`describe-fast-launch-images`

Il seguente esempio di codice mostra come utilizzare`describe-fast-launch-images`.

AWS CLI

Per descrivere i dettagli relativi a Windows AMIs configurati per un avvio più rapido

L'`describe-fast-launch-images`esempio seguente descrive i dettagli di ciascuno degli elementi dell' AMIs account configurati per un avvio più rapido, tra cui il tipo di risorsa, la configurazione dell'istantanea, i dettagli del modello di avvio, il numero massimo di avvii paralleli, l'ID proprietario dell'AMI, lo stato della configurazione di avvio rapido, il motivo per cui lo stato è stato modificato e l'ora in cui è avvenuta la modifica dello stato.

```
aws ec2 describe-fast-launch-images
```

Output:

```
{
    "FastLaunchImages": [
        {
            "ImageId": "ami-01234567890abcedf",
            "ResourceType": "snapshot",
            "SnapshotConfiguration": {},
            "LaunchTemplate": {
                "LaunchTemplateId": "lt-01234567890abcedf",
                "LaunchTemplateName": "EC2FastLaunchDefaultResourceCreation-
a8c6215d-94e6-441b-9272-dbd1f87b07e2",
                "Version": "1"
            },
            "MaxParallelLaunches": 6,
            "OwnerId": "0123456789123",
            "State": "enabled",
            "StateTransitionReason": "Client.UserInitiated",
            "StateTransitionTime": "2022-01-27T22:20:06.552000+00:00"
        }
    ]
}
```

Per ulteriori informazioni sulla configurazione di un'AMI Windows per un avvio più rapido, consulta Configura la tua AMI per un avvio più rapido nella Amazon EC2 User Guide.

- Per i dettagli sull'API, consulta Command DescribeFastLaunchImagesReference AWS CLI .

`describe-fast-snapshot-restores`

Il seguente esempio di codice mostra come utilizzare`describe-fast-snapshot-restores`.

AWS CLI

Come descrivere i ripristini rapidi degli snapshot

L'esempio `describe-fast-snapshot-restores` seguente visualizza i dettagli di tutti i ripristini rapidi degli snapshot con uno stato di `disabled`.

```
aws ec2 describe-fast-snapshot-restores \
    --filters Name=state,Values=disabled
```

Output:

```
{
    "FastSnapshotRestores": [
        {
            "SnapshotId": "snap-1234567890abcdef0",
            "AvailabilityZone": "us-west-2c",
            "State": "disabled",
            "StateTransitionReason": "Client.UserInitiated - Lifecycle state
 transition",
            "OwnerId": "123456789012",
            "EnablingTime": "2020-01-25T23:57:49.596Z",
            "OptimizingTime": "2020-01-25T23:58:25.573Z",
            "EnabledTime": "2020-01-25T23:59:29.852Z",
            "DisablingTime": "2020-01-26T00:40:56.069Z",
            "DisabledTime": "2020-01-26T00:41:27.390Z"
        }
    ]
}
```

L'esempio `describe-fast-snapshot-restores` seguente descrive tutti i ripristini rapidi degli snapshot.

```
aws ec2 describe-fast-snapshot-restores
```

- Per i dettagli sull'API, consulta DescribeFastSnapshotRestores AWS CLICommand Reference.

`describe-fleet-history`

Il seguente esempio di codice mostra come utilizzare`describe-fleet-history`.

AWS CLI

Per descrivere la storia EC2 della flotta

L'`describe-fleet-history`esempio seguente restituisce la cronologia della EC2 flotta specificata a partire dall'ora specificata. L'output è per un EC2 Fleet con due istanze in esecuzione.

```
aws ec2 describe-fleet-history \
    --fleet-id fleet-12a34b55-67cd-8ef9-ba9b-9208dEXAMPLE \
    --start-time 2020-09-01T00:00:00Z
```

Output:


```
{
    "HistoryRecords": [
        {
            "EventInformation": {
                "EventSubType": "submitted"
            },
            "EventType": "fleetRequestChange",
            "Timestamp": "2020-09-01T18:26:05.000Z"
        },
        {
            "EventInformation": {
                "EventSubType": "active"
            },
            "EventType": "fleetRequestChange",
            "Timestamp": "2020-09-01T18:26:15.000Z"
        },
        {
            "EventInformation": {
                "EventDescription": "t2.small, ami-07c8bc5c1ce9598c3, ...",
                "EventSubType": "progress"
            },
            "EventType": "fleetRequestChange",
            "Timestamp": "2020-09-01T18:26:17.000Z"
        },
        {
            "EventInformation": {
                "EventDescription": "{\"instanceType\":\"t2.small\", ...}",
                "EventSubType": "launched",
                "InstanceId": "i-083a1c446e66085d2"
            },
            "EventType": "instanceChange",
            "Timestamp": "2020-09-01T18:26:17.000Z"
        },
        {
            "EventInformation": {
                "EventDescription": "{\"instanceType\":\"t2.small\", ...}",
                "EventSubType": "launched",
                "InstanceId": "i-090db02406cc3c2d6"
            },
            "EventType": "instanceChange",
            "Timestamp": "2020-09-01T18:26:17.000Z"
        }
```

```
    ],
    "LastEvaluatedTime": "2020-09-01T19:10:19.000Z",
    "FleetId": "fleet-12a34b55-67cd-8ef9-ba9b-9208dEXAMPLE",
    "StartTime": "2020-08-31T23:53:20.000Z"
}
```

Per ulteriori informazioni, consulta Managing an EC2 Fleet nella Amazon Elastic Compute Cloud User Guide for Linux Instances.

- Per i dettagli sull'API, consulta AWS CLI Command DescribeFleetHistoryReference.

`describe-fleet-instances`

Il seguente esempio di codice mostra come utilizzare`describe-fleet-instances`.

AWS CLI

Per descrivere le istanze in esecuzione per un EC2 parco istanze

L'`describe-fleet-instances`esempio seguente descrive le istanze in esecuzione per il Fleet specificato EC2 .

```
aws ec2 describe-fleet-instances \
    --fleet-id 12a34b55-67cd-8ef9-ba9b-9208dEXAMPLE
```

Output:

```
{
    "ActiveInstances": [
        {
            "InstanceId": "i-090db02406cc3c2d6",
            "InstanceType": "t2.small",
            "SpotInstanceRequestId": "sir-a43gtpfk",
            "InstanceHealth": "healthy"
        },
        {
            "InstanceId": "i-083a1c446e66085d2",
            "InstanceType": "t2.small",
            "SpotInstanceRequestId": "sir-iwcit2nj",
            "InstanceHealth": "healthy"
        }
```

```
    ],
    "FleetId": "fleet-12a34b55-67cd-8ef9-ba9b-9208dEXAMPLE"
}
```

Per ulteriori informazioni, consulta Managing an EC2 Fleet nella Amazon Elastic Compute Cloud User Guide for Linux Instances.

- Per i dettagli sull'API, consulta AWS CLI Command DescribeFleetInstancesReference.

`describe-fleets`

Il seguente esempio di codice mostra come utilizzare`describe-fleets`.

AWS CLI

Per descrivere una EC2 flotta

L'`describe-fleets`esempio seguente descrive la EC2 flotta specificata.

```
aws ec2 describe-fleets \
    --fleet-ids fleet-12a34b55-67cd-8ef9-ba9b-9208dEXAMPLE
```

Output:

```
{
    "Fleets": [
        {
            "ActivityStatus": "pending_fulfillment",
            "CreateTime": "2020-09-01T18:26:05.000Z",
            "FleetId": "fleet-12a34b55-67cd-8ef9-ba9b-9208dEXAMPLE",
            "FleetState": "active",
            "ExcessCapacityTerminationPolicy": "termination",
            "FulfilledCapacity": 0.0,
            "FulfilledOnDemandCapacity": 0.0,
            "LaunchTemplateConfigs": [
                {
                    "LaunchTemplateSpecification": {
                        "LaunchTemplateId": "lt-0e632f2855a979cd5",
                        "Version": "1"
                    }
                }
            ],
```

```
            "TargetCapacitySpecification": {
                "TotalTargetCapacity": 2,
                "OnDemandTargetCapacity": 0,
                "SpotTargetCapacity": 2,
                "DefaultTargetCapacityType": "spot"
            },
            "TerminateInstancesWithExpiration": false,
            "Type": "maintain",
            "ReplaceUnhealthyInstances": false,
            "SpotOptions": {
                "AllocationStrategy": "lowestPrice",
                "InstanceInterruptionBehavior": "terminate",
                "InstancePoolsToUseCount": 1
            },
            "OnDemandOptions": {
                "AllocationStrategy": "lowestPrice"
            }
        }
    ]
}
```

Per ulteriori informazioni, consulta Managing an EC2 Fleet nella Amazon Elastic Compute Cloud User Guide for Linux Instances.

- Per i dettagli sull'API, consulta AWS CLI Command DescribeFleetsReference.

`describe-flow-logs`

Il seguente esempio di codice mostra come utilizzare`describe-flow-logs`.

AWS CLI

Esempio 1: come descrivere tutti i log di flusso

L'esempio `describe-flow-logs` seguente visualizza i dettagli di tutti i log di flusso.

```
aws ec2 describe-flow-logs
```

Output:

```
{
    "FlowLogs": [
```

```
        {
            "CreationTime": "2018-02-21T13:22:12.644Z",
            "DeliverLogsPermissionArn": "arn:aws:iam::123456789012:role/flow-logs-
role",
            "DeliverLogsStatus": "SUCCESS",
            "FlowLogId": "fl-aabbccdd112233445",
            "MaxAggregationInterval": 600,
            "FlowLogStatus": "ACTIVE",
            "LogGroupName": "FlowLogGroup",
            "ResourceId": "subnet-12345678901234567",
            "TrafficType": "ALL",
            "LogDestinationType": "cloud-watch-logs",
            "LogFormat": "${version} ${account-id} ${interface-id} ${srcaddr}
 ${dstaddr} ${srcport} ${dstport} ${protocol} ${packets} ${bytes} ${start} ${end}
 ${action} ${log-status}"
        },
        {
            "CreationTime": "2020-02-04T15:22:29.986Z",
            "DeliverLogsStatus": "SUCCESS",
            "FlowLogId": "fl-01234567890123456",
            "MaxAggregationInterval": 60,
            "FlowLogStatus": "ACTIVE",
            "ResourceId": "vpc-00112233445566778",
            "TrafficType": "ACCEPT",
            "LogDestinationType": "s3",
            "LogDestination": "arn:aws:s3:::my-flow-log-bucket/custom",
            "LogFormat": "${version} ${vpc-id} ${subnet-id} ${instance-id}
 ${interface-id} ${account-id} ${type} ${srcaddr} ${dstaddr} ${srcport} ${dstport}
 ${pkt-srcaddr} ${pkt-dstaddr} ${protocol} ${bytes} ${packets} ${start} ${end}
 ${action} ${tcp-flags} ${log-status}"
        }
    ]
}
```

Esempio 2: come descrivere un subset di log di flusso

L'`describe-flow-logs`esempio seguente utilizza un filtro per visualizzare i dettagli solo per i log di flusso che si trovano nel gruppo di log specificato in Amazon CloudWatch Logs.

```
aws ec2 describe-flow-logs \
    --filter "Name=log-group-name,Values=MyFlowLogs"
```

- Per i dettagli sull'API, consulta AWS CLI Command DescribeFlowLogsReference.

`describe-fpga-image-attribute`

Il seguente esempio di codice mostra come utilizzare`describe-fpga-image-attribute`.

AWS CLI

Come descrivere gli attributi di un'immagine FPGA di Amazon (AFI)

Questo esempio descrive le autorizzazioni di caricamento per l'AFI specificato.

Comando:

```
aws ec2 describe-fpga-image-attribute --fpga-image-id afi-0d123e123bfc85abc --
attribute loadPermission
```

Output:

```
{
  "FpgaImageAttribute": {
      "FpgaImageId": "afi-0d123e123bfc85abc",
      "LoadPermissions": [
          {
              "UserId": "123456789012"
          }
      ]
  }
}
```

- Per i dettagli sull'API, consulta DescribeFpgaImageAttribute AWS CLICommand Reference.

`describe-fpga-images`

Il seguente esempio di codice mostra come utilizzare`describe-fpga-images`.

AWS CLI

Come descrivere le immagini FPGA di Amazon (AFI)

Questo esempio descrive AFIs che sono di proprietà dell'account`123456789012`.

Comando:

```
aws ec2 describe-fpga-images --filters Name=owner-id,Values=123456789012
```

Output:

```
{
  "FpgaImages": [
      {
          "UpdateTime": "2017-12-22T12:09:14.000Z",
          "Name": "my-afi",
          "PciId": {
              "SubsystemVendorId": "0xfedd",
              "VendorId": "0x1d0f",
              "DeviceId": "0xf000",
              "SubsystemId": "0x1d51"
          },
          "FpgaImageGlobalId": "agfi-123cb27b5e84a0abc",
          "Public": false,
          "State": {
              "Code": "available"
          },
          "ShellVersion": "0x071417d3",
          "OwnerId": "123456789012",
          "FpgaImageId": "afi-0d123e123bfc85abc",
          "CreateTime": "2017-12-22T11:43:33.000Z",
          "Description": "my-afi"
      }
  ]
}
```

- Per i dettagli sull'API, consulta DescribeFpgaImages AWS CLICommand Reference.

`describe-host-reservation-offerings`

Il seguente esempio di codice mostra come utilizzare`describe-host-reservation-offerings`.

AWS CLI

Come descrivere le offerte per le prenotazioni degli host dedicati

Questo esempio descrive le prenotazioni di host dedicati per la famiglia di istanze M4 disponibili per l'acquisto.

Comando:

```
aws ec2 describe-host-reservation-offerings --filter Name=instance-family,Values=m4
```

Output:


```
{
  "OfferingSet": [
      {
          "HourlyPrice": "1.499",
          "OfferingId": "hro-03f707bf363b6b324",
          "InstanceFamily": "m4",
          "PaymentOption": "NoUpfront",
          "UpfrontPrice": "0.000",
          "Duration": 31536000
      },
      {
          "HourlyPrice": "1.045",
          "OfferingId": "hro-0ef9181cabdef7a02",
          "InstanceFamily": "m4",
          "PaymentOption": "NoUpfront",
          "UpfrontPrice": "0.000",
          "Duration": 94608000
      },
      {
          "HourlyPrice": "0.714",
          "OfferingId": "hro-04567a15500b92a51",
          "InstanceFamily": "m4",
          "PaymentOption": "PartialUpfront",
          "UpfrontPrice": "6254.000",
          "Duration": 31536000
      },
      {
          "HourlyPrice": "0.484",
          "OfferingId": "hro-0d5d7a9d23ed7fbfe",
          "InstanceFamily": "m4",
          "PaymentOption": "PartialUpfront",
          "UpfrontPrice": "12720.000",
          "Duration": 94608000
      },
      {
          "HourlyPrice": "0.000",
          "OfferingId": "hro-05da4108ca998c2e5",
```

```
            "InstanceFamily": "m4",
            "PaymentOption": "AllUpfront",
            "UpfrontPrice": "23913.000",
            "Duration": 94608000
        },
        {
            "HourlyPrice": "0.000",
            "OfferingId": "hro-0a9f9be3b95a3dc8f",
            "InstanceFamily": "m4",
            "PaymentOption": "AllUpfront",
            "UpfrontPrice": "12257.000",
            "Duration": 31536000
        }
    ]
}
```

- Per i dettagli sull'API, consulta [DescribeHostReservationOfferings AWS CLI](#)Command Reference.

`describe-host-reservations`

Il seguente esempio di codice mostra come utilizzare`describe-host-reservations`.

AWS CLI

Come descrivere le prenotazioni di host dedicati nell'account in uso

Questo esempio descrive le prenotazioni di host dedicati nell'account in uso.

Comando:

```
aws ec2 describe-host-reservations
```

Output:

```
{
    "HostReservationSet": [
        {
            "Count": 1,
            "End": "2019-01-10T12:14:09Z",
            "HourlyPrice": "1.499",
```

```
            "InstanceFamily": "m4",
            "OfferingId": "hro-03f707bf363b6b324",
            "PaymentOption": "NoUpfront",
            "State": "active",
            "HostIdSet": [
                "h-013abcd2a00cbd123"
            ],
            "Start": "2018-01-10T12:14:09Z",
            "HostReservationId": "hr-0d418a3a4ffc669ae",
            "UpfrontPrice": "0.000",
            "Duration": 31536000
        }
    ]
}
```

- Per i dettagli sull'API, consulta DescribeHostReservations AWS CLICommand Reference.

`describe-hosts`

Il seguente esempio di codice mostra come utilizzare`describe-hosts`.

AWS CLI

Come visualizzare i dettagli degli host dedicati

L'`describe-hosts`esempio seguente mostra i dettagli degli host `available` dedicati del tuo AWS account.

```
aws ec2 describe-hosts --filter "Name=state,Values=available"
```

Output:

```
{
    "Hosts": [
        {
            "HostId": "h-07879acf49EXAMPLE",
            "Tags": [
                {
                    "Value": "production",
                    "Key": "purpose"
                }
            ],
```

```
            "HostProperties": {
                "Cores": 48,
                "TotalVCpus": 96,
                "InstanceType": "m5.large",
                "Sockets": 2
            },
            "Instances": [],
            "State": "available",
            "AvailabilityZone": "eu-west-1a",
            "AvailableCapacity": {
                "AvailableInstanceCapacity": [
                    {
                        "AvailableCapacity": 48,
                        "InstanceType": "m5.large",
                        "TotalCapacity": 48
                    }
                ],
                "AvailableVCpus": 96
            },
            "HostRecovery": "on",
            "AllocationTime": "2019-08-19T08:57:44.000Z",
            "AutoPlacement": "off"
        }
    ]
}
```

Per ulteriori informazioni, consulta Visualizzazione di host dedicati nella Guida per l'utente di Amazon Elastic Compute Cloud per le istanze Linux.

- Per i dettagli sull'API, consulta DescribeHosts AWS CLICommand Reference.

`describe-iam-instance-profile-associations`

Il seguente esempio di codice mostra come utilizzare`describe-iam-instance-profile-associations`.

AWS CLI

Come descrivere le associazioni del profilo dell'istanza IAM

In questo esempio vengono descritte tutte le associazioni del profilo dell'istanza IAM.

Comando:

```
aws ec2 describe-iam-instance-profile-associations
```

Output:

```
{
  "IamInstanceProfileAssociations": [
      {
          "InstanceId": "i-09eb09efa73ec1dee",
          "State": "associated",
          "AssociationId": "iip-assoc-0db249b1f25fa24b8",
          "IamInstanceProfile": {
              "Id": "AIPAJVQN4F5WVLGCJDRGM",
              "Arn": "arn:aws:iam::123456789012:instance-profile/admin-role"
          }
      },
      {
          "InstanceId": "i-0402909a2f4dffd14",
          "State": "associating",
          "AssociationId": "iip-assoc-0d1ec06278d29f44a",
          "IamInstanceProfile": {
              "Id": "AGJAJVQN4F5WVLGCJABCM",
              "Arn": "arn:aws:iam::123456789012:instance-profile/user1-role"
          }
      }
   ]
}
```

- Per i dettagli sull'API, consulta DescribeIamInstanceProfileAssociations AWS CLICommand Reference.

`describe-id-format`

Il seguente esempio di codice mostra come utilizzare`describe-id-format`.

AWS CLI

Esempio 1: come descrivere il formato di ID di una risorsa

L'esempio `describe-id-format` seguente descrive il formato ID del gruppo di sicurezza.

```
aws ec2 describe-id-format \
```

```
    --resource security-group
```

Nell'output di esempio seguente, il valore `Deadline` indica che per questo tipo di risorsa il tempo massimo per il passaggio definitivo dal formato ID breve al formato ID lungo è scaduto alle 00:00 UTC del 15 agosto 2018.

```
{
    "Statuses": [
        {
            "Deadline": "2018-08-15T00:00:00.000Z",
            "Resource": "security-group",
            "UseLongIds": true
        }
    ]
}
```

Esempio 2: come descrivere il formato di ID di tutte le risorse

L'esempio `describe-id-format` seguente descrive il formato ID di tutti i tipi di risorse. Tutti i tipi di risorse che supportavano il formato ID breve sono stati convertiti al formato ID lungo.

```
aws ec2 describe-id-format
```

- Per i dettagli sull'API, consulta DescribeIdFormat AWS CLICommand Reference.

`describe-identity-id-format`

Il seguente esempio di codice mostra come utilizzare`describe-identity-id-format`.

AWS CLI

Come descrivere il formato di ID per un ruolo IAM

L'`describe-identity-id-format`esempio seguente descrive il formato ID ricevuto dalle istanze create dal ruolo IAM `EC2Role` nel tuo AWS account.

```
aws ec2 describe-identity-id-format \
    --principal-arn arn:aws:iam::123456789012:role/my-iam-role \
    --resource instance
```

L'output seguente indica che le istanze create da questo ruolo ricevono IDs in formato ID lungo.

```
{
    "Statuses": [
        {
            "Deadline": "2016-12-15T00:00:00Z",
            "Resource": "instance",
            "UseLongIds": true
        }
    ]
}
```

Come descrivere il formato di ID per un utente IAM

L'`describe-identity-id-format`esempio seguente descrive il formato ID ricevuto dagli snapshot creati dall'utente IAM `AdminUser` nel tuo AWS account.

```
aws ec2 describe-identity-id-format \
    --principal-arn arn:aws:iam::123456789012:user/AdminUser \
    --resource snapshot
```

L'output indica che le istantanee create da questo utente vengono ricevute IDs in formato ID lungo.

```
{
    "Statuses": [
        {
            "Deadline": "2016-12-15T00:00:00Z",
            "Resource": "snapshot",
            "UseLongIds": true
        }
    ]
}
```

- Per i dettagli sull'API, consulta DescribeIdentityIdFormat AWS CLICommand Reference.

`describe-image-attribute`

Il seguente esempio di codice mostra come utilizzare`describe-image-attribute`.

AWS CLI

Come descrivere le autorizzazioni di avvio di un'AMI

Questo esempio descrive le autorizzazioni di avvio per l'AMI specificata.

Comando:

```
aws ec2 describe-image-attribute --image-id ami-5731123e --
attribute launchPermission
```

Output:

```
{
    "LaunchPermissions": [
        {
            "UserId": "123456789012"
        }
    ],
    "ImageId": "ami-5731123e",
}
```

Come descrivere i codici prodotto di un'AMI

Questo esempio descrive i codici prodotto per l'AMI specificata. Nota che questa AMI non dispone di codici prodotto.

Comando:

```
aws ec2 describe-image-attribute --image-id ami-5731123e --attribute productCodes
```

Output:

```
{
    "ProductCodes": [],
    "ImageId": "ami-5731123e",
}
```

- Per i dettagli sull'API, consulta DescribeImageAttribute AWS CLICommand Reference.

`describe-images`

Il seguente esempio di codice mostra come utilizzare`describe-images`.

AWS CLI

Esempio 1: come descrivere un'AMI

Nell'esempio di `describe-images` seguente viene descritta l'AMI specificata nella regione specificata.

```
aws ec2 describe-images \
    --region us-east-1 \
    --image-ids ami-1234567890EXAMPLE
```

Output:

```
{
    "Images": [
        {
            "VirtualizationType": "hvm",
            "Description": "Provided by Red Hat, Inc.",
            "PlatformDetails": "Red Hat Enterprise Linux",
            "EnaSupport": true,
            "Hypervisor": "xen",
            "State": "available",
            "SriovNetSupport": "simple",
            "ImageId": "ami-1234567890EXAMPLE",
            "UsageOperation": "RunInstances:0010",
            "BlockDeviceMappings": [
                {
                    "DeviceName": "/dev/sda1",
                    "Ebs": {
                        "SnapshotId": "snap-111222333444aaabb",
                        "DeleteOnTermination": true,
                        "VolumeType": "gp2",
                        "VolumeSize": 10,
                        "Encrypted": false
                    }
                }
            ],
            "Architecture": "x86_64",
            "ImageLocation": "123456789012/RHEL-8.0.0_HVM-20190618-x86_64-1-Hourly2-
GP2",
            "RootDeviceType": "ebs",
            "OwnerId": "123456789012",
            "RootDeviceName": "/dev/sda1",
```

```
            "CreationDate": "2019-05-10T13:17:12.000Z",
            "Public": true,
            "ImageType": "machine",
            "Name": "RHEL-8.0.0_HVM-20190618-x86_64-1-Hourly2-GP2"
        }
    ]
}
```

Per ulteriori informazioni, consulta Amazon Machine Images (AMI) nella Amazon EC2 User Guide.

Esempio 2: descrivere AMIs in base ai filtri

L'`describe-images`esempio seguente descrive Windows AMIs fornito da Amazon e supportato da Amazon EBS.

```
aws ec2 describe-images \
    --owners amazon \
    --filters "Name=platform,Values=windows" "Name=root-device-type,Values=ebs"
```

Per un esempio dell'output di `describe-images`, consulta l'Esempio 1.

Per ulteriori esempi di utilizzo dei filtri, consulta Elencare e filtrare le risorse nella Amazon EC2 User Guide.

Esempio 3: descrivere in AMIs base ai tag

L'`describe-images`esempio seguente descrive tutto ciò AMIs che ha il tag`Type=Custom`. L'esempio utilizza il `--query` parametro per visualizzare solo l'AMI IDs.

```
aws ec2 describe-images \
    --filters "Name=tag:Type,Values=Custom" \
    --query 'Images[*].[ImageId]' \
    --output text
```

Output:

```
ami-1234567890EXAMPLE
ami-0abcdef1234567890
```

Per ulteriori esempi di utilizzo dei filtri di tag, consulta Lavorare con i tag nella Amazon EC2 User Guide.

- Per i dettagli sull'API, consulta DescribeImages AWS CLICommand Reference.

`describe-import-image-tasks`

Il seguente esempio di codice mostra come utilizzare`describe-import-image-tasks`.

AWS CLI

Come monitorare un'attività di importazione di immagini

L'esempio `describe-import-image-tasks` seguente controlla lo stato dell'attività di importazione di immagini specificata.

```
aws ec2 describe-import-image-tasks \
    --import-task-ids import-ami-1234567890abcdef0
```

Output di un'attività di importazione di immagini in corso.

```
{
    "ImportImageTasks": [
        {
            "ImportTaskId": "import-ami-1234567890abcdef0",
            "Progress": "28",
            "SnapshotDetails": [
                {
                    "DiskImageSize": 705638400.0,
                    "Format": "ova",
                    "Status": "completed",
                    "UserBucket": {
                        "S3Bucket": "my-import-bucket",
                        "S3Key": "vms/my-server-vm.ova"
                    }
                }
            ],
            "Status": "active",
            "StatusMessage": "converting"
        }
    ]
}
```

Output di un'attività di importazione di immagini completata. L'ID dell'AMI risultante è fornito da `ImageId`.

```
{
    "ImportImageTasks": [
        {
            "ImportTaskId": "import-ami-1234567890abcdef0",
            "ImageId": "ami-1234567890abcdef0",
            "SnapshotDetails": [
                {
                    "DiskImageSize": 705638400.0,
                    "Format": "ova",
                    "SnapshotId": "snap-1234567890abcdef0"
                    "Status": "completed",
                    "UserBucket": {
                        "S3Bucket": "my-import-bucket",
                        "S3Key": "vms/my-server-vm.ova"
                    }
                }
            ],
            "Status": "completed"
        }
    ]
}
```

- Per i dettagli sull'API, consulta DescribeImportImageTasks AWS CLICommand Reference.

`describe-import-snapshot-tasks`

Il seguente esempio di codice mostra come utilizzare`describe-import-snapshot-tasks`.

AWS CLI

Come monitorare un'attività di importazione di snapshot

L'esempio `describe-import-snapshot-tasks` seguente controlla lo stato dell'attività di importazione di snapshot specificata.

```
aws ec2 describe-import-snapshot-tasks \
    --import-task-ids import-snap-1234567890abcdef0
```

Output di un'attività di importazione di snapshot in corso:

```
{
    "ImportSnapshotTasks": [
```

```
        {
            "Description": "My server VMDK",
            "ImportTaskId": "import-snap-1234567890abcdef0",
            "SnapshotTaskDetail": {
                "Description": "My server VMDK",
                "DiskImageSize": "705638400.0",
                "Format": "VMDK",
                "Progress": "42",
                "Status": "active",
                "StatusMessage": "downloading/converting",
                "UserBucket": {
                    "S3Bucket": "my-import-bucket",
                    "S3Key": "vms/my-server-vm.vmdk"
                }
            }
        }
    ]
}
```

Output di un'attività di importazione di snapshot completata. L'ID dello snapshot risultante è fornito da `SnapshotId`.

```
{
    "ImportSnapshotTasks": [
        {
            "Description": "My server VMDK",
            "ImportTaskId": "import-snap-1234567890abcdef0",
            "SnapshotTaskDetail": {
                "Description": "My server VMDK",
                "DiskImageSize": "705638400.0",
                "Format": "VMDK",
                "SnapshotId": "snap-1234567890abcdef0"
                "Status": "completed",
                "UserBucket": {
                    "S3Bucket": "my-import-bucket",
                    "S3Key": "vms/my-server-vm.vmdk"
                }
            }
        }
    ]
}
```

- Per i dettagli sull'API, consulta DescribeImportSnapshotTasks AWS CLICommand Reference.

`describe-instance-attribute`

Il seguente esempio di codice mostra come utilizzare`describe-instance-attribute`.

AWS CLI

Come descrivere il tipo di istanza

Questo esempio descrive il tipo di istanza dell'istanza specificata.

Comando:

```
aws ec2 describe-instance-attribute --instance-id i-1234567890abcdef0 --
attribute instanceType
```

Output:

```
{
    "InstanceId": "i-1234567890abcdef0"
    "InstanceType": {
        "Value": "t1.micro"
    }
}
```

Per descrivere l' disableApiTermination attributo

Questo esempio descrive l'attributo `disableApiTermination` dell'istanza specificata.

Comando:

```
aws ec2 describe-instance-attribute --instance-id i-1234567890abcdef0 --
attribute disableApiTermination
```

Output:

```
{
"InstanceId": "i-1234567890abcdef0"
    "DisableApiTermination": {
        "Value": "false"
    }
}
```

Come descrivere la mappatura dei dispositivi a blocchi per un'istanza

Questo esempio descrive l'attributo `blockDeviceMapping` dell'istanza specificata.

Comando:

```
aws ec2 describe-instance-attribute --instance-id i-1234567890abcdef0 --
attribute blockDeviceMapping
```

Output:

```
{
    "InstanceId": "i-1234567890abcdef0"
    "BlockDeviceMappings": [
        {
            "DeviceName": "/dev/sda1",
            "Ebs": {
                "Status": "attached",
                "DeleteOnTermination": true,
                "VolumeId": "vol-049df61146c4d7901",
                "AttachTime": "2013-05-17T22:42:34.000Z"
            }
        },
        {
            "DeviceName": "/dev/sdf",
            "Ebs": {
                "Status": "attached",
                "DeleteOnTermination": false,
                "VolumeId": "vol-049df61146c4d7901",
                "AttachTime": "2013-09-10T23:07:00.000Z"
            }
        }
    ],
}
```

- Per i dettagli sull'API, consulta DescribeInstanceAttribute AWS CLICommand Reference.

`describe-instance-connect-endpoints`

Il seguente esempio di codice mostra come utilizzare`describe-instance-connect-endpoints`.

AWS CLI

Per descrivere un endpoint EC2 Instance Connect

L'`describe-instance-connect-endpoints`esempio seguente descrive l'endpoint EC2 Instance Connect specificato.

```
aws ec2 describe-instance-connect-endpoints \
    --region us-east-1 \
    --instance-connect-endpoint-ids eice-0123456789example
```

Output:

```
{
    "InstanceConnectEndpoints": [
        {
            "OwnerId": "111111111111",
            "InstanceConnectEndpointId": "eice-0123456789example",
            "InstanceConnectEndpointArn": "arn:aws:ec2:us-
east-1:111111111111:instance-connect-endpoint/eice-0123456789example",
            "State": "create-complete",
            "StateMessage": "",
            "DnsName": "eice-0123456789example.b67b86ba.ec2-instance-connect-
endpoint.us-east-1.amazonaws.com",
            "NetworkInterfaceIds": [
                "eni-0123456789example"
            ],
            "VpcId": "vpc-0123abcd",
            "AvailabilityZone": "us-east-1d",
            "CreatedAt": "2023-02-07T12:05:37+00:00",
            "SubnetId": "subnet-0123abcd",
            "Tags": []
        }
    ]
}
```

Per ulteriori informazioni, consulta Create an EC2 Instance Connect Endpoint nella Amazon EC2 User Guide.

- Per i dettagli sull'API, consulta DescribeInstanceConnectEndpoints AWS CLICommand Reference.

`describe-instance-credit-specifications`

Il seguente esempio di codice mostra come utilizzare`describe-instance-credit-specifications`.

AWS CLI

Come descrivere l'opzione di credito per l'utilizzo della CPU di una o più istanze

L'esempio `describe-instance-credit-specifications` seguente descrive l'opzione di credito CPU per l'istanza specificata.

```
aws ec2 describe-instance-credit-specifications \
    --instance-ids i-1234567890abcdef0
```

Output:

```
{
    "InstanceCreditSpecifications": [
        {
            "InstanceId": "i-1234567890abcdef0",
            "CpuCredits": "unlimited"
        }
    ]
}
```

Per ulteriori informazioni, consulta Lavora con istanze a prestazioni espandibili nella Amazon EC2 User Guide.

- Per i dettagli sull'API, consulta AWS CLI Command DescribeInstanceCreditSpecificationsReference.

`describe-instance-event-notification-attributes`

Il seguente esempio di codice mostra come utilizzare`describe-instance-event-notification-attributes`.

AWS CLI

Come descrivere i tag per le notifiche di eventi pianificati

L'esempio `describe-instance-event-notification-attributes` seguente descrive i tag da visualizzare nelle notifiche di eventi pianificati.

```
aws ec2 describe-instance-event-notification-attributes
```

Output:

```
{
    "InstanceTagAttribute": {
        "InstanceTagKeys": [],
        "IncludeAllTagsOfInstance": true
    }
}
```

Per ulteriori informazioni, consulta Eventi pianificati per le istanze nella Guida per l'utente di Amazon Elastic Compute Cloud per le istanze Linux.

- Per i dettagli sull'API, consulta DescribeInstanceEventNotificationAttributes AWS CLICommand Reference.

`describe-instance-event-windows`

Il seguente esempio di codice mostra come utilizzare`describe-instance-event-windows`.

AWS CLI

Esempio 1: come descrivere tutte le finestre di eventi

L'esempio `describe-instance-event-windows` seguente descrive tutte le finestre di eventi nella Regione specificata.

```
aws ec2 describe-instance-event-windows \
    --region us-east-1
```

Output:

```
{
    "InstanceEventWindows": [
        {
            "InstanceEventWindowId": "iew-0abcdef1234567890",
            "Name": "myEventWindowName",
            "CronExpression": "* 21-23 * * 2,3",
            "AssociationTarget": {
                "InstanceIds": [
                    "i-1234567890abcdef0",
                    "i-0598c7d356eba48d7"
                ],
                "Tags": [],
```

```
                "DedicatedHostIds": []
            },
            "State": "active",
            "Tags": []
        }

        ...

    ],
    "NextToken": "9d624e0c-388b-4862-a31e-a85c64fc1d4a"
}
```

Esempio 2: come descrivere una finestra di eventi specifica

L'esempio `describe-instance-event-windows` seguente descrive un evento specifico utilizzando il parametro `instance-event-window` per descrivere una finestra di eventi specifica.

```
aws ec2 describe-instance-event-windows \
    --region us-east-1 \
    --instance-event-window-ids iew-0abcdef1234567890
```

Output:

```
{
    "InstanceEventWindows": [
        {
            "InstanceEventWindowId": "iew-0abcdef1234567890",
            "Name": "myEventWindowName",
            "CronExpression": "* 21-23 * * 2,3",
            "AssociationTarget": {
                "InstanceIds": [
                    "i-1234567890abcdef0",
                    "i-0598c7d356eba48d7"
                ],
                "Tags": [],
                "DedicatedHostIds": []
            },
            "State": "active",
            "Tags": []
        }
}
```

Esempio 3: come descrivere le finestre di eventi che corrispondono a uno o più filtri

L'esempio `describe-instance-event-windows` seguente descrive le finestre di eventi che corrispondono a uno o più filtri tramite il parametro `filter`. Il filtro `instance-id` viene utilizzato per descrivere tutte le finestre di eventi associate all'istanza specificata. Quando viene utilizzato un filtro, si stabilisce una corrispondenza diretta. Tuttavia, il filtro `instance-id` è diverso. Se non esiste una corrispondenza diretta con l'ID istanza, viene restituito alle associazioni indirette con la finestra di eventi, ad esempio i tag dell'istanza o l'ID host dedicato (se l'istanza si trova su un host dedicato).

```
aws ec2 describe-instance-event-windows \
    --region us-east-1 \
    --filters Name=instance-id,Values=i-1234567890abcdef0 \
    --max-results 100 \
    --next-token <next-token-value>
```

Output:

```
{
    "InstanceEventWindows": [
        {
            "InstanceEventWindowId": "iew-0dbc0adb66f235982",
            "TimeRanges": [
                {
                    "StartWeekDay": "sunday",
                    "StartHour": 2,
                    "EndWeekDay": "sunday",
                    "EndHour": 8
                }
            ],
            "Name": "myEventWindowName",
            "AssociationTarget": {
                "InstanceIds": [],
                "Tags": [],
                "DedicatedHostIds": [
                    "h-0140d9a7ecbd102dd"
                ]
            },
            "State": "active",
            "Tags": []
        }
    ]
```

```
}
```

Nell'output di esempio, l'istanza si trova su un host dedicato, associato alla finestra di eventi.

Per i vincoli relativi alla finestra degli eventi, consulta Considerazioni nella Amazon EC2 User Guide.

- Per i dettagli sull'API, consulta Command DescribeInstanceEventWindowsReference AWS CLI .

`describe-instance-image-metadata`

Il seguente esempio di codice mostra come utilizzare`describe-instance-image-metadata`.

AWS CLI

Esempio 1: come descrivere i metadati AMI per tutte le istanze

L'`describe-instance-image-metadata`esempio seguente descrive i metadati AMI di tutte le istanze del tuo AWS account nella regione specificata.

```
aws ec2 describe-instance-image-metadata \
    --region us-east-1
```

Output:

```
{
    "InstanceImageMetadata": [
        {
            "InstanceId": "i-1234567890EXAMPLE",
            "InstanceType": "t2.micro",
            "LaunchTime": "2024-08-28T11:25:45+00:00",
            "AvailabilityZone": "us-east-1a",
            "State": {
                "Code": 16,
                "Name": "running"
            },
            "OwnerId": "123412341234",
            "Tags": [
                {
                    "Key": "MyTagName",
                    "Value": "my-tag-value"
                }
            ],
```

```
            "ImageMetadata": {
                "ImageId": "ami-0b752bf1df193a6c4",
                "Name": "al2023-ami-2023.5.20240819.0-kernel-6.1-x86_64",
                "OwnerId": "137112412989",
                "State": "available",
                "ImageOwnerAlias": "amazon",
                "CreationDate": "2023-01-25T17:20:40Z",
                "DeprecationTime": "2025-01-25T17:20:40Z",
                "IsPublic": true
            }
        }
    ],
    "NextToken": "...EXAMPLEwIAABAA2JHaFxLnEXAMPLE..."
}
```

Per ulteriori informazioni, consulta Amazon Machine Images in Amazon EC2 nella Amazon EC2 User Guide.

Esempio 2: come descrivere i metadati AMI per le istanze specificate

L'esempio `describe-instance-image-metadata` seguente descrive i metadati AMI per le istanze specificate.

```
aws ec2 describe-instance-image-metadata \
    --region us-east-1 \
    --instance-ids i-1234567890EXAMPLE i-0987654321EXAMPLE
```

Output:

```
{
    "InstanceImageMetadata": [
        {
            "InstanceId": "i-1234567890EXAMPLE",
            "InstanceType": "t2.micro",
            "LaunchTime": "2024-08-28T11:25:45+00:00",
            "AvailabilityZone": "us-east-1a",
            "State": {
                "Code": 16,
                "Name": "running"
            },
            "OwnerId": "123412341234",
            "Tags": [
                {
```

```
                    "Key": "MyTagName",
                    "Value": "my-tag-value"
                }
            ],
            "ImageMetadata": {
                "ImageId": "ami-0b752bf1df193a6c4",
                "Name": "al2023-ami-2023.5.20240819.0-kernel-6.1-x86_64",
                "OwnerId": "137112412989",
                "State": "available",
                "ImageOwnerAlias": "amazon",
                "CreationDate": "2023-01-25T17:20:40Z",
                "DeprecationTime": "2025-01-25T17:20:40Z",
                "IsPublic": true
            }
        },
        {
            "InstanceId": "i-0987654321EXAMPLE",
            "InstanceType": "t2.micro",
            "LaunchTime": "2024-08-28T11:25:45+00:00",
            "AvailabilityZone": "us-east-1a",
            "State": {
                "Code": 16,
                "Name": "running"
            },
            "OwnerId": "123412341234",
            "Tags": [
                {
                    "Key": "MyTagName",
                    "Value": "my-tag-value"
                }
            ],
            "ImageMetadata": {
                "ImageId": "ami-0b752bf1df193a6c4",
                "Name": "al2023-ami-2023.5.20240819.0-kernel-6.1-x86_64",
                "OwnerId": "137112412989",
                "State": "available",
                "ImageOwnerAlias": "amazon",
                "CreationDate": "2023-01-25T17:20:40Z",
                "DeprecationTime": "2025-01-25T17:20:40Z",
                "IsPublic": true
            }
        }
    ]
```

```
}
```

Per ulteriori informazioni, consulta Amazon Machine Images in Amazon EC2 nella Amazon EC2 User Guide.

Esempio 3: come descrivere i metadati AMI per le istanze basate sui filtri

L'esempio `describe-instance-image-metadata` seguente descrive i metadati AMI per le istanze `t2.nano` e `t2.micro` nella zona di disponibilità `us-east-1a`.

```
aws ec2 describe-instance-image-metadata \
    --region us-east-1 \
    --filters Name=availability-zone,Values=us-east-1a Name=instance-
type,Values=t2.nano,t2.micro
```

Output:

```
{
    "InstanceImageMetadata": [
        {
            "InstanceId": "i-1234567890EXAMPLE",
            "InstanceType": "t2.micro",
            "LaunchTime": "2024-08-28T11:25:45+00:00",
            "AvailabilityZone": "us-east-1a",
            "State": {
                "Code": 16,
                "Name": "running"
            },
            "OwnerId": "123412341234",
            "Tags": [
                {
                    "Key": "MyTagName",
                    "Value": "my-tag-value"
                }
            ],
            "ImageMetadata": {
                "ImageId": "ami-0b752bf1df193a6c4",
                "Name": "al2023-ami-2023.5.20240819.0-kernel-6.1-x86_64",
                "OwnerId": "137112412989",
                "State": "available",
                "ImageOwnerAlias": "amazon",
                "CreationDate": "2023-01-25T17:20:40Z",
                "DeprecationTime": "2025-01-25T17:20:40Z",
```

```
                "IsPublic": true
            }
        },
        {
            "InstanceId": "i-0987654321EXAMPLE",
            "InstanceType": "t2.micro",
            "LaunchTime": "2024-08-28T11:25:45+00:00",
            "AvailabilityZone": "us-east-1a",
            "State": {
                "Code": 16,
                "Name": "running"
            },
            "OwnerId": "123412341234",
            "Tags": [
                {
                    "Key": "MyTagName",
                    "Value": "my-tag-value"
                }
            ],
            "ImageMetadata": {
                "ImageId": "ami-0b752bf1df193a6c4",
                "Name": "al2023-ami-2023.5.20240819.0-kernel-6.1-x86_64",
                "OwnerId": "137112412989",
                "State": "available",
                "ImageOwnerAlias": "amazon",
                "CreationDate": "2023-01-25T17:20:40Z",
                "DeprecationTime": "2025-01-25T17:20:40Z",
                "IsPublic": true
            }
        }
    ],
    "NextToken": "...EXAMPLEV7ixRYHwIAABAA2JHaFxLnDAzpatfEXAMPLE..."
}
```


Per ulteriori informazioni, consulta Amazon Machine Images in Amazon EC2 nella Amazon EC2 User Guide.

- Per i dettagli sull'API, consulta DescribeInstanceImageMetadata AWS CLICommand Reference.

`describe-instance-status`

Il seguente esempio di codice mostra come utilizzare`describe-instance-status`.

AWS CLI

Come descrivere lo stato delle istanze

Nell'esempio di `describe-instance-status` seguente viene descritto lo stato attuale dell'istanza specificata.

```
aws ec2 describe-instance-status \
    --instance-ids i-1234567890abcdef0
```

Output:

```
{
    "InstanceStatuses": [
        {
            "InstanceId": "i-1234567890abcdef0",
            "InstanceState": {
                "Code": 16,
                "Name": "running"
            },
            "AvailabilityZone": "us-east-1d",
            "SystemStatus": {
                "Status": "ok",
                "Details": [
                    {
                        "Status": "passed",
                        "Name": "reachability"
                    }
                ]
            },
            "InstanceStatus": {
                "Status": "ok",
                "Details": [
                    {
                        "Status": "passed",
                        "Name": "reachability"
                    }
                ]
            }
        }
    ]
}
```

Per ulteriori informazioni, consulta Monitora lo stato delle tue istanze nella Amazon EC2 User Guide.

- Per i dettagli sull'API, consulta DescribeInstanceStatus AWS CLICommand Reference.

`describe-instance-topology`

Il seguente esempio di codice mostra come utilizzare`describe-instance-topology`.

AWS CLI

Come descrivere la topologia dell'istanza di tutte le istanze

L'esempio `describe-instance-topology` seguente descrive la topologia di tutte le istanze che corrispondono ai tipi di istanze supportati per questo comando.

```
aws ec2 describe-instance-topology \
    --region us-west-2
```

Output:

```
{
    "Instances": [
        {
            "InstanceId": "i-1111111111example",
            "InstanceType": "p4d.24xlarge",
            "GroupName": "my-ml-cpg",
            "NetworkNodes": [
                "nn-1111111111example",
                "nn-2222222222example",
                "nn-3333333333example"
            ],
            "ZoneId": "usw2-az2",
            "AvailabilityZone": "us-west-2a"
        },
        {
            "InstanceId": "i-2222222222example",
            "InstanceType": "p4d.24xlarge",
            "NetworkNodes": [
                "nn-1111111111example",
                "nn-2222222222example",
                "nn-3333333333example"
```

```
            ],
            "ZoneId": "usw2-az2",
            "AvailabilityZone": "us-west-2a"
        },
        {
            "InstanceId": "i-3333333333example",
            "InstanceType": "trn1.32xlarge",
            "NetworkNodes": [
                "nn-1212121212example",
                "nn-1211122211example",
                "nn-1311133311example"
            ],
            "ZoneId": "usw2-az4",
            "AvailabilityZone": "us-west-2d"
        },
        {
            "InstanceId": "i-444444444example",
            "InstanceType": "trn1.2xlarge",
            "NetworkNodes": [
                "nn-1111111111example",
                "nn-5434334334example",
                "nn-1235301234example"
            ],
            "ZoneId": "usw2-az2",
            "AvailabilityZone": "us-west-2a"
        }
    ],
    "NextToken": "SomeEncryptedToken"
}
```

Per ulteriori informazioni, inclusi altri esempi, consulta la topologia delle EC2 istanze Amazon nella Amazon EC2 User Guide.

- Per i dettagli sull'API, consulta DescribeInstanceTopology AWS CLICommand Reference.

`describe-instance-type-offerings`

Il seguente esempio di codice mostra come utilizzare`describe-instance-type-offerings`.

AWS CLI

Esempio 1: come elencare i tipi di istanze offerti in una Regione

L'`describe-instance-type-offerings`esempio seguente elenca i tipi di istanza offerti nella regione configurata come regione predefinita per la AWS CLI.

```
aws ec2 describe-instance-type-offerings
```

Per elencare i tipi di istanza offerti in una Regione differente, specifica la Regione utilizzando il parametro `--region`.

```
aws ec2 describe-instance-type-offerings \
    --region us-east-2
```

Output:

```
{
  "InstanceTypeOfferings": [
      {
          "InstanceType": "m5.2xlarge",
          "LocationType": "region",
          "Location": "us-east-2"
      },
      {
          "InstanceType": "t3.micro",
          "LocationType": "region",
          "Location": "us-east-2"
      },
      ...
  ]
}
```

Esempio 2: come elencare i tipi di istanza offerti in una zona di disponibilità

L'esempio `describe-instance-type-offerings` seguente visualizza i tipi di istanza offerti nella zona di disponibilità specificata. La zona di disponibilità deve trovarsi nella Regione specificata.

```
aws ec2 describe-instance-type-offerings \
    --location-type availability-zone \
    --filters Name=location,Values=us-east-2a \
    --region us-east-2
```

Esempio 3: come verificare se un tipo di istanza è supportato

Il comando `describe-instance-type-offerings` seguente indica se il tipo di istanza `c5.xlarge` è supportato nella Regione specificata.

```
aws ec2 describe-instance-type-offerings \
    --filters Name=instance-type,Values=c5.xlarge \
    --region us-east-2
```

L'esempio `describe-instance-type-offerings` seguente elenca tutti i tipi di istanza C5 supportati nella Regione specificata.

```
aws ec2 describe-instance-type-offerings \
    --filters Name=instance-type,Values=c5* \
    --query "InstanceTypeOfferings[].InstanceType" \
    --region us-east-2
```

Output:

```
[
    "c5d.12xlarge",
    "c5d.9xlarge",
    "c5n.xlarge",
    "c5.xlarge",
    "c5d.metal",
    "c5n.metal",
    "c5.large",
    "c5d.2xlarge",
    "c5n.4xlarge",
    "c5.2xlarge",
    "c5n.large",
    "c5n.9xlarge",
    "c5d.large",
    "c5.18xlarge",
    "c5d.18xlarge",
    "c5.12xlarge",
    "c5n.18xlarge",
    "c5.metal",
    "c5d.4xlarge",
    "c5.24xlarge",
    "c5d.xlarge",
    "c5n.2xlarge",
    "c5d.24xlarge",
    "c5.9xlarge",
```

```
    "c5.4xlarge"
]
```

- Per i dettagli sull'API, consulta DescribeInstanceTypeOfferings AWS CLICommand Reference.

`describe-instance-types`

Il seguente esempio di codice mostra come utilizzare`describe-instance-types`.

AWS CLI

Esempio 1: come descrivere un tipo di istanza

Nell'esempio di `describe-instance-types` seguente vengono visualizzati i dettagli del tipo di istanza specificato.

```
aws ec2 describe-instance-types \
    --instance-types t2.micro
```

Output:

```
{
    "InstanceTypes": [
        {
            "InstanceType": "t2.micro",
            "CurrentGeneration": true,
            "FreeTierEligible": true,
            "SupportedUsageClasses": [
                "on-demand",
                "spot"
            ],
            "SupportedRootDeviceTypes": [
                "ebs"
            ],
            "BareMetal": false,
            "Hypervisor": "xen",
            "ProcessorInfo": {
                "SupportedArchitectures": [
                    "i386",
                    "x86_64"
                ],
                "SustainedClockSpeedInGhz": 2.5
```

```
            },
            "VCpuInfo": {
                "DefaultVCpus": 1,
                "DefaultCores": 1,
                "DefaultThreadsPerCore": 1,
                "ValidCores": [
                    1
                ],
                "ValidThreadsPerCore": [
                    1
                ]
            },
            "MemoryInfo": {
                "SizeInMiB": 1024
            },
            "InstanceStorageSupported": false,
            "EbsInfo": {
                "EbsOptimizedSupport": "unsupported",
                "EncryptionSupport": "supported"
            },
            "NetworkInfo": {
                "NetworkPerformance": "Low to Moderate",
                "MaximumNetworkInterfaces": 2,
                "Ipv4AddressesPerInterface": 2,
                "Ipv6AddressesPerInterface": 2,
                "Ipv6Supported": true,
                "EnaSupport": "unsupported"
            },
            "PlacementGroupInfo": {
                "SupportedStrategies": [
                    "partition",
                    "spread"
                ]
            },
            "HibernationSupported": false,
            "BurstablePerformanceSupported": true,
            "DedicatedHostsSupported": false,
            "AutoRecoverySupported": true
        }
    ]
}
```

Per ulteriori informazioni, consulta Tipi di istanza nella Guida per l'utente di Amazon Elastic Compute Cloud per le istanze Linux.

Esempio 2: come filtrare i tipi di istanza disponibili

È possibile specificare un filtro per rifinire i risultati in base ai tipi di istanza che hanno una caratteristica specifica. Nell'esempio di `describe-instance-types` seguente vengono elencati i tipi di istanza che supportano l'ibernazione.

```
aws ec2 describe-instance-types \
    --filters Name=hibernation-supported,Values=true --query
 'InstanceTypes[*].InstanceType'
```

Output:

```
[
    "m5.8xlarge",
    "r3.large",
    "c3.8xlarge",
    "r5.large",
    "m4.4xlarge",
    "c4.large",
    "m5.xlarge",
    "m4.xlarge",
    "c3.large",
    "c4.8xlarge",
    "c4.4xlarge",
    "c5.xlarge",
    "c5.12xlarge",
    "r5.4xlarge",
    "c5.4xlarge"
]
```

Per ulteriori informazioni, consulta Tipi di istanza nella Guida per l'utente di Amazon Elastic Compute Cloud per le istanze Linux.

- Per i dettagli sull'API, consulta DescribeInstanceTypes AWS CLICommand Reference.

`describe-instances`

Il seguente esempio di codice mostra come utilizzare`describe-instances`.

AWS CLI

Esempio 1: come descrivere un'istanza

Nell'esempio di `describe-instances` seguente viene descritta l'istanza specificata.

```
aws ec2 describe-instances \
    --instance-ids i-1234567890abcdef0
```

Output:

```
{
    "Reservations": [
        {
            "Groups": [],
            "Instances": [
                {
                    "AmiLaunchIndex": 0,
                    "ImageId": "ami-0abcdef1234567890",
                    "InstanceId": "i-1234567890abcdef0",
                    "InstanceType": "t3.nano",
                    "KeyName": "my-key-pair",
                    "LaunchTime": "2022-11-15T10:48:59+00:00",
                    "Monitoring": {
                        "State": "disabled"
                    },
                    "Placement": {
                        "AvailabilityZone": "us-east-2a",
                        "GroupName": "",
                        "Tenancy": "default"
                    },
                    "PrivateDnsName": "ip-10-0-0-157.us-east-2.compute.internal",
                    "PrivateIpAddress": "10-0-0-157",
                    "ProductCodes": [],
                    "PublicDnsName": "ec2-34-253-223-13.us-
east-2.compute.amazonaws.com",
                    "PublicIpAddress": "34.253.223.13",
                    "State": {
                        "Code": 16,
                        "Name": "running"
                    },
                    "StateTransitionReason": "",
                    "SubnetId": "subnet-04a636d18e83cfacb",
```

```
                    "VpcId": "vpc-1234567890abcdef0",
                    "Architecture": "x86_64",
                    "BlockDeviceMappings": [
                        {
                            "DeviceName": "/dev/xvda",
                            "Ebs": {
                                "AttachTime": "2022-11-15T10:49:00+00:00",
                                "DeleteOnTermination": true,
                                "Status": "attached",
                                "VolumeId": "vol-02e6ccdca7de29cf2"
                            }
                        }
                    ],
                    "ClientToken": "1234abcd-1234-abcd-1234-d46a8903e9bc",
                    "EbsOptimized": true,
                    "EnaSupport": true,
                    "Hypervisor": "xen",
                    "IamInstanceProfile": {
                        "Arn": "arn:aws:iam::111111111111:instance-profile/
AmazonSSMRoleForInstancesQuickSetup",
                        "Id": "11111111111111111111"
                    },
                    "NetworkInterfaces": [
                        {
                            "Association": {
                                "IpOwnerId": "amazon",
                                "PublicDnsName": "ec2-34-253-223-13.us-
east-2.compute.amazonaws.com",
                                "PublicIp": "34.253.223.13"
                            },
                            "Attachment": {
                                "AttachTime": "2022-11-15T10:48:59+00:00",
                                "AttachmentId": "eni-attach-1234567890abcdefg",
                                "DeleteOnTermination": true,
                                "DeviceIndex": 0,
                                "Status": "attached",
                                "NetworkCardIndex": 0
                            },
                            "Description": "",
                            "Groups": [
                                {
                                    "GroupName": "launch-wizard-146",
                                    "GroupId": "sg-1234567890abcdefg"
                                }
```

```
                            ],
                            "Ipv6Addresses": [],
                            "MacAddress": "00:11:22:33:44:55",
                            "NetworkInterfaceId": "eni-1234567890abcdefg",
                            "OwnerId": "104024344472",
                            "PrivateDnsName": "ip-10-0-0-157.us-
east-2.compute.internal",
                            "PrivateIpAddress": "10-0-0-157",
                            "PrivateIpAddresses": [
                                {
                                    "Association": {
                                        "IpOwnerId": "amazon",
                                        "PublicDnsName": "ec2-34-253-223-13.us-
east-2.compute.amazonaws.com",
                                        "PublicIp": "34.253.223.13"
                                    },
                                    "Primary": true,
                                    "PrivateDnsName": "ip-10-0-0-157.us-
east-2.compute.internal",
                                    "PrivateIpAddress": "10-0-0-157"
                                }
                            ],
                            "SourceDestCheck": true,
                            "Status": "in-use",
                            "SubnetId": "subnet-1234567890abcdefg",
                            "VpcId": "vpc-1234567890abcdefg",
                            "InterfaceType": "interface"
                        }
                    ],
                    "RootDeviceName": "/dev/xvda",
                    "RootDeviceType": "ebs",
                    "SecurityGroups": [
                        {
                            "GroupName": "launch-wizard-146",
                            "GroupId": "sg-1234567890abcdefg"
                        }
                    ],
                    "SourceDestCheck": true,
                    "Tags": [
                        {
                            "Key": "Name",
                            "Value": "my-instance"
                        }
                    ],
```

```
                    "VirtualizationType": "hvm",
                    "CpuOptions": {
                        "CoreCount": 1,
                        "ThreadsPerCore": 2
                    },
                    "CapacityReservationSpecification": {
                        "CapacityReservationPreference": "open"
                    },
                    "HibernationOptions": {
                        "Configured": false
                    },
                    "MetadataOptions": {
                        "State": "applied",
                        "HttpTokens": "optional",
                        "HttpPutResponseHopLimit": 1,
                        "HttpEndpoint": "enabled",
                        "HttpProtocolIpv6": "disabled",
                        "InstanceMetadataTags": "enabled"
                    },
                    "EnclaveOptions": {
                        "Enabled": false
                    },
                    "PlatformDetails": "Linux/UNIX",
                    "UsageOperation": "RunInstances",
                    "UsageOperationUpdateTime": "2022-11-15T10:48:59+00:00",
                    "PrivateDnsNameOptions": {
                        "HostnameType": "ip-name",
                        "EnableResourceNameDnsARecord": true,
                        "EnableResourceNameDnsAAAARecord": false
                    },
                    "MaintenanceOptions": {
                        "AutoRecovery": "default"
                    }
                }
            ],
            "OwnerId": "111111111111",
            "ReservationId": "r-1234567890abcdefg"
        }
    ]
}
```

Esempio 2: come filtrare per istanze secondo il tipo specificato

Nell'esempio di `describe-instances` seguente vengono utilizzati filtri per rifinire i risultati in base alle istanze del tipo specificato.

```
aws ec2 describe-instances \
    --filters Name=instance-type,Values=m5.large
```

Per un output di esempio, consulta l'Esempio 1.

Per ulteriori informazioni, consulta Elenca e filtra utilizzando la CLI nella Amazon EC2 User Guide.

Esempio 3: come filtrare per istanze secondo il tipo e la zona di disponibilità specificati

Nell'esempio di `describe-instances` seguente vengono utilizzati più filtri per rifinire i risultati in base alle istanze del tipo specificato che si trovano anche nella zona di disponibilità specificata.

```
aws ec2 describe-instances \
    --filters Name=instance-type,Values=t2.micro,t3.micro Name=availability-
zone,Values=us-east-2c
```

Per un output di esempio, vedi l'Esempio 1.

Esempio 4: come filtrare per istanze secondo il tipo e la zona di disponibilità specificati utilizzando un file JSON

Nell'esempio di `describe-instances` seguente viene utilizzato un file di input JSON per eseguire gli stessi filtri definiti nell'esempio precedente. Quando i filtri diventano più complicati, può essere più facile specificarli in un file JSON.

```
aws ec2 describe-instances \
    --filters file://filters.json
```

Contenuto di `filters.json`:

```
[
    {
        "Name": "instance-type",
        "Values": ["t2.micro", "t3.micro"]
    },
    {
        "Name": "availability-zone",
        "Values": ["us-east-2c"]
```

```
    }
]
```

Per un output di esempio, vedi l'Esempio 1.

Esempio 5: come filtrare per istanze secondo il tag Proprietario specificato

Nell'esempio di `describe-instances` seguente vengono utilizzati filtri tag per rifinire i risultati in base alle istanze che hanno un tag con la chiave tag specificata (Proprietario), a prescindere dal valore del tag.

```
aws ec2 describe-instances \
    --filters "Name=tag-key,Values=Owner"
```

Per un output di esempio, vedi l'Esempio 1.

Esempio 6: come filtrare per istanze secondo il valore tag my-team specificato

Nell'esempio di `describe-instances` seguente vengono utilizzati filtri tag per rifinire i risultati in base alle istanze che hanno un tag con il valore tag specificato (my-team), a prescindere dalla chiave del tag.

```
aws ec2 describe-instances \
    --filters "Name=tag-value,Values=my-team"
```

Per un output di esempio, vedi l'Esempio 1.

Esempio 7: come filtrare le istanze secondo i valori tag Proprietario e my-team specificati

Nell'esempio di `describe-instances` seguente vengono utilizzati filtri tag per rifinire i risultati in base alle istanze che hanno il tag specificato (Proprietario=my-team).

```
aws ec2 describe-instances \
    --filters "Name=tag:Owner,Values=my-team"
```

Per un output di esempio, vedi l'Esempio 1.

Esempio 8: per visualizzare solo l'istanza e la sottorete IDs per tutte le istanze

`describe-instances`Gli esempi seguenti utilizzano il `--query` parametro per visualizzare solo l'istanza e la sottorete IDs per tutte le istanze, in formato JSON.

Linux e macOS:

```
aws ec2 describe-instances \
    --query 'Reservations[*].Instances[*].{Instance:InstanceId,Subnet:SubnetId}' \
    --output json
```

Windows:

```
aws ec2 describe-instances ^
    --query "Reservations[*].Instances[*].{Instance:InstanceId,Subnet:SubnetId}" ^
    --output json
```

Output:

```
[
    {
        "Instance": "i-057750d42936e468a",
        "Subnet": "subnet-069beee9b12030077"
    },
    {
        "Instance": "i-001efd250faaa6ffa",
        "Subnet": "subnet-0b715c6b7db68927a"
    },
    {
        "Instance": "i-027552a73f021f3bd",
        "Subnet": "subnet-0250c25a1f4e15235"
    }
    ...
]
```

Esempio 9: per filtrare le istanze del tipo specificato e visualizzare solo le relative istanze IDs

L'`describe-instances`esempio seguente utilizza i filtri per limitare i risultati alle istanze del tipo specificato e il `--query` parametro per visualizzare solo l'istanza. IDs

```
aws ec2 describe-instances \
    --filters "Name=instance-type,Values=t2.micro" \
    --query "Reservations[*].Instances[*].[InstanceId]" \
    --output text
```

Output:

```
i-031c0dc19de2fb70c
i-00d8bff789a736b75
i-0b715c6b7db68927a
i-0626d4edd54f1286d
i-00b8ae04f9f99908e
i-0fc71c25d2374130c
```

Esempio 10: per filtrare le istanze del tipo specificato e visualizzare solo la relativa istanza IDs, la zona di disponibilità e il valore del tag specificato

Negli esempi di `describe-instances` seguenti vengono visualizzati l'ID dell'istanza, la zona di disponibilità e il valore del tag `Name` per le istanze che hanno un tag con il nome `tag-key`, in formato tabella.

Linux e macOS:

```
aws ec2 describe-instances \
    --filters Name=tag-key,Values=Name \
    --query 'Reservations[*].Instances[*].
{Instance:InstanceId,AZ:Placement.AvailabilityZone,Name:Tags[?Key==`Name`]|
[0].Value}' \
    --output table
```

Windows:

```
aws ec2 describe-instances ^
    --filters Name=tag-key,Values=Name ^
    --query "Reservations[*].Instances[*].
{Instance:InstanceId,AZ:Placement.AvailabilityZone,Name:Tags[?Key=='Name']|
[0].Value}" ^
    --output table
```

Output:

```
-------------------------------------------------------------
|                    DescribeInstances                      |
+--------------+-----------------------+--------------------+
|      AZ      |       Instance        |        Name        |
+--------------+-----------------------+--------------------+
|  us-east-2b  |  i-057750d42936e468a  |  my-prod-server    |
|  us-east-2a  |  i-001efd250faaa6ffa  |  test-server-1     |
```

```
|  us-east-2a  |  i-027552a73f021f3bd  |  test-server-2     |
+--------------+-----------------------+--------------------+
```

Esempio 11: come descrivere le istanze in un gruppo di posizionamento delle partizioni

L'esempio `describe-instances` seguente descrive l'istanza specificata. L'output include le informazioni di collocamento dell'istanza, che a loro volta comprendono il nome del gruppo di collocamento e il numero di partizioni per l'istanza.

```
aws ec2 describe-instances \
    --instance-ids i-0123a456700123456 \
    --query "Reservations[*].Instances[*].Placement"
```

Output:

```
[
    [
        {
            "AvailabilityZone": "us-east-1c",
            "GroupName": "HDFS-Group-A",
            "PartitionNumber": 3,
            "Tenancy": "default"
        }

    ]
]
```

Per ulteriori informazioni, consulta Descrivere le istanze in un gruppo di collocamento nella Amazon EC2 User Guide.

Esempio 12: come filtrare le istanze secondo il gruppo di collocamento e il numero di partizioni specificati

Nell'esempio di `describe-instances` seguente i risultati vengono filtrati solamente in base alle istanze con il gruppo di collocamento e il numero di partizioni specificati.

```
aws ec2 describe-instances \
    --filters "Name=placement-group-name,Values=HDFS-Group-A" "Name=placement-
partition-number,Values=7"
```

Di seguito vengono mostrate solo le informazioni rilevanti contenute nell'output.

```
"Instances": [
    {
        "InstanceId": "i-0123a456700123456",
        "InstanceType": "r4.large",
        "Placement": {
            "AvailabilityZone": "us-east-1c",
            "GroupName": "HDFS-Group-A",
            "PartitionNumber": 7,
            "Tenancy": "default"
        }
    },
    {
        "InstanceId": "i-9876a543210987654",
        "InstanceType": "r4.large",
        "Placement": {
            "AvailabilityZone": "us-east-1c",
            "GroupName": "HDFS-Group-A",
            "PartitionNumber": 7,
            "Tenancy": "default"
        }
    ],
```

Per ulteriori informazioni, consulta Descrivere le istanze in un gruppo di collocamento nella Amazon EC2 User Guide.

Esempio 13: come filtrare le istanze configurate per consentire l'accesso ai tag dai metadati dell'istanza

Nell'esempio di `describe-instances` seguente i risultati vengono filtrati solamente in base alle istanze configurate per consentire l'accesso ai tag dell'istanza dai metadati dell'istanza stessa.

```
aws ec2 describe-instances \
    --filters "Name=metadata-options.instance-metadata-tags,Values=enabled" \
    --query "Reservations[*].Instances[*].InstanceId" \
    --output text
```

Di seguito è riportato l'output previsto.

```
i-1234567890abcdefg
i-abcdefg1234567890
i-11111111aaaaaaaaa
i-aaaaaaaa111111111
```

Per ulteriori informazioni, consulta Lavora con i tag di istanza nei metadati delle istanze nella Amazon EC2 User Guide.

- Per i dettagli sull'API, consulta DescribeInstances AWS CLICommand Reference.

`describe-internet-gateways`

Il seguente esempio di codice mostra come utilizzare`describe-internet-gateways`.

AWS CLI

Come descrivere un gateway Internet

L'esempio `describe-internet-gateways` seguente descrive il gateway Internet specificato.

```
aws ec2 describe-internet-gateways \
    --internet-gateway-ids igw-0d0fb496b3EXAMPLE
```

Output:

```
{
    "InternetGateways": [
        {
            "Attachments": [
                {
                    "State": "available",
                    "VpcId": "vpc-0a60eb65b4EXAMPLE"
                }
            ],
            "InternetGatewayId": "igw-0d0fb496b3EXAMPLE",
            "OwnerId": "123456789012",
            "Tags": [
                {
                    "Key": "Name",
                    "Value": "my-igw"
                }
            ]
        }
    ]
}
```

Per ulteriori informazioni, consulta Gateway Internet nella Guida per l'utente di Amazon VPC.

- Per i dettagli sull'API, consulta DescribeInternetGateways AWS CLICommand Reference.

`describe-ipam-pools`

Il seguente esempio di codice mostra come utilizzare`describe-ipam-pools`.

AWS CLI

Come visualizzare i dettagli di un pool IPAM

L'esempio `describe-ipam-pools` seguente mostra i dettagli per i pool.

Linux:

```
aws ec2 describe-ipam-pools \
    --filters Name=owner-id,Values=123456789012 Name=ipam-scope-id,Values=ipam-scope-02fc38cd4c48e7d38
```

Windows:

```
aws ec2 describe-ipam-pools ^
    --filters Name=owner-id,Values=123456789012 Name=ipam-scope-id,Values=ipam-scope-02fc38cd4c48e7d38
```

Output:

```
{
    "IpamPools": [
        {
            "OwnerId": "123456789012",
            "IpamPoolId": "ipam-pool-02ec043a19bbe5d08",
            "IpamPoolArn": "arn:aws:ec2::123456789012:ipam-pool/ipam-pool-02ec043a19bbe5d08",
            "IpamScopeArn": "arn:aws:ec2::123456789012:ipam-scope/ipam-scope-02fc38cd4c48e7d38",
            "IpamScopeType": "private",
            "IpamArn": "arn:aws:ec2::123456789012:ipam/ipam-08440e7a3acde3908",
            "IpamRegion": "us-east-1",
            "Locale": "None",
            "PoolDepth": 1,
            "State": "create-complete",
            "AutoImport": true,
```

```
            "AddressFamily": "ipv4",
            "AllocationMinNetmaskLength": 16,
            "AllocationMaxNetmaskLength": 26,
            "AllocationDefaultNetmaskLength": 24,
            "AllocationResourceTags": [
                {
                    "Key": "Environment",
                    "Value": "Preprod"
                }
            ],
            "Tags": [
                {
                    "Key": "Name",
                    "Value": "Preprod pool"
                }
            ]
        }
    ]
}
```

- Per i dettagli sull'API, consulta DescribeIpamPools AWS CLICommand Reference.

`describe-ipam-resource-discoveries`

Il seguente esempio di codice mostra come utilizzare`describe-ipam-resource-discoveries`.

AWS CLI

Esempio 1: come visualizzare i dettagli completi delle Individuazioni di risorse

In questo esempio, sei un amministratore IPAM delegato che desidera creare e condividere un rilevamento delle risorse con l'amministratore IPAM di un'altra AWS organizzazione in modo che l'amministratore possa gestire e monitorare gli indirizzi IP delle risorse dell'organizzazione.

Questo esempio può essere utile se:

Si è cercato di creare un rilevamento di risorse, ma è stato visualizzato un errore che indica che è stato raggiunto il limite di 1. Ci si rende conto di aver già creato un rilevamento di risorse e si desidera visualizzarlo nell'account in uso. Si dispone di risorse in una Regione che non vengono rilevate dall'IPAM. Si desidera visualizzare le `--operating-regions` definite per la risorsa e assicurarti di aver aggiunto la Regione corretta come Regione operativa, in modo che le risorse presenti possano essere rilevate.

L'`describe-ipam-resource-discoveries`esempio seguente elenca i dettagli della scoperta delle risorse nel tuo account. AWS È possibile effettuare una sola ricerca di risorse per AWS regione.

```
aws ec2 describe-ipam-resource-discoveries \
    --region us-east-1
```

Output:

```
{
    "IpamResourceDiscoveries": [
        {
            "OwnerId": "149977607591",
            "IpamResourceDiscoveryId": "ipam-res-disco-0f8bdee9067137c0d",
            "IpamResourceDiscoveryArn": "arn:aws:ec2::149977607591:ipam-resource-
discovery/ipam-res-disco-0f8bdee9067137c0d",
            "IpamResourceDiscoveryRegion": "us-east-1",
            "OperatingRegions": [
                {
                    "RegionName": "us-east-1"
                }
            ],
            "IsDefault": false,
            "State": "create-complete",
            "Tags": []
    }
]
}
```

Per ulteriori informazioni, consulta Come integrare IPAM con account esterni alla tua organizzazione nella Guida per l'utente IPAM di Amazon VPC.

Esempio 2: Visualizza solo l'individuazione delle risorse IDs

L'`describe-ipam-resource-discoveries`esempio seguente elenca l'ID della scoperta delle risorse nel tuo AWS account. È possibile effettuare una sola ricerca di risorse per AWS regione.

```
aws ec2 describe-ipam-resource-discoveries \
    --query "IpamResourceDiscoveries[*].IpamResourceDiscoveryId" \
    --output text
```

Output:

```
ipam-res-disco-0481e39b242860333
```

Per ulteriori informazioni, consulta Come integrare IPAM con account esterni alla tua organizzazione nella Guida per l'utente IPAM di Amazon VPC.

- Per i dettagli sull'API, consulta DescribeIpamResourceDiscoveries AWS CLICommand Reference.

`describe-ipam-resource-discovery-associations`

Il seguente esempio di codice mostra come utilizzare`describe-ipam-resource-discovery-associations`.

AWS CLI

Come visualizzare tutte le associazioni di rilevamento delle risorse con l'IPAM in uso

In questo esempio, un amministratore delegato IPAM dispone di rilevamenti di risorse associati all'IPAM in uso per integrare altri account con l'IPAM in uso. IPAM non rileva le risorse nelle Regioni operative del rilevamento di risorse come previsto. Si vuole controllare lo stato e la situazione del rilevamento di risorse per assicurarsi che l'account utilizzato per crearla sia ancora attivo e che il rilevamento di risorse sia ancora condiviso.

`--region` deve essere la Regione di origine dell'IPAM.

L'`describe-ipam-resource-discovery-associations`esempio seguente elenca le associazioni di ricerca delle risorse nel tuo AWS account.

```
aws ec2 describe-ipam-resource-discovery-associations \
    --region us-east-1
```

Output:

```
{
    "IpamResourceDiscoveryAssociations": [
        {
            "OwnerId": "320805250157",
            "IpamResourceDiscoveryAssociationId": "ipam-res-disco-
assoc-05e6b45eca5bf5cf7",
            "IpamResourceDiscoveryAssociationArn": "arn:aws:ec2::320805250157:ipam-
resource-discovery-association/ipam-res-disco-assoc-05e6b45eca5bf5cf7",
```

```
            "IpamResourceDiscoveryId": "ipam-res-disco-0f4ef577a9f37a162",
            "IpamId": "ipam-005f921c17ebd5107",
            "IpamArn": "arn:aws:ec2::320805250157:ipam/ipam-005f921c17ebd5107",
            "IpamRegion": "us-east-1",
            "IsDefault": true,
            "ResourceDiscoveryStatus": "active",
            "State": "associate-complete",
            "Tags": []
        },
        {
            "OwnerId": "149977607591",
            "IpamResourceDiscoveryAssociationId": "ipam-res-disco-
assoc-0dfd21ae189ab5f62",
            "IpamResourceDiscoveryAssociationArn": "arn:aws:ec2::149977607591:ipam-
resource-discovery-association/ipam-res-disco-assoc-0dfd21ae189ab5f62",
            "IpamResourceDiscoveryId": "ipam-res-disco-0365d2977fc1672fe",
            "IpamId": "ipam-005f921c17ebd5107",
            "IpamArn": "arn:aws:ec2::149977607591:ipam/ipam-005f921c17ebd5107",
            "IpamRegion": "us-east-1",
            "IsDefault": false,
            "ResourceDiscoveryStatus": "active",
            "State": "create-complete",
            "Tags": []
        }
    ]
}
```

In questo esempio, dopo aver eseguito questo comando, si nota che è disponibile un rilevamento di risorse non predefinito (`"IsDefault": false ``) that is ``"ResourceDiscoveryStatus": "not-found"` e `"State": "create-complete"`. L'account del proprietario del rilevamento di risorse è stato chiuso. Se, in un altro caso, si nota che è `"ResourceDiscoveryStatus": "not-found"` e `"State": "associate-complete"`, ciò indica che si è verificata una delle seguenti situazioni:

Il rilevamento delle risorse è stato eliminato dal proprietario del rilevamento. Il proprietario del rilevamento delle risorse ha annullato la condivisione del rilevamento delle risorse.

Per ulteriori informazioni, consulta Come integrare IPAM con account esterni alla tua organizzazione nella Guida per l'utente IPAM di Amazon VPC.

- Per i dettagli sull'API, consulta DescribeIpamResourceDiscoveryAssociations AWS CLICommand Reference.

`describe-ipam-scopes`

Il seguente esempio di codice mostra come utilizzare`describe-ipam-scopes`.

AWS CLI

Come visualizzare i dettagli di un ambito IPAM

L'esempio `describe-ipam-scopes` seguente mostra i dettagli per gli ambiti.

```
aws ec2 describe-ipam-scopes \
    --filters Name=owner-id,Values=123456789012 Name=ipam-
id,Values=ipam-08440e7a3acde3908
```

Output:

```
{
    "IpamScopes": [
        {
            "OwnerId": "123456789012",
            "IpamScopeId": "ipam-scope-02fc38cd4c48e7d38",
            "IpamScopeArn": "arn:aws:ec2::123456789012:ipam-scope/ipam-
scope-02fc38cd4c48e7d38",
            "IpamArn": "arn:aws:ec2::123456789012:ipam/ipam-08440e7a3acde3908",
            "IpamRegion": "us-east-1",
            "IpamScopeType": "private",
            "IsDefault": true,
            "PoolCount": 2,
            "State": "create-complete",
            "Tags": []
        },
        {
            "OwnerId": "123456789012",
            "IpamScopeId": "ipam-scope-0b9eed026396dbc16",
            "IpamScopeArn": "arn:aws:ec2::123456789012:ipam-scope/ipam-
scope-0b9eed026396dbc16",
            "IpamArn": "arn:aws:ec2::123456789012:ipam/ipam-08440e7a3acde3908",
            "IpamRegion": "us-east-1",
            "IpamScopeType": "public",
            "IsDefault": true,
            "PoolCount": 0,
            "State": "create-complete",
            "Tags": []
        },
```

```
        {
            "OwnerId": "123456789012",
            "IpamScopeId": "ipam-scope-0f1aff29486355c22",
            "IpamScopeArn": "arn:aws:ec2::123456789012:ipam-scope/ipam-
scope-0f1aff29486355c22",
            "IpamArn": "arn:aws:ec2::123456789012:ipam/ipam-08440e7a3acde3908",
            "IpamRegion": "us-east-1",
            "IpamScopeType": "private",
            "IsDefault": false,
            "Description": "Example description",
            "PoolCount": 0,
            "State": "create-complete",
            "Tags": [
                {
                    "Key": "Name",
                    "Value": "Example name value"
                }
            ]
        }
    ]
}
```

- Per i dettagli sull'API, consulta DescribeIpamScopes AWS CLICommand Reference.

`describe-ipams`

Il seguente esempio di codice mostra come utilizzare`describe-ipams`.

AWS CLI

Come visualizzare i dettagli IPAM

L'esempio `describe-ipams` seguente mostra i dettagli di un'IPAM.

```
aws ec2 describe-ipams \
    --filters Name=owner-id,Values=123456789012
```

Output:

```
{
    "Ipams": [
        {
            "OwnerId": "123456789012",
```

```
            "IpamId": "ipam-08440e7a3acde3908",
            "IpamArn": "arn:aws:ec2::123456789012:ipam/ipam-08440e7a3acde3908",
            "IpamRegion": "us-east-1",
            "PublicDefaultScopeId": "ipam-scope-0b9eed026396dbc16",
            "PrivateDefaultScopeId": "ipam-scope-02fc38cd4c48e7d38",
            "ScopeCount": 3,
            "OperatingRegions": [
                {
                    "RegionName": "us-east-1"
                },
                {
                    "RegionName": "us-east-2"
                },
                {
                    "RegionName": "us-west-1"
                }
            ],
            "State": "create-complete",
            "Tags": [
                {
                    "Key": "Name",
                    "Value": "ExampleIPAM"
                }
            ]
        }
    ]
}
```

- Per i dettagli sull'API, consulta DescribeIpams AWS CLICommand Reference.

`describe-ipv6-pools`

Il seguente esempio di codice mostra come utilizzare`describe-ipv6-pools`.

AWS CLI

Per descrivere i tuoi pool di IPv6 indirizzi

L'`describe-ipv6-pools`esempio seguente mostra i dettagli di tutti i tuoi pool di IPv6 indirizzi.

```
aws ec2 describe-ipv6-pools
```

Output:

```
{
    "Ipv6Pools": [
        {
            "PoolId": "ipv6pool-ec2-012345abc12345abc",
            "PoolCidrBlocks": [
                {
                    "Cidr": "2001:db8:123::/48"
                }
            ],
            "Tags": [
                {
                    "Key": "pool-1",
                    "Value": "public"
                }
            ]
        }
    ]
}
```

- Per i dettagli sull'API, consulta DescribeIpv6Pools in AWS CLI Command Reference.

`describe-key-pairs`

Il seguente esempio di codice mostra come utilizzare`describe-key-pairs`.

AWS CLI

Come visualizzare una coppia di chiavi

Nell'esempio di `describe-key-pairs` seguente vengono visualizzate informazioni sulla coppia di chiavi specificata.

```
aws ec2 describe-key-pairs \
    --key-names my-key-pair
```

Output:

```
{
    "KeyPairs": [
        {
            "KeyPairId": "key-0b94643da6EXAMPLE",
```

```
            "KeyFingerprint":
 "1f:51:ae:28:bf:89:e9:d8:1f:25:5d:37:2d:7d:b8:ca:9f:f5:f1:6f",
            "KeyName": "my-key-pair",
            "KeyType": "rsa",
            "Tags": [],
            "CreateTime": "2022-05-27T21:51:16.000Z"
        }
    ]
}
```

Per ulteriori informazioni, consulta Descrivi le chiavi pubbliche nella Amazon EC2 User Guide.

- Per i dettagli sull'API, consulta DescribeKeyPairs AWS CLICommand Reference.

`describe-launch-template-versions`

Il seguente esempio di codice mostra come utilizzare`describe-launch-template-versions`.

AWS CLI

Come descrivere le versioni del modello di avvio

Questo esempio descrive le versioni del modello di avvio specificato.

Comando:

```
aws ec2 describe-launch-template-versions --launch-template-id lt-068f72b72934aff71
```

Output:

```
{
  "LaunchTemplateVersions": [
      {
          "LaunchTemplateId": "lt-068f72b72934aff71",
          "LaunchTemplateName": "Webservers",
          "VersionNumber": 3,
          "CreatedBy": "arn:aws:iam::123456789102:root",
          "LaunchTemplateData": {
              "KeyName": "kp-us-east",
              "ImageId": "ami-6057e21a",
              "InstanceType": "t2.small",
              "NetworkInterfaces": [
                  {
```

```
                    "SubnetId": "subnet-7b16de0c",
                    "DeviceIndex": 0,
                    "Groups": [
                        "sg-7c227019"
                    ]
                }
            ]
        },
        "DefaultVersion": false,
        "CreateTime": "2017-11-20T13:19:54.000Z"
    },
    {
        "LaunchTemplateId": "lt-068f72b72934aff71",
        "LaunchTemplateName": "Webservers",
        "VersionNumber": 2,
        "CreatedBy": "arn:aws:iam::123456789102:root",
        "LaunchTemplateData": {
            "KeyName": "kp-us-east",
            "ImageId": "ami-6057e21a",
            "InstanceType": "t2.medium",
            "NetworkInterfaces": [
                {
                    "SubnetId": "subnet-1a2b3c4d",
                    "DeviceIndex": 0,
                    "Groups": [
                        "sg-7c227019"
                    ]
                }
            ]
        },
        "DefaultVersion": false,
        "CreateTime": "2017-11-20T13:12:32.000Z"
    },
    {
        "LaunchTemplateId": "lt-068f72b72934aff71",
        "LaunchTemplateName": "Webservers",
        "VersionNumber": 1,
        "CreatedBy": "arn:aws:iam::123456789102:root",
        "LaunchTemplateData": {
            "UserData": "",
            "KeyName": "kp-us-east",
            "ImageId": "ami-aabbcc11",
            "InstanceType": "t2.medium",
            "NetworkInterfaces": [
```

```
                {
                    "SubnetId": "subnet-7b16de0c",
                    "DeviceIndex": 0,
                    "DeleteOnTermination": false,
                    "Groups": [
                        "sg-7c227019"
                    ],
                    "AssociatePublicIpAddress": true
                }
            ]
        },
        "DefaultVersion": true,
        "CreateTime": "2017-11-20T12:52:33.000Z"
    }
  ]
}
```

- Per i dettagli sull'API, consulta DescribeLaunchTemplateVersions AWS CLICommand Reference.

`describe-launch-templates`

Il seguente esempio di codice mostra come utilizzare`describe-launch-templates`.

AWS CLI

Come descrivere i modelli di avvio

Questo esempio descrive i modelli di avvio in uso.

Comando:

```
aws ec2 describe-launch-templates
```

Output:

```
{
  "LaunchTemplates": [
      {
          "LatestVersionNumber": 2,
          "LaunchTemplateId": "lt-0e06d290751193123",
          "LaunchTemplateName": "TemplateForWebServer",
```

```
            "DefaultVersionNumber": 2,
            "CreatedBy": "arn:aws:iam::123456789012:root",
            "CreateTime": "2017-11-27T09:30:23.000Z"
        },
        {
            "LatestVersionNumber": 6,
            "LaunchTemplateId": "lt-0c45b5e061ec98456",
            "LaunchTemplateName": "DBServersTemplate",
            "DefaultVersionNumber": 1,
            "CreatedBy": "arn:aws:iam::123456789012:root",
            "CreateTime": "2017-11-20T09:25:22.000Z"
        },
        {
            "LatestVersionNumber": 1,
            "LaunchTemplateId": "lt-0d47d774e8e52dabc",
            "LaunchTemplateName": "MyLaunchTemplate2",
            "DefaultVersionNumber": 1,
            "CreatedBy": "arn:aws:iam::123456789012:root",
            "CreateTime": "2017-11-02T12:06:21.000Z"
        },
        {
            "LatestVersionNumber": 3,
            "LaunchTemplateId": "lt-01e5f948eb4f589d6",
            "LaunchTemplateName": "testingtemplate2",
            "DefaultVersionNumber": 1,
            "CreatedBy": "arn:aws:sts::123456789012:assumed-role/AdminRole/
i-03ee35176e2e5aabc",
            "CreateTime": "2017-12-01T08:19:48.000Z"
        },
    ]
}
```

- Per i dettagli sull'API, consulta DescribeLaunchTemplates AWS CLICommand Reference.

`describe-local-gateway-route-table-virtual-interface-group-associations`

Il seguente esempio di codice mostra come utilizzare`describe-local-gateway-route-table-virtual-interface-group-associations`.

AWS CLI

Come descrivere le associazioni tra gruppi di interfacce virtuali e tabelle di routing del gateway locale

L'`describe-local-gateway-route-table-virtual-interface-group-associations`esempio seguente descrive le associazioni tra i gruppi di interfacce virtuali e le tabelle di routing del gateway locale nell' AWS account.

```
aws ec2 describe-local-gateway-route-table-virtual-interface-group-associations
```

Output:

```
{
    "LocalGatewayRouteTableVirtualInterfaceGroupAssociations": [
        {
            "LocalGatewayRouteTableVirtualInterfaceGroupAssociationId": "lgw-vif-grp-assoc-07145b276bEXAMPLE",
            "LocalGatewayVirtualInterfaceGroupId": "lgw-vif-grp-07145b276bEXAMPLE",
            "LocalGatewayId": "lgw-0ab1c23d4eEXAMPLE",
            "LocalGatewayRouteTableId": "lgw-rtb-059615ef7dEXAMPLE",
            "LocalGatewayRouteTableArn": "arn:aws:ec2:us-west-2:123456789012:local-gateway-route-table/lgw-rtb-059615ef7dEXAMPLE",
            "OwnerId": "123456789012",
            "State": "associated",
            "Tags": []
        }
    ]
}
```

Per ulteriori informazioni, consulta Utilizzo di gateway locali nella Guida per l'utente di AWS Outposts.

- Per i dettagli sull'API, consulta DescribeLocalGatewayRouteTableVirtualInterfaceGroupAssociations AWS CLICommand Reference.

`describe-local-gateway-route-table-vpc-associations`

Il seguente esempio di codice mostra come utilizzare`describe-local-gateway-route-table-vpc-associations`.

AWS CLI

Per descrivere le associazioni tra le tabelle di routing del gateway locale VPCs e le tabelle di routing

L'`describe-local-gateway-route-table-vpc-associations`esempio seguente visualizza informazioni sull'associazione specificata tra le tabelle di routing del gateway locale VPCs e quelle relative all'associazione specificata.

```
aws ec2 describe-local-gateway-route-table-vpc-associations \
    --local-gateway-route-table-vpc-association-ids lgw-vpc-assoc-0e0f27af15EXAMPLE
```

Output:

```
{
    "LocalGatewayRouteTableVpcAssociation": {
        "LocalGatewayRouteTableVpcAssociationId": "lgw-vpc-assoc-0e0f27af1EXAMPLE",
        "LocalGatewayRouteTableId": "lgw-rtb-059615ef7dEXAMPLE",
        "LocalGatewayId": "lgw-09b493aa7cEXAMPLE",
        "VpcId": "vpc-0efe9bde08EXAMPLE",
        "State": "associated"
    }
}
```

Per ulteriori informazioni, consulta Tabelle di routing del gateway locale nella Guida per l'utente di Outposts.

- Per i dettagli sull'API, vedere DescribeLocalGatewayRouteTableVpcAssociationsin AWS CLI Command Reference.

`describe-local-gateway-route-tables`

Il seguente esempio di codice mostra come utilizzare`describe-local-gateway-route-tables`.

AWS CLI

Come descrivere le tabelle di routing del gateway locale

L'esempio `describe-local-gateway-route-tables` seguente visualizza i dettagli relativi alle tabelle di routing del gateway locale.

```
aws ec2 describe-local-gateway-route-tables
```

Output:

```
{
```

```
    "LocalGatewayRouteTables": [
        {
            "LocalGatewayRouteTableId": "lgw-rtb-059615ef7deEXAMPLE",
            "LocalGatewayId": "lgw-09b493aa7cEXAMPLE",
            "OutpostArn": "arn:aws:outposts:us-west-2:111122223333:outpost/
op-0dc11b66edEXAMPLE",
            "State": "available"
        }
    ]
}
```

- Per i dettagli sull'API, consulta DescribeLocalGatewayRouteTables AWS CLICommand Reference.

`describe-local-gateway-virtual-interface-groups`

Il seguente esempio di codice mostra come utilizzare`describe-local-gateway-virtual-interface-groups`.

AWS CLI

Come descrivere i gruppi di interfacce virtuali del gateway locale

L'`describe-local-gateway-virtual-interface-groups`esempio seguente descrive i gruppi di interfaccia virtuale del gateway locale presenti nell' AWS account.

```
aws ec2 describe-local-gateway-virtual-interface-groups
```

Output:

```
{
    "LocalGatewayVirtualInterfaceGroups": [
        {
            "LocalGatewayVirtualInterfaceGroupId": "lgw-vif-grp-07145b276bEXAMPLE",
            "LocalGatewayVirtualInterfaceIds": [
                "lgw-vif-01a23bc4d5EXAMPLE",
                "lgw-vif-543ab21012EXAMPLE"
            ],
            "LocalGatewayId": "lgw-0ab1c23d4eEXAMPLE",
            "OwnerId": "123456789012",
            "Tags": []
```

```
        }
    ]
}
```

Per ulteriori informazioni, consulta Utilizzo di gateway locali nella Guida per l'utente di AWS Outposts.

- Per i dettagli sull'API, consulta DescribeLocalGatewayVirtualInterfaceGroups AWS CLICommand Reference.

`describe-local-gateway-virtual-interfaces`

Il seguente esempio di codice mostra come utilizzare`describe-local-gateway-virtual-interfaces`.

AWS CLI

Come descrivere le interfacce virtuali del gateway locale

L'`describe-local-gateway-virtual-interfaces`esempio seguente descrive le interfacce virtuali del gateway locale del tuo AWS account.

```
aws ec2 describe-local-gateway-virtual-interfaces
```

Output:

```
{
    "LocalGatewayVirtualInterfaces": [
        {
            "LocalGatewayVirtualInterfaceId": "lgw-vif-01a23bc4d5EXAMPLE",
            "LocalGatewayId": "lgw-0ab1c23d4eEXAMPLE",
            "Vlan": 2410,
            "LocalAddress": "0.0.0.0/0",
            "PeerAddress": "0.0.0.0/0",
            "LocalBgpAsn": 65010,
            "PeerBgpAsn": 65000,
            "OwnerId": "123456789012",
            "Tags": []
        },
        {
            "LocalGatewayVirtualInterfaceId": "lgw-vif-543ab21012EXAMPLE",
            "LocalGatewayId": "lgw-0ab1c23d4eEXAMPLE",
```

```
            "Vlan": 2410,
            "LocalAddress": "0.0.0.0/0",
            "PeerAddress": "0.0.0.0/0",
            "LocalBgpAsn": 65010,
            "PeerBgpAsn": 65000,
            "OwnerId": "123456789012",
            "Tags": []
        }
    ]
}
```

Per ulteriori informazioni, consulta Utilizzo di gateway locali nella Guida per l'utente di AWS Outposts.

- Per i dettagli sull'API, consulta DescribeLocalGatewayVirtualInterfaces AWS CLICommand Reference.

`describe-local-gateways`

Il seguente esempio di codice mostra come utilizzare`describe-local-gateways`.

AWS CLI

Come descrivere i gateway locali

L'esempio `describe-local-gateways` seguente visualizza i dettagli dei gateway locali disponibili per l'utente.

```
aws ec2 describe-local-gateways
```

Output:

```
{
    "LocalGateways": [
        {
            "LocalGatewayId": "lgw-09b493aa7cEXAMPLE",
            "OutpostArn": "arn:aws:outposts:us-west-2:123456789012:outpost/
op-0dc11b66ed59f995a",
            "OwnerId": "123456789012",
            "State": "available"
        }
    ]
```

```
}
```

- Per i dettagli sull'API, consulta DescribeLocalGateways AWS CLICommand Reference.

`describe-locked-snapshots`

Il seguente esempio di codice mostra come utilizzare`describe-locked-snapshots`.

AWS CLI

Come descrivere lo stato di blocco di uno snapshot

L'esempio `describe-locked-snapshots` seguente descrive lo stato di blocco dello snapshot specificato.

```
aws ec2 describe-locked-snapshots \
    --snapshot-ids snap-0b5e733b4a8df6e0d
```

Output:

```
{
    "Snapshots": [
        {
            "OwnerId": "123456789012",
            "SnapshotId": "snap-0b5e733b4a8df6e0d",
            "LockState": "governance",
            "LockDuration": 365,
            "LockCreatedOn": "2024-05-05T00:56:06.208000+00:00",
            "LockDurationStartTime": "2024-05-05T00:56:06.208000+00:00",
            "LockExpiresOn": "2025-05-05T00:56:06.208000+00:00"
        }
    ]
}
```

Per ulteriori informazioni, consulta Snapshot Lock nella Guida per l'utente di Amazon EBS.

- Per i dettagli sull'API, consulta DescribeLockedSnapshots AWS CLICommand Reference.

`describe-managed-prefix-lists`

Il seguente esempio di codice mostra come utilizzare`describe-managed-prefix-lists`.

AWS CLI

Come descrivere gli elenchi di prefissi gestiti

L'`describe-managed-prefix-lists`esempio seguente descrive gli elenchi di prefissi di proprietà AWS dell'account`123456789012`.

```
aws ec2 describe-managed-prefix-lists \
    --filters Name=owner-id,Values=123456789012
```

Output:

```
{
    "PrefixLists": [
        {
            "PrefixListId": "pl-11223344556677aab",
            "AddressFamily": "IPv6",
            "State": "create-complete",
            "PrefixListArn": "arn:aws:ec2:us-west-2:123456789012:prefix-list/
pl-11223344556677aab",
            "PrefixListName": "vpc-ipv6-cidrs",
            "MaxEntries": 25,
            "Version": 1,
            "Tags": [],
            "OwnerId": "123456789012"
        },
        {
            "PrefixListId": "pl-0123456abcabcabc1",
            "AddressFamily": "IPv4",
            "State": "active",
            "PrefixListArn": "arn:aws:ec2:us-west-2:123456789012:prefix-list/
pl-0123456abcabcabc1",
            "PrefixListName": "vpc-cidrs",
            "MaxEntries": 10,
            "Version": 1,
            "Tags": [],
            "OwnerId": "123456789012"
      }
  ]
}
```

Per ulteriori informazioni, consulta Elenchi di prefissi gestiti nella Guida per l'utente di Amazon VPC.

- Per i dettagli sull'API, vedere DescribeManagedPrefixListsin AWS CLI Command Reference.

`describe-moving-addresses`

Il seguente esempio di codice mostra come utilizzare`describe-moving-addresses`.

AWS CLI

Come descrivere gli indirizzi di spostamento in uso

Questo esempio descrive tutti gli indirizzi IP elastici di spostamento.

Comando:

```
aws ec2 describe-moving-addresses
```

Output:

```
{
  "MovingAddressStatuses": [
    {
      "PublicIp": "198.51.100.0",
      "MoveStatus": "MovingToVpc"
    }
  ]
}
```

Questo esempio descrive tutti gli indirizzi che stanno passando alla piattaforma EC2 -VPC.

Comando:

```
aws ec2 describe-moving-addresses --filters Name=moving-status,Values=MovingToVpc
```

- Per i dettagli sull'API, consulta AWS CLI Command DescribeMovingAddressesReference.

`describe-nat-gateways`

Il seguente esempio di codice mostra come utilizzare`describe-nat-gateways`.

AWS CLI

Esempio 1: come descrivere un gateway NAT pubblico

L'esempio `describe-nat-gateways` seguente descrive il gateway NAT pubblico specificato.

```
aws ec2 describe-nat-gateways \
    --nat-gateway-id nat-01234567890abcdef
```

Output:

```
{
    "NatGateways": [
        {
            "CreateTime": "2023-08-25T01:56:51.000Z",
            "NatGatewayAddresses": [
                {
                    "AllocationId": "eipalloc-0790180cd2EXAMPLE",
                    "NetworkInterfaceId": "eni-09cc4b2558794f7f9",
                    "PrivateIp": "10.0.0.211",
                    "PublicIp": "54.85.121.213",
                    "AssociationId": "eipassoc-04d295cc9b8815b24",
                    "IsPrimary": true,
                    "Status": "succeeded"
                },
                {
                    "AllocationId": "eipalloc-0be6ecac95EXAMPLE",
                    "NetworkInterfaceId": "eni-09cc4b2558794f7f9",
                    "PrivateIp": "10.0.0.74",
                    "PublicIp": "3.211.231.218",
                    "AssociationId": "eipassoc-0f96bdca17EXAMPLE",
                    "IsPrimary": false,
                    "Status": "succeeded"
                }
            ],
            "NatGatewayId": "nat-01234567890abcdef",
            "State": "available",
            "SubnetId": "subnet-655eab5f08EXAMPLE",
            "VpcId": "vpc-098eb5ef58EXAMPLE",
            "Tags": [
                {
                    "Key": "Name",
                    "Value": "public-nat"
                }
            ],
            "ConnectivityType": "public"
        }
```

```
    ]
}
```

Esempio 2: come descrivere un gateway NAT privato

L'esempio `describe-nat-gateways` seguente descrive il gateway NAT privato specificato.

```
aws ec2 describe-nat-gateways \
    --nat-gateway-id nat-1234567890abcdef0
```

Output:

```
{
    "NatGateways": [
        {
            "CreateTime": "2023-08-25T00:50:05.000Z",
            "NatGatewayAddresses": [
                {
                    "NetworkInterfaceId": "eni-0065a61b324d1897a",
                    "PrivateIp": "10.0.20.240",
                    "IsPrimary": true,
                    "Status": "succeeded"
                },
                {
                    "NetworkInterfaceId": "eni-0065a61b324d1897a",
                    "PrivateIp": "10.0.20.33",
                    "IsPrimary": false,
                    "Status": "succeeded"
                },
                {
                    "NetworkInterfaceId": "eni-0065a61b324d1897a",
                    "PrivateIp": "10.0.20.197",
                    "IsPrimary": false,
                    "Status": "succeeded"
                }
            ],
            "NatGatewayId": "nat-1234567890abcdef0",
            "State": "available",
            "SubnetId": "subnet-08fc749671EXAMPLE",
            "VpcId": "vpc-098eb5ef58EXAMPLE",
            "Tags": [
                {
                    "Key": "Name",
```

```
                    "Value": "private-nat"
                }
            ],
            "ConnectivityType": "private"
        }
    ]
}
```

Per ulteriori informazioni, consulta Gateway NAT nella Guida per l'utente di Amazon VPC.

- Per i dettagli sull'API, consulta DescribeNatGateways AWS CLICommand Reference.

`describe-network-acls`

Il seguente esempio di codice mostra come utilizzare`describe-network-acls`.

AWS CLI

Per descrivere la rete ACLs

L'`describe-network-acls`esempio seguente recupera i dettagli sulla rete ACLs.

```
aws ec2 describe-network-acls
```

Output:

```
{
    "NetworkAcls": [
        {
            "Associations": [
                {
                    "NetworkAclAssociationId": "aclassoc-0c1679dc41EXAMPLE",
                    "NetworkAclId": "acl-0ea1f54ca7EXAMPLE",
                    "SubnetId": "subnet-0931fc2fa5EXAMPLE"
                }
            ],
            "Entries": [
                {
                    "CidrBlock": "0.0.0.0/0",
                    "Egress": true,
                    "Protocol": "-1",
                    "RuleAction": "allow",
                    "RuleNumber": 100
```

```
                },
                {
                    "CidrBlock": "0.0.0.0/0",
                    "Egress": true,
                    "Protocol": "-1",
                    "RuleAction": "deny",
                    "RuleNumber": 32767
                },
                {
                    "CidrBlock": "0.0.0.0/0",
                    "Egress": false,
                    "Protocol": "-1",
                    "RuleAction": "allow",
                    "RuleNumber": 100
                },
                {
                    "CidrBlock": "0.0.0.0/0",
                    "Egress": false,
                    "Protocol": "-1",
                    "RuleAction": "deny",
                    "RuleNumber": 32767
                }
            ],
            "IsDefault": true,
            "NetworkAclId": "acl-0ea1f54ca7EXAMPLE",
            "Tags": [],
            "VpcId": "vpc-06e4ab6c6cEXAMPLE",
            "OwnerId": "111122223333"
        },
        {
            "Associations": [],
            "Entries": [
                {
                    "CidrBlock": "0.0.0.0/0",
                    "Egress": true,
                    "Protocol": "-1",
                    "RuleAction": "allow",
                    "RuleNumber": 100
                },
                {
                    "Egress": true,
                    "Ipv6CidrBlock": "::/0",
                    "Protocol": "-1",
                    "RuleAction": "allow",
```

```
            "RuleNumber": 101
        },
        {
            "CidrBlock": "0.0.0.0/0",
            "Egress": true,
            "Protocol": "-1",
            "RuleAction": "deny",
            "RuleNumber": 32767
        },
        {
            "Egress": true,
            "Ipv6CidrBlock": "::/0",
            "Protocol": "-1",
            "RuleAction": "deny",
            "RuleNumber": 32768
        },
        {
            "CidrBlock": "0.0.0.0/0",
            "Egress": false,
            "Protocol": "-1",
            "RuleAction": "allow",
            "RuleNumber": 100
        },
        {
            "Egress": false,
            "Ipv6CidrBlock": "::/0",
            "Protocol": "-1",
            "RuleAction": "allow",
            "RuleNumber": 101
        },
        {
            "CidrBlock": "0.0.0.0/0",
            "Egress": false,
            "Protocol": "-1",
            "RuleAction": "deny",
            "RuleNumber": 32767
        },
        {
            "Egress": false,
            "Ipv6CidrBlock": "::/0",
            "Protocol": "-1",
            "RuleAction": "deny",
            "RuleNumber": 32768
        }
```

```
            ],
            "IsDefault": true,
            "NetworkAclId": "acl-0e2a78e4e2EXAMPLE",
            "Tags": [],
            "VpcId": "vpc-03914afb3eEXAMPLE",
            "OwnerId": "111122223333"
        }
    ]
}
```

Per ulteriori informazioni, consulta Network ACLs in the AWS VPC User Guide.

- Per i dettagli sull'API, consulta DescribeNetworkAcls AWS CLICommand Reference.

`describe-network-insights-access-scope-analyses`

Il seguente esempio di codice mostra come utilizzare`describe-network-insights-access-scope-analyses`.

AWS CLI

Come descrivere le analisi dell'ambito di accesso alle informazioni di rete

L'`describe-network-insights-access-scope-analyses`esempio seguente descrive l'analisi dell'ambito di accesso nel tuo AWS account.

```
aws ec2 describe-network-insights-access-scope-analyses \
    --region us-east-1
```

Output:

```
{
    "NetworkInsightsAccessScopeAnalyses": [
        {
            "NetworkInsightsAccessScopeAnalysisId": "nisa-123456789111",
            "NetworkInsightsAccessScopeAnalysisArn": "arn:aws:ec2:us-
east-1:123456789012:network-insights-access-scope-analysis/nisa-123456789111",
            "NetworkInsightsAccessScopeId": "nis-123456789222",
            "Status": "succeeded",
            "StartDate": "2022-01-25T19:45:36.842000+00:00",
            "FindingsFound": "true",
            "Tags": []
        }
```

```
    ]
}
```

Per ulteriori informazioni, consulta Guida introduttiva a Network Access Analyzer utilizzando la AWS CLI nella Guida Network Access Analyzer.

- Per i dettagli sull'API, consulta Command DescribeNetworkInsightsAccessScopeAnalysesReference AWS CLI .

`describe-network-insights-access-scopes`

Il seguente esempio di codice mostra come utilizzare`describe-network-insights-access-scopes`.

AWS CLI

Come descrivere gli ambiti di accesso alle informazioni di rete

L'`describe-network-insights-access-scopes`esempio seguente descrive le analisi dell'ambito di accesso nel tuo account. AWS

```
aws ec2 describe-network-insights-access-scopes \
    --region us-east-1
```

Output:

```
{
    "NetworkInsightsAccessScopes": [
        {
            "NetworkInsightsAccessScopeId": "nis-123456789111",
            "NetworkInsightsAccessScopeArn": "arn:aws:ec2:us-
east-1:123456789012:network-insights-access-scope/nis-123456789111",
            "CreatedDate": "2021-11-29T21:12:41.416000+00:00",
            "UpdatedDate": "2021-11-29T21:12:41.416000+00:00",
            "Tags": []
        }
    ]
}
```

Per ulteriori informazioni, consulta Guida introduttiva a Network Access Analyzer utilizzando la AWS CLI nella Guida Network Access Analyzer.

- Per i dettagli sull'API, consulta Command DescribeNetworkInsightsAccessScopesReference AWS CLI .

`describe-network-insights-analyses`

Il seguente esempio di codice mostra come utilizzare`describe-network-insights-analyses`.

AWS CLI

Come visualizzare i risultati di un'analisi del percorso

L'esempio `describe-network-insights-analyses` seguente descrive l'analisi specificata. In questo esempio, l'origine è un gateway Internet, la destinazione è un' EC2 istanza e il protocollo è TCP. L'analisi è riuscita (`Status` è `succeeded`) e il percorso non è raggiungibile (`NetworkPathFound` è `false`). Il codice di spiegazione `ENI_SG_RULES_MISMATCH` indica che il gruppo di sicurezza per l'istanza non contiene una regola che consente il traffico sulla porta di destinazione.

```
aws ec2 describe-network-insights-analyses \
    --network-insights-analysis-ids nia-02207aa13eb480c7a
```

Output:

```
{
    "NetworkInsightsAnalyses": [
        {
            "NetworkInsightsAnalysisId": "nia-02207aa13eb480c7a",
            "NetworkInsightsAnalysisArn": "arn:aws:ec2:us-
east-1:123456789012:network-insights-analysis/nia-02207aa13eb480c7a",
            "NetworkInsightsPathId": "nip-0b26f224f1d131fa8",
            "StartDate": "2021-01-20T22:58:37.495Z",
            "Status": "succeeded",
            "NetworkPathFound": false,
            "Explanations": [
                {
                    "Direction": "ingress",
                    "ExplanationCode": "ENI_SG_RULES_MISMATCH",
                    "NetworkInterface": {
                        "Id": "eni-0a25edef15a6cc08c",
                        "Arn": "arn:aws:ec2:us-east-1:123456789012:network-
interface/eni-0a25edef15a6cc08c"
```

```
                    },
                    "SecurityGroups": [
                        {
                            "Id": "sg-02f0d35a850ba727f",
                            "Arn": "arn:aws:ec2:us-east-1:123456789012:security-
group/sg-02f0d35a850ba727f"
                        }
                    ],
                    "Subnet": {
                        "Id": "subnet-004ff41eccb4d1194",
                        "Arn": "arn:aws:ec2:us-east-1:123456789012:subnet/
subnet-004ff41eccb4d1194"
                    },
                    "Vpc": {
                        "Id": "vpc-f1663d98ad28331c7",
                        "Arn": "arn:aws:ec2:us-east-1:123456789012:vpc/vpc-
f1663d98ad28331c7"
                    }
                }
            ],
            "Tags": []
        }
    ]
}
```

Per ulteriori informazioni, consulta Guida introduttiva all'uso della AWS CLI nella Reachability Analyzer Guide.

- Per i dettagli sull'API, consulta Command DescribeNetworkInsightsAnalysesReference AWS CLI .

`describe-network-insights-paths`

Il seguente esempio di codice mostra come utilizzare`describe-network-insights-paths`.

AWS CLI

Come descrivere un percorso

L'esempio `describe-network-insights-paths` seguente descrive il percorso specificato.

```
aws ec2 describe-network-insights-paths \
    --network-insights-path-ids nip-0b26f224f1d131fa8
```

Output:

```
{
    "NetworkInsightsPaths": [
        {
            "NetworkInsightsPathId": "nip-0b26f224f1d131fa8",
            "NetworkInsightsPathArn": "arn:aws:ec2:us-east-1:123456789012:network-
insights-path/nip-0b26f224f1d131fa8",
            "CreatedDate": "2021-01-20T22:43:46.933Z",
            "Source": "igw-0797cccdc9d73b0e5",
            "Destination": "i-0495d385ad28331c7",
            "Protocol": "tcp"
        }
    ]
}
```

Per ulteriori informazioni, consulta Guida introduttiva all'uso della AWS CLI nella Reachability Analyzer Guide.

- Per i dettagli sull'API, consulta Command DescribeNetworkInsightsPathsReference AWS CLI .

`describe-network-interface-attribute`

Il seguente esempio di codice mostra come utilizzare`describe-network-interface-attribute`.

AWS CLI

Come descrivere l'attributo attachment di un'interfaccia di rete

Questo comando di esempio descrive l'attributo `attachment` dell'interfaccia di rete specificata.

Comando:

```
aws ec2 describe-network-interface-attribute --network-interface-id eni-686ea200 --
attribute attachment
```

Output:

```
{
  "NetworkInterfaceId": "eni-686ea200",
  "Attachment": {
      "Status": "attached",
```

```
      "DeviceIndex": 0,
      "AttachTime": "2015-05-21T20:02:20.000Z",
      "InstanceId": "i-1234567890abcdef0",
      "DeleteOnTermination": true,
      "AttachmentId": "eni-attach-43348162",
      "InstanceOwnerId": "123456789012"
  }
}
```

Come descrivere l'attributo description di un'interfaccia di rete

Questo comando di esempio descrive l'attributo `description` dell'interfaccia di rete specificata.

Comando:

```
aws ec2 describe-network-interface-attribute --network-interface-id eni-686ea200 --
attribute description
```

Output:

```
{
  "NetworkInterfaceId": "eni-686ea200",
  "Description": {
      "Value": "My description"
  }
}
```

Come descrivere l'attributo groupSet di un'interfaccia di rete

Questo comando di esempio descrive l'attributo `groupSet` dell'interfaccia di rete specificata.

Comando:

```
aws ec2 describe-network-interface-attribute --network-interface-id eni-686ea200 --
attribute groupSet
```

Output:

```
{
  "NetworkInterfaceId": "eni-686ea200",
  "Groups": [
      {
          "GroupName": "my-security-group",
```

```
            "GroupId": "sg-903004f8"
        }
    ]
}
```

Per descrivere l' sourceDestCheck attributo di un'interfaccia di rete

Questo comando di esempio descrive l'attributo `sourceDestCheck` dell'interfaccia di rete specificata.

Comando:

```
aws ec2 describe-network-interface-attribute --network-interface-id eni-686ea200 --attribute sourceDestCheck
```

Output:

```
{
  "NetworkInterfaceId": "eni-686ea200",
  "SourceDestCheck": {
      "Value": true
  }
}
```

- Per i dettagli sull'API, consulta DescribeNetworkInterfaceAttribute AWS CLICommand Reference.

`describe-network-interface-permissions`

Il seguente esempio di codice mostra come utilizzare`describe-network-interface-permissions`.

AWS CLI

Come descrivere le autorizzazioni dell'interfaccia di rete

Questo esempio descrive tutte le autorizzazione dell'interfaccia di rete.

Comando:

```
aws ec2 describe-network-interface-permissions
```

Output:

```
{
  "NetworkInterfacePermissions": [
      {
          "PermissionState": {
              "State": "GRANTED"
          },
          "NetworkInterfacePermissionId": "eni-perm-06fd19020ede149ea",
          "NetworkInterfaceId": "eni-b909511a",
          "Permission": "INSTANCE-ATTACH",
          "AwsAccountId": "123456789012"
      }
  ]
}
```

- Per i dettagli sull'API, consulta DescribeNetworkInterfacePermissions AWS CLICommand Reference.

`describe-network-interfaces`

Il seguente esempio di codice mostra come utilizzare`describe-network-interfaces`.

AWS CLI

Come descrivere le interfacce di rete

Questo esempio descrive tutte le interfacce di rete.

Comando:

```
aws ec2 describe-network-interfaces
```

Output:

```
{
  "NetworkInterfaces": [
      {
          "Status": "in-use",
          "MacAddress": "02:2f:8f:b0:cf:75",
          "SourceDestCheck": true,
          "VpcId": "vpc-a01106c2",
```

```
            "Description": "my network interface",
            "Association": {
                "PublicIp": "203.0.113.12",
                "AssociationId": "eipassoc-0fbb766a",
                "PublicDnsName": "ec2-203-0-113-12.compute-1.amazonaws.com",
                "IpOwnerId": "123456789012"
            },
            "NetworkInterfaceId": "eni-e5aa89a3",
            "PrivateIpAddresses": [
                {
                    "PrivateDnsName": "ip-10-0-1-17.ec2.internal",
                    "Association": {
                        "PublicIp": "203.0.113.12",
                        "AssociationId": "eipassoc-0fbb766a",
                        "PublicDnsName": "ec2-203-0-113-12.compute-1.amazonaws.com",
                        "IpOwnerId": "123456789012"
                    },
                    "Primary": true,
                    "PrivateIpAddress": "10.0.1.17"
                }
            ],
            "RequesterManaged": false,
            "Ipv6Addresses": [],
            "PrivateDnsName": "ip-10-0-1-17.ec2.internal",
            "AvailabilityZone": "us-east-1d",
            "Attachment": {
                "Status": "attached",
                "DeviceIndex": 1,
                "AttachTime": "2013-11-30T23:36:42.000Z",
                "InstanceId": "i-1234567890abcdef0",
                "DeleteOnTermination": false,
                "AttachmentId": "eni-attach-66c4350a",
                "InstanceOwnerId": "123456789012"
            },
            "Groups": [
                {
                    "GroupName": "default",
                    "GroupId": "sg-8637d3e3"
                }
            ],
            "SubnetId": "subnet-b61f49f0",
            "OwnerId": "123456789012",
            "TagSet": [],
            "PrivateIpAddress": "10.0.1.17"
```

```
        },
        {
            "Status": "in-use",
            "MacAddress": "02:58:f5:ef:4b:06",
            "SourceDestCheck": true,
            "VpcId": "vpc-a01106c2",
            "Description": "Primary network interface",
            "Association": {
                "PublicIp": "198.51.100.0",
                "IpOwnerId": "amazon"
            },
            "NetworkInterfaceId": "eni-f9ba99bf",
            "PrivateIpAddresses": [
                {
                    "Association": {
                        "PublicIp": "198.51.100.0",
                        "IpOwnerId": "amazon"
                    },
                    "Primary": true,
                    "PrivateIpAddress": "10.0.1.149"
                }
            ],
            "RequesterManaged": false,
            "Ipv6Addresses": [],
            "AvailabilityZone": "us-east-1d",
            "Attachment": {
                "Status": "attached",
                "DeviceIndex": 0,
                "AttachTime": "2013-11-30T23:35:33.000Z",
                "InstanceId": "i-0598c7d356eba48d7",
                "DeleteOnTermination": true,
                "AttachmentId": "eni-attach-1b9db777",
                "InstanceOwnerId": "123456789012"
            },
            "Groups": [
                {
                    "GroupName": "default",
                    "GroupId": "sg-8637d3e3"
                }
            ],
            "SubnetId": "subnet-b61f49f0",
            "OwnerId": "123456789012",
            "TagSet": [],
            "PrivateIpAddress": "10.0.1.149"
```

```
        }
    ]
}
```

Questo esempio descrive le interfacce di rete che hanno un tag con la chiave `Purpose` e il valore `Prod`.

Comando:

```
aws ec2 describe-network-interfaces --filters Name=tag:Purpose,Values=Prod
```

Output:

```
{
    "NetworkInterfaces": [
        {
            "Status": "available",
            "MacAddress": "12:2c:bd:f9:bf:17",
            "SourceDestCheck": true,
            "VpcId": "vpc-8941ebec",
            "Description": "ProdENI",
            "NetworkInterfaceId": "eni-b9a5ac93",
            "PrivateIpAddresses": [
                {
                    "PrivateDnsName": "ip-10-0-1-55.ec2.internal",
                    "Primary": true,
                    "PrivateIpAddress": "10.0.1.55"
                },
                {
                    "PrivateDnsName": "ip-10-0-1-117.ec2.internal",
                    "Primary": false,
                    "PrivateIpAddress": "10.0.1.117"
                }
            ],
            "RequesterManaged": false,
            "PrivateDnsName": "ip-10-0-1-55.ec2.internal",
            "AvailabilityZone": "us-east-1d",
            "Ipv6Addresses": [],
            "Groups": [
                {
                    "GroupName": "MySG",
                    "GroupId": "sg-905002f5"
                }
```

```
            ],
            "SubnetId": "subnet-31d6c219",
            "OwnerId": "123456789012",
            "TagSet": [
                {
                    "Value": "Prod",
                    "Key": "Purpose"
                }
            ],
            "PrivateIpAddress": "10.0.1.55"
        }
    ]
}
```

- Per i dettagli sull'API, consulta DescribeNetworkInterfaces AWS CLICommand Reference.

`describe-placement-groups`

Il seguente esempio di codice mostra come utilizzare`describe-placement-groups`.

AWS CLI

Come descrivere i gruppi di posizionamento

Questo comando di esempio descrive tutti i gruppi di posizionamento.

Comando:

```
aws ec2 describe-placement-groups
```

Output:

```
{
    "PlacementGroups": [
        {
            "GroupName": "my-cluster",
            "State": "available",
            "Strategy": "cluster"
        },
        ...
    ]
}
```

- Per i dettagli sull'API, consulta DescribePlacementGroups AWS CLICommand Reference.

`describe-prefix-lists`

Il seguente esempio di codice mostra come utilizzare`describe-prefix-lists`.

AWS CLI

Come descrivere gli elenchi di prefissi

Questo esempio elenca tutti gli elenchi di prefissi disponibili per la Regione.

Comando:

```
aws ec2 describe-prefix-lists
```

Output:

```
{
  "PrefixLists": [
    {
      "PrefixListName": "com.amazonaws.us-east-1.s3",
      "Cidrs": [
        "54.231.0.0/17"
      ],
      "PrefixListId": "pl-63a5400a"
    }
  ]
}
```

- Per i dettagli sull'API, consulta DescribePrefixLists AWS CLICommand Reference.

`describe-principal-id-format`

Il seguente esempio di codice mostra come utilizzare`describe-principal-id-format`.

AWS CLI

Come descrivere il formato ID per gli utenti e i ruoli IAM con il formato ID lungo abilitato

L'esempio `describe-principal-id-format` seguente descrive il formato ID per l'utente root, tutti i ruoli IAM e tutti gli utenti IAM con il formato ID lungo abilitato.

```
aws ec2 describe-principal-id-format \
    --resource instance
```

Output:

```
{
    "Principals": [
        {
            "Arn": "arn:aws:iam::123456789012:root",
            "Statuses": [
                {
                    "Deadline": "2016-12-15T00:00:00.000Z",
                    "Resource": "reservation",
                    "UseLongIds": true
                },
                {
                    "Deadline": "2016-12-15T00:00:00.000Z",
                    "Resource": "instance",
                    "UseLongIds": true
                },
                {
                    "Deadline": "2016-12-15T00:00:00.000Z",
                    "Resource": "volume",
                    "UseLongIds": true
                },
            ]
        },
        ...
    ]
}
```

- Per i dettagli sull'API, consulta DescribePrincipalIdFormat AWS CLICommand Reference.

`describe-public-ipv4-pools`

Il seguente esempio di codice mostra come utilizzare`describe-public-ipv4-pools`.

AWS CLI

Per descrivere i pool di IPv4 indirizzi pubblici

L'`describe-public-ipv4-pools`esempio seguente mostra i dettagli sui pool di indirizzi che sono stati creati quando è stato eseguito il provisioning di intervalli di IPv4 indirizzi pubblici utilizzando Bring Your Own IP Addresses (BYOIP).

```
aws ec2 describe-public-ipv4-pools
```

Output:

```
{
    "PublicIpv4Pools": [
        {
            "PoolId": "ipv4pool-ec2-1234567890abcdef0",
            "PoolAddressRanges": [
                {
                    "FirstAddress": "203.0.113.0",
                    "LastAddress": "203.0.113.255",
                    "AddressCount": 256,
                    "AvailableAddressCount": 256
                }
            ],
            "TotalAddressCount": 256,
            "TotalAvailableAddressCount": 256
        }
    ]
}
```

- Per i dettagli sull'API, consulta DescribePublicIpv4Pools in Command Reference.AWS CLI

`describe-regions`

Il seguente esempio di codice mostra come utilizzare`describe-regions`.

AWS CLI

Esempio 1: come descrivere tutte le Regioni abilitate

Nell'esempio di `describe-regions` seguente vengono descritte tutte le regioni abilitate per l'account.

```
aws ec2 describe-regions
```

Output:

```
{
    "Regions": [
        {
            "Endpoint": "ec2.eu-north-1.amazonaws.com",
            "RegionName": "eu-north-1",
            "OptInStatus": "opt-in-not-required"
        },
        {
            "Endpoint": "ec2.ap-south-1.amazonaws.com",
            "RegionName": "ap-south-1",
            "OptInStatus": "opt-in-not-required"
        },
        {
            "Endpoint": "ec2.eu-west-3.amazonaws.com",
            "RegionName": "eu-west-3",
            "OptInStatus": "opt-in-not-required"
        },
        {
            "Endpoint": "ec2.eu-west-2.amazonaws.com",
            "RegionName": "eu-west-2",
            "OptInStatus": "opt-in-not-required"
        },
        {
            "Endpoint": "ec2.eu-west-1.amazonaws.com",
            "RegionName": "eu-west-1",
            "OptInStatus": "opt-in-not-required"
        },
        {
            "Endpoint": "ec2.ap-northeast-3.amazonaws.com",
            "RegionName": "ap-northeast-3",
            "OptInStatus": "opt-in-not-required"
        },
        {
            "Endpoint": "ec2.ap-northeast-2.amazonaws.com",
            "RegionName": "ap-northeast-2",
            "OptInStatus": "opt-in-not-required"
        },
        {
            "Endpoint": "ec2.ap-northeast-1.amazonaws.com",
            "RegionName": "ap-northeast-1",
            "OptInStatus": "opt-in-not-required"
        },
```

```
        {
            "Endpoint": "ec2.sa-east-1.amazonaws.com",
            "RegionName": "sa-east-1",
            "OptInStatus": "opt-in-not-required"
        },
        {
            "Endpoint": "ec2.ca-central-1.amazonaws.com",
            "RegionName": "ca-central-1",
            "OptInStatus": "opt-in-not-required"
        },
        {
            "Endpoint": "ec2.ap-southeast-1.amazonaws.com",
            "RegionName": "ap-southeast-1",
            "OptInStatus": "opt-in-not-required"
        },
        {
            "Endpoint": "ec2.ap-southeast-2.amazonaws.com",
            "RegionName": "ap-southeast-2",
            "OptInStatus": "opt-in-not-required"
        },
        {
            "Endpoint": "ec2.eu-central-1.amazonaws.com",
            "RegionName": "eu-central-1",
            "OptInStatus": "opt-in-not-required"
        },
        {
            "Endpoint": "ec2.us-east-1.amazonaws.com",
            "RegionName": "us-east-1",
            "OptInStatus": "opt-in-not-required"
        },
        {
            "Endpoint": "ec2.us-east-2.amazonaws.com",
            "RegionName": "us-east-2",
            "OptInStatus": "opt-in-not-required"
        },
        {
            "Endpoint": "ec2.us-west-1.amazonaws.com",
            "RegionName": "us-west-1",
            "OptInStatus": "opt-in-not-required"
        },
        {
            "Endpoint": "ec2.us-west-2.amazonaws.com",
            "RegionName": "us-west-2",
            "OptInStatus": "opt-in-not-required"
```

```
        }
    ]
}
```

Per ulteriori informazioni, consulta Regioni e zone nella Amazon EC2 User Guide.

Esempio 2: come descrivere le Regioni abilitate all'interno di un endpoint il cui nome contiene una stringa specifica

Nell'esempio di `describe-regions` seguente vengono descritte tutte le regioni abilitate che contengono la stringa "us" nell'endpoint.

```
aws ec2 describe-regions \
    --filters "Name=endpoint,Values=*us*"
```

Output:

```
{
    "Regions": [
        {
            "Endpoint": "ec2.us-east-1.amazonaws.com",
            "RegionName": "us-east-1"
        },
        {
            "Endpoint": "ec2.us-east-2.amazonaws.com",
            "RegionName": "us-east-2"
        },
        {
            "Endpoint": "ec2.us-west-1.amazonaws.com",
            "RegionName": "us-west-1"
        },
        {
            "Endpoint": "ec2.us-west-2.amazonaws.com",
            "RegionName": "us-west-2"
        }
    ]
}
```

Per ulteriori informazioni, consulta Regioni e zone nella Amazon EC2 User Guide.

Esempio 3: come descrivere tutte le Regioni

Nell'esempio di `describe-regions` seguente vengono descritte tutte le regioni disponibili, comprese le regioni disabilitate.

```
aws ec2 describe-regions \
    --all-regions
```

Output:

```
{
    "Regions": [
        {
            "Endpoint": "ec2.eu-north-1.amazonaws.com",
            "RegionName": "eu-north-1",
            "OptInStatus": "opt-in-not-required"
        },
        {
            "Endpoint": "ec2.ap-south-1.amazonaws.com",
            "RegionName": "ap-south-1",
            "OptInStatus": "opt-in-not-required"
        },
        {
            "Endpoint": "ec2.eu-west-3.amazonaws.com",
            "RegionName": "eu-west-3",
            "OptInStatus": "opt-in-not-required"
        },
        {
            "Endpoint": "ec2.eu-west-2.amazonaws.com",
            "RegionName": "eu-west-2",
            "OptInStatus": "opt-in-not-required"
        },
        {
            "Endpoint": "ec2.eu-west-1.amazonaws.com",
            "RegionName": "eu-west-1",
            "OptInStatus": "opt-in-not-required"
        },
        {
            "Endpoint": "ec2.ap-northeast-3.amazonaws.com",
            "RegionName": "ap-northeast-3",
            "OptInStatus": "opt-in-not-required"
        },
        {
            "Endpoint": "ec2.me-south-1.amazonaws.com",
            "RegionName": "me-south-1",
```

```
            "OptInStatus": "not-opted-in"
        },
        {
            "Endpoint": "ec2.ap-northeast-2.amazonaws.com",
            "RegionName": "ap-northeast-2",
            "OptInStatus": "opt-in-not-required"
        },
        {
            "Endpoint": "ec2.ap-northeast-1.amazonaws.com",
            "RegionName": "ap-northeast-1",
            "OptInStatus": "opt-in-not-required"
        },
        {
            "Endpoint": "ec2.sa-east-1.amazonaws.com",
            "RegionName": "sa-east-1",
            "OptInStatus": "opt-in-not-required"
        },
        {
            "Endpoint": "ec2.ca-central-1.amazonaws.com",
            "RegionName": "ca-central-1",
            "OptInStatus": "opt-in-not-required"
        },
        {
            "Endpoint": "ec2.ap-east-1.amazonaws.com",
            "RegionName": "ap-east-1",
            "OptInStatus": "not-opted-in"
        },
        {
            "Endpoint": "ec2.ap-southeast-1.amazonaws.com",
            "RegionName": "ap-southeast-1",
            "OptInStatus": "opt-in-not-required"
        },
        {
            "Endpoint": "ec2.ap-southeast-2.amazonaws.com",
            "RegionName": "ap-southeast-2",
            "OptInStatus": "opt-in-not-required"
        },
        {
            "Endpoint": "ec2.eu-central-1.amazonaws.com",
            "RegionName": "eu-central-1",
            "OptInStatus": "opt-in-not-required"
        },
        {
            "Endpoint": "ec2.us-east-1.amazonaws.com",
```

```
            "RegionName": "us-east-1",
            "OptInStatus": "opt-in-not-required"
        },
        {
            "Endpoint": "ec2.us-east-2.amazonaws.com",
            "RegionName": "us-east-2",
            "OptInStatus": "opt-in-not-required"
        },
        {
            "Endpoint": "ec2.us-west-1.amazonaws.com",
            "RegionName": "us-west-1",
            "OptInStatus": "opt-in-not-required"
        },
        {
            "Endpoint": "ec2.us-west-2.amazonaws.com",
            "RegionName": "us-west-2",
            "OptInStatus": "opt-in-not-required"
        }
    ]
}
```

Per ulteriori informazioni, consulta Regioni e zone nella Amazon EC2 User Guide.

Esempio 4: come elencare solo i nomi delle Regioni

Nell'esempio di `describe-regions` seguente viene utilizzato il parametro `--query` per filtrare l'output e restituire solo i nomi delle regioni come testo.

```
aws ec2 describe-regions \
    --all-regions \
    --query "Regions[].{Name:RegionName}" \
    --output text
```

Output:

```
eu-north-1
ap-south-1
eu-west-3
eu-west-2
eu-west-1
ap-northeast-3
ap-northeast-2
me-south-1
```

```
ap-northeast-1
sa-east-1
ca-central-1
ap-east-1
ap-southeast-1
ap-southeast-2
eu-central-1
us-east-1
us-east-2
us-west-1
us-west-2
```

Per ulteriori informazioni, consulta Regioni e zone nella Amazon EC2 User Guide.

- Per i dettagli sull'API, consulta DescribeRegions AWS CLICommand Reference.

`describe-replace-root-volume-tasks`

Il seguente esempio di codice mostra come utilizzare`describe-replace-root-volume-tasks`.

AWS CLI

Esempio 1: come visualizzare informazioni su una specifica attività di sostituzione del volume root

L'esempio `describe-replace-root-volume-tasks` seguente descrive l'attività di sostituzione del volume root replacevol-0111122223333abcd.

```
aws ec2 describe-replace-root-volume-tasks \
    --replace-root-volume-task-ids replacevol-0111122223333abcd
```

Output:

```
{
    "ReplaceRootVolumeTasks": [
        {
            "ReplaceRootVolumeTaskId": "replacevol-0111122223333abcd",
            "Tags": [],
            "InstanceId": "i-0123456789abcdefa",
            "TaskState": "succeeded",
            "StartTime": "2022-03-14T15:16:28Z",
            "CompleteTime": "2022-03-14T15:16:52Z"
        }
    ]
```

```
}
```

Per ulteriori informazioni, consulta Sostituzione di un volume root nella Guida per l'utente di Amazon Elastic Compute Cloud.

Esempio 2: come visualizzare informazioni su tutte le attività di sostituzione del volume root per un'istanza specifica

L'esempio `describe-replace-root-volume-tasks` seguente descrive tutte le attività di sostituzione del volume root, ad esempio i-0123456789abcdefa.

```
aws ec2 describe-replace-root-volume-tasks \
    --filters Name=instance-id,Values=i-0123456789abcdefa
```

Output:

```
{
    "ReplaceRootVolumeTasks": [
        {
            "ReplaceRootVolumeTaskId": "replacevol-0111122223333abcd",
            "Tags": [],
            "InstanceId": "i-0123456789abcdefa",
            "TaskState": "succeeded",
            "StartTime": "2022-03-14T15:06:38Z",
            "CompleteTime": "2022-03-14T15:07:03Z"
        },
        {
            "ReplaceRootVolumeTaskId": "replacevol-0444455555555abcd",
            "Tags": [],
            "InstanceId": "i-0123456789abcdefa",
            "TaskState": "succeeded",
            "StartTime": "2022-03-14T15:16:28Z",
            "CompleteTime": "2022-03-14T15:16:52Z"
        }
    ]
}
```

Per ulteriori informazioni, consulta Sostituzione di un volume root nella Guida per l'utente di Amazon Elastic Compute Cloud.

- Per i dettagli sull'API, consulta DescribeReplaceRootVolumeTasks AWS CLICommand Reference.

`describe-reserved-instances-listings`

Il seguente esempio di codice mostra come utilizzare`describe-reserved-instances-listings`.

AWS CLI

Come descrivere un elenco di istanze riservate

L'esempio `describe-reserved-instances-listings` seguente recupera le informazioni sull'elenco di istanze riservate.

```
aws ec2 describe-reserved-instances-listings \
    --reserved-instances-listing-id 5ec28771-05ff-4b9b-aa31-9e57dexample
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta DescribeReservedInstancesListings AWS CLICommand Reference.

`describe-reserved-instances-modifications`

Il seguente esempio di codice mostra come utilizzare`describe-reserved-instances-modifications`.

AWS CLI

Come descrivere le modifiche delle istanze riservate

Questo comando di esempio descrive tutte le richieste di modifica delle istanze riservate che sono state inviate per l'account in uso.

Comando:

```
aws ec2 describe-reserved-instances-modifications
```

Output:

```
{
    "ReservedInstancesModifications": [
```

```
        {
            "Status": "fulfilled",
            "ModificationResults": [
                {
                    "ReservedInstancesId": "93bbbca2-62f1-4d9d-b225-16bada29e6c7",
                    "TargetConfiguration": {
                        "AvailabilityZone": "us-east-1b",
                        "InstanceType": "m1.large",
                        "InstanceCount": 3
                    }
                },
                {
                     "ReservedInstancesId": "1ba8e2e3-aabb-46c3-bcf5-3fe2fda922e6",
                     "TargetConfiguration": {
                         "AvailabilityZone": "us-east-1d",
                         "InstanceType": "m1.xlarge",
                         "InstanceCount": 1
                     }
                 }
            ],
            "EffectiveDate": "2015-08-12T17:00:00.000Z",
            "CreateDate": "2015-08-12T17:52:52.630Z",
            "UpdateDate": "2015-08-12T18:08:06.698Z",
            "ClientToken": "c9adb218-3222-4889-8216-0cf0e52dc37e:
            "ReservedInstancesModificationId": "rimod-d3ed4335-b1d3-4de6-
ab31-0f13aaf46687",
            "ReservedInstancesIds": [
                {
                    "ReservedInstancesId": "b847fa93-e282-4f55-b59a-1342f5bd7c02"
                }
            ]
        }
    ]
}
```

- Per i dettagli sull'API, consulta DescribeReservedInstancesModifications AWS CLICommand Reference.

`describe-reserved-instances-offerings`

Il seguente esempio di codice mostra come utilizzare`describe-reserved-instances-offerings`.

AWS CLI

Come descrivere le offerte di istanze riservate

Questo comando di esempio descrive tutte le istanze riservate disponibili per l'acquisto nella Regione.

Comando:

```
aws ec2 describe-reserved-instances-offerings
```

Output:

```
{
  "ReservedInstancesOfferings": [
      {
          "OfferingType": "Partial Upfront",
          "AvailabilityZone": "us-east-1b",
          "InstanceTenancy": "default",
          "PricingDetails": [],
          "ProductDescription": "Red Hat Enterprise Linux",
          "UsagePrice": 0.0,
          "RecurringCharges": [
              {
                  "Amount": 0.088,
                  "Frequency": "Hourly"
              }
          ],
          "Marketplace": false,
          "CurrencyCode": "USD",
          "FixedPrice": 631.0,
          "Duration": 94608000,
          "ReservedInstancesOfferingId": "9a06095a-bdc6-47fe-a94a-2a382f016040",
          "InstanceType": "c1.medium"
      },
      {
          "OfferingType": "PartialUpfront",
          "AvailabilityZone": "us-east-1b",
          "InstanceTenancy": "default",
          "PricingDetails": [],
          "ProductDescription": "Linux/UNIX",
          "UsagePrice": 0.0,
          "RecurringCharges": [
```

```
            {
                "Amount": 0.028,
                "Frequency": "Hourly"
            }
        ],
        "Marketplace": false,
        "CurrencyCode": "USD",
        "FixedPrice": 631.0,
        "Duration": 94608000,
        "ReservedInstancesOfferingId": "bfbefc6c-0d10-418d-b144-7258578d329d",
        "InstanceType": "c1.medium"
    },
  ...
}
```

Come descrivere le offerte di istanze riservate utilizzando le opzioni

Questo esempio elenca le istanze riservate offerte da AWS con le seguenti specifiche: tipi di istanze t1.micro, prodotto Windows (Amazon VPC) e offerte Heavy Utilization.

Comando:

```
aws ec2 describe-reserved-instances-offerings --no-include-marketplace --instance-
type "t1.micro" --product-description "Windows (Amazon VPC)" --offering-type "no
 upfront"
```

Output:

```
{
    "ReservedInstancesOfferings": [
      {
        "OfferingType": "No Upfront",
        "AvailabilityZone": "us-east-1b",
        "InstanceTenancy": "default",
        "PricingDetails": [],
        "ProductDescription": "Windows",
        "UsagePrice": 0.0,
        "RecurringCharges": [
            {
                "Amount": 0.015,
                "Frequency": "Hourly"
            }
```

```
            ],
            "Marketplace": false,
            "CurrencyCode": "USD",
            "FixedPrice": 0.0,
            "Duration": 31536000,
            "ReservedInstancesOfferingId": "c48ab04c-fe69-4f94-8e39-a23842292823",
            "InstanceType": "t1.micro"
        },

                ...
        {
            "OfferingType": "No Upfront",
            "AvailabilityZone": "us-east-1d",
            "InstanceTenancy": "default",
            "PricingDetails": [],
            "ProductDescription": "Windows (Amazon VPC)",
            "UsagePrice": 0.0,
            "RecurringCharges": [
                {
                    "Amount": 0.015,
                    "Frequency": "Hourly"
                }
            ],
            "Marketplace": false,
            "CurrencyCode": "USD",
            "FixedPrice": 0.0,
            "Duration": 31536000,
            "ReservedInstancesOfferingId": "3a98bf7d-2123-42d4-b4f5-8dbec4b06dc6",
            "InstanceType": "t1.micro"
        }
    ]
}
```

- Per i dettagli sull'API, consulta Command Reference. DescribeReservedInstancesOfferingsAWS CLI

`describe-reserved-instances`

Il seguente esempio di codice mostra come utilizzare`describe-reserved-instances`.

AWS CLI

Come descrivere le istanze riservate in uso

Questo comando di esempio descrive le istanze riservate di proprietà dell'utente.

Comando:

```
aws ec2 describe-reserved-instances
```

Output:

```
{
  "ReservedInstances": [
      {
          "ReservedInstancesId": "b847fa93-e282-4f55-b59a-1342fexample",
          "OfferingType": "No Upfront",
          "AvailabilityZone": "us-west-1c",
          "End": "2016-08-14T21:34:34.000Z",
          "ProductDescription": "Linux/UNIX",
          "UsagePrice": 0.00,
          "RecurringCharges": [
              {
                  "Amount": 0.104,
                  "Frequency": "Hourly"
              }
          ],
          "Start": "2015-08-15T21:34:35.086Z",
          "State": "active",
          "FixedPrice": 0.0,
          "CurrencyCode": "USD",
          "Duration": 31536000,
          "InstanceTenancy": "default",
          "InstanceType": "m3.medium",
          "InstanceCount": 2
      },
      ...
  ]
}
```

Come descrivere le istanze riservate utilizzando i filtri

Questo esempio filtra la risposta per includere solo le istanze Linux/UNIX riservate t2.micro di tre anni in us-west-1c.

Comando:

```
aws ec2 describe-reserved-instances --
filters Name=duration,Values=94608000 Name=instance-
type,Values=t2.micro Name=product-description,Values=Linux/UNIX Name=availability-
zone,Values=us-east-1e
```

Output:

```
{
    "ReservedInstances": [
        {
            "ReservedInstancesId": "f127bd27-edb7-44c9-a0eb-0d7e09259af0",
            "OfferingType": "All Upfront",
            "AvailabilityZone": "us-east-1e",
            "End": "2018-03-26T21:34:34.000Z",
            "ProductDescription": "Linux/UNIX",
            "UsagePrice": 0.00,
            "RecurringCharges": [],
            "Start": "2015-03-27T21:34:35.848Z",
            "State": "active",
            "FixedPrice": 151.0,
            "CurrencyCode": "USD",
            "Duration": 94608000,
            "InstanceTenancy": "default",
            "InstanceType": "t2.micro",
            "InstanceCount": 1
        }
    ]
}
```

Per ulteriori informazioni, consulta Using Amazon EC2 Instances nella AWS Command Line Interface User Guide.

- Per i dettagli sull'API, consulta DescribeReservedInstances AWS CLICommand Reference.

`describe-route-tables`

Il seguente esempio di codice mostra come utilizzare`describe-route-tables`.

AWS CLI

Come descrivere le tabelle di routing

L'esempio `describe-route-tables` seguente recupera i dettagli di tutte le tabelle di routing.

```
aws ec2 describe-route-tables
```

Output:

```
{
    "RouteTables": [
        {
            "Associations": [
                {
                    "Main": true,
                    "RouteTableAssociationId": "rtbassoc-0df3f54e06EXAMPLE",
                    "RouteTableId": "rtb-09ba434c1bEXAMPLE"
                }
            ],
            "PropagatingVgws": [],
            "RouteTableId": "rtb-09ba434c1bEXAMPLE",
            "Routes": [
                {
                    "DestinationCidrBlock": "10.0.0.0/16",
                    "GatewayId": "local",
                    "Origin": "CreateRouteTable",
                    "State": "active"
                },
                {
                    "DestinationCidrBlock": "0.0.0.0/0",
                    "NatGatewayId": "nat-06c018cbd8EXAMPLE",
                    "Origin": "CreateRoute",
                    "State": "blackhole"
                }
            ],
            "Tags": [],
            "VpcId": "vpc-0065acced4EXAMPLE",
            "OwnerId": "111122223333"
        },
        {
            "Associations": [
                {
                    "Main": true,
                    "RouteTableAssociationId": "rtbassoc-9EXAMPLE",
                    "RouteTableId": "rtb-a1eec7de"
                }
```

```
            ],
            "PropagatingVgws": [],
            "RouteTableId": "rtb-a1eec7de",
            "Routes": [
                {
                    "DestinationCidrBlock": "172.31.0.0/16",
                    "GatewayId": "local",
                    "Origin": "CreateRouteTable",
                    "State": "active"
                },
                {
                    "DestinationCidrBlock": "0.0.0.0/0",
                    "GatewayId": "igw-fEXAMPLE",
                    "Origin": "CreateRoute",
                    "State": "active"
                }
            ],
            "Tags": [],
            "VpcId": "vpc-3EXAMPLE",
            "OwnerId": "111122223333"
        },
        {
            "Associations": [
                {
                    "Main": false,
                    "RouteTableAssociationId": "rtbassoc-0b100c28b2EXAMPLE",
                    "RouteTableId": "rtb-07a98f76e5EXAMPLE",
                    "SubnetId": "subnet-0d3d002af8EXAMPLE"
                }
            ],
            "PropagatingVgws": [],
            "RouteTableId": "rtb-07a98f76e5EXAMPLE",
            "Routes": [
                {
                    "DestinationCidrBlock": "10.0.0.0/16",
                    "GatewayId": "local",
                    "Origin": "CreateRouteTable",
                    "State": "active"
                },
                {
                    "DestinationCidrBlock": "0.0.0.0/0",
                    "GatewayId": "igw-06cf664d80EXAMPLE",
                    "Origin": "CreateRoute",
                    "State": "active"
```

```
                }
            ],
            "Tags": [],
            "VpcId": "vpc-0065acced4EXAMPLE",
            "OwnerId": "111122223333"
        }
    ]
}
```

Per ulteriori informazioni, consulta Utilizzo delle tabelle di routing nella Guida per l'utente di AWS VPC.

- Per i dettagli sull'API, consulta DescribeRouteTables AWS CLICommand Reference.

`describe-scheduled-instance-availability`

Il seguente esempio di codice mostra come utilizzare`describe-scheduled-instance-availability`.

AWS CLI

Come descrivere una pianificazione disponibile

Questo esempio descrive una pianificazione che si verifica ogni settimana di domenica, a partire dalla data specificata.

Comando:

```
aws ec2 describe-scheduled-instance-availability --
recurrence Frequency=Weekly,Interval=1,OccurrenceDays=[1] --first-slot-start-time-
range EarliestTime=2016-01-31T00:00:00Z,LatestTime=2016-01-31T04:00:00Z
```

Output:

```
{
  "ScheduledInstanceAvailabilitySet": [
    {
        "AvailabilityZone": "us-west-2b",
        "TotalScheduledInstanceHours": 1219,
        "PurchaseToken": "eyJ2IjoiMSIsInMiOjEsImMiOi...",
        "MinTermDurationInDays": 366,
```

```
        "AvailableInstanceCount": 20,
        "Recurrence": {
            "OccurrenceDaySet": [
                1
            ],
            "Interval": 1,
            "Frequency": "Weekly",
            "OccurrenceRelativeToEnd": false
        },
        "Platform": "Linux/UNIX",
        "FirstSlotStartTime": "2016-01-31T00:00:00Z",
        "MaxTermDurationInDays": 366,
        "SlotDurationInHours": 23,
        "NetworkPlatform": "EC2-VPC",
        "InstanceType": "c4.large",
        "HourlyPrice": "0.095"
    },
    ...
  ]
}
```

Per limitare i risultati, è possibile aggiungere filtri che specificano il sistema operativo, la rete e il tipo di istanza.

Comando:

--filters name=platform, values=Linux/UNIX nome=Piattaforma di rete, Valori= -VPCName=tipo-istanza, valori=C4.large EC2

- Per i dettagli sull'API, consulta Command Reference DescribeScheduledInstanceAvailability.AWS CLI

`describe-scheduled-instances`

Il seguente esempio di codice mostra come utilizzare`describe-scheduled-instances`.

AWS CLI

Come descrivere le istanze pianificate

Questo esempio descrive l'istanza pianificata specificata.

Comando:

```
aws ec2 describe-scheduled-instances --scheduled-instance-
ids sci-1234-1234-1234-1234-123456789012
```

Output:

```
{
  "ScheduledInstanceSet": [
      {
          "AvailabilityZone": "us-west-2b",
          "ScheduledInstanceId": "sci-1234-1234-1234-1234-123456789012",
          "HourlyPrice": "0.095",
          "CreateDate": "2016-01-25T21:43:38.612Z",
          "Recurrence": {
              "OccurrenceDaySet": [
                  1
              ],
              "Interval": 1,
              "Frequency": "Weekly",
              "OccurrenceRelativeToEnd": false,
              "OccurrenceUnit": ""
          },
          "Platform": "Linux/UNIX",
          "TermEndDate": "2017-01-31T09:00:00Z",
          "InstanceCount": 1,
          "SlotDurationInHours": 32,
          "TermStartDate": "2016-01-31T09:00:00Z",
          "NetworkPlatform": "EC2-VPC",
          "TotalScheduledInstanceHours": 1696,
          "NextSlotStartTime": "2016-01-31T09:00:00Z",
          "InstanceType": "c4.large"
      }
  ]
}
```

Questo esempio descrive tutte le istanze pianificate.

Comando:

```
aws ec2 describe-scheduled-instances
```

- Per i dettagli sull'API, consulta DescribeScheduledInstances AWS CLICommand Reference.

`describe-security-group-references`

Il seguente esempio di codice mostra come utilizzare`describe-security-group-references`.

AWS CLI

Come descrivere i riferimenti dei gruppi di sicurezza

Questo esempio descrive i riferimenti dei gruppi di sicurezza per`sg-bbbb2222`. La risposta indica che un gruppo di sicurezza `sg-bbbb2222` fa riferimento a un gruppo di sicurezza nel VPC `vpc-aaaaaaaa`:

Comando:

```
aws ec2 describe-security-group-references --group-id sg-bbbbb22222
```

Output:

```
{
  "SecurityGroupsReferenceSet": [
    {
      "ReferencingVpcId": "vpc-aaaaaaaa ",
      "GroupId": "sg-bbbbb22222",
      "VpcPeeringConnectionId": "pcx-b04deed9"
    }
  ]
}
```

- Per i dettagli sull'API, consulta [DescribeSecurityGroupReferences AWS CLI](Command Reference).

`describe-security-group-rules`

Il seguente esempio di codice mostra come utilizzare`describe-security-group-rules`.

AWS CLI

Esempio 1: come descrivere le regole del gruppo di sicurezza per un gruppo di sicurezza

L'esempio `describe-security-group-rules` seguente descrive le regole del gruppo di sicurezza di un gruppo di sicurezza specificato. Utilizza l'opzione `filters` per limitare i risultati a un gruppo di sicurezza specifico.

```
aws ec2 describe-security-group-rules \
    --filters Name="group-id",Values="sg-1234567890abcdef0"
```

Output:


```
{
    "SecurityGroupRules": [
        {
            "SecurityGroupRuleId": "sgr-abcdef01234567890",
            "GroupId": "sg-1234567890abcdef0",
            "GroupOwnerId": "111122223333",
            "IsEgress": false,
            "IpProtocol": "-1",
            "FromPort": -1,
            "ToPort": -1,
            "ReferencedGroupInfo": {
                "GroupId": "sg-1234567890abcdef0",
                "UserId": "111122223333"
            },
            "Tags": []
        },
        {
            "SecurityGroupRuleId": "sgr-bcdef01234567890a",
            "GroupId": "sg-1234567890abcdef0",
            "GroupOwnerId": "111122223333",
            "IsEgress": true,
            "IpProtocol": "-1",
            "FromPort": -1,
            "ToPort": -1,
            "CidrIpv6": "::/0",
            "Tags": []
        },
        {
            "SecurityGroupRuleId": "sgr-cdef01234567890ab",
            "GroupId": "sg-1234567890abcdef0",
            "GroupOwnerId": "111122223333",
            "IsEgress": true,
            "IpProtocol": "-1",
            "FromPort": -1,
            "ToPort": -1,
            "CidrIpv4": "0.0.0.0/0",
            "Tags": []
        }
```

```
    ]
}
```

Esempio 2: come descrivere una regola del gruppo di sicurezza

L'esempio `describe-security-group-rules` seguente descrive la regola del gruppo di sicurezza specificata.

```
aws ec2 describe-security-group-rules \
    --security-group-rule-ids sgr-cdef01234567890ab
```

Output:

```
{
    "SecurityGroupRules": [
        {
            "SecurityGroupRuleId": "sgr-cdef01234567890ab",
            "GroupId": "sg-1234567890abcdef0",
            "GroupOwnerId": "111122223333",
            "IsEgress": true,
            "IpProtocol": "-1",
            "FromPort": -1,
            "ToPort": -1,
            "CidrIpv4": "0.0.0.0/0",
            "Tags": []
        }
    ]
}
```

Per ulteriori informazioni, consulta Regole del gruppo di sicurezza nella Guida per l'utente di Amazon VPC.

- Per i dettagli sull'API, consulta DescribeSecurityGroupRules AWS CLICommand Reference.

`describe-security-group-vpc-associations`

Il seguente esempio di codice mostra come utilizzare`describe-security-group-vpc-associations`.

AWS CLI

Come descrivere le associazioni VPC

L'esempio `describe-security-group-vpc-associations` seguente descrive le associazioni VPC per il gruppo di sicurezza specificato.

```
aws ec2 describe-security-group-vpc-associations \
    --filters Name=group-id,Values=sg-04dbb43907d3f8a78
```

Output:

```
{
    "SecurityGroupVpcAssociations": [
        {
            "GroupId": "sg-04dbb43907d3f8a78",
            "VpcId": "vpc-0bf4c2739bc05a694",
            "VpcOwnerId": "123456789012",
            "State": "associated"
        }
    ]
}
```

Per ulteriori informazioni, consulta Associare gruppi di sicurezza a più gruppi VPCs nella Amazon VPC User Guide.

- Per i dettagli sull'API, consulta DescribeSecurityGroupVpcAssociations AWS CLICommand Reference.

`describe-security-groups`

Il seguente esempio di codice mostra come utilizzare`describe-security-groups`.

AWS CLI

Esempio 1: come descrivere un gruppo di sicurezza

Nell'esempio di `describe-security-groups` seguente viene descritto il gruppo di sicurezza specificato.

```
aws ec2 describe-security-groups \
    --group-ids sg-903004f8
```

Output:

```
{
```

```
    "SecurityGroups": [
        {
            "IpPermissionsEgress": [
                {
                    "IpProtocol": "-1",
                    "IpRanges": [
                        {
                            "CidrIp": "0.0.0.0/0"
                        }
                    ],
                    "UserIdGroupPairs": [],
                    "PrefixListIds": []
                }
            ],
            "Description": "My security group",
            "Tags": [
                {
                    "Value": "SG1",
                    "Key": "Name"
                }
            ],
            "IpPermissions": [
                {
                    "IpProtocol": "-1",
                    "IpRanges": [],
                    "UserIdGroupPairs": [
                        {
                            "UserId": "123456789012",
                            "GroupId": "sg-903004f8"
                        }
                    ],
                    "PrefixListIds": []
                },
                {
                    "PrefixListIds": [],
                    "FromPort": 22,
                    "IpRanges": [
                        {
                            "Description": "Access from NY office",
                            "CidrIp": "203.0.113.0/24"
                        }
                    ],
                    "ToPort": 22,
                    "IpProtocol": "tcp",
```

```
                    "UserIdGroupPairs": []
                    }
            ],
            "GroupName": "MySecurityGroup",
            "VpcId": "vpc-1a2b3c4d",
            "OwnerId": "123456789012",
            "GroupId": "sg-903004f8",
        }
    ]
}
```

Esempio 2: come descrivere gruppi di sicurezza con regole specifiche

L'esempio `describe-security-groups` seguente utilizza filtri per definire l'ambito dei risultati in base ai gruppi di sicurezza che includono una regola che consente il traffico SSH (porta 22) e una regola che consente il traffico da tutti gli indirizzi (`0.0.0.0/0`). Nell'esempio viene utilizzato il parametro `--query` per visualizzare solamente i nomi dei gruppi di sicurezza. I gruppi di sicurezza devono corrispondere a tutti i filtri per essere restituiti nei risultati; tuttavia, una singola regola non deve corrispondere a tutti i filtri. Per esempio, l'output restituisce un gruppo di sicurezza con una regola che consente il traffico SSH da un indirizzo IP specifico e un'altra regola che consente il traffico HTTP da tutti gli indirizzi.

```
aws ec2 describe-security-groups \
    --filters Name=ip-permission.from-port,Values=22 Name=ip-permission.to-
port,Values=22 Name=ip-permission.cidr,Values='0.0.0.0/0' \
    --query "SecurityGroups[*].[GroupName]" \
    --output text
```

Output:

```
default
my-security-group
web-servers
launch-wizard-1
```

Esempio 3: come descrivere gruppi di sicurezza in base ai tag

Nell'esempio di `describe-security-groups` seguente vengono utilizzati filtri per rifinire i risultati in base ai gruppi di sicurezza che includono `test` nel nome del gruppo di sicurezza e che hanno il tag `Test=To-delete`. L'esempio utilizza il `--query` parametro per visualizzare solo i nomi e i gruppi IDs di sicurezza.

```
aws ec2 describe-security-groups \
    --filters Name=group-name,Values=*test* Name=tag:Test,Values=To-delete \
    --query "SecurityGroups[*].{Name:GroupName,ID:GroupId}"
```

Output:

```
[
    {
        "Name": "testfornewinstance",
        "ID": "sg-33bb22aa"
    },
    {
        "Name": "newgrouptest",
        "ID": "sg-1a2b3c4d"
    }
]
```

Per ulteriori esempi di utilizzo dei filtri di tag, consulta Lavorare con i tag nella Amazon EC2 User Guide.

- Per i dettagli sull'API, consulta DescribeSecurityGroups AWS CLICommand Reference.

`describe-snapshot-attribute`

Il seguente esempio di codice mostra come utilizzare`describe-snapshot-attribute`.

AWS CLI

Come descrivere gli attributi di uno snapshot

L'esempio `describe-snapshot-attribute` seguente elenca gli account con cui è condiviso uno snapshot.

```
aws ec2 describe-snapshot-attribute \
    --snapshot-id snap-01234567890abcedf \
    --attribute createVolumePermission
```

Output:

```
{
    "SnapshotId": "snap-01234567890abcedf",
```

```
    "CreateVolumePermissions": [
        {
            "UserId": "123456789012"
        }
    ]
}
```

Per ulteriori informazioni, consulta Condivisione di uno snapshot Amazon EBS nella Guida per l'utente di Amazon Elastic Compute Cloud.

- Per i dettagli sull'API, consulta DescribeSnapshotAttribute AWS CLICommand Reference.

`describe-snapshot-tier-status`

Il seguente esempio di codice mostra come utilizzare`describe-snapshot-tier-status`.

AWS CLI

Come visualizzare le informazioni di archiviazione su uno snapshot archiviato

L'esempio `describe-snapshot-tier-status` seguente fornisce informazioni di archiviazione su uno snapshot archiviato.

```
aws ec2 describe-snapshot-tier-status \
    --filters "Name=snapshot-id, Values=snap-01234567890abcedf"
```

Output:

```
{
    "SnapshotTierStatuses": [
        {
            "Status": "completed",
            "ArchivalCompleteTime": "2021-09-15T17:33:16.147Z",
            "LastTieringProgress": 100,
            "Tags": [],
            "VolumeId": "vol-01234567890abcedf",
            "LastTieringOperationState": "archival-completed",
            "StorageTier": "archive",
            "OwnerId": "123456789012",
            "SnapshotId": "snap-01234567890abcedf",
            "LastTieringStartTime": "2021-09-15T16:44:37.574Z"
        }
```

```
    ]
}
```

Per ulteriori informazioni, consulta Visualizzazione degli snapshot archiviati nella Guida per l'utente di Amazon Elastic Compute Cloud.

- Per i dettagli sull'API, consulta DescribeSnapshotTierStatus AWS CLICommand Reference.

`describe-snapshots`

Il seguente esempio di codice mostra come utilizzare`describe-snapshots`.

AWS CLI

Esempio 1: come descrivere uno snapshot

Nell'esempio di `describe-snapshots` seguente viene descritto lo snapshot specificato.

```
aws ec2 describe-snapshots \
    --snapshot-ids snap-1234567890abcdef0
```

Output:

```
{
    "Snapshots": [
        {
            "Description": "This is my snapshot",
            "Encrypted": false,
            "VolumeId": "vol-049df61146c4d7901",
            "State": "completed",
            "VolumeSize": 8,
            "StartTime": "2019-02-28T21:28:32.000Z",
            "Progress": "100%",
            "OwnerId": "012345678910",
            "SnapshotId": "snap-01234567890abcdef",
            "Tags": [
                {
                    "Key": "Stack",
                    "Value": "test"
                }
            ]
        }
    ]
```

```
}
```

Per ulteriori informazioni, consulta gli snapshot di Amazon EBS nella Amazon EC2 User Guide.

Esempio 2: come descrivere snapshot in base ai filtri

L'`describe-snapshots`esempio seguente utilizza filtri per limitare i risultati agli snapshot di proprietà del tuo AWS account che si trovano nello stato. `pending` L'esempio utilizza il `--query` parametro per visualizzare solo l'istantanea IDs e l'ora di avvio dell'istantanea.

```
aws ec2 describe-snapshots \
    --owner-ids self \
    --filters Name=status,Values=pending \
    --query "Snapshots[*].{ID:SnapshotId,Time:StartTime}"
```

Output:

```
[
    {
        "ID": "snap-1234567890abcdef0",
        "Time": "2019-08-04T12:48:18.000Z"
    },
    {
        "ID": "snap-066877671789bd71b",
        "Time": "2019-08-04T02:45:16.000Z
    },
    ...
]
```

Nell'esempio di `describe-snapshots` seguente vengono utilizzati filtri per rifinire i risultati in base agli snapshot creati dal volume specificato. L'esempio utilizza il `--query` parametro per visualizzare solo l'istantanea. IDs

```
aws ec2 describe-snapshots \
    --filters Name=volume-id,Values=049df61146c4d7901 \
    --query "Snapshots[*].[SnapshotId]" \
    --output text
```

Output:

```
snap-1234567890abcdef0
```

```
snap-08637175a712c3fb9
...
```

Per ulteriori esempi di utilizzo dei filtri, consulta Elencare e filtrare le risorse nella Amazon EC2 User Guide.

Esempio 3: come descrivere snapshot in base ai tag

Nell'esempio di `describe-snapshots` seguente vengono utilizzati filtri per rifinire i risultati in base agli snapshot che hanno il tag `Stack=Prod`.

```
aws ec2 describe-snapshots \
    --filters Name=tag:Stack,Values=prod
```

Per un esempio dell'output di `describe-snapshots`, vedi l'Esempio 1.

Per ulteriori esempi di utilizzo dei filtri di tag, consulta Lavorare con i tag nella Amazon EC2 User Guide.

Esempio 4: come descrivere snapshot in base all'età

L'`describe-snapshots`esempio seguente utilizza JMESPath espressioni per descrivere tutte le istantanee create dal tuo AWS account prima della data specificata. Visualizza solo l'istantanea IDs.

```
aws ec2 describe-snapshots \
    --owner-ids 012345678910 \
    --query "Snapshots[?(StartTime<='2020-03-31')].[SnapshotId]"
```

Per ulteriori esempi di utilizzo dei filtri, consulta Elencare e filtrare le risorse nella Amazon EC2 User Guide.

Esempio 5: come visualizzare solo gli snapshot archiviati

Nell'esempio di `describe-snapshots` seguente vengono elencati solo gli snapshot memorizzati nel livello archivio.

```
aws ec2 describe-snapshots \
    --filters "Name=storage-tier,Values=archive"
```

Output:

```
{
    "Snapshots": [
        {
            "Description": "Snap A",
            "Encrypted": false,
            "VolumeId": "vol-01234567890aaaaaa",
            "State": "completed",
            "VolumeSize": 8,
            "StartTime": "2021-09-07T21:00:00.000Z",
            "Progress": "100%",
            "OwnerId": "123456789012",
            "SnapshotId": "snap-01234567890aaaaaa",
            "StorageTier": "archive",
            "Tags": []
        },
    ]
}
```

Per ulteriori informazioni, consulta Visualizzazione degli snapshot archiviati nella Guida per l'utente di Amazon Elastic Compute Cloud.

- Per i dettagli sull'API, consulta DescribeSnapshots AWS CLICommand Reference.

`describe-spot-datafeed-subscription`

Il seguente esempio di codice mostra come utilizzare`describe-spot-datafeed-subscription`.

AWS CLI

Come descrivere la sottoscrizione ai feed di dati dell'istanza spot per un account

Questo comando di esempio descrive il feed di dati per l'account.

Comando:

```
aws ec2 describe-spot-datafeed-subscription
```

Output:

```
{
    "SpotDatafeedSubscription": {
        "OwnerId": "123456789012",
        "Prefix": "spotdata",
```

```
        "Bucket": "amzn-s3-demo-bucket",
        "State": "Active"
    }
}
```

- Per i dettagli sull'API, consulta DescribeSpotDatafeedSubscription AWS CLICommand Reference.

`describe-spot-fleet-instances`

Il seguente esempio di codice mostra come utilizzare`describe-spot-fleet-instances`.

AWS CLI

Come descrivere le istanze spot associate a un parco istanze spot

Questo comando di esempio elenca le istanze spot associate al parco istanze spot specificato.

Comando:

```
aws ec2 describe-spot-fleet-instances --spot-fleet-request-id sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE
```

Output:

```
{
  "ActiveInstances": [
      {
          "InstanceId": "i-1234567890abcdef0",
          "InstanceType": "m3.medium",
          "SpotInstanceRequestId": "sir-08b93456"
      },
      ...
  ],
  "SpotFleetRequestId": "sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE"
}
```

- Per i dettagli sull'API, consulta DescribeSpotFleetInstances AWS CLICommand Reference.

`describe-spot-fleet-request-history`

Il seguente esempio di codice mostra come utilizzare`describe-spot-fleet-request-history`.

AWS CLI

Come descrivere la cronologia del parco istanze spot

Questo comando di esempio restituisce la cronologia del parco istanze spot specificato a partire dall'ora specificata.

Comando:

```
aws ec2 describe-spot-fleet-request-history --spot-fleet-request-id sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE --start-time 2015-05-26T00:00:00Z
```

L'output di esempio seguente mostra gli avvii riusciti di due istanze spot del parco istanze spot.

Output:

```
{
  "HistoryRecords": [
      {
          "Timestamp": "2015-05-26T23:17:20.697Z",
          "EventInformation": {
              "EventSubType": "submitted"
          },
          "EventType": "fleetRequestChange"
      },
      {
          "Timestamp": "2015-05-26T23:17:20.873Z",
          "EventInformation": {
              "EventSubType": "active"
          },
          "EventType": "fleetRequestChange"
      },
      {
          "Timestamp": "2015-05-26T23:21:21.712Z",
          "EventInformation": {
              "InstanceId": "i-1234567890abcdef0",
              "EventSubType": "launched"
          },
          "EventType": "instanceChange"
      },
      {
          "Timestamp": "2015-05-26T23:21:21.816Z",
          "EventInformation": {
              "InstanceId": "i-1234567890abcdef1",
```

```
                "EventSubType": "launched"
            },
            "EventType": "instanceChange"
        }
    ],
    "SpotFleetRequestId": "sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE",
    "NextToken": "CpHNsscimcV5oH7bSbub03CI2Qms5+ypNpNm
+53MNlR0YcXAkp0xFlfKf91yVxSExmbtma3awYxMFzNA663ZskT0AHtJ6TCb2Z8bQC2EnZgyELbymtWPfpZ1ZbauVg
+P+TfGlWxWWB/Vr5dk5d4LfdgA/DRAHUrYgxzrEXAMPLE=",
    "StartTime": "2015-05-26T00:00:00Z"
}
```

- Per i dettagli sull'API, consulta [DescribeSpotFleetRequestHistory AWS CLI](Command Reference.

`describe-spot-fleet-requests`

Il seguente esempio di codice mostra come utilizzare`describe-spot-fleet-requests`.

AWS CLI

Come descrivere le richieste del parco istanze spot

Questo esempio descrive tutte le richieste del parco istanze spot.

Comando:

```
aws ec2 describe-spot-fleet-requests
```

Output:

```
{
    "SpotFleetRequestConfigs": [
        {
            "SpotFleetRequestId": "sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE",
            "SpotFleetRequestConfig": {
                "TargetCapacity": 20,
                "LaunchSpecifications": [
                    {
                        "EbsOptimized": false,
                        "NetworkInterfaces": [
                            {
                                "SubnetId": "subnet-a61dafcf",
```

```
                            "DeviceIndex": 0,
                            "DeleteOnTermination": false,
                            "AssociatePublicIpAddress": true,
                            "SecondaryPrivateIpAddressCount": 0
                        }
                    ],
                    "InstanceType": "cc2.8xlarge",
                    "ImageId": "ami-1a2b3c4d"
                },
                {
                    "EbsOptimized": false,
                    "NetworkInterfaces": [
                        {
                            "SubnetId": "subnet-a61dafcf",
                            "DeviceIndex": 0,
                            "DeleteOnTermination": false,
                            "AssociatePublicIpAddress": true,
                            "SecondaryPrivateIpAddressCount": 0
                        }
                    ],
                    "InstanceType": "r3.8xlarge",
                    "ImageId": "ami-1a2b3c4d"
                }
            ],
            "SpotPrice": "0.05",
            "IamFleetRole": "arn:aws:iam::123456789012:role/my-spot-fleet-role"
        },
        "SpotFleetRequestState": "active"
    },
    {
        "SpotFleetRequestId": "sfr-306341ed-9739-402e-881b-ce47bEXAMPLE",
        "SpotFleetRequestConfig": {
            "TargetCapacity": 20,
            "LaunchSpecifications": [
                {
                    "EbsOptimized": false,
                    "NetworkInterfaces": [
                        {
                            "SubnetId": "subnet-6e7f829e",
                            "DeviceIndex": 0,
                            "DeleteOnTermination": false,
                            "AssociatePublicIpAddress": true,
                            "SecondaryPrivateIpAddressCount": 0
                        }
```

```
                    ],
                    "InstanceType": "m3.medium",
                    "ImageId": "ami-1a2b3c4d"
                }
            ],
            "SpotPrice": "0.05",
            "IamFleetRole": "arn:aws:iam::123456789012:role/my-spot-fleet-role"
        },
        "SpotFleetRequestState": "active"
    }
  ]
}
```

Come descrivere la richiesta del parco istanze spot

Questo esempio descrive la richiesta del parco istanze spot specificata.

Comando:

```
aws ec2 describe-spot-fleet-requests --spot-fleet-request-ids sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE
```

Output:

```
{
  "SpotFleetRequestConfigs": [
      {
          "SpotFleetRequestId": "sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE",
          "SpotFleetRequestConfig": {
              "TargetCapacity": 20,
              "LaunchSpecifications": [
                  {
                      "EbsOptimized": false,
                      "NetworkInterfaces": [
                          {
                              "SubnetId": "subnet-a61dafcf",
                              "DeviceIndex": 0,
                              "DeleteOnTermination": false,
                              "AssociatePublicIpAddress": true,
                              "SecondaryPrivateIpAddressCount": 0
                          }
                      ],
                      "InstanceType": "cc2.8xlarge",
```

```
                    "ImageId": "ami-1a2b3c4d"
                },
                {
                    "EbsOptimized": false,
                    "NetworkInterfaces": [
                        {
                            "SubnetId": "subnet-a61dafcf",
                            "DeviceIndex": 0,
                            "DeleteOnTermination": false,
                            "AssociatePublicIpAddress": true,
                            "SecondaryPrivateIpAddressCount": 0
                        }
                    ],
                    "InstanceType": "r3.8xlarge",
                    "ImageId": "ami-1a2b3c4d"
                }
            ],
            "SpotPrice": "0.05",
            "IamFleetRole": "arn:aws:iam::123456789012:role/my-spot-fleet-role"
        },
        "SpotFleetRequestState": "active"
    }
  ]
}
```

- Per i dettagli sull'API, consulta DescribeSpotFleetRequests AWS CLICommand Reference.

`describe-spot-instance-requests`

Il seguente esempio di codice mostra come utilizzare`describe-spot-instance-requests`.

AWS CLI

Esempio 1: come descrivere una richiesta di istanze spot

L'esempio `describe-spot-instance-requests` seguente descrive la richiesta di istanza spot specificata.

```
aws ec2 describe-spot-instance-requests \
    --spot-instance-request-ids sir-08b93456
```

Output:

```
{
    "SpotInstanceRequests": [
        {
            "CreateTime": "2018-04-30T18:14:55.000Z",
            "InstanceId": "i-1234567890abcdef1",
            "LaunchSpecification": {
                "InstanceType": "t2.micro",
                "ImageId": "ami-003634241a8fcdec0",
                "KeyName": "my-key-pair",
                "SecurityGroups": [
                    {
                        "GroupName": "default",
                        "GroupId": "sg-e38f24a7"
                    }
                ],
                "BlockDeviceMappings": [
                    {
                        "DeviceName": "/dev/sda1",
                        "Ebs": {
                            "DeleteOnTermination": true,
                            "SnapshotId": "snap-0e54a519c999adbbd",
                            "VolumeSize": 8,
                            "VolumeType": "standard",
                            "Encrypted": false
                        }
                    }
                ],
                "NetworkInterfaces": [
                    {
                        "DeleteOnTermination": true,
                        "DeviceIndex": 0,
                        "SubnetId": "subnet-049df61146c4d7901"
                    }
                ],
                "Placement": {
                    "AvailabilityZone": "us-east-2b",
                    "Tenancy": "default"
                },
                "Monitoring": {
                    "Enabled": false
                }
            },
            "LaunchedAvailabilityZone": "us-east-2b",
```

```
            "ProductDescription": "Linux/UNIX",
            "SpotInstanceRequestId": "sir-08b93456",
            "SpotPrice": "0.010000"
            "State": "active",
            "Status": {
                "Code": "fulfilled",
                "Message": "Your Spot request is fulfilled.",
                "UpdateTime": "2018-04-30T18:16:21.000Z"
            },
            "Tags": [],
            "Type": "one-time",
            "InstanceInterruptionBehavior": "terminate"
        }
    ]
}
```

Esempio 2: come descrivere le richieste di istanze spot in base ai filtri

L'esempio `describe-spot-instance-requests` seguente utilizza i filtri per definire l'ambito dei risultati in base alle richieste di istanze spot con il tipo di istanza specificato nella zona di disponibilità specificata. L'esempio utilizza il `--query` parametro per visualizzare solo l'istanza IDs.

```
aws ec2 describe-spot-instance-requests \
    --filters Name=launch.instance-type,Values=m3.medium Name=launched-availability-zone,Values=us-east-2a \
    --query "SpotInstanceRequests[*].[InstanceId]" \
    --output text
```

Output:

```
i-057750d42936e468a
i-001efd250faaa6ffa
i-027552a73f021f3bd
...
```

Per ulteriori esempi di utilizzo dei filtri, consulta Elenco e filtraggio delle risorse nella Guida per l'utente di Amazon Elastic Compute Cloud.

Esempio 3: come descrivere le richieste di istanze spot in base ai tag

L'esempio `describe-spot-instance-requests` seguente utilizza i filtri di tag per definire l'ambito dei risultati in base alle richieste di istanze spot associate al tag `cost-center=cc123`.

```
aws ec2 describe-spot-instance-requests \
    --filters Name=tag:cost-center,Values=cc123
```

Per un esempio dell'output di `describe-spot-instance-requests`, vedi l'Esempio 1.

Per ulteriori esempi di utilizzo dei filtri di tag, consulta Lavorare con i tag nella Amazon EC2 User Guide.

- Per i dettagli sull'API, consulta DescribeSpotInstanceRequests AWS CLICommand Reference.

`describe-spot-price-history`

Il seguente esempio di codice mostra come utilizzare`describe-spot-price-history`.

AWS CLI

Come descrivere la cronologia dei prezzi spot

Questo comando di esempio restituisce la cronologia dei prezzi spot per le istanze m1.xlarge per un giorno di gennaio specifico.

Comando:

```
aws ec2 describe-spot-price-history --instance-types m1.xlarge --start-
time 2014-01-06T07:08:09 --end-time 2014-01-06T08:09:10
```

Output:

```
{
  "SpotPriceHistory": [
          {
              "Timestamp": "2014-01-06T07:10:55.000Z",
              "ProductDescription": "SUSE Linux",
              "InstanceType": "m1.xlarge",
              "SpotPrice": "0.087000",
              "AvailabilityZone": "us-west-1b"
          },
          {
```

```
            "Timestamp": "2014-01-06T07:10:55.000Z",
            "ProductDescription": "SUSE Linux",
            "InstanceType": "m1.xlarge",
            "SpotPrice": "0.087000",
            "AvailabilityZone": "us-west-1c"
        },
        {
            "Timestamp": "2014-01-06T05:42:36.000Z",
            "ProductDescription": "SUSE Linux (Amazon VPC)",
            "InstanceType": "m1.xlarge",
            "SpotPrice": "0.087000",
            "AvailabilityZone": "us-west-1a"
    },
    ...
}
```

Per descrivere la cronologia dei prezzi Spot per Linux/UNIX Amazon VPC

Questo comando di esempio restituisce lo storico dei prezzi Spot per m1.xlarge, istanze Amazon Linux/UNIX VPC per un determinato giorno di gennaio.

Comando:

```
aws ec2 describe-spot-price-history --instance-types m1.xlarge --product-
description "Linux/UNIX (Amazon VPC)" --start-time 2014-01-06T07:08:09 --end-
time 2014-01-06T08:09:10
```

Output:

```
{
  "SpotPriceHistory": [
      {
          "Timestamp": "2014-01-06T04:32:53.000Z",
          "ProductDescription": "Linux/UNIX (Amazon VPC)",
          "InstanceType": "m1.xlarge",
          "SpotPrice": "0.080000",
          "AvailabilityZone": "us-west-1a"
      },
      {
          "Timestamp": "2014-01-05T11:28:26.000Z",
          "ProductDescription": "Linux/UNIX (Amazon VPC)",
          "InstanceType": "m1.xlarge",
          "SpotPrice": "0.080000",
```

```
            "AvailabilityZone": "us-west-1c"
        }
    ]
}
```

- Per i dettagli sull'API, consulta Command Reference. DescribeSpotPriceHistoryAWS CLI

`describe-stale-security-groups`

Il seguente esempio di codice mostra come utilizzare`describe-stale-security-groups`.

AWS CLI

Come descrivere i gruppi di sicurezza obsoleti

Questo esempio descrive le regole del gruppo di sicurezza obsoleto per `vpc-11223344`. La risposta mostra che sg-5fa68d3a nell'account in uso ha una regola SSH di ingresso obsoleta che fa riferimento a `sg-279ab042` nel VPC peer e che `sg-fe6fba9a` nell'account in uso ha una regola SSH in uscita non aggiornata che fa riferimento a `sg-ef6fba8b` nel VPC peer.

Comando:

```
aws ec2 describe-stale-security-groups --vpc-id vpc-11223344
```

Output:

```
{
  "StaleSecurityGroupSet": [
      {
          "VpcId": "vpc-11223344",
          "StaleIpPermissionsEgress": [
              {
                  "ToPort": 22,
                  "FromPort": 22,
                  "UserIdGroupPairs": [
                      {
                          "VpcId": "vpc-7a20e51f",
                          "GroupId": "sg-ef6fba8b",
                          "VpcPeeringConnectionId": "pcx-b04deed9",
                          "PeeringStatus": "active"
                      }
                  ],
```

```
                "IpProtocol": "tcp"
            }
        ],
        "GroupName": "MySG1",
        "StaleIpPermissions": [],
        "GroupId": "sg-fe6fba9a",
        "Description": MySG1"
    },
    {
        "VpcId": "vpc-11223344",
        "StaleIpPermissionsEgress": [],
        "GroupName": "MySG2",
        "StaleIpPermissions": [
            {
                "ToPort": 22,
                "FromPort": 22,
                "UserIdGroupPairs": [
                    {
                        "VpcId": "vpc-7a20e51f",
                        "GroupId": "sg-279ab042",
                        "Description": "Access from pcx-b04deed9",
                        "VpcPeeringConnectionId": "pcx-b04deed9",
                        "PeeringStatus": "active"
                    }
                ],
                "IpProtocol": "tcp"
            }
        ],
        "GroupId": "sg-5fa68d3a",
        "Description": "MySG2"
    }
  ]
}
```

- Per i dettagli sull'API, consulta DescribeStaleSecurityGroups AWS CLICommand Reference.

`describe-store-image-tasks`

Il seguente esempio di codice mostra come utilizzare`describe-store-image-tasks`.

AWS CLI

Come descrivere lo stato di avanzamento di un'attività di archiviazione AMI

L'esempio `describe-store-image-tasks` seguente descrive lo stato di avanzamento di un'attività di archiviazione AMI.

```
aws ec2 describe-store-image-tasks
```

Output:

```
{
    "StoreImageTaskResults": [
        {
            "AmiId": "ami-1234567890abcdef0",
            "Bucket": "my-ami-bucket",
            "ProgressPercentage": 17,
            "S3objectKey": "ami-1234567890abcdef0.bin",
            "StoreTaskState": "InProgress",
            "StoreTaskFailureReason": null,
            "TaskStartTime": "2022-01-01T01:01:01.001Z"
        }
    ]
}
```

Per ulteriori informazioni sull'archiviazione e il ripristino di un'AMI utilizzando S3, consulta Archiviazione e ripristino di un'AMI utilizzando S3 <https://docs.aws.amazon.com/AWS EC2/latest/UserGuide/ami-store-restore.html> nella Amazon User Guide. EC2

- Per i DescribeStoreImageTasksdettagli sull'API, consulta Command Reference.AWS CLI

`describe-subnets`

Il seguente esempio di codice mostra come utilizzare`describe-subnets`.

AWS CLI

Esempio 1: come descrivere tutte le sottoreti

Nell'esempio di `describe-subnets` seguente vengono visualizzati i dettagli delle sottoreti.

```
aws ec2 describe-subnets
```

Output:

```
{
```

```
    "Subnets": [
        {
            "AvailabilityZone": "us-east-1d",
            "AvailabilityZoneId": "use1-az2",
            "AvailableIpAddressCount": 4089,
            "CidrBlock": "172.31.80.0/20",
            "DefaultForAz": true,
            "MapPublicIpOnLaunch": false,
            "MapCustomerOwnedIpOnLaunch": true,
            "State": "available",
            "SubnetId": "subnet-0bb1c79de3EXAMPLE",
            "VpcId": "vpc-0ee975135dEXAMPLE",
            "OwnerId": "111122223333",
            "AssignIpv6AddressOnCreation": false,
            "Ipv6CidrBlockAssociationSet": [],
            "CustomerOwnedIpv4Pool:": 'pool-2EXAMPLE',
            "SubnetArn": "arn:aws:ec2:us-east-2:111122223333:subnet/
subnet-0bb1c79de3EXAMPLE",
            "EnableDns64": false,
            "Ipv6Native": false,
            "PrivateDnsNameOptionsOnLaunch": {
                "HostnameType": "ip-name",
                "EnableResourceNameDnsARecord": false,
                "EnableResourceNameDnsAAAARecord": false
            }
        },
        {
            "AvailabilityZone": "us-east-1d",
            "AvailabilityZoneId": "use1-az2",
            "AvailableIpAddressCount": 4089,
            "CidrBlock": "172.31.80.0/20",
            "DefaultForAz": true,
            "MapPublicIpOnLaunch": true,
            "MapCustomerOwnedIpOnLaunch": false,
            "State": "available",
            "SubnetId": "subnet-8EXAMPLE",
            "VpcId": "vpc-3EXAMPLE",
            "OwnerId": "1111222233333",
            "AssignIpv6AddressOnCreation": false,
            "Ipv6CidrBlockAssociationSet": [],
            "Tags": [
                {
                    "Key": "Name",
                    "Value": "MySubnet"
```

```
                }
            ],
            "SubnetArn": "arn:aws:ec2:us-east-1:111122223333:subnet/
subnet-8EXAMPLE",
            "EnableDns64": false,
            "Ipv6Native": false,
            "PrivateDnsNameOptionsOnLaunch": {
                "HostnameType": "ip-name",
                "EnableResourceNameDnsARecord": false,
                "EnableResourceNameDnsAAAARecord": false
            }
        }
    ]
}
```

Per ulteriori informazioni, consulta Working with VPCs and Subnet nella AWS VPC User Guide.

Esempio 2: come descrivere le sottoreti di un VPC specifico

Nell'esempio `describe-subnets` seguente viene utilizzato un filtro per recuperare i dettagli per le sottoreti del VPC specificato.

```
aws ec2 describe-subnets \
    --filters "Name=vpc-id,Values=vpc-3EXAMPLE"
```

Output:

```
{
    "Subnets": [
        {
            "AvailabilityZone": "us-east-1d",
            "AvailabilityZoneId": "use1-az2",
            "AvailableIpAddressCount": 4089,
            "CidrBlock": "172.31.80.0/20",
            "DefaultForAz": true,
            "MapPublicIpOnLaunch": true,
            "MapCustomerOwnedIpOnLaunch": false,
            "State": "available",
            "SubnetId": "subnet-8EXAMPLE",
            "VpcId": "vpc-3EXAMPLE",
            "OwnerId": "1111222233333",
            "AssignIpv6AddressOnCreation": false,
```

```
            "Ipv6CidrBlockAssociationSet": [],
            "Tags": [
                {
                    "Key": "Name",
                    "Value": "MySubnet"
                }
            ],
            "SubnetArn": "arn:aws:ec2:us-east-1:111122223333:subnet/
subnet-8EXAMPLE",
            "EnableDns64": false,
            "Ipv6Native": false,
            "PrivateDnsNameOptionsOnLaunch": {
                "HostnameType": "ip-name",
                "EnableResourceNameDnsARecord": false,
                "EnableResourceNameDnsAAAARecord": false
            }
        }
    ]
}
```

Per ulteriori informazioni, consulta Working with VPCs and Subnet nella AWS VPC User Guide.

Esempio 3: come descrivere le sottoreti con un tag specifico

L'`describe-subnets`esempio seguente utilizza un filtro per recuperare i dettagli di quelle sottoreti con il tag `CostCenter=123` e il `--query` parametro per visualizzare la sottorete delle IDs sottoreti con questo tag.

```
aws ec2 describe-subnets \
    --filters "Name=tag:CostCenter,Values=123" \
    --query "Subnets[*].SubnetId" \
    --output text
```

Output:

```
subnet-0987a87c8b37348ef
subnet-02a95061c45f372ee
subnet-03f720e7de2788d73
```

Per ulteriori informazioni, consulta Working with VPCs and Subnet nella Amazon VPC User Guide.

- Per i dettagli sull'API, consulta Command DescribeSubnetsReference AWS CLI .

`describe-tags`

Il seguente esempio di codice mostra come utilizzare`describe-tags`.

AWS CLI

Esempio 1: come descrivere tutti i tag di un'unica risorsa

L'esempio `describe-tags` seguente descrive i tag per l'istanza specificata.

```
aws ec2 describe-tags \
    --filters "Name=resource-id,Values=i-1234567890abcdef8"
```

Output:

```
{
    "Tags": [
        {
            "ResourceType": "instance",
            "ResourceId": "i-1234567890abcdef8",
            "Value": "Test",
            "Key": "Stack"
        },
        {
            "ResourceType": "instance",
            "ResourceId": "i-1234567890abcdef8",
            "Value": "Beta Server",
            "Key": "Name"
        }
    ]
}
```

Esempio 2: come descrivere tutti i tag di un tipo di risorsa

L'esempio `describe-tags` seguente descrive i tag per i volumi in uso.

```
aws ec2 describe-tags \
    --filters "Name=resource-type,Values=volume"
```

Output:

```
{
    "Tags": [
        {
            "ResourceType": "volume",
            "ResourceId": "vol-1234567890abcdef0",
            "Value": "Project1",
            "Key": "Purpose"
        },
        {
            "ResourceType": "volume",
            "ResourceId": "vol-049df61146c4d7901",
            "Value": "Logs",
            "Key": "Purpose"
        }
    ]
}
```

Esempio 3: come descrivere tutti tag

L'esempio `describe-tags` seguente descrive i tag per tutte le risorse.

```
aws ec2 describe-tags
```

Esempio 4: come descrivere i tag delle risorse in base a una chiave di tag

L'esempio `describe-tags` seguente descrive i tag per le risorse che hanno un tag con la chiave `Stack`.

```
aws ec2 describe-tags \
    --filters Name=key,Values=Stack
```

Output:

```
{
    "Tags": [
        {
            "ResourceType": "volume",
            "ResourceId": "vol-027552a73f021f3b",
            "Value": "Production",
            "Key": "Stack"
```

```
        },
        {
            "ResourceType": "instance",
            "ResourceId": "i-1234567890abcdef8",
            "Value": "Test",
            "Key": "Stack"
        }
    ]
}
```

Esempio 5: come descrivere i tag delle risorse in base a una chiave di tag e a un valore di tag

L'esempio `describe-tags` seguente descrive i tag per le risorse che hanno il tag `Stack=Test`.

```
aws ec2 describe-tags \
    --filters Name=key,Values=Stack Name=value,Values=Test
```

Output:

```
{
    "Tags": [
        {
            "ResourceType": "image",
            "ResourceId": "ami-3ac336533f021f3bd",
            "Value": "Test",
            "Key": "Stack"
        },
        {
            "ResourceType": "instance",
            "ResourceId": "i-1234567890abcdef8",
            "Value": "Test",
            "Key": "Stack"
        }
    ]
}
```

L'esempio `describe-tags` seguente utilizza una sintassi alternativa per descrivere le risorse con il tag `Stack=Test`.

```
aws ec2 describe-tags \
    --filters "Name=tag:Stack,Values=Test"
```

L'esempio `describe-tags` seguente descrive i tag per tutte le istanze che hanno un tag con la chiave `Purpose` e nessun valore.

```
aws ec2 describe-tags \
    --filters "Name=resource-
type,Values=instance" "Name=key,Values=Purpose" "Name=value,Values="
```

Output:

```
{
    "Tags": [
        {
            "ResourceType": "instance",
            "ResourceId": "i-1234567890abcdef5",
            "Value": null,
            "Key": "Purpose"
        }
    ]
}
```

- Per i dettagli sull'API, consulta DescribeTags AWS CLICommand Reference.

`describe-traffic-mirror-filters`

Il seguente esempio di codice mostra come utilizzare`describe-traffic-mirror-filters`.

AWS CLI

Come visualizzare i filtri di mirroring del traffico

L'esempio `describe-traffic-mirror-filters` seguente mostra i dettagli di tutti i filtri di mirroring del traffico.

```
aws ec2 describe-traffic-mirror-filters
```

Output:

```
{
    "TrafficMirrorFilters": [
        {
            "TrafficMirrorFilterId": "tmf-0293f26e86EXAMPLE",
```

```
            "IngressFilterRules": [
                {
                    "TrafficMirrorFilterRuleId": "tmfr-0ca76e0e08EXAMPLE",
                    "TrafficMirrorFilterId": "tmf-0293f26e86EXAMPLE",
                    "TrafficDirection": "ingress",
                    "RuleNumber": 100,
                    "RuleAction": "accept",
                    "Protocol": 6,
                    "DestinationCidrBlock": "10.0.0.0/24",
                    "SourceCidrBlock": "10.0.0.0/24",
                    "Description": "TCP Rule"
                }
            ],
            "EgressFilterRules": [],
            "NetworkServices": [],
            "Description": "Example filter",
            "Tags": []
        }
    ]
}
```

Per ulteriori informazioni, consulta Visualizzare i filtri del mirroring del traffico nella Guida di Mirroring del traffico.

- Per i dettagli sull'API, consulta DescribeTrafficMirrorFilters AWS CLICommand Reference.

`describe-traffic-mirror-sessions`

Il seguente esempio di codice mostra come utilizzare`describe-traffic-mirror-sessions`.

AWS CLI

Come descrivere una sessione di mirroring del traffico

L'esempio `describe-traffic-mirror-sessions` seguente visualizza i dettagli delle sessioni di mirroring del traffico.

```
aws ec2 describe-traffic-mirror-sessions
```

Output:

```
{
    "TrafficMirrorSessions": [
```

```
        {
            "Tags": [],
            "VirtualNetworkId": 42,
            "OwnerId": "111122223333",
            "Description": "TCP Session",
            "NetworkInterfaceId": "eni-0a471a5cf3EXAMPLE",
            "TrafficMirrorTargetId": "tmt-0dabe9b0a6EXAMPLE",
            "TrafficMirrorFilterId": "tmf-083e18f985EXAMPLE",
            "PacketLength": 20,
            "SessionNumber": 1,
            "TrafficMirrorSessionId": "tms-0567a4c684EXAMPLE"
        },
        {
            "Tags": [
                {
                    "Key": "Name",
                    "Value": "tag test"
                }
            ],
            "VirtualNetworkId": 13314501,
            "OwnerId": "111122223333",
            "Description": "TCP Session",
            "NetworkInterfaceId": "eni-0a471a5cf3EXAMPLE",
            "TrafficMirrorTargetId": "tmt-03665551cbEXAMPLE",
            "TrafficMirrorFilterId": "tmf-06c787846cEXAMPLE",
            "SessionNumber": 2,
            "TrafficMirrorSessionId": "tms-0060101cf8EXAMPLE"
        }
    ]
}
```


Per ulteriori informazioni, consulta Visualizzare i dettagli della sessione di mirroring del traffico nella Guida di Mirroring del traffico AWS .

- Per i dettagli sull'API, consulta DescribeTrafficMirrorSessions AWS CLICommand Reference.

`describe-traffic-mirror-targets`

Il seguente esempio di codice mostra come utilizzare`describe-traffic-mirror-targets`.

AWS CLI

Come descrivere una destinazione di mirroring del traffico

L'esempio `describe-traffic-mirror-targets` seguente visualizza le informazioni sulla destinazione di mirroring del traffico specificata.

```
aws ec2 describe-traffic-mirror-targets \
    --traffic-mirror-target-ids tmt-0dabe9b0a6EXAMPLE
```

Output:

```
{
    "TrafficMirrorTargets": [
        {
            "TrafficMirrorTargetId": "tmt-0dabe9b0a6EXAMPLE",
            "NetworkLoadBalancerArn": "arn:aws:elasticloadbalancing:us-
east-1:111122223333:loadbalancer/net/NLB/7cdec873fEXAMPLE",
            "Type": "network-load-balancer",
            "Description": "Example Network Load Balancer target",
            "OwnerId": "111122223333",
            "Tags": []
        }
    ]
}
```

Per ulteriori informazioni, consulta Destinazioni del mirroring del traffico nella Guida di Mirroring del traffico Amazon VPC.

- Per i dettagli sull'API, consulta DescribeTrafficMirrorTargets AWS CLICommand Reference.

`describe-transit-gateway-attachments`

Il seguente esempio di codice mostra come utilizzare`describe-transit-gateway-attachments`.

AWS CLI

Come visualizzare i collegamenti del gateway di transito alla VPN

L'esempio `describe-transit-gateway-attachments` seguente visualizza i dettagli per i collegamenti del gateway di transito alla VPN.

```
aws ec2 describe-transit-gateway-attachments
```

Output:


```
{
    "TransitGatewayAttachments": [
        {
            "TransitGatewayAttachmentId": "tgw-attach-01f8100bc7EXAMPLE",
            "TransitGatewayId": "tgw-02f776b1a7EXAMPLE",
            "TransitGatewayOwnerId": "123456789012",
            "ResourceOwnerId": "123456789012",
            "ResourceType": "vpc",
            "ResourceId": "vpc-3EXAMPLE",
            "State": "available",
            "Association": {
                "TransitGatewayRouteTableId": "tgw-rtb-002573ed1eEXAMPLE",
                "State": "associated"
            },
            "CreationTime": "2019-08-26T14:59:25.000Z",
            "Tags": [
                {
                    "Key": "Name",
                    "Value": "Example"
                }
            ]
        },
        {
            "TransitGatewayAttachmentId": "tgw-attach-0b5968d3b6EXAMPLE",
            "TransitGatewayId": "tgw-02f776b1a7EXAMPLE",
            "TransitGatewayOwnerId": "123456789012",
            "ResourceOwnerId": "123456789012",
            "ResourceType": "vpc",
            "ResourceId": "vpc-0065acced4EXAMPLE",
            "State": "available",
            "Association": {
                "TransitGatewayRouteTableId": "tgw-rtb-002573ed1eEXAMPLE",
                "State": "associated"
            },
            "CreationTime": "2019-08-07T17:03:07.000Z",
            "Tags": []
        },
        {
            "TransitGatewayAttachmentId": "tgw-attach-08e0bc912cEXAMPLE",
            "TransitGatewayId": "tgw-02f776b1a7EXAMPLE",
            "TransitGatewayOwnerId": "123456789012",
            "ResourceOwnerId": "123456789012",
```

```
            "ResourceType": "direct-connect-gateway",
            "ResourceId": "11460968-4ac1-4fd3-bdb2-00599EXAMPLE",
            "State": "available",
            "Association": {
                "TransitGatewayRouteTableId": "tgw-rtb-002573ed1eEXAMPLE",
                "State": "associated"
            },
            "CreationTime": "2019-08-14T20:27:44.000Z",
            "Tags": []
        },
        {
            "TransitGatewayAttachmentId": "tgw-attach-0a89069f57EXAMPLE",
            "TransitGatewayId": "tgw-02f776b1a7EXAMPLE",
            "TransitGatewayOwnerId": "123456789012",
            "ResourceOwnerId": "123456789012",
            "ResourceType": "direct-connect-gateway",
            "ResourceId": "8384da05-13ce-4a91-aada-5a1baEXAMPLE",
            "State": "available",
            "Association": {
                "TransitGatewayRouteTableId": "tgw-rtb-002573ed1eEXAMPLE",
                "State": "associated"
            },
            "CreationTime": "2019-08-14T20:33:02.000Z",
            "Tags": []
        }
    ]
}
```

Per ulteriori informazioni, consulta Utilizzo dei gateway di transito nella Guida di Transit Gateway.

- Per i dettagli sull'API, consulta DescribeTransitGatewayAttachments AWS CLICommand Reference.

`describe-transit-gateway-connect-peers`

Il seguente esempio di codice mostra come utilizzare`describe-transit-gateway-connect-peers`.

AWS CLI

Come descrivere un peer di Transit Gateway Connect

L'esempio `describe-transit-gateway-connect-peers` seguente descrive il peer Connect specificato.

```
aws ec2 describe-transit-gateway-connect-peers \
    --transit-gateway-connect-peer-ids tgw-connect-peer-0666adbac4EXAMPLE
```

Output:

```
{
    "TransitGatewayConnectPeers": [
        {
            "TransitGatewayAttachmentId": "tgw-attach-0f0927767cEXAMPLE",
            "TransitGatewayConnectPeerId": "tgw-connect-peer-0666adbac4EXAMPLE",
            "State": "available",
            "CreationTime": "2021-10-13T03:35:17.000Z",
            "ConnectPeerConfiguration": {
                "TransitGatewayAddress": "10.0.0.234",
                "PeerAddress": "172.31.1.11",
                "InsideCidrBlocks": [
                    "169.254.6.0/29"
                ],
                "Protocol": "gre",
                "BgpConfigurations": [
                    {
                        "TransitGatewayAsn": 64512,
                        "PeerAsn": 64512,
                        "TransitGatewayAddress": "169.254.6.2",
                        "PeerAddress": "169.254.6.1",
                        "BgpStatus": "down"
                    },
                    {
                        "TransitGatewayAsn": 64512,
                        "PeerAsn": 64512,
                        "TransitGatewayAddress": "169.254.6.3",
                        "PeerAddress": "169.254.6.1",
                        "BgpStatus": "down"
                    }
                ]
            },
            "Tags": []
        }
    ]
```

```
}
```

Per ulteriori informazioni, consulta Collegamenti Transit gateway Connect e peer Connect nella Guida di Transit Gateway.

- Per i dettagli sull'API, consulta DescribeTransitGatewayConnectPeers AWS CLICommand Reference.

`describe-transit-gateway-connects`

Il seguente esempio di codice mostra come utilizzare`describe-transit-gateway-connects`.

AWS CLI

Come descrivere un collegamento Connect del gateway di transito

L'esempio `describe-transit-gateway-connects` seguente descrive il collegamento Connect specificato.

```
aws ec2 describe-transit-gateway-connects \
    --transit-gateway-attachment-ids tgw-attach-037012e5dcEXAMPLE
```

Output:

```
{
    "TransitGatewayConnects": [
        {
            "TransitGatewayAttachmentId": "tgw-attach-037012e5dcEXAMPLE",
            "TransportTransitGatewayAttachmentId": "tgw-attach-0a89069f57EXAMPLE",
            "TransitGatewayId": "tgw-02f776b1a7EXAMPLE",
            "State": "available",
            "CreationTime": "2021-03-09T19:59:17+00:00",
            "Options": {
                "Protocol": "gre"
            },
            "Tags": []
        }
    ]
}
```

Per ulteriori informazioni, consulta Collegamenti Transit gateway Connect e peer Connect nella Guida di Transit Gateway.

- Per i dettagli sull'API, consulta DescribeTransitGatewayConnects AWS CLICommand Reference.

`describe-transit-gateway-multicast-domains`

Il seguente esempio di codice mostra come utilizzare`describe-transit-gateway-multicast-domains`.

AWS CLI

Come descrivere i domini multicast (trasmissione uno a molti) del gateway di transito

L'esempio `describe-transit-gateway-multicast-domains` seguente visualizza i dettagli per tutti i domini multicast (trasmissione uno a molti) del gateway di transito.

```
aws ec2 describe-transit-gateway-multicast-domains
```

Output:

```
{

    "TransitGatewayMulticastDomains": [
        {
            "TransitGatewayMulticastDomainId": "tgw-mcast-domain-000fb24d04EXAMPLE",
            "TransitGatewayId": "tgw-0bf0bffefaEXAMPLE",
            "TransitGatewayMulticastDomainArn": "arn:aws:ec2:us-
east-1:123456789012:transit-gateway-multicast-domain/tgw-mcast-
domain-000fb24d04EXAMPLE",
            "OwnerId": "123456789012",
            "Options": {
                "Igmpv2Support": "disable",
                "StaticSourcesSupport": "enable",
                "AutoAcceptSharedAssociations": "disable"
            },
            "State": "available",
            "CreationTime": "2019-12-10T18:32:50+00:00",
            "Tags": [
                {
                    "Key": "Name",
                    "Value": "mc1"
                }
```

```
            ]
        }
    ]
}
```

Per ulteriori informazioni, consulta Gestione dei domini multicast (trasmissione uno a molti) nella Guida di Transit Gateway.

- Per i dettagli sull'API, consulta DescribeTransitGatewayMulticastDomains AWS CLICommand Reference.

`describe-transit-gateway-peering-attachments`

Il seguente esempio di codice mostra come utilizzare`describe-transit-gateway-peering-attachments`.

AWS CLI

Come descrivere i collegamenti peering del gateway di transito

L'esempio `describe-transit-gateway-peering-attachments` seguente visualizza i dettagli di tutti i collegamenti peering del gateway di transito.

```
aws ec2 describe-transit-gateway-peering-attachments
```

Output:

```
{
    "TransitGatewayPeeringAttachments": [
        {
            "TransitGatewayAttachmentId": "tgw-attach-4455667788aabbccd",
            "RequesterTgwInfo": {
                "TransitGatewayId": "tgw-123abc05e04123abc",
                "OwnerId": "123456789012",
                "Region": "us-west-2"
            },
            "AccepterTgwInfo": {
                "TransitGatewayId": "tgw-11223344aabbcc112",
                "OwnerId": "123456789012",
                "Region": "us-east-2"
            },
```

```
            "State": "pendingAcceptance",
            "CreationTime": "2019-12-09T11:38:05.000Z",
            "Tags": []
        }
    ]
}
```

Per ulteriori informazioni, consulta Accessori di peering del gateway di transito nella Guida di Transit Gateway.

- Per i dettagli sull'API, consulta DescribeTransitGatewayPeeringAttachments AWS CLICommand Reference.

`describe-transit-gateway-policy-tables`

Il seguente esempio di codice mostra come utilizzare`describe-transit-gateway-policy-tables`.

AWS CLI

Come descrivere una tabella di policy del gateway di transito

L'esempio `describe-transit-gateway-policy-tables` seguente descrive la tabella di policy del gateway di transito specificato.

```
aws ec2 describe-transit-gateway-policy-tables \
    --transit-gateway-policy-table-ids tgw-ptb-0a16f134b78668a81
```

Output:

```
{
    "TransitGatewayPolicyTables": [
        {
            "TransitGatewayPolicyTableId": "tgw-ptb-0a16f134b78668a81",
            "TransitGatewayId": "tgw-067f8505c18f0bd6e",
            "State": "available",
            "CreationTime": "2023-11-28T16:36:43+00:00",
            "Tags": []
        }
    ]
}
```

Per ulteriori informazioni, consulta Tabelle di policy del gateway di transito nella Guida per l'utente di Transit Gateway.

- Per i dettagli sull'API, consulta DescribeTransitGatewayPolicyTables AWS CLICommand Reference.

`describe-transit-gateway-route-tables`

Il seguente esempio di codice mostra come utilizzare`describe-transit-gateway-route-tables`.

AWS CLI

Come descrivere le tabelle di routing del gateway di transito

L'esempio `describe-transit-gateway-route-tables` seguente visualizza i dettagli per le tabella di routing del gateway di transito.

```
aws ec2 describe-transit-gateway-route-tables
```

Output:

```
{
    "TransitGatewayRouteTables": [
        {
            "TransitGatewayRouteTableId": "tgw-rtb-0ca78a549EXAMPLE",
            "TransitGatewayId": "tgw-0bc994abffEXAMPLE",
            "State": "available",
            "DefaultAssociationRouteTable": true,
            "DefaultPropagationRouteTable": true,
            "CreationTime": "2018-11-28T14:24:49.000Z",
            "Tags": []
        },
        {
            "TransitGatewayRouteTableId": "tgw-rtb-0e8f48f148EXAMPLE",
            "TransitGatewayId": "tgw-0043d72bb4EXAMPLE",
            "State": "available",
            "DefaultAssociationRouteTable": true,
            "DefaultPropagationRouteTable": true,
            "CreationTime": "2018-11-28T14:24:00.000Z",
            "Tags": []
        }
```

```
    ]
}
```

Per ulteriori informazioni, consulta Visualizzare le tabelle di routing del gateway di transito nella Guida di Transit Gateway.

- Per i dettagli sull'API, consulta DescribeTransitGatewayRouteTables AWS CLICommand Reference.

`describe-transit-gateway-vpc-attachments`

Il seguente esempio di codice mostra come utilizzare`describe-transit-gateway-vpc-attachments`.

AWS CLI

Come descrivere gli allegati VPC del gateway di transito

L'esempio `describe-transit-gateway-vpc-attachments` seguente visualizza i dettagli per gli allegati VPC del gateway di transito.

```
aws ec2 describe-transit-gateway-vpc-attachments
```

Output:

```
{
    "TransitGatewayVpcAttachments": [
        {
            "TransitGatewayAttachmentId": "tgw-attach-0a08e88308EXAMPLE",
            "TransitGatewayId": "tgw-0043d72bb4EXAMPLE",
            "VpcId": "vpc-0f501f7ee8EXAMPLE",
            "VpcOwnerId": "111122223333",
            "State": "available",
            "SubnetIds": [
                "subnet-045d586432EXAMPLE",
                "subnet-0a0ad478a6EXAMPLE"
            ],
            "CreationTime": "2019-02-13T11:04:02.000Z",
            "Options": {
                "DnsSupport": "enable",
                "Ipv6Support": "disable"
            },
```

```
            "Tags": [
                {
                    "Key": "Name",
                    "Value": "attachment name"
                }
            ]
        }
    ]
}
```

Per ulteriori informazioni, consulta Visualizzare i collegamenti al VPC nella Guida di Transit Gateway.

- Per i dettagli sull'API, consulta DescribeTransitGatewayVpcAttachments AWS CLICommand Reference.

`describe-transit-gateways`

Il seguente esempio di codice mostra come utilizzare`describe-transit-gateways`.

AWS CLI

Come descrivere i gateway di transito

L'esempio `describe-transit-gateways` seguente recupera i dettagli di tutti i gateway di transito.

```
aws ec2 describe-transit-gateways
```

Output:

```
{
    "TransitGateways": [
        {
            "TransitGatewayId": "tgw-0262a0e521EXAMPLE",
            "TransitGatewayArn": "arn:aws:ec2:us-east-2:111122223333:transit-
gateway/tgw-0262a0e521EXAMPLE",
            "State": "available",
            "OwnerId": "111122223333",
            "Description": "MyTGW",
            "CreationTime": "2019-07-10T14:02:12.000Z",
            "Options": {
```

```
                "AmazonSideAsn": 64516,
                "AutoAcceptSharedAttachments": "enable",
                "DefaultRouteTableAssociation": "enable",
                "AssociationDefaultRouteTableId": "tgw-rtb-018774adf3EXAMPLE",
                "DefaultRouteTablePropagation": "enable",
                "PropagationDefaultRouteTableId": "tgw-rtb-018774adf3EXAMPLE",
                "VpnEcmpSupport": "enable",
                "DnsSupport": "enable"
            },
            "Tags": []
        },
        {
            "TransitGatewayId": "tgw-0fb8421e2dEXAMPLE",
            "TransitGatewayArn": "arn:aws:ec2:us-east-2:111122223333:transit-
gateway/tgw-0fb8421e2da853bf3",
            "State": "available",
            "OwnerId": "111122223333",
            "CreationTime": "2019-03-15T22:57:33.000Z",
            "Options": {
                "AmazonSideAsn": 65412,
                "AutoAcceptSharedAttachments": "disable",
                "DefaultRouteTableAssociation": "enable",
                "AssociationDefaultRouteTableId": "tgw-rtb-06a241a3d8EXAMPLE",
                "DefaultRouteTablePropagation": "enable",
                "PropagationDefaultRouteTableId": "tgw-rtb-06a241a3d8EXAMPLE",
                "VpnEcmpSupport": "enable",
                "DnsSupport": "enable"
            },
            "Tags": [
                {
                    "Key": "Name",
                    "Value": "TGW1"
                }
            ]
        }
    ]
}
```

- Per i dettagli sull'API, consulta DescribeTransitGateways AWS CLICommand Reference.

`describe-verified-access-endpoints`

Il seguente esempio di codice mostra come utilizzare`describe-verified-access-endpoints`.

AWS CLI

Come descrivere un endpoint di accesso verificato

L'esempio `describe-verified-access-endpoints` seguente descrive l'endpoint di accesso verificato specificato.

```
aws ec2 describe-verified-access-endpoints \
    --verified-access-endpoint-ids vae-066fac616d4d546f2
```

Output:

```
{
    "VerifiedAccessEndpoints": [
        {
            "VerifiedAccessInstanceId": "vai-0ce000c0b7643abea",
            "VerifiedAccessGroupId": "vagr-0dbe967baf14b7235",
            "VerifiedAccessEndpointId": "vae-066fac616d4d546f2",
            "ApplicationDomain": "example.com",
            "EndpointType": "network-interface",
            "AttachmentType": "vpc",
            "DomainCertificateArn": "arn:aws:acm:us-east-2:123456789012:certificate/
eb065ea0-26f9-4e75-a6ce-0a1a7EXAMPLE",
            "EndpointDomain": "my-ava-
app.edge-00c3372d53b1540bb.vai-0ce000c0b7643abea.prod.verified-access.us-
east-2.amazonaws.com",
            "SecurityGroupIds": [
                "sg-004915970c4c8f13a"
            ],
            "NetworkInterfaceOptions": {
                "NetworkInterfaceId": "eni-0aec70418c8d87a0f",
                "Protocol": "https",
                "Port": 443
            },
            "Status": {
                "Code": "active"
            },
            "Description": "",
            "CreationTime": "2023-08-25T20:54:43",
            "LastUpdatedTime": "2023-08-25T22:17:26",
            "Tags": [
                {
                    "Key": "Name",
```

```
                    "Value": "my-va-endpoint"
                }
            ]
        }
    ]
}
```

Per ulteriori informazioni, consulta Endpoint di Accesso verificato nella Guida per l'utente di Accesso verificato da AWS .

- Per i dettagli sull'API, consulta DescribeVerifiedAccessEndpoints AWS CLICommand Reference.

`describe-verified-access-groups`

Il seguente esempio di codice mostra come utilizzare`describe-verified-access-groups`.

AWS CLI

Come descrivere un gruppo di accesso verificato

L'esempio `describe-verified-access-groups` seguente descrive il gruppo di accesso verificato specificato.

```
aws ec2 describe-verified-access-groups \
    --verified-access-group-ids vagr-0dbe967baf14b7235
```

Output:

```
{
    "VerifiedAccessGroups": [
        {
            "VerifiedAccessGroupId": "vagr-0dbe967baf14b7235",
            "VerifiedAccessInstanceId": "vai-0ce000c0b7643abea",
            "Description": "Testing Verified Access",
            "Owner": "123456789012",
            "VerifiedAccessGroupArn": "arn:aws:ec2:us-east-2:123456789012:verified-
access-group/vagr-0dbe967baf14b7235",
            "CreationTime": "2023-08-25T19:55:19",
            "LastUpdatedTime": "2023-08-25T22:17:25",
            "Tags": [
```

```
                {
                    "Key": "Name",
                    "Value": "my-va-group"
                }
            ]
        }
    ]
}
```

Per ulteriori informazioni, consulta Gruppi di Accesso verificato nella Guida per l'utente di Accesso verificato da AWS .

- Per i dettagli sull'API, consulta DescribeVerifiedAccessGroups AWS CLICommand Reference.

`describe-verified-access-instance-logging-configurations`

Il seguente esempio di codice mostra come utilizzare`describe-verified-access-instance-logging-configurations`.

AWS CLI

Come descrivere la configurazione di registrazione per un'istanza di accesso verificato

L'esempio `describe-verified-access-instance-logging-configurations` seguente descrive la configurazione di registrazione dell'istanza di accesso verificato specificata.

```
aws ec2 describe-verified-access-instance-logging-configurations \
    --verified-access-instance-ids vai-0ce000c0b7643abea
```

Output:

```
{
    "LoggingConfigurations": [
        {
            "VerifiedAccessInstanceId": "vai-0ce000c0b7643abea",
            "AccessLogs": {
                "S3": {
                    "Enabled": false
                },
                "CloudWatchLogs": {
                    "Enabled": true,
```

```
                    "DeliveryStatus": {
                        "Code": "success"
                    },
                    "LogGroup": "my-log-group"
                },
                "KinesisDataFirehose": {
                    "Enabled": false
                },
                "LogVersion": "ocsf-1.0.0-rc.2",
                "IncludeTrustContext": false
            }
        }
    ]
}
```

Per ulteriori informazioni, consulta Log di accesso verificato nella Guida per l'utente di Accesso verificato da AWS .

- Per i dettagli sull'API, consulta DescribeVerifiedAccessInstanceLoggingConfigurations AWS CLICommand Reference.

`describe-verified-access-instances`

Il seguente esempio di codice mostra come utilizzare`describe-verified-access-instances`.

AWS CLI

Come descrivere un'istanza di accesso verificato

L'esempio `describe-verified-access-instances` seguente descrive l'istanza di accesso verificato specificata.

```
aws ec2 describe-verified-access-instances \
    --verified-access-instance-ids vai-0ce000c0b7643abea
```

Output:

```
{
    "VerifiedAccessInstances": [
        {
            "VerifiedAccessInstanceId": "vai-0ce000c0b7643abea",
```

```
            "Description": "Testing Verified Access",
            "VerifiedAccessTrustProviders": [
                {
                    "VerifiedAccessTrustProviderId": "vatp-0bb32de759a3e19e7",
                    "TrustProviderType": "user",
                    "UserTrustProviderType": "iam-identity-center"
                }
            ],
            "CreationTime": "2023-08-25T18:27:56",
            "LastUpdatedTime": "2023-08-25T19:03:32",
            "Tags": [
                {
                    "Key": "Name",
                    "Value": "my-ava-instance"
                }
            ]
        }
    ]
}
```

Per ulteriori informazioni, consulta Istanze di Accesso verificato nella Guida per l'utente di Accesso verificato da AWS .

- Per i dettagli sull'API, consulta DescribeVerifiedAccessInstances AWS CLICommand Reference.

`describe-verified-access-trust-providers`

Il seguente esempio di codice mostra come utilizzare`describe-verified-access-trust-providers`.

AWS CLI

Come descrivere un provider di attendibilità di accesso verificato

L'esempio `describe-verified-access-trust-providers` seguente descrive il provider di attendibilità di accesso verificato specificato.

```
aws ec2 describe-verified-access-trust-providers \
    --verified-access-trust-provider-ids vatp-0bb32de759a3e19e7
```

Output:

```
{
    "VerifiedAccessTrustProviders": [
        {
            "VerifiedAccessTrustProviderId": "vatp-0bb32de759a3e19e7",
            "Description": "Testing Verified Access",
            "TrustProviderType": "user",
            "UserTrustProviderType": "iam-identity-center",
            "PolicyReferenceName": "idc",
            "CreationTime": "2023-08-25T19:00:38",
            "LastUpdatedTime": "2023-08-25T19:03:32",
            "Tags": [
                {
                    "Key": "Name",
                    "Value": "my-va-trust-provider"
                }
            ]
        }
    ]
}
```

Per ulteriori informazioni, consulta Provider di attendibilità per Accesso verificato nella Guida per l'utente di Accesso verificato da AWS .

- Per i dettagli sull'API, consulta DescribeVerifiedAccessTrustProviders AWS CLICommand Reference.

`describe-volume-attribute`

Il seguente esempio di codice mostra come utilizzare`describe-volume-attribute`.

AWS CLI

Come descrivere un attributo del volume

Questo esempio imposta l'attributo `autoEnableIo` del volume con l'ID `vol-049df61146c4d7901`.

Comando:

```
aws ec2 describe-volume-attribute --volume-id vol-049df61146c4d7901 --
attribute autoEnableIO
```

Output:

```
{
    "AutoEnableIO": {
        "Value": false
    },
    "VolumeId": "vol-049df61146c4d7901"
}
```

- Per i dettagli sull'API, consulta DescribeVolumeAttribute AWS CLICommand Reference.

`describe-volume-status`

Il seguente esempio di codice mostra come utilizzare`describe-volume-status`.

AWS CLI

Come descrivere lo stato di un volume

Questo comando di esempio descrive lo stato del volume `vol-1234567890abcdef0`.

Comando:

```
aws ec2 describe-volume-status --volume-ids vol-1234567890abcdef0
```

Output:

```
{
    "VolumeStatuses": [
        {
            "VolumeStatus": {
                "Status": "ok",
                "Details": [
                    {
                        "Status": "passed",
                        "Name": "io-enabled"
                    },
                    {
                        "Status": "not-applicable",
                        "Name": "io-performance"
                    }
```

```
                ]
            },
            "AvailabilityZone": "us-east-1a",
            "VolumeId": "vol-1234567890abcdef0",
            "Actions": [],
            "Events": []
        }
    ]
}
```

Come descrivere lo stato dei volumi danneggiati

Questo comando di esempio descrive lo stato di tutti i volumi danneggiati. In questo output di esempio non sono presenti volumi danneggiati.

Comando:

```
aws ec2 describe-volume-status --filters Name=volume-status.status,Values=impaired
```

Output:

```
{
    "VolumeStatuses": []
}
```

Se hai un volume con un controllo dello stato non riuscito (lo stato è compromesso), consulta Working with an Impaired Volume nella Amazon EC2 User Guide.

- Per i dettagli sull'API, consulta AWS CLI Command DescribeVolumeStatusReference.

`describe-volumes-modifications`

Il seguente esempio di codice mostra come utilizzare`describe-volumes-modifications`.

AWS CLI

Come descrivere lo stato di modifica di un volume

L'esempio `describe-volumes-modifications` seguente descrive lo stato di modifica del volume specificato.

```
aws ec2 describe-volumes-modifications \
    --volume-ids vol-1234567890abcdef0
```

Output:

```
{
    "VolumeModification": {
        "TargetSize": 150,
        "TargetVolumeType": "io1",
        "ModificationState": "optimizing",
        "VolumeId": " vol-1234567890abcdef0",
        "TargetIops": 100,
        "StartTime": "2019-05-17T11:27:19.000Z",
        "Progress": 70,
        "OriginalVolumeType": "io1",
        "OriginalIops": 100,
        "OriginalSize": 100
    }
}
```

- Per i dettagli sull'API, consulta DescribeVolumesModifications AWS CLICommand Reference.

`describe-volumes`

Il seguente esempio di codice mostra come utilizzare`describe-volumes`.

AWS CLI

Esempio 1: come descrivere un volume

L'esempio `describe-volumes` seguente descrive i volumi specificati nella Regione corrente.

```
aws ec2 describe-volumes \
    --volume-ids vol-049df61146c4d7901 vol-1234567890abcdef0
```

Output:

```
{
    "Volumes": [
        {
```

```
            "AvailabilityZone": "us-east-1a",
            "Attachments": [
                {
                    "AttachTime": "2013-12-18T22:35:00.000Z",
                    "InstanceId": "i-1234567890abcdef0",
                    "VolumeId": "vol-049df61146c4d7901",
                    "State": "attached",
                    "DeleteOnTermination": true,
                    "Device": "/dev/sda1"
                }
            ],
            "Encrypted": true,
            "KmsKeyId": "arn:aws:kms:us-east-2a:123456789012:key/8c5b2c63-b9bc-45a3-
a87a-5513eEXAMPLE,
            "VolumeType": "gp2",
            "VolumeId": "vol-049df61146c4d7901",
            "State": "in-use",
            "Iops": 100,
            "SnapshotId": "snap-1234567890abcdef0",
            "CreateTime": "2019-12-18T22:35:00.084Z",
            "Size": 8
        },
        {
            "AvailabilityZone": "us-east-1a",
            "Attachments": [],
            "Encrypted": false,
            "VolumeType": "gp2",
            "VolumeId": "vol-1234567890abcdef0",
            "State": "available",
            "Iops": 300,
            "SnapshotId": "",
            "CreateTime": "2020-02-27T00:02:41.791Z",
            "Size": 100
        }
    ]
}
```

Esempio 2: come descrivere i volumi collegati a un'istanza specifica

L'esempio `describe-volumes` seguente descrive tutti i volumi collegati all'istanza specificata e impostati per essere eliminati alla terminazione dell'istanza.

```
aws ec2 describe-volumes \
```

```
    --region us-east-1 \
    --filters Name=attachment.instance-
id,Values=i-1234567890abcdef0 Name=attachment.delete-on-termination,Values=true
```

Per un esempio dell'output di `describe-volumes`, consulta l'Esempio 1.

Esempio 3: come descrivere i volumi disponibili in una zona di disponibilità specifica

L'esempio `describe-volumes` seguente descrive tutti i volumi con stato `available` e che si trovano nella zona di disponibilità specificata.

```
aws ec2 describe-volumes \
    --filters Name=status,Values=available Name=availability-zone,Values=us-east-1a
```

Per un esempio dell'output di `describe-volumes`, vedi l'Esempio 1.

Esempio 4: come descrivere i volumi in base ai tag

L'esempio `describe-volumes` seguente descrive tutti i volumi con la chiave di tag `Name` e un valore che inizia con `Test`. L'output viene quindi filtrato con una query che visualizza solo i tag e IDs i volumi.

```
aws ec2 describe-volumes \
    --filters Name=tag:Name,Values=Test* \
    --query "Volumes[*].{ID:VolumeId,Tag:Tags}"
```

Output:

```
[
    {
        "Tag": [
            {
                "Value": "Test2",
                "Key": "Name"
            }
        ],
        "ID": "vol-1234567890abcdef0"
    },
    {
        "Tag": [
```

```
            {
                "Value": "Test1",
                "Key": "Name"
            }
        ],
        "ID": "vol-049df61146c4d7901"
    }
]
```

Per ulteriori esempi di utilizzo dei filtri di tag, consulta Lavorare con i tag nella Amazon EC2 User Guide.

- Per i dettagli sull'API, consulta DescribeVolumes AWS CLICommand Reference.

`describe-vpc-attribute`

Il seguente esempio di codice mostra come utilizzare`describe-vpc-attribute`.

AWS CLI

Per descrivere l' enableDnsSupport attributo

Questo esempio descrive l'attributo `enableDnsSupport`. Questo attributo indica se la risoluzione DNS è abilitata per il VPC. Se questo attributo è `true`, il server Amazon DNS risolve i nomi host DNS per le istanze negli indirizzi IP corrispondenti, ma solo in quel caso.

Comando:

```
aws ec2 describe-vpc-attribute --vpc-id vpc-a01106c2 --attribute enableDnsSupport
```

Output:

```
{
    "VpcId": "vpc-a01106c2",
    "EnableDnsSupport": {
        "Value": true
    }
}
```

Per descrivere l' enableDnsHostnames attributo

Questo esempio descrive l'attributo `enableDnsHostnames`. Questo attributo indica se le istanze avviate nel VPC ricevono nomi di host DNS. Se questo attributo è `true`, le istanze nel VPC ottengono nomi host DNS, altrimenti no.

Comando:

```
aws ec2 describe-vpc-attribute --vpc-id vpc-a01106c2 --attribute enableDnsHostnames
```

Output:

```
{
    "VpcId": "vpc-a01106c2",
    "EnableDnsHostnames": {
        "Value": true
    }
}
```

- Per i dettagli sull'API, consulta DescribeVpcAttribute AWS CLICommand Reference.

`describe-vpc-classic-link-dns-support`

Il seguente esempio di codice mostra come utilizzare`describe-vpc-classic-link-dns-support`.

AWS CLI

Per descrivere il supporto ClassicLink DNS per VPCs

Questo esempio descrive lo stato del supporto ClassicLink DNS di tutti i tuoi. VPCs

Comando:

```
aws ec2 describe-vpc-classic-link-dns-support
```

Output:

```
{
  "Vpcs": [
    {
```

```
      "VpcId": "vpc-88888888",
      "ClassicLinkDnsSupported": true
    },
    {
      "VpcId": "vpc-1a2b3c4d",
      "ClassicLinkDnsSupported": false
    }
  ]
}
```

- Per i dettagli sull'API, consulta DescribeVpcClassicLinkDnsSupport AWS CLICommand Reference.

`describe-vpc-classic-link`

Il seguente esempio di codice mostra come utilizzare`describe-vpc-classic-link`.

AWS CLI

Per descrivere lo ClassicLink stato del tuo VPCs

Questo esempio elenca lo ClassicLink stato di vpc-88888888.

Comando:

```
aws ec2 describe-vpc-classic-link --vpc-id vpc-88888888
```

Output:

```
{
  "Vpcs": [
    {
      "ClassicLinkEnabled": true,
      "VpcId": "vpc-88888888",
      "Tags": [
        {
          "Value": "classiclinkvpc",
          "Key": "Name"
        }
      ]
    }
```

```
  ]
}
```

Questo esempio elenca solo VPCs quelli abilitati per Classiclink (il valore del `is-classic-link-enabled` filtro è impostato su). `true`

Comando:

```
aws ec2 describe-vpc-classic-link --filter "Name=is-classic-link-enabled,Values=true"
```

- Per i dettagli sull'API, consulta AWS CLI Command DescribeVpcClassicLinkReference.

`describe-vpc-endpoint-associations`

Il seguente esempio di codice mostra come utilizzare`describe-vpc-endpoint-associations`.

AWS CLI

Come descrivere le associazioni di endpoint VPC

L'esempio `describe-vpc-endpoint-associations` seguente descrive le associazioni di endpoint VPC.

```
aws ec2 describe-vpc-endpoint-associations
```

Output:

```
{
    "VpcEndpointAssociations": [
        {
            "Id": "vpce-rsc-asc-0a810ca6ac8866bf9",
            "VpcEndpointId": "vpce-019b90d6f16d4f958",
            "AssociatedResourceAccessibility": "Accessible",
            "DnsEntry": {
                "DnsName":
 "vpce-019b90d6f16d4f958.rcfg-07129f3acded87625.4232ccc.vpc-lattice-rsc.us-east-2.on.aws",
                "HostedZoneId": "Z03265862FOUNWMZOKUF4"
            },
```

```
            "AssociatedResourceArn": "arn:aws:vpc-lattice:us-
east-1:123456789012:resourceconfiguration/rcfg-07129f3acded87625"
        }
    ]
}
```

Per ulteriori informazioni, consulta Gestire le associazioni degli endpoint VPC nella Guida per l'AWS PrivateLink utente.

- Per i dettagli sulle API, consulta AWS CLI Command DescribeVpcEndpointAssociationsReference.

`describe-vpc-endpoint-connection-notifications`

Il seguente esempio di codice mostra come utilizzare`describe-vpc-endpoint-connection-notifications`.

AWS CLI

Come descrivere le notifica della connessione endpoint

L'esempio `describe-vpc-endpoint-connection-notifications` seguente descrive tutte le notifiche della connessione endpoint.

```
aws ec2 describe-vpc-endpoint-connection-notifications
```

Output:

```
{
   "ConnectionNotificationSet": [
       {
           "ConnectionNotificationState": "Enabled",
           "ConnectionNotificationType": "Topic",
           "ConnectionEvents": [
               "Accept",
               "Reject",
               "Delete",
               "Connect"
           ],
           "ConnectionNotificationId": "vpce-nfn-04bcb952bc8af7abc",
           "ConnectionNotificationArn": "arn:aws:sns:us-
east-1:123456789012:VpceNotification",
```

```
            "VpcEndpointId": "vpce-0324151a02f327123"
        }
    ]
}
```

- Per i dettagli sull'API, consulta DescribeVpcEndpointConnectionNotifications AWS CLICommand Reference.

`describe-vpc-endpoint-connections`

Il seguente esempio di codice mostra come utilizzare`describe-vpc-endpoint-connections`.

AWS CLI

Come descrivere le connessioni endpoint VPC

Questo esempio descrive le connessioni endpoint dell'interfaccia al servizio endpoint e filtra i risultati per visualizzare gli endpoint che sono nello stato `PendingAcceptance`.

Comando:

```
aws ec2 describe-vpc-endpoint-connections --filters Name=vpc-endpoint-state,Values=pendingAcceptance
```

Output:

```
{
  "VpcEndpointConnections": [
      {
          "VpcEndpointId": "vpce-0abed31004e618123",
          "ServiceId": "vpce-svc-0abced088d20def56",
          "CreationTimestamp": "2017-11-30T10:00:24.350Z",
          "VpcEndpointState": "pendingAcceptance",
          "VpcEndpointOwner": "123456789012"
      }
  ]
}
```

- Per i dettagli sull'API, consulta DescribeVpcEndpointConnections AWS CLICommand Reference.

`describe-vpc-endpoint-service-configurations`

Il seguente esempio di codice mostra come utilizzare`describe-vpc-endpoint-service-configurations`.

AWS CLI

Come descrivere le configurazioni del servizio endpoint

L'esempio `describe-vpc-endpoint-service-configurations` seguente descrive le configurazioni del servizio endpoint.

```
aws ec2 describe-vpc-endpoint-service-configurations
```

Output:

```
{
    "ServiceConfigurations": [
        {
            "ServiceType": [
                {
                    "ServiceType": "GatewayLoadBalancer"
                }
            ],
            "ServiceId": "vpce-svc-012d33a1c4321cabc",
            "ServiceName": "com.amazonaws.vpce.us-east-1.vpce-
svc-012d33a1c4321cabc",
            "ServiceState": "Available",
            "AvailabilityZones": [
                "us-east-1d"
            ],
            "AcceptanceRequired": false,
            "ManagesVpcEndpoints": false,
            "GatewayLoadBalancerArns": [
                "arn:aws:elasticloadbalancing:us-east-1:123456789012:loadbalancer/
gwy/GWLBService/123210844e429123"
            ],
            "Tags": []
        },
        {
            "ServiceType": [
                {
```

```
                    "ServiceType": "Interface"
                }
            ],
            "ServiceId": "vpce-svc-123cabc125efa123",
            "ServiceName": "com.amazonaws.vpce.us-east-1.vpce-svc-123cabc125efa123",
            "ServiceState": "Available",
            "AvailabilityZones": [
                "us-east-1a"
            ],
            "AcceptanceRequired": true,
            "ManagesVpcEndpoints": false,
            "NetworkLoadBalancerArns": [
                "arn:aws:elasticloadbalancing:us-east-1:123456789012:loadbalancer/
net/NLBforService/1238753950b25123"
            ],
            "BaseEndpointDnsNames": [
                "vpce-svc-123cabc125efa123.us-east-1.vpce.amazonaws.com"
            ],
            "PrivateDnsName": "example.com",
            "PrivateDnsNameConfiguration": {
                "State": "failed",
                "Type": "TXT",
                "Value": "vpce:qUAth3FdeABCApUiXabc",
                "Name": "_1d367jvbg34znqvyefrj"
            },
            "Tags": []
        }
    ]
}
```

Per ulteriori informazioni, vedere Concepts nella Guida AWS PrivateLink per l'utente.

- Per i dettagli sull'API, consulta DescribeVpcEndpointServiceConfigurations AWS CLICommand Reference.

`describe-vpc-endpoint-service-permissions`

Il seguente esempio di codice mostra come utilizzare`describe-vpc-endpoint-service-permissions`.

AWS CLI

Come descrivere le autorizzazioni del servizio endpoint

Questo esempio descrive le autorizzazioni per il servizio endpoint specificato.

Comando:

```
aws ec2 describe-vpc-endpoint-service-permissions --service-id vpce-svc-03d5ebb7d9579a2b3
```

Output:

```
{
   "AllowedPrincipals": [
       {
           "PrincipalType": "Account",
           "Principal": "arn:aws:iam::123456789012:root"
       }
   ]
}
```

- Per i dettagli sull'API, consulta DescribeVpcEndpointServicePermissions AWS CLICommand Reference.

`describe-vpc-endpoint-services`

Il seguente esempio di codice mostra come utilizzare`describe-vpc-endpoint-services`.

AWS CLI

Esempio 1: come descrivere tutti i servizi endpoint VPC

L'`describe-vpc-endpoint-services`esempio seguente elenca tutti i servizi endpoint VPC per una regione. AWS

```
aws ec2 describe-vpc-endpoint-services
```

Output:

```
{
    "ServiceDetails": [
        {
            "ServiceType": [
                {
                    "ServiceType": "Gateway"
```

```
            }
        ],
        "AcceptanceRequired": false,
        "ServiceName": "com.amazonaws.us-east-1.dynamodb",
        "VpcEndpointPolicySupported": true,
        "Owner": "amazon",
        "AvailabilityZones": [
            "us-east-1a",
            "us-east-1b",
            "us-east-1c",
            "us-east-1d",
            "us-east-1e",
            "us-east-1f"
        ],
        "BaseEndpointDnsNames": [
            "dynamodb.us-east-1.amazonaws.com"
        ]
    },
    {
        "ServiceType": [
            {
                "ServiceType": "Interface"
            }
        ],
        "PrivateDnsName": "ec2.us-east-1.amazonaws.com",
        "ServiceName": "com.amazonaws.us-east-1.ec2",
        "VpcEndpointPolicySupported": false,
        "Owner": "amazon",
        "AvailabilityZones": [
            "us-east-1a",
            "us-east-1b",
            "us-east-1c",
            "us-east-1d",
            "us-east-1e",
            "us-east-1f"
        ],
        "AcceptanceRequired": false,
        "BaseEndpointDnsNames": [
            "ec2.us-east-1.vpce.amazonaws.com"
        ]
    },
    {
        "ServiceType": [
            {
```

```
                    "ServiceType": "Interface"
                }
            ],
            "PrivateDnsName": "ssm.us-east-1.amazonaws.com",
            "ServiceName": "com.amazonaws.us-east-1.ssm",
            "VpcEndpointPolicySupported": true,
            "Owner": "amazon",
            "AvailabilityZones": [
                "us-east-1a",
                "us-east-1b",
                "us-east-1c",
                "us-east-1d",
                "us-east-1e"
            ],
            "AcceptanceRequired": false,
            "BaseEndpointDnsNames": [
                "ssm.us-east-1.vpce.amazonaws.com"
            ]
        }
    ],
    "ServiceNames": [
        "com.amazonaws.us-east-1.dynamodb",
        "com.amazonaws.us-east-1.ec2",
        "com.amazonaws.us-east-1.ec2messages",
        "com.amazonaws.us-east-1.elasticloadbalancing",
        "com.amazonaws.us-east-1.kinesis-streams",
        "com.amazonaws.us-east-1.s3",
        "com.amazonaws.us-east-1.ssm"
    ]
}
```

Esempio 2: come descrivere i dettagli di un servizio endpoint

L'esempio `describe-vpc-endpoint-services` seguente elenca i dettagli del servizio endpoint dell'interfaccia Amazon S3.

```
aws ec2 describe-vpc-endpoint-services \
    --filter 'Name=service-type,Values=Interface' Name=service-
name,Values=com.amazonaws.us-east-1.s3
```

Output:

```
{
```

```
    "ServiceDetails": [
        {
            "ServiceName": "com.amazonaws.us-east-1.s3",
            "ServiceId": "vpce-svc-081d84efcdEXAMPLE",
            "ServiceType": [
                {
                    "ServiceType": "Interface"
                }
            ],
            "AvailabilityZones": [
                "us-east-1a",
                "us-east-1b",
                "us-east-1c",
                "us-east-1d",
                "us-east-1e",
            "us-east-1f"
            ],
            "Owner": "amazon",
            "BaseEndpointDnsNames": [
                "s3.us-east-1.vpce.amazonaws.com"
            ],
            "VpcEndpointPolicySupported": true,
            "AcceptanceRequired": false,
            "ManagesVpcEndpoints": false,
            "Tags": []
        }
    ],
    "ServiceNames": [
        "com.amazonaws.us-east-1.s3"
    ]
}
```

Per ulteriori informazioni, consulta Visualizza i nomi dei AWS servizi disponibili nella Guida per l'AWS PrivateLink utente.

- Per i dettagli sull'API, consulta DescribeVpcEndpointServices AWS CLICommand Reference.

`describe-vpc-endpoints`

Il seguente esempio di codice mostra come utilizzare`describe-vpc-endpoints`.

AWS CLI

Come descrivere gli endpoint VPC

L'esempio `describe-vpc-endpoints` seguente visualizza i dettagli di tutti gli endpoint VPC.

```
aws ec2 describe-vpc-endpoints
```

Output:

```
{
    "VpcEndpoints": [
        {
            "PolicyDocument": "{\"Version\":\"2008-10-17\",\"Statement\":[{\"Effect
\":\"Allow\",\"Principal\":\"*\",\"Action\":\"*\",\"Resource\":\"*\"}]}",
            "VpcId": "vpc-aabb1122",
            "NetworkInterfaceIds": [],
            "SubnetIds": [],
            "PrivateDnsEnabled": true,
            "State": "available",
            "ServiceName": "com.amazonaws.us-east-1.dynamodb",
            "RouteTableIds": [
                "rtb-3d560345"
            ],
            "Groups": [],
            "VpcEndpointId": "vpce-032a826a",
            "VpcEndpointType": "Gateway",
            "CreationTimestamp": "2017-09-05T20:41:28Z",
            "DnsEntries": [],
            "OwnerId": "123456789012"
        },
        {
            "PolicyDocument": "{\n  \"Statement\": [\n    {\n      \"Action\": \"*
\", \n      \"Effect\": \"Allow\", \n      \"Principal\": \"*\", \n      \"Resource
\": \"*\"\n    }\n  ]\n}",
            "VpcId": "vpc-1a2b3c4d",
            "NetworkInterfaceIds": [
                "eni-2ec2b084",
                "eni-1b4a65cf"
            ],
            "SubnetIds": [
                "subnet-d6fcaa8d",
                "subnet-7b16de0c"
```

```
            ],
            "PrivateDnsEnabled": false,
            "State": "available",
            "ServiceName": "com.amazonaws.us-east-1.elasticloadbalancing",
            "RouteTableIds": [],
            "Groups": [
                {
                    "GroupName": "default",
                    "GroupId": "sg-54e8bf31"
                }
            ],
            "VpcEndpointId": "vpce-0f89a33420c1931d7",
            "VpcEndpointType": "Interface",
            "CreationTimestamp": "2017-09-05T17:55:27.583Z",
            "DnsEntries": [
                {
                    "HostedZoneId": "Z7HUB22UULQXV",
                    "DnsName": "vpce-0f89a33420c1931d7-
bluzidnv.elasticloadbalancing.us-east-1.vpce.amazonaws.com"
                },
                {
                    "HostedZoneId": "Z7HUB22UULQXV",
                    "DnsName": "vpce-0f89a33420c1931d7-bluzidnv-us-
east-1b.elasticloadbalancing.us-east-1.vpce.amazonaws.com"
                },
                {
                    "HostedZoneId": "Z7HUB22UULQXV",
                    "DnsName": "vpce-0f89a33420c1931d7-bluzidnv-us-
east-1a.elasticloadbalancing.us-east-1.vpce.amazonaws.com"
                }
            ],
            "OwnerId": "123456789012"
        },
        {
            "VpcEndpointId": "vpce-aabbaabbaabbaabba",
            "VpcEndpointType": "GatewayLoadBalancer",
            "VpcId": "vpc-111122223333aabbc",
            "ServiceName": "com.amazonaws.vpce.us-east-1.vpce-
svc-123123a1c43abc123",
            "State": "available",
            "SubnetIds": [
                "subnet-0011aabbcc2233445"
            ],
            "RequesterManaged": false,
```

```
            "NetworkInterfaceIds": [
                "eni-01010120203030405"
            ],
            "CreationTimestamp": "2020-11-11T08:06:03.522Z",
            "Tags": [],
            "OwnerId": "123456789012"
        }
    ]
}
```

Per ulteriori informazioni, vedere Concepts nella Guida AWS PrivateLink per l'utente.

- Per i dettagli sull'API, consulta DescribeVpcEndpoints AWS CLICommand Reference.

`describe-vpc-peering-connections`

Il seguente esempio di codice mostra come utilizzare`describe-vpc-peering-connections`.

AWS CLI

Come descrivere le connessioni peering VPC

Questo esempio descrive tutte le connessioni peering VPC.

Comando:

```
aws ec2 describe-vpc-peering-connections
```

Output:

```
{
    "VpcPeeringConnections": [
        {
            "Status": {
                "Message": "Active",
                "Code": "active"
            },
            "Tags": [
                {
                    "Value": "Peering-1",
                    "Key": "Name"
                }
```

```
            ],
            "AccepterVpcInfo": {
                "OwnerId": "111122223333",
                "VpcId": "vpc-1a2b3c4d",
                "CidrBlock": "10.0.1.0/28"
            },
            "VpcPeeringConnectionId": "pcx-11122233",
            "RequesterVpcInfo": {
                "PeeringOptions": {
                    "AllowEgressFromLocalVpcToRemoteClassicLink": false,
                    "AllowEgressFromLocalClassicLinkToRemoteVpc": false
                },
                "OwnerId": "444455556666",
                "VpcId": "vpc-123abc45",
                "CidrBlock": "192.168.0.0/16"
            }
        },
        {
            "Status": {
                "Message": "Pending Acceptance by 444455556666",
                "Code": "pending-acceptance"
            },
            "Tags": [],
            "RequesterVpcInfo": {
                "PeeringOptions": {
                    "AllowEgressFromLocalVpcToRemoteClassicLink": false,
                    "AllowEgressFromLocalClassicLinkToRemoteVpc": false
                },
                "OwnerId": "444455556666",
                "VpcId": "vpc-11aa22bb",
                "CidrBlock": "10.0.0.0/28"
            },
            "VpcPeeringConnectionId": "pcx-abababab",
            "ExpirationTime": "2014-04-03T09:12:43.000Z",
            "AccepterVpcInfo": {
                "OwnerId": "444455556666",
                "VpcId": "vpc-33cc44dd"
            }
        }
    ]
}
```

Come descrivere connessioni peering VPC specifiche

Questo esempio descrive tutte le connessioni peering VPC che si trovano nello stato di accettazione in sospeso.

Comando:

```
aws ec2 describe-vpc-peering-connections --filters Name=status-code,Values=pending-acceptance
```

Questo esempio descrive tutte le connessioni peering VPC con il tag Owner=Finance.

Comando:

```
aws ec2 describe-vpc-peering-connections --filters Name=tag:Owner,Values=Finance
```

Questo esempio descrive tutte le connessioni peering VPC richieste per il VPC specificato, vpc-1a2b3c4d.

Comando:

```
aws ec2 describe-vpc-peering-connections --filters Name=requester-vpc-info.vpc-id,Values=vpc-1a2b3c4d
```

- Per i dettagli sull'API, consulta DescribeVpcPeeringConnections AWS CLICommand Reference.

`describe-vpcs`

Il seguente esempio di codice mostra come utilizzare`describe-vpcs`.

AWS CLI

Esempio 1: Per descrivere tutti i tuoi VPCs

L'`describe-vpcs`esempio seguente recupera i dettagli sul tuo VPCs.

```
aws ec2 describe-vpcs
```

Output:

```
{
    "Vpcs": [
```

```
        {
            "CidrBlock": "30.1.0.0/16",
            "DhcpOptionsId": "dopt-19edf471",
            "State": "available",
            "VpcId": "vpc-0e9801d129EXAMPLE",
            "OwnerId": "111122223333",
            "InstanceTenancy": "default",
            "CidrBlockAssociationSet": [
                {
                    "AssociationId": "vpc-cidr-assoc-062c64cfafEXAMPLE",
                    "CidrBlock": "30.1.0.0/16",
                    "CidrBlockState": {
                        "State": "associated"
                    }
                }
            ],
            "IsDefault": false,
            "Tags": [
                {
                    "Key": "Name",
                    "Value": "Not Shared"
                }
            ]
        },
        {
            "CidrBlock": "10.0.0.0/16",
            "DhcpOptionsId": "dopt-19edf471",
            "State": "available",
            "VpcId": "vpc-06e4ab6c6cEXAMPLE",
            "OwnerId": "222222222222",
            "InstanceTenancy": "default",
            "CidrBlockAssociationSet": [
                {
                    "AssociationId": "vpc-cidr-assoc-00b17b4eddEXAMPLE",
                    "CidrBlock": "10.0.0.0/16",
                    "CidrBlockState": {
                        "State": "associated"
                    }
                }
            ],
            "IsDefault": false,
            "Tags": [
                {
                    "Key": "Name",
```

```
                    "Value": "Shared VPC"
                }
            ]
        }
    ]
}
```

Esempio 2: come descrivere un VPC specificato

Nell'esempio `describe-vpcs` seguente vengono recuperati i dettagli per il VPC specificato.

```
aws ec2 describe-vpcs \
    --vpc-ids vpc-06e4ab6c6cEXAMPLE
```

Output:

```
{
    "Vpcs": [
        {
            "CidrBlock": "10.0.0.0/16",
            "DhcpOptionsId": "dopt-19edf471",
            "State": "available",
            "VpcId": "vpc-06e4ab6c6cEXAMPLE",
            "OwnerId": "111122223333",
            "InstanceTenancy": "default",
            "CidrBlockAssociationSet": [
                {
                    "AssociationId": "vpc-cidr-assoc-00b17b4eddEXAMPLE",
                    "CidrBlock": "10.0.0.0/16",
                    "CidrBlockState": {
                        "State": "associated"
                    }
                }
            ],
            "IsDefault": false,
            "Tags": [
                {
                    "Key": "Name",
                    "Value": "Shared VPC"
                }
            ]
        }
    ]
```

```
}
```

- Per i dettagli sull'API, vedere DescribeVpcsin AWS CLI Command Reference.

`describe-vpn-connections`

Il seguente esempio di codice mostra come utilizzare`describe-vpn-connections`.

AWS CLI

Esempio 1: come descrivere le connessioni VPN

L'`describe-vpn-connections`esempio seguente descrive tutte le connessioni Site-to-Site VPN.

```
aws ec2 describe-vpn-connections
```

Output:

```
{
    "VpnConnections": [
        {
            "CustomerGatewayConfiguration": "...configuration information...",
            "CustomerGatewayId": "cgw-01234567abcde1234",
            "Category": "VPN",
            "State": "available",
            "Type": "ipsec.1",
            "VpnConnectionId": "vpn-1122334455aabbccd",
            "TransitGatewayId": "tgw-00112233445566aab",
            "Options": {
                "EnableAcceleration": false,
                "StaticRoutesOnly": true,
                "LocalIpv4NetworkCidr": "0.0.0.0/0",
                "RemoteIpv4NetworkCidr": "0.0.0.0/0",
                "TunnelInsideIpVersion": "ipv4"
            },
            "Routes": [],
            "Tags": [
                {
                    "Key": "Name",
                    "Value": "CanadaVPN"
```

```
                }
            ],
            "VgwTelemetry": [
                {
                    "AcceptedRouteCount": 0,
                    "LastStatusChange": "2020-07-29T10:35:11.000Z",
                    "OutsideIpAddress": "203.0.113.3",
                    "Status": "DOWN",
                    "StatusMessage": ""
                },
                {
                    "AcceptedRouteCount": 0,
                    "LastStatusChange": "2020-09-02T09:09:33.000Z",
                    "OutsideIpAddress": "203.0.113.5",
                    "Status": "UP",
                    "StatusMessage": ""
                }
            ]
        }
    ]
}
```

Per ulteriori informazioni, consulta Come funziona la AWS Site-to-Site VPN nella Guida per l'utente della AWS Site-to-Site VPN.

Esempio 2: come descrivere le connessioni VPN disponibili

L'`describe-vpn-connections`esempio seguente descrive le tue connessioni Site-to-Site VPN con uno stato di`available`.

```
aws ec2 describe-vpn-connections \
    --filters "Name=state,Values=available"
```

Per ulteriori informazioni, consulta Come funziona la AWS Site-to-Site VPN nella Guida per l'utente della AWS Site-to-Site VPN.

- Per i dettagli sull'API, consulta DescribeVpnConnections AWS CLICommand Reference.

`describe-vpn-gateways`

Il seguente esempio di codice mostra come utilizzare`describe-vpn-gateways`.

AWS CLI

Come descrivere i gateway privati virtuali

Questo esempio descrive i gateway privati virtuali.

Comando:

```
aws ec2 describe-vpn-gateways
```

Output:

```
{
    "VpnGateways": [
        {
            "State": "available",
            "Type": "ipsec.1",
            "VpnGatewayId": "vgw-f211f09b",
            "VpcAttachments": [
                {
                    "State": "attached",
                    "VpcId": "vpc-98eb5ef5"
                }
            ]
        },
        {
            "State": "available",
            "Type": "ipsec.1",
            "VpnGatewayId": "vgw-9a4cacf3",
            "VpcAttachments": [
                {
                    "State": "attaching",
                    "VpcId": "vpc-a01106c2"
                }
            ]
        }
    ]
}
```

- Per i dettagli sull'API, consulta DescribeVpnGateways AWS CLICommand Reference.

`detach-classic-link-vpc`

Il seguente esempio di codice mostra come utilizzare`detach-classic-link-vpc`.

AWS CLI

Per scollegare (scollegare) un'istanza EC2 -Classic da un VPC

Questo esempio scollega l'istanza i-0598c7d356eba48d7 dal VPC vpc-88888888.

Comando:

```
aws ec2 detach-classic-link-vpc --instance-id i-0598c7d356eba48d7 --vpc-
id vpc-88888888
```

Output:

```
{
  "Return": true
}
```

- Per i dettagli sull'API, consulta DetachClassicLinkVpcCommand Reference.AWS CLI

`detach-internet-gateway`

Il seguente esempio di codice mostra come utilizzare`detach-internet-gateway`.

AWS CLI

Come scollegare un gateway Internet dal VPC

L'esempio `detach-internet-gateway` seguente scollega il gateway Internet specificato dal VPC specifico.

```
aws ec2 detach-internet-gateway \
    --internet-gateway-id igw-0d0fb496b3EXAMPLE \
    --vpc-id vpc-0a60eb65b4EXAMPLE
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Gateway Internet nella Guida per l'utente di Amazon VPC.

- Per i dettagli sull'API, consulta DetachInternetGateway AWS CLICommand Reference.

`detach-network-interface`

Il seguente esempio di codice mostra come utilizzare`detach-network-interface`.

AWS CLI

Come scollegare un'interfaccia di rete da un'istanza

Questo esempio scollega l'interfaccia di rete specificata dall'istanza specificata. Se il comando va a buon fine, non viene restituito alcun output.

Comando:

```
aws ec2 detach-network-interface --attachment-id eni-attach-66c4350a
```

- Per i dettagli sull'API, consulta DetachNetworkInterface AWS CLICommand Reference.

`detach-verified-access-trust-provider`

Il seguente esempio di codice mostra come utilizzare`detach-verified-access-trust-provider`.

AWS CLI

Come scollegare un provider di attendibilità da un'istanza

L'esempio `detach-verified-access-trust-provider` seguente scollega il provider di attendibilità di accesso verificato specificato dall'istanza di accesso verificato specificata.

```
aws ec2 detach-verified-access-trust-provider \
    --verified-access-instance-id vai-0ce000c0b7643abea \
    --verified-access-trust-provider-id vatp-0bb32de759a3e19e7
```

Output:

```
{
    "VerifiedAccessTrustProvider": {
        "VerifiedAccessTrustProviderId": "vatp-0bb32de759a3e19e7",
        "Description": "Testing Verified Access",
        "TrustProviderType": "user",
        "UserTrustProviderType": "iam-identity-center",
        "PolicyReferenceName": "idc",
```

```
        "CreationTime": "2023-08-25T19:00:38",
        "LastUpdatedTime": "2023-08-25T19:00:38"
    },
    "VerifiedAccessInstance": {
        "VerifiedAccessInstanceId": "vai-0ce000c0b7643abea",
        "Description": "Testing Verified Access",
        "VerifiedAccessTrustProviders": [],
        "CreationTime": "2023-08-25T18:27:56",
        "LastUpdatedTime": "2023-08-25T18:27:56"
    }
}
```

Per ulteriori informazioni, consulta Istanze di Accesso verificato nella Guida per l'utente di Accesso verificato da AWS .

- Per i dettagli sull'API, consulta DetachVerifiedAccessTrustProvider AWS CLICommand Reference.

`detach-volume`

Il seguente esempio di codice mostra come utilizzare`detach-volume`.

AWS CLI

Come scollegare un volume da un'istanza

Questo comando di esempio scollega il volume (`vol-049df61146c4d7901`) dall'istanza a cui è collegato.

Comando:

```
aws ec2 detach-volume --volume-id vol-1234567890abcdef0
```

Output:

```
{
    "AttachTime": "2014-02-27T19:23:06.000Z",
    "InstanceId": "i-1234567890abcdef0",
    "VolumeId": "vol-049df61146c4d7901",
    "State": "detaching",
    "Device": "/dev/sdb"
}
```

- Per i dettagli sull'API, consulta DetachVolume AWS CLICommand Reference.

`detach-vpn-gateway`

Il seguente esempio di codice mostra come utilizzare`detach-vpn-gateway`.

AWS CLI

Come scollegare un gateway privato virtuale da un VPC

Questo esempio scollega il gateway privato virtuale specificato dal VPC specificato. Se il comando va a buon fine, non viene restituito alcun output.

Comando:

```
aws ec2 detach-vpn-gateway --vpn-gateway-id vgw-9a4cacf3 --vpc-id vpc-a01106c2
```

- Per i dettagli sull'API, consulta DetachVpnGateway AWS CLICommand Reference.

`disable-address-transfer`

Il seguente esempio di codice mostra come utilizzare`disable-address-transfer`.

AWS CLI

Come disabilitare il trasferimento degli indirizzi IP elastici

L'esempio `disable-address-transfer` seguente disabilita il trasferimento degli indirizzi IP elastici per l'indirizzo IP elastico specificato.

```
aws ec2 disable-address-transfer \
    --allocation-id eipalloc-09ad461b0d03f6aaf
```

Output:

```
{
    "AddressTransfer": {
        "PublicIp": "100.21.184.216",
        "AllocationId": "eipalloc-09ad461b0d03f6aaf",
        "AddressTransferStatus": "disabled"
    }
```

```
}
```

Per ulteriori informazioni, consulta Trasferimento degli indirizzi IP elastici nella Guida per l'utente di Amazon VPC.

- Per i dettagli sull'API, consulta DisableAddressTransfer AWS CLICommand Reference.

`disable-aws-network-performance-metric-subscription`

Il seguente esempio di codice mostra come utilizzare`disable-aws-network-performance-metric-subscription`.

AWS CLI

Come disabilitare una sottoscrizioni di parametri di prestazioni

L'esempio `disable-aws-network-performance-metric-subscription` seguente disabilita il monitoraggio della latenza di rete aggregata tra le Regioni di origine e di destinazione specificate.

```
aws ec2 disable-aws-network-performance-metric-subscription \
    --source us-east-1 \
    --destination eu-west-1 \
    --metric aggregate-latency \
    --statistic p50
```

Output:

```
{
    "Output": true
}
```

Per ulteriori informazioni, consulta Gestire CloudWatch gli abbonamenti utilizzando la CLI nella Infrastructure Performance User Guide.

- Per i dettagli sull'API, consulta AWS CLI Command DisableAwsNetworkPerformanceMetricSubscriptionReference.

`disable-ebs-encryption-by-default`

Il seguente esempio di codice mostra come utilizzare`disable-ebs-encryption-by-default`.

AWS CLI

Come disabilitare la crittografia EBS per impostazione predefinita

L'`disable-ebs-encryption-by-default`esempio seguente disabilita la crittografia EBS per impostazione predefinita per il tuo AWS account nella regione corrente.

```
aws ec2 disable-ebs-encryption-by-default
```

Output:

```
{
    "EbsEncryptionByDefault": false
}
```

- Per i dettagli sull'API, consulta AWS CLI Command DisableEbsEncryptionByDefaultReference.

`disable-fast-launch`

Il seguente esempio di codice mostra come utilizzare`disable-fast-launch`.

AWS CLI

Come interrompere l'avvio rapido di un'immagine

L'esempio `disable-fast-launch` seguente interrompe l'avvio rapido per l'AMI specificata e rimuove gli snapshot con pre-provisioning esistenti.

```
aws ec2 disable-fast-launch \
    --image-id ami-01234567890abcedf
```

Output:

```
{
    "ImageId": "ami-01234567890abcedf",
    "ResourceType": "snapshot",
    "SnapshotConfiguration": {},
    "LaunchTemplate": {
        "LaunchTemplateId": "lt-01234567890abcedf",
        "LaunchTemplateName": "EC2FastLaunchDefaultResourceCreation-
a8c6215d-94e6-441b-9272-dbd1f87b07e2",
```

```
        "Version": "1"
    },
    "MaxParallelLaunches": 6,
    "OwnerId": "0123456789123",
    "State": "disabling",
    "StateTransitionReason": "Client.UserInitiated",
    "StateTransitionTime": "2022-01-27T22:47:29.265000+00:00"
}
```

Per ulteriori informazioni, consulta Configurare le impostazioni di EC2 Fast Launch per la tua AMI Windows nella Amazon EC2 User Guide.

- Per i dettagli sull'API, consulta DisableFastLaunch AWS CLICommand Reference.

`disable-fast-snapshot-restores`

Il seguente esempio di codice mostra come utilizzare`disable-fast-snapshot-restores`.

AWS CLI

Come disabilitare il ripristino rapido degli snapshot

L'esempio `disable-fast-snapshot-restores` seguente disabilita il ripristino rapido degli snapshot per gli snapshot specificati nella zona di disponibilità specificata.

```
aws ec2 disable-fast-snapshot-restores \
    --availability-zones us-east-2a \
    --source-snapshot-ids snap-1234567890abcdef0
```

Output:

```
{
    "Successful": [
        {
            "SnapshotId": "snap-1234567890abcdef0"
            "AvailabilityZone": "us-east-2a",
            "State": "disabling",
            "StateTransitionReason": "Client.UserInitiated",
            "OwnerId": "123456789012",
            "EnablingTime": "2020-01-25T23:57:49.602Z"
        }
    ],
```

```
    "Unsuccessful": []
}
```

- Per i dettagli sull'API, consulta DisableFastSnapshotRestores AWS CLICommand Reference.

`disable-image-block-public-access`

Il seguente esempio di codice mostra come utilizzare`disable-image-block-public-access`.

AWS CLI

Per disabilitare l'accesso pubblico a blocchi AMIs nella regione specificata

L'`disable-image-block-public-access`esempio seguente disabilita l'accesso pubblico a blocchi AMIs a livello di account nella regione specificata.

```
aws ec2 disable-image-block-public-access \
    --region us-east-1
```

Output:

```
{
    "ImageBlockPublicAccessState": "unblocked"
}
```

Per ulteriori informazioni, consulta Bloccare l'accesso pubblico al tuo AMIs nella Amazon EC2 User Guide.

- Per i dettagli sull'API, consulta DisableImageBlockPublicAccess AWS CLICommand Reference.

`disable-image-deprecation`

Il seguente esempio di codice mostra come utilizzare`disable-image-deprecation`.

AWS CLI

Come annullare la deprecazione di un'AMI

L'esempio `disable-image-deprecation` seguente annulla l'obsolescenza di un'AMI, che rimuove il campo `DeprecationTime` dall'output `describe-images`. Per eseguire questa procedura, è necessario essere il proprietario dell'AMI.

```
aws ec2 disable-image-deprecation \
    --image-id ami-1234567890abcdef0
```

Output:

```
{
    "RequestID": "11aabb229-4eac-35bd-99ed-be587EXAMPLE",
    "Return": "true"
}
```

Per ulteriori informazioni, consulta Deprecare un'AMI nella Amazon EC2 User Guide.

- Per i dettagli sull'API, consulta AWS CLI Command DisableImageDeprecationReference.

`disable-image-deregistration-protection`

Il seguente esempio di codice mostra come utilizzare`disable-image-deregistration-protection`.

AWS CLI

Come disabilitare la protezione da annullamento della registrazione

L'esempio `disable-image-deregistration-protection` seguente disabilita la protezione da annullamento della registrazione per l'immagine specificata.

```
aws ec2 disable-image-deregistration-protection \
    --image-id ami-0b1a928a144a74ec9
```

Output:

```
{
    "Return": "disabled"
}
```

Per ulteriori informazioni, consulta Proteggere un AMI dall'annullamento della registrazione nella Amazon EC2 User Guide.

- Per i dettagli sull'API, consulta Command DisableImageDeregistrationProtectionReference AWS CLI .

`disable-image`

Il seguente esempio di codice mostra come utilizzare`disable-image`.

AWS CLI

Come disabilitare un'AMI

L'esempio `disable-image` seguente disabilita l'AMI specificata.

```
aws ec2 disable-image \
    --image-id ami-1234567890abcdef0
```

Output:

```
{
    "Return": "true"
}
```

Per ulteriori informazioni, consulta Disabilitare un'AMI nella Amazon EC2 User Guide.

- Per i dettagli sull'API, consulta DisableImage AWS CLICommand Reference.

`disable-ipam-organization-admin-account`

Il seguente esempio di codice mostra come utilizzare`disable-ipam-organization-admin-account`.

AWS CLI

Come disabilitare l'amministratore IPAM delegato

In alcuni scenari, integrerai IPAM con AWS Organizations. Quando si esegue questa operazione, l'account di gestione AWS Organizations delega un account membro di AWS Organizations come amministratore IPAM.

In questo esempio, sei l'account di gestione AWS Organizations che ha delegato l'account amministratore IPAM e desideri disabilitare tale account dall'essere amministratore IPAM.

È possibile utilizzare qualsiasi AWS regione per effettuare questa `--region` richiesta. Non è necessario utilizzare la Regione in cui è stato originariamente delegato l'amministratore, dove è

stato creato l'IPAM o una Regione operativa dell'IPAM. Se si disattiva l'account amministratore delegato, è possibile riattivarlo in qualsiasi momento o delegare un nuovo account come amministratore IPAM.

L'`disable-ipam-organization-admin-account`esempio seguente disabilita l'amministratore IPAM delegato nel tuo account. AWS

```
aws ec2 disable-ipam-organization-admin-account \
    --delegated-admin-account-id 320805250157 \
    --region ap-south-1
```

Output:

```
{
    "Success": true
}
```

Per ulteriori informazioni, consulta Integrare IPAM con gli account di un' AWS organizzazione nella Guida per l'utente IPAM di Amazon VPC.

- Per i dettagli sull'API, consulta Command DisableIpamOrganizationAdminAccountReference AWS CLI .

`disable-serial-console-access`

Il seguente esempio di codice mostra come utilizzare`disable-serial-console-access`.

AWS CLI

Per disabilitare l'accesso alla console EC2 seriale per il tuo account

L'esempio `disable-serial-console-access` seguente disabilita l'accesso account alla console seriale.

```
aws ec2 disable-serial-console-access
```

Output:

```
{
    "SerialConsoleAccessEnabled": false
```

```
}
```

Per ulteriori informazioni, consulta EC2 Serial Console nella Amazon EC2 User Guide.

- Per i dettagli sull'API, consulta DisableSerialConsoleAccess AWS CLICommand Reference.

`disable-snapshot-block-public-access`

Il seguente esempio di codice mostra come utilizzare`disable-snapshot-block-public-access`.

AWS CLI

Come disabilitare il blocco dell'accesso pubblico per gli snapshot

L'esempio `disable-snapshot-block-public-access` seguente disabilita il blocco dell'accesso pubblico per gli snapshot per consentire la condivisione pubblica degli snapshot.

```
aws ec2 disable-snapshot-block-public-access
```

Output:

```
{
    "State": "unblocked"
}
```

Per ulteriori informazioni, consulta Blocco dell'accesso pubblico per gli snapshot nella Guida per l'utente di Amazon EBS.

- Per i dettagli sull'API, consulta DisableSnapshotBlockPublicAccess AWS CLICommand Reference.

`disable-transit-gateway-route-table-propagation`

Il seguente esempio di codice mostra come utilizzare`disable-transit-gateway-route-table-propagation`.

AWS CLI

Come disabilitare un collegamento del gateway di transito alla VPN per propagare le route alla tabella di routing di propagazione specificata

L'esempio `disable-transit-gateway-route-table-propagation` seguente disabilita l'allegato specificato per propagare le route alla tabella di routing di propagazione specificata.

```
aws ec2 disable-transit-gateway-route-table-propagation \
    --transit-gateway-route-table-id tgw-rtb-0a823edbdeEXAMPLE \
    --transit-gateway-attachment-id tgw-attach-09b52ccdb5EXAMPLE
```

Output:

```
{
    "Propagation": {
        "TransitGatewayAttachmentId": "tgw-attach-09b52ccdb5EXAMPLE",
        "ResourceId": "vpc-4d7de228",
        "ResourceType": "vpc",
        "TransitGatewayRouteTableId": "tgw-rtb-0a823edbdeEXAMPLE",
        "State": "disabled"
    }
}
```

Per ulteriori informazioni, consulta Tabelle di routing del gateway di transito nella Guida di Transit Gateway.

- Per i dettagli sull'API, consulta DisableTransitGatewayRouteTablePropagation AWS CLICommand Reference.

`disable-vgw-route-propagation`

Il seguente esempio di codice mostra come utilizzare`disable-vgw-route-propagation`.

AWS CLI

Come disabilitare la propagazione di route

Questo esempio disabilita il gateway privato virtuale specificato per propagare route statiche alla tabella di routing specificata. Se il comando va a buon fine, non viene restituito alcun output.

Comando:

```
aws ec2 disable-vgw-route-propagation --route-table-id rtb-22574640 --gateway-
id vgw-9a4cacf3
```

- Per i dettagli sull'API, consulta DisableVgwRoutePropagation AWS CLICommand Reference.

`disable-vpc-classic-link-dns-support`

Il seguente esempio di codice mostra come utilizzare`disable-vpc-classic-link-dns-support`.

AWS CLI

Per disabilitare il supporto ClassicLink DNS per un VPC

Questo esempio disabilita il supporto ClassicLink DNS per. `vpc-88888888`

Comando:

```
aws ec2 disable-vpc-classic-link-dns-support --vpc-id vpc-88888888
```

Output:

```
{
  "Return": true
}
```

- Per i dettagli sull'API, vedere DisableVpcClassicLinkDnsSupportin AWS CLI Command Reference.

`disable-vpc-classic-link`

Il seguente esempio di codice mostra come utilizzare`disable-vpc-classic-link`.

AWS CLI

Da disattivare ClassicLink per un VPC

Questo esempio disabilita ClassicLink vpc-8888888.

Comando:

```
aws ec2 disable-vpc-classic-link --vpc-id vpc-88888888
```

Output:

```
{
  "Return": true
```

```
}
```

- Per i dettagli sull'API, consulta DisableVpcClassicLinkCommand Reference.AWS CLI

`disassociate-address`

Il seguente esempio di codice mostra come utilizzare`disassociate-address`.

AWS CLI

Per dissociare un indirizzo IP elastico in EC2 -Classic

Questo esempio dissocia un indirizzo IP elastico da un'istanza in -Classic. EC2 Se il comando va a buon fine, non viene restituito alcun output.

Comando:

```
aws ec2 disassociate-address --public-ip 198.51.100.0
```

Per dissociare un indirizzo IP elastico in EC2 -VPC

Nell'esempio seguente viene rimossa l'associazione di un indirizzo IP elastico a un'istanza in un VPC. Se il comando va a buon fine, non viene restituito alcun output.

Comando:

```
aws ec2 disassociate-address --association-id eipassoc-2bebb745
```

- Per i dettagli sull'API, consulta Command DisassociateAddressReference AWS CLI .

`disassociate-client-vpn-target-network`

Il seguente esempio di codice mostra come utilizzare`disassociate-client-vpn-target-network`.

AWS CLI

Come annullare l'associazione di una rete da un endpoint VPN Client

L'esempio `disassociate-client-vpn-target-network` seguente annulla l'associazione della rete di destinazione associata all'ID associazione `cvpn-assoc-12312312312312312` per l'endpoint VPN Client specificato.

```
aws ec2 disassociate-client-vpn-target-network \
    --client-vpn-endpoint-id cvpn-endpoint-123456789123abcde \
    --association-id cvpn-assoc-12312312312312312
```

Output:

```
{
    "AssociationId": "cvpn-assoc-12312312312312312",
    "Status": {
        "Code": "disassociating"
    }
}
```

Per ulteriori informazioni, consulta Reti target nella Guida per l'amministratore di VPN Client di AWS .

- Per i dettagli sull'API, consulta DisassociateClientVpnTargetNetwork AWS CLICommand Reference.

`disassociate-iam-instance-profile`

Il seguente esempio di codice mostra come utilizzare`disassociate-iam-instance-profile`.

AWS CLI

Come annullare l'associazione di un profilo dell'istanza IAM

Questo esempio annulla l'associazione di un profilo dell'istanza IAM dall'ID associazione `iip-assoc-05020b59952902f5f`.

Comando:

```
aws ec2 disassociate-iam-instance-profile --association-id iip-
assoc-05020b59952902f5f
```

Output:

```
{
  "IamInstanceProfileAssociation": {
      "InstanceId": "i-123456789abcde123",
      "State": "disassociating",
      "AssociationId": "iip-assoc-05020b59952902f5f",
```

```
        "IamInstanceProfile": {
            "Id": "AIPAI5IVIHMFFYY2DKV5Y",
            "Arn": "arn:aws:iam::123456789012:instance-profile/admin-role"
        }
    }
}
```

- Per i dettagli sull'API, consulta DisassociateIamInstanceProfile AWS CLICommand Reference.

`disassociate-instance-event-window`

Il seguente esempio di codice mostra come utilizzare`disassociate-instance-event-window`.

AWS CLI

Esempio 1: come dissociare una o più istanze da una finestra di eventi

L'esempio `disassociate-instance-event-window` seguente annulla l'associazione di una o più istanze da una finestra di eventi. Imposta il parametro `instance-event-window-id` per specificare la finestra di eventi. Per dissociare le istanze, specificate il `association-target` parametro e, per i valori dei parametri, specificate una o più istanze. IDs

```
aws ec2 disassociate-instance-event-window \
    --region us-east-1 \
    --instance-event-window-id iew-0abcdef1234567890 \
    --association-target "InstanceIds=i-1234567890abcdef0,i-0598c7d356eba48d7"
```

Output:

```
{
    "InstanceEventWindow": {
        "InstanceEventWindowId": "iew-0abcdef1234567890",
        "Name": "myEventWindowName",
        "CronExpression": "* 21-23 * * 2,3",
        "AssociationTarget": {
            "InstanceIds": [],
            "Tags": [],
            "DedicatedHostIds": []
        },
        "State": "creating"
    }
}
```

Per i vincoli relativi alla finestra degli eventi, consulta Considerazioni nella sezione Eventi pianificati della Amazon EC2 User Guide.

Esempio 2: come dissociare i tag di istanza da una finestra di eventi

L'esempio `disassociate-instance-event-window` seguente annulla l'associazione dei tag di istanza da una finestra di eventi. Imposta il parametro `instance-event-window-id` per specificare la finestra di eventi. Per dissociare i tag di istanza, specifica il parametro `association-target` e per i valori dei parametri specifica uno o più tag.

```
aws ec2 disassociate-instance-event-window \
    --region us-east-1 \
    --instance-event-window-id iew-0abcdef1234567890 \
    --association-target "InstanceTags=[{Key=k2,Value=v2},{Key=k1,Value=v1}]"
```

Output:

```
{
    "InstanceEventWindow": {
        "InstanceEventWindowId": "iew-0abcdef1234567890",
        "Name": "myEventWindowName",
        "CronExpression": "* 21-23 * * 2,3",
        "AssociationTarget": {
            "InstanceIds": [],
            "Tags": [],
            "DedicatedHostIds": []
        },
        "State": "creating"
    }
}
```

Per i vincoli relativi alla finestra degli eventi, consulta Considerazioni nella sezione Eventi pianificati della Amazon EC2 User Guide.

Esempio 3: come dissociare un host dedicato da una finestra di eventi

L'esempio `disassociate-instance-event-window` seguente dissocia un host dedicato da una finestra di eventi. Imposta il parametro `instance-event-window-id` per specificare la finestra di eventi. Per dissociare un host dedicato, specifica il `association-target` parametro e, per i valori dei parametri, specifica uno o più host dedicati. IDs

```
aws ec2 disassociate-instance-event-window \
```

```
    --region us-east-1 \
    --instance-event-window-id iew-0abcdef1234567890 \
    --association-target DedicatedHostIds=h-029fa35a02b99801d
```

Output:

```
{
    "InstanceEventWindow": {
        "InstanceEventWindowId": "iew-0abcdef1234567890",
        "Name": "myEventWindowName",
        "CronExpression": "* 21-23 * * 2,3",
        "AssociationTarget": {
            "InstanceIds": [],
            "Tags": [],
            "DedicatedHostIds": []
        },
        "State": "creating"
    }
}
```

Per i vincoli relativi alla finestra degli eventi, consulta Considerazioni nella sezione Eventi pianificati della Amazon EC2 User Guide.

- Per i dettagli sull'API, consulta Command DisassociateInstanceEventWindowReference AWS CLI .

`disassociate-ipam-resource-discovery`

Il seguente esempio di codice mostra come utilizzare`disassociate-ipam-resource-discovery`.

AWS CLI

Come annullare l'associazione di un rilevamento delle risorse da un'IPAM

In questo esempio, l'utente è un account amministratore delegato IPAM e desidera dissociare un rilevamento di risorse IPAM dall'IPAM in uso. L'utente ha eseguito il comando describe e ha notato che `"ResourceDiscoveryStatus": "not-found"` e desidera dissociarlo dall'IPAM per fare spazio ad altre associazioni.

L'`disassociate-ipam-resource-discovery`esempio seguente dissocia l'individuazione di una risorsa IPAM nell'account. AWS

```
aws ec2 disassociate-ipam-resource-discovery \
    --ipam-resource-discovery-association-id ipam-res-disco-assoc-04382a6346357cf82
 \
    --region us-east-1
```

Output:

```
{
    "IpamResourceDiscoveryAssociation": {
        "OwnerId": "320805250157",
        "IpamResourceDiscoveryAssociationId": "ipam-res-disco-
assoc-04382a6346357cf82",
        "IpamResourceDiscoveryAssociationArn":
 "arn:aws:ec2::320805250157:ipam-resource-discovery-association/ipam-res-disco-
assoc-04382a6346357cf82",
        "IpamResourceDiscoveryId": "ipam-res-disco-0365d2977fc1672fe",
        "IpamId": "ipam-005f921c17ebd5107",
        "IpamArn": "arn:aws:ec2::320805250157:ipam/ipam-005f921c17ebd5107",
        "IpamRegion": "us-east-1",
        "IsDefault": false,
        "ResourceDiscoveryStatus": "not-found",
        "State": "disassociate-in-progress"
    }
}
```

Per ulteriori informazioni, consulta Come integrare IPAM con account esterni alla tua organizzazione nella Guida per l'utente IPAM di Amazon VPC.

- Per i dettagli sull'API, consulta Command DisassociateIpamResourceDiscoveryReference AWS CLI .

`disassociate-nat-gateway-address`

Il seguente esempio di codice mostra come utilizzare`disassociate-nat-gateway-address`.

AWS CLI

Come dissociare un indirizzo IP elastico da un gateway NAT pubblico

L'esempio `disassociate-nat-gateway-address` seguente dissocia l'indirizzo IP elastico specificato dal gateway NAT pubblico specificato.

```
aws ec2 disassociate-nat-gateway-address \
    --nat-gateway-id nat-1234567890abcdef0 \
    --association-ids eipassoc-0f96bdca17EXAMPLE
```

Output:

```
{
    "NatGatewayId": "nat-1234567890abcdef0",
    "NatGatewayAddresses": [
        {
            "AllocationId": "eipalloc-0be6ecac95EXAMPLE",
            "NetworkInterfaceId": "eni-09cc4b2558794f7f9",
            "PrivateIp": "10.0.0.74",
            "PublicIp": "3.211.231.218",
            "AssociationId": "eipassoc-0f96bdca17EXAMPLE",
            "IsPrimary": false,
            "Status": "disassociating"
        }
    ]
}
```

Per ulteriori informazioni, consulta Gateway NAT nella Guida per l'utente di Amazon VPC.

- Per i dettagli sull'API, consulta DisassociateNatGatewayAddress AWS CLICommand Reference.

`disassociate-route-table`

Il seguente esempio di codice mostra come utilizzare`disassociate-route-table`.

AWS CLI

Come dissociare una tabella di routing

Questo esempio dissocia la tabella di routing specificata dalla sottorete specificata. Se il comando va a buon fine, non viene restituito alcun output.

Comando:

```
aws ec2 disassociate-route-table --association-id rtbassoc-781d0d1a
```

- Per i dettagli sull'API, consulta DisassociateRouteTable AWS CLICommand Reference.

`disassociate-security-group-vpc`

Il seguente esempio di codice mostra come utilizzare`disassociate-security-group-vpc`.

AWS CLI

Come dissociare un gruppo di sicurezza da un VPC

L'esempio `disassociate-security-group-vpc` seguente dissocia il gruppo di sicurezza specificato dal VPC specificato.

```
aws ec2 disassociate-security-group-vpc \
    --group-id sg-04dbb43907d3f8a78 \
    --vpc-id vpc-0bf4c2739bc05a694
```

Output:

```
{
    "State": "disassociating"
}
```

Per ulteriori informazioni, consulta Associare gruppi di sicurezza a più gruppi VPCs nella Amazon VPC User Guide.

- Per i dettagli sull'API, consulta DisassociateSecurityGroupVpc AWS CLICommand Reference.

`disassociate-subnet-cidr-block`

Il seguente esempio di codice mostra come utilizzare`disassociate-subnet-cidr-block`.

AWS CLI

Per dissociare un blocco IPv6 CIDR da una sottorete

Questo esempio dissocia un blocco IPv6 CIDR da una sottorete utilizzando l'ID di associazione per il blocco CIDR.

Comando:

```
aws ec2 disassociate-subnet-cidr-block --association-id subnet-cidr-assoc-3aa54053
```

Output:

```
{
  "SubnetId": "subnet-5f46ec3b",
  "Ipv6CidrBlockAssociation": {
      "Ipv6CidrBlock": "2001:db8:1234:1a00::/64",
      "AssociationId": "subnet-cidr-assoc-3aa54053",
      "Ipv6CidrBlockState": {
          "State": "disassociating"
      }
  }
}
```

- Per i dettagli sull'API, consulta Command Reference. DisassociateSubnetCidrBlockAWS CLI

`disassociate-transit-gateway-multicast-domain`

Il seguente esempio di codice mostra come utilizzare`disassociate-transit-gateway-multicast-domain`.

AWS CLI

Come dissociare sottoreti da un dominio multicast (trasmissione uno a molti)

L'esempio `disassociate-transit-gateway-multicast-domain` seguente dissocia una sottorete dal dominio multicast (trasmissione uno a molti) specificato.

```
aws ec2 disassociate-transit-gateway-multicast-domain \
    --transit-gateway-attachment-id tgw-attach-070e571cd1EXAMPLE \
    --subnet-id subnet-000de86e3bEXAMPLE \
    --transit-gateway-multicast-domain-id tgw-mcast-domain-0c4905cef7EXAMPLE
```

Output:

```
{
    "Associations": {
        "TransitGatewayMulticastDomainId": "tgw-mcast-domain-0c4905cef7EXAMPLE",
        "TransitGatewayAttachmentId": "tgw-attach-070e571cd1EXAMPLE",
        "ResourceId": "vpc-7EXAMPLE",
        "ResourceType": "vpc",
        "Subnets": [
```

```
            {
                "SubnetId": "subnet-000de86e3bEXAMPLE",
                "State": "disassociating"
            }
        ]
    }
}
```

Per ulteriori informazioni, consulta Domini multicast (trasmissione uno a molti) nella Guida di Transit Gateway.

- Per i dettagli sull'API, consulta DisassociateTransitGatewayMulticastDomain AWS CLICommand Reference.

`disassociate-transit-gateway-route-table`

Il seguente esempio di codice mostra come utilizzare`disassociate-transit-gateway-route-table`.

AWS CLI

Come dissociare una tabella di routing del gateway di transito da un allegato delle risorse

L'esempio `disassociate-transit-gateway-route-table` seguente dissocia l'allegato specificato dalla tabella di routing del gateway di transito.

```
aws ec2 disassociate-transit-gateway-route-table \
    --transit-gateway-route-table-id tgw-rtb-002573ed1eEXAMPLE \
    --transit-gateway-attachment-id tgw-attach-08e0bc912cEXAMPLE
```

Output:

```
{
    "Association": {
        "TransitGatewayRouteTableId": "tgw-rtb-002573ed1eEXAMPLE",
        "TransitGatewayAttachmentId": "tgw-attach-08e0bc912cEXAMPLE",
        "ResourceId": "11460968-4ac1-4fd3-bdb2-00599EXAMPLE",
        "ResourceType": "direct-connect-gateway",
        "State": "disassociating"
    }
}
```

Per ulteriori informazioni, consulta Tabelle di routing del gateway di transito nella Guida di Transit Gateway.

- Per i dettagli sull'API, consulta DisassociateTransitGatewayRouteTable AWS CLICommand Reference.

`disassociate-vpc-cidr-block`

Il seguente esempio di codice mostra come utilizzare`disassociate-vpc-cidr-block`.

AWS CLI

Per dissociare un blocco IPv6 CIDR da un VPC

Questo esempio dissocia un blocco IPv6 CIDR da un VPC utilizzando l'ID di associazione per il blocco CIDR.

Comando:

```
aws ec2 disassociate-vpc-cidr-block --association-id vpc-cidr-assoc-eca54085
```

Output:

```
{
  "Ipv6CidrBlockAssociation": {
      "Ipv6CidrBlock": "2001:db8:1234:1a00::/56",
      "AssociationId": "vpc-cidr-assoc-eca54085",
      "Ipv6CidrBlockState": {
          "State": "disassociating"
      }
  },
  "VpcId": "vpc-a034d6c4"
}
```

Per dissociare un blocco IPv4 CIDR da un VPC

Questo esempio dissocia un blocco IPv4 CIDR da un VPC.

Comando:

```
aws ec2 disassociate-vpc-cidr-block --association-id vpc-cidr-assoc-0287ac6b
```

Output:

```
{
  "CidrBlockAssociation": {
      "AssociationId": "vpc-cidr-assoc-0287ac6b",
      "CidrBlock": "172.18.0.0/16",
      "CidrBlockState": {
          "State": "disassociating"
      }
  },
  "VpcId": "vpc-27621243"
}
```

- Per i dettagli sull'API, consulta DisassociateVpcCidrBlockCommand Reference.AWS CLI

`enable-address-transfer`

Il seguente esempio di codice mostra come utilizzare`enable-address-transfer`.

AWS CLI

Come abilitare il trasferimento degli indirizzi IP elastici

L'esempio `enable-address-transfer` seguente abilita il trasferimento degli indirizzi IP elastici per l'indirizzo IP elastico specificato nell'account specificato.

```
aws ec2 enable-address-transfer \
    --allocation-id eipalloc-09ad461b0d03f6aaf \
    --transfer-account-id 123456789012
```

Output:

```
{
     "AddressTransfer": {
        "PublicIp": "100.21.184.216",
        "AllocationId": "eipalloc-09ad461b0d03f6aaf",
        "TransferAccountId": "123456789012",
        "TransferOfferExpirationTimestamp": "2023-02-22T20:51:01.000Z",
        "AddressTransferStatus": "pending"
    }
}
```

Per ulteriori informazioni, consulta Trasferimento degli indirizzi IP elastici nella Guida per l'utente di Amazon VPC.

- Per i dettagli sull'API, consulta EnableAddressTransfer AWS CLICommand Reference.

`enable-aws-network-performance-metric-subscription`

Il seguente esempio di codice mostra come utilizzare`enable-aws-network-performance-metric-subscription`.

AWS CLI

Come abilitare una sottoscrizioni di parametri di prestazioni

L'esempio `enable-aws-network-performance-metric-subscription` seguente abilita il monitoraggio della latenza di rete aggregata tra le Regioni di origine e di destinazione specificate.

```
aws ec2 enable-aws-network-performance-metric-subscription \
    --source us-east-1 \
    --destination eu-west-1 \
    --metric aggregate-latency \
    --statistic p50
```

Output:

```
{
    "Output": true
}
```

Per ulteriori informazioni, consulta Gestione delle sottoscrizioni nella Guida per l'utente di Infrastructure Performance.

- Per i dettagli sull'API, consulta EnableAwsNetworkPerformanceMetricSubscription AWS CLICommand Reference.

`enable-ebs-encryption-by-default`

Il seguente esempio di codice mostra come utilizzare`enable-ebs-encryption-by-default`.

AWS CLI

Come abilitare la crittografia EBS per impostazione predefinita

L'`enable-ebs-encryption-by-default`esempio seguente abilita la crittografia EBS per impostazione predefinita per il tuo AWS account nella regione corrente.

```
aws ec2 enable-ebs-encryption-by-default
```

Output:

```
{
    "EbsEncryptionByDefault": true
}
```

- Per i dettagli sull'API, consulta EnableEbsEncryptionByDefault AWS CLICommand Reference.

`enable-fast-launch`

Il seguente esempio di codice mostra come utilizzare`enable-fast-launch`.

AWS CLI

Come attivare l'avvio rapido di un'immagine

L'esempio `enable-fast-launch` seguente configura l'AMI specificata per l'avvio rapido e imposta il numero massimo di istanze parallele da avviare su 6. Il tipo di risorsa da utilizzare per il pre-provisioning dell'AMI è impostato su `snapshot`, che è anche il valore predefinito.

```
aws ec2 enable-fast-launch \
    --image-id ami-01234567890abcedf \
    --max-parallel-launches 6 \
    --resource-type snapshot
```

Output:

```
{
    "ImageId": "ami-01234567890abcedf",
    "ResourceType": "snapshot",
    "SnapshotConfiguration": {
        "TargetResourceCount": 10
    },
    "LaunchTemplate": {},
    "MaxParallelLaunches": 6,
    "OwnerId": "0123456789123",
```

```
    "State": "enabling",
    "StateTransitionReason": "Client.UserInitiated",
    "StateTransitionTime": "2022-01-27T22:16:03.199000+00:00"
}
```

Per ulteriori informazioni, consulta Configurare le impostazioni di EC2 Fast Launch per la tua AMI Windows nella Amazon EC2 User Guide.

- Per i dettagli sull'API, consulta EnableFastLaunch AWS CLICommand Reference.

`enable-fast-snapshot-restores`

Il seguente esempio di codice mostra come utilizzare`enable-fast-snapshot-restores`.

AWS CLI

Come abilitare il ripristino rapido degli snapshot

L'esempio `enable-fast-snapshot-restores` seguente abilita il ripristino rapido degli snapshot per gli snapshot specificati nelle zone di disponibilità specificate.

```
aws ec2 enable-fast-snapshot-restores \
    --availability-zones us-east-2a us-east-2b \
    --source-snapshot-ids snap-1234567890abcdef0
```

Output:

```
{
    "Successful": [
        {
            "SnapshotId": "snap-1234567890abcdef0"
            "AvailabilityZone": "us-east-2a",
            "State": "enabling",
            "StateTransitionReason": "Client.UserInitiated",
            "OwnerId": "123456789012",
            "EnablingTime": "2020-01-25T23:57:49.602Z"
        },
        {
            "SnapshotId": "snap-1234567890abcdef0"
            "AvailabilityZone": "us-east-2b",
            "State": "enabling",
            "StateTransitionReason": "Client.UserInitiated",
```

```
            "OwnerId": "123456789012",
            "EnablingTime": "2020-01-25T23:57:49.596Z"
        }
    ],
    "Unsuccessful": []
}
```

- Per i dettagli sull'API, consulta EnableFastSnapshotRestores AWS CLICommand Reference.

`enable-image-block-public-access`

Il seguente esempio di codice mostra come utilizzare`enable-image-block-public-access`.

AWS CLI

Per abilitare il blocco dell'accesso pubblico AMIs nella regione specificata

L'`enable-image-block-public-access`esempio seguente abilita il blocco dell'accesso pubblico AMIs a livello di account nella regione specificata.

```
aws ec2 enable-image-block-public-access \
    --region us-east-1 \
    --image-block-public-access-state block-new-sharing
```

Output:

```
{
    "ImageBlockPublicAccessState": "block-new-sharing"
}
```

Per ulteriori informazioni, consulta Bloccare l'accesso pubblico al tuo AMIs nella Amazon EC2 User Guide.

- Per i dettagli sull'API, consulta EnableImageBlockPublicAccess AWS CLICommand Reference.

`enable-image-deprecation`

Il seguente esempio di codice mostra come utilizzare`enable-image-deprecation`.

AWS CLI

Come dichiarare un'AMI come obsoleta

L'esempio `enable-image-deprecation` seguente dichiara un'AMI come obsoleta in una data e un'ora specifiche. Se specifichi un valore in secondi, Amazon EC2 arrotonda i secondi al minuto più vicino. Per eseguire questa procedura, è necessario essere il proprietario dell'AMI.

```
aws ec2 enable-image-deprecation \
    --image-id ami-1234567890abcdef0 \
    --deprecate-at '2022-10-15T13:17:12.000Z'
```

Output:

```
{
    "RequestID": "59dbff89-35bd-4eac-99ed-be587EXAMPLE",
    "Return": "true"
}
```

Per ulteriori informazioni, consulta Deprecare un'AMI nella Amazon EC2 User Guide.

- Per i dettagli sull'API, consulta AWS CLI Command EnableImageDeprecationReference.

`enable-image-deregistration-protection`

Il seguente esempio di codice mostra come utilizzare`enable-image-deregistration-protection`.

AWS CLI

Come abilitare la protezione da annullamento della registrazione

L'esempio `enable-image-deregistration-protection` seguente abilita la protezione da annullamento della registrazione per l'immagine specificata.

```
aws ec2 enable-image-deregistration-protection \
    --image-id ami-0b1a928a144a74ec9
```

Output:

```
{
    "Return": "enabled-without-cooldown"
}
```

Per ulteriori informazioni, consulta Proteggere un EC2 AMI dall'annullamento della registrazione nella Amazon EC2 User Guide.

- Per i dettagli sull'API, consulta Command EnableImageDeregistrationProtectionReference AWS CLI .

`enable-image`

Il seguente esempio di codice mostra come utilizzare`enable-image`.

AWS CLI

Come abilitare un'AMI

L'esempio `enable-image` seguente abilita l'AMI specificata.

```
aws ec2 enable-image \
    --image-id ami-1234567890abcdef0
```

Output:

```
{
    "Return": "true"
}
```

Per ulteriori informazioni, consulta Disabilitare un'AMI nella Amazon EC2 User Guide.

- Per i dettagli sull'API, consulta EnableImage AWS CLICommand Reference.

`enable-ipam-organization-admin-account`

Il seguente esempio di codice mostra come utilizzare`enable-ipam-organization-admin-account`.

AWS CLI

Integrarsi con AWS Organizations e delegare un account membro come account IPAM

L'`enable-ipam-organization-admin-account`esempio seguente integra IPAM con AWS Organizations e delega un account membro come account IPAM.

```
aws ec2 enable-ipam-organization-admin-account \
    --delegated-admin-account-id 320805250157
```

Output:

```
{
    "Success": true
}
```

Per ulteriori informazioni, consulta Integrate IPAM with AWS Organizations nella Amazon VPC IPAM User Guide.

- Per i dettagli sull'API, consulta Command EnableIpamOrganizationAdminAccountReference AWS CLI .

`enable-reachability-analyzer-organization-sharing`

Il seguente esempio di codice mostra come utilizzare`enable-reachability-analyzer-organization-sharing`.

AWS CLI

Come abilitare l'accesso attendibile per Sistema di analisi della reperibilità

L'esempio `enable-reachability-analyzer-organization-sharing` seguente abilita l'accesso attendibile per Sistema di analisi della reperibilità.

```
aws ec2 enable-reachability-analyzer-organization-sharing
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Analisi tra account nella Guida per l'utente di Sistema di analisi della reperibilità.

- Per i dettagli sull'API, consulta EnableReachabilityAnalyzerOrganizationSharing AWS CLICommand Reference.

`enable-serial-console-access`

Il seguente esempio di codice mostra come utilizzare`enable-serial-console-access`.

AWS CLI

Come abilitare l'accesso alla console seriale per l'account in uso

L'esempio `enable-serial-console-access` seguente abilita l'accesso account alla console seriale.

```
aws ec2 enable-serial-console-access
```

Output:

```
{
    "SerialConsoleAccessEnabled": true
}
```

Per ulteriori informazioni, consulta EC2 Serial Console nella Amazon EC2 User Guide.

- Per i dettagli sull'API, consulta EnableSerialConsoleAccess AWS CLICommand Reference.

`enable-snapshot-block-public-access`

Il seguente esempio di codice mostra come utilizzare`enable-snapshot-block-public-access`.

AWS CLI

Come abilitare il blocco dell'accesso pubblico per gli snapshot

L'esempio `enable-snapshot-block-public-access` seguente blocca tutte le condivisioni pubbliche degli snapshot.

```
aws ec2 enable-snapshot-block-public-access \
    --state block-all-sharing
```

Output:

```
{
    "State": "block-all-sharing"
}
```

Per ulteriori informazioni, consulta Blocco dell'accesso pubblico per gli snapshot nella Guida per l'utente di Amazon EBS.

- Per i dettagli sull'API, consulta EnableSnapshotBlockPublicAccess AWS CLICommand Reference.

`enable-transit-gateway-route-table-propagation`

Il seguente esempio di codice mostra come utilizzare`enable-transit-gateway-route-table-propagation`.

AWS CLI

Come abilitare un collegamento del gateway di transito alla VPN per propagare le route alla tabella di routing di propagazione specificata

L'esempio `enable-transit-gateway-route-table-propagation` seguente abilita l'allegato specificato per propagare le route alla tabella di routing di propagazione specificata.

```
aws ec2 enable-transit-gateway-route-table-propagation \
    --transit-gateway-route-table-id tgw-rtb-0a823edbdeEXAMPLE \
    --transit-gateway-attachment-id tgw-attach-09b52ccdb5EXAMPLE
```

Output:

```
{
    "Propagation": {
        "TransitGatewayAttachmentId": "tgw-attach-09b52ccdb5EXAMPLE",
        "ResourceId": "vpc-4d7de228",
        "ResourceType": "vpc",
        "TransitGatewayRouteTableId": "tgw-rtb-0a823edbdeEXAMPLE",
        "State": "disabled"
    }
}
```

Per ulteriori informazioni, consulta Tabelle di routing del gateway di transito nella Guida di Transit Gateway.

- Per i dettagli sull'API, consulta EnableTransitGatewayRouteTablePropagation AWS CLICommand Reference.

`enable-vgw-route-propagation`

Il seguente esempio di codice mostra come utilizzare`enable-vgw-route-propagation`.

AWS CLI

Come abilitare la propagazione di route

Questo esempio abilita il gateway privato virtuale specificato per propagare route statiche alla tabella di routing specificata. Se il comando va a buon fine, non viene restituito alcun output.

Comando:

```
aws ec2 enable-vgw-route-propagation --route-table-id rtb-22574640 --gateway-
id vgw-9a4cacf3
```

- Per i dettagli sull'API, consulta EnableVgwRoutePropagation AWS CLICommand Reference.

`enable-volume-io`

Il seguente esempio di codice mostra come utilizzare`enable-volume-io`.

AWS CLI

I/O Per abilitare un volume

Questo esempio abilita I/O on volume`vol-1234567890abcdef0`.

Comando:

```
aws ec2 enable-volume-io --volume-id vol-1234567890abcdef0
```

Output:

```
{
  "Return": true
}
```

- Per i dettagli sull'API, consulta EnableVolumeIo AWS CLICommand Reference.

`enable-vpc-classic-link-dns-support`

Il seguente esempio di codice mostra come utilizzare`enable-vpc-classic-link-dns-support`.

AWS CLI

Per abilitare il supporto ClassicLink DNS per un VPC

Questo esempio abilita il supporto ClassicLink DNS per. `vpc-88888888`

Comando:

```
aws ec2 enable-vpc-classic-link-dns-support --vpc-id vpc-88888888
```

Output:

```
{
  "Return": true
}
```

- Per i dettagli sull'API, consulta EnableVpcClassicLinkDnsSupport AWS CLICommand Reference.

`enable-vpc-classic-link`

Il seguente esempio di codice mostra come utilizzare`enable-vpc-classic-link`.

AWS CLI

Per abilitare un VPC per ClassicLink

Questo esempio abilita vpc-8888888 per. ClassicLink

Comando:

```
aws ec2 enable-vpc-classic-link --vpc-id vpc-88888888
```

Output:

```
{
  "Return": true
}
```

- Per i dettagli sull'API, consulta Command EnableVpcClassicLinkReference AWS CLI .

`export-client-vpn-client-certificate-revocation-list`

Il seguente esempio di codice mostra come utilizzare`export-client-vpn-client-certificate-revocation-list`.

AWS CLI

Come esportare un elenco di revoche di certificati client

L'esempio `export-client-vpn-client-certificate-revocation-list` seguente esporta l'elenco di revoche di certificati client per l'endpoint VPN Client specificato. In questo esempio, l'output viene restituito in formato di testo per semplificarne la lettura.

```
aws ec2 export-client-vpn-client-certificate-revocation-list \
    --client-vpn-endpoint-id cvpn-endpoint-123456789123abcde \
    --output text
```

Output:

```
-----BEGIN X509 CRL-----
MIICiTCCAfICCQD6m7oRw0uXOjANBgkqhkiG9w0BAQUFADCBiDELMAkGA1UEBhMC
VVMxCzAJBgNVBAgTAldBMRAwDgYDVQQHEwdTZWF0dGxlMQ8wDQYDVQQKEwZBbWF6
b24xFDASBgNVBAsTC0lBTSBDb25zb2xlMRIwEAYDVQQDEwlUZXN0Q2lsYWMxHzAd
BgkqhkiG9w0BCQEWEG5vb25lQGFtYXpvbi5jb20wHhcNMTEwNDI1MjA0NTIxWhcN
MTIwNDI0MjA0NTIxWjCBiDELMAkGA1UEBhMCVVMxCzAJBgNVBAgTAldBMRAwDgYD
VQQHEwdTZWF0dGxlMQ8wDQYDVQQKEwZBbWF6b24xFDASBgNVBAsTC0lBTSBDb25z
b2xlMRIwEAYDVQQDEwlUZXN0Q2lsYWMxHzAdBgkqhkiG9w0BCQEWEG5vb25lQGFt
YXpvbi5jb20wgZ8wDQYJKoZIhvcNAQEBBQADgY0AMIGJAoGBAMaK0dn+a4GmWIWJ
21uUSfwfEvySWtC2XADZ4nB+BLYgVIk60CpiwsZ3G93vUEIO3IyNoH/f0wYK8m9T
rDHudUZg3qX4waLG5M43q7Wgc/MbQITxOUSQv7c7ugFFDzQGBzZswY6786m86gpE
Ibb3OhjZnzcvQAaRHhdlQWIMm2nrAgMBAAEwDQYJKoZIhvcNAQEFBQADgYEAtCu4
nUhVVxYUntneD9+h8Mg9q6q+auNKyExzyLwaxlAoo7TJHidbtS4J5iNmZgXL0Fkb
FFBjvSfpJIlJ00zbhNYS5f6GuoEDmFJl0ZxBHjJnyp378OD8uTs7fLvjx79LjSTb
NYiytVbZPQUQ5Yaxu2jXnimvw3rrszlaEXAMPLE=
-----END X509 CRL-----
STATUS      pending
```

Per ulteriori informazioni, consulta Elenchi di revoche di certificati client nella Guida per l'amministratore di VPN Client di AWS .

- Per i dettagli sull'API, consulta ExportClientVpnClientCertificateRevocationList AWS CLICommand Reference.

`export-client-vpn-client-configuration`

Il seguente esempio di codice mostra come utilizzare`export-client-vpn-client-configuration`.

AWS CLI

Come esportare la configurazione del client

L'esempio `export-client-vpn-client-configuration` seguente esporta la configurazione del client per l'endpoint VPN Client specificato. In questo esempio, l'output viene restituito in formato di testo per semplificarne la lettura.

```
aws ec2 export-client-vpn-client-configuration \
    --client-vpn-endpoint-id cvpn-endpoint-123456789123abcde \
    --output text
```

Output:

```
client
dev tun
proto udp
remote cvpn-endpoint-123456789123abcde.prod.clientvpn.ap-south-1.amazonaws.com 443
remote-random-hostname
resolv-retry infinite
nobind
persist-key
persist-tun
remote-cert-tls server
cipher AES-256-GCM
verb 3
<ca>
-----BEGIN CERTIFICATE-----
MIICiTCCAfICCQD6m7oRw0uXOjANBgkqhkiG9w0BAQUFADCBiDELMAkGA1UEBhMC
VVMxCzAJBgNVBAgTAldBMRAwDgYDVQQHEwdTZWF0dGxlMQ8wDQYDVQQKEwZBbWF6
b24xFDASBgNVBAsTC0lBTSBDb25zb2xlMRIwEAYDVQQDEwlUZXN0Q2lsYWMxHzAd
BgkqhkiG9w0BCQEWEG5vb25lQGFtYXpvbi5jb20wHhcNMTEwNDI1MjA0NTIxWhcN
MTIwNDI0MjA0NTIxWjCBiDELMAkGA1UEBhMCVVMxCzAJBgNVBAgTAldBMRAwDgYD
VQQHEwdTZWF0dGxlMQ8wDQYDVQQKEwZBbWF6b24xFDASBgNVBAsTC0lBTSBDb25z
b2xlMRIwEAYDVQQDEwlUZXN0Q2lsYWMxHzAdBgkqhkiG9w0BCQEWEG5vb25lQGFt
YXpvbi5jb20wgZ8wDQYJKoZIhvcNAQEBBQADgY0AMIGJAoGBAMaK0dn+a4GmWIWJ
21uUSfwfEvySWtC2XADZ4nB+BLYgVIk60CpiwsZ3G93vUEIO3IyNoH/f0wYK8m9T
rDHudUZg3qX4waLG5M43q7Wgc/MbQITxOUSQv7c7ugFFDzQGBzZswY6786m86gpE
```

```
Ibb3OhjZnzcvQAaRHhdlQWIMm2nrAgMBAAEwDQYJKoZIhvcNAQEFBQADgYEAtCu4
nUhVVxYUntneD9+h8Mg9q6q+auNKyExzyLwaxlAoo7TJHidbtS4J5iNmZgXL0Fkb
FFBjvSfpJIlJ00zbhNYS5f6GuoEDmFJl0ZxBHjJnyp378OD8uTs7fLvjx79LjSTb
NYiytVbZPQUQ5Yaxu2jXnimvw3rrszlaEXAMPLE=
-----END CERTIFICATE-----
</ca>
reneg-sec 0
```

Per ulteriori informazioni, consulta Esportazione del file di configurazione dell'endpoint VPN client nella Guida per l'amministratore di VPN Client di AWS .

- Per i dettagli sull'API, consulta ExportClientVpnClientConfiguration AWS CLICommand Reference.

`export-image`

Il seguente esempio di codice mostra come utilizzare`export-image`.

AWS CLI

Come esportare una VM da un'AMI

L'esempio `export-image` seguente esporta l'AMI specificata nel bucket specificato nel formato specificato.

```
aws ec2 export-image \
    --image-id ami-1234567890abcdef0 \
    --disk-image-format VMDK \
    --s3-export-location S3Bucket=my-export-bucket,S3Prefix=exports/
```

Output:

```
{
    "DiskImageFormat": "vmdk",
    "ExportImageTaskId": "export-ami-1234567890abcdef0"
    "ImageId": "ami-1234567890abcdef0",
    "RoleName": "vmimport",
    "Progress": "0",
    "S3ExportLocation": {
        "S3Bucket": "my-export-bucket",
        "S3Prefix": "exports/"
    },
    "Status": "active",
```

```
    "StatusMessage": "validating"
}
```

- Per i dettagli sull'API, consulta ExportImage AWS CLICommand Reference.

`get-associated-ipv6-pool-cidrs`

Il seguente esempio di codice mostra come utilizzare`get-associated-ipv6-pool-cidrs`.

AWS CLI

Per ottenere le associazioni per un pool di IPv6 indirizzi

L'`get-associated-ipv6-pool-cidrs`esempio seguente ottiene le associazioni per il pool di IPv6 indirizzi specificato.

```
aws ec2 get-associated-ipv6-pool-cidrs \
    --pool-id ipv6pool-ec2-012345abc12345abc
```

Output:

```
{
    "Ipv6CidrAssociations": [
        {
            "Ipv6Cidr": "2001:db8:1234:1a00::/56",
            "AssociatedResource": "vpc-111111222222333ab"
        }
    ]
}
```

- Per i dettagli sull'API, vedere GetAssociatedIpv6 PoolCidrs in AWS CLI Command Reference.

`get-aws-network-performance-data`

Il seguente esempio di codice mostra come utilizzare`get-aws-network-performance-data`.

AWS CLI

Come ottenere dati sulle prestazioni di rete

L'esempio `get-aws-network-performance-data` seguente recupera i dati sulle prestazioni di rete tra le Regioni specificate nel periodo di tempo specificato.

```
aws ec2 get-aws-network-performance-data \
    --start-time 2022-10-26T12:00:00.000Z \
    --end-time 2022-10-26T12:30:00.000Z \
    --data-queries Id=my-query,Source=us-east-1,Destination=eu-
west-1,Metric=aggregate-latency,Statistic=p50,Period=five-minutes
```

Output:


```
{
    "DataResponses": [
        {
            "Id": "my-query",
            "Source": "us-east-1",
            "Destination": "eu-west-1",
            "Metric": "aggregate-latency",
            "Statistic": "p50",
            "Period": "five-minutes",
            "MetricPoints": [
                {
                    "StartDate": "2022-10-26T12:00:00+00:00",
                    "EndDate": "2022-10-26T12:05:00+00:00",
                    "Value": 62.44349,
                    "Status": "OK"
                },
                {
                    "StartDate": "2022-10-26T12:05:00+00:00",
                    "EndDate": "2022-10-26T12:10:00+00:00",
                    "Value": 62.483498,
                    "Status": "OK"
                },
                {
                    "StartDate": "2022-10-26T12:10:00+00:00",
                    "EndDate": "2022-10-26T12:15:00+00:00",
                    "Value": 62.51248,
                    "Status": "OK"
                },
                {
                    "StartDate": "2022-10-26T12:15:00+00:00",
                    "EndDate": "2022-10-26T12:20:00+00:00",
                    "Value": 62.635475,
                    "Status": "OK"
                },
                {
```

```
                    "StartDate": "2022-10-26T12:20:00+00:00",
                    "EndDate": "2022-10-26T12:25:00+00:00",
                    "Value": 62.733974,
                    "Status": "OK"
                },
                {
                    "StartDate": "2022-10-26T12:25:00+00:00",
                    "EndDate": "2022-10-26T12:30:00+00:00",
                    "Value": 62.773975,
                    "Status": "OK"
                },
                {
                    "StartDate": "2022-10-26T12:30:00+00:00",
                    "EndDate": "2022-10-26T12:35:00+00:00",
                    "Value": 62.75349,
                    "Status": "OK"
                }
            ]
        }
    ]
}
```

Per ulteriori informazioni, consulta Monitoraggio delle prestazioni di rete nella Guida per l'utente di Infrastructure Performance.

- Per i dettagli sull'API, consulta GetAwsNetworkPerformanceData AWS CLICommand Reference.

`get-capacity-reservation-usage`

Il seguente esempio di codice mostra come utilizzare`get-capacity-reservation-usage`.

AWS CLI

Per visualizzare l'utilizzo della prenotazione della capacità tra AWS gli account

L'esempio `get-capacity-reservation-usage` seguente visualizza le informazioni di utilizzo per la prenotazione della capacità specificata.

```
aws ec2 get-capacity-reservation-usage \
    --capacity-reservation-id cr-1234abcd56EXAMPLE
```

Output:

```
{
    "CapacityReservationId": "cr-1234abcd56EXAMPLE ",
    "InstanceUsages": [
        {
            "UsedInstanceCount": 1,
            "AccountId": "123456789012"
        }
    ],
    "AvailableInstanceCount": 4,
    "TotalInstanceCount": 5,
    "State": "active",
    "InstanceType": "t2.medium"
}
```

Per ulteriori informazioni, consulta Shared Capacity Reservations nella Amazon EC2 User Guide.

- Per i dettagli sull'API, consulta GetCapacityReservationUsage AWS CLICommand Reference.

`get-coip-pool-usage`

Il seguente esempio di codice mostra come utilizzare`get-coip-pool-usage`.

AWS CLI

Come ottenere l'utilizzo del pool di indirizzi IP di proprietà del cliente

L'esempio `get-coip-pool-usage` seguente recupera i dettagli di utilizzo del pool di indirizzi IP specificato di proprietà del cliente.

```
aws ec2 get-coip-pool-usage \
    --pool-id ipv4pool-coip-123a45678bEXAMPLE
```

Output:

```
{
    "CoipPoolId": "ipv4pool-coip-123a45678bEXAMPLE",
    "CoipAddressUsages": [
        {
            "CoIp": "0.0.0.0"
        },
        {
```

```
            "AllocationId": "eipalloc-123ab45c6dEXAMPLE",
            "AwsAccountId": "123456789012",
            "CoIp": "0.0.0.0"
        },
        {
            "AllocationId": "eipalloc-123ab45c6dEXAMPLE",
            "AwsAccountId": "123456789111",
            "CoIp": "0.0.0.0"
        }
    ],
    "LocalGatewayRouteTableId": "lgw-rtb-059615ef7dEXAMPLE"
}
```

Per ulteriori informazioni, consulta Indirizzi IP di proprietà del cliente nella Guida per l'utente di AWS Outposts per i rack Outposts.

- Per i dettagli sull'API, consulta GetCoipPoolUsage AWS CLICommand Reference.

`get-console-output`

Il seguente esempio di codice mostra come utilizzare`get-console-output`.

AWS CLI

Esempio 1: come ottenere l'output della console

L'esempio `get-console-output` seguente ottiene l'output della console per l'istanza Linux specificata.

```
aws ec2 get-console-output \
    --instance-id i-1234567890abcdef0
```

Output:

```
{
    "InstanceId": "i-1234567890abcdef0",
    "Timestamp": "2013-07-25T21:23:53.000Z",
    "Output": "..."
}
```

Per ulteriori informazioni, consulta l'output della console di istanza nella Amazon EC2 User Guide.

Esempio 2: come ottenere l'output più recente della console

L'esempio `get-console-output` seguente ottiene l'output più recente della console per l'istanza Linux specificata.

```
aws ec2 get-console-output \
    --instance-id i-1234567890abcdef0 \
    --latest \
    --output text
```

Output:

```
i-1234567890abcdef0 [    0.000000] Command line: root=LABEL=/ console=tty1
 console=ttyS0 selinux=0 nvme_core.io_timeout=4294967295
[    0.000000] x86/fpu: Supporting XSAVE feature 0x001: 'x87 floating point
 registers'
[    0.000000] x86/fpu: Supporting XSAVE feature 0x002: 'SSE registers'
[    0.000000] x86/fpu: Supporting XSAVE feature 0x004: 'AVX registers'
...
Cloud-init v. 0.7.6 finished at Wed, 09 May 2018 19:01:13 +0000. Datasource
 DataSourceEc2.  Up 21.50 seconds
Amazon Linux AMI release 2018.03
Kernel 4.14.26-46.32.amzn1.x
```

Per ulteriori informazioni, consulta l'output della console di istanza nella Amazon EC2 User Guide.

- Per i dettagli sull'API, consulta GetConsoleOutput AWS CLICommand Reference.

`get-console-screenshot`

Il seguente esempio di codice mostra come utilizzare`get-console-screenshot`.

AWS CLI

Come recuperare uno screenshot di un'istanza in esecuzione

L'esempio `get-console-screenshot` seguente recupera uno screenshot dell'istanza specificata in formato .jpg. Lo screenshot viene restituito come stringa codificata Base64.

```
aws ec2 get-console-screenshot \
    --instance-id i-1234567890abcdef0
```

Output:

```
{
    "ImageData": "997987/8kgj49ikjhewkwwe0008084EXAMPLE",
    "InstanceId": "i-1234567890abcdef0"
}
```

- Per i dettagli sull'API, consulta GetConsoleScreenshot AWS CLICommand Reference.

`get-default-credit-specification`

Il seguente esempio di codice mostra come utilizzare`get-default-credit-specification`.

AWS CLI

Come descrivere l'opzione di credito predefinita

L'esempio `get-default-credit-specification` seguente descrive l'opzione di credito predefinita per le istanze T2.

```
aws ec2 get-default-credit-specification \
    --instance-family t2
```

Output:

```
{
    "InstanceFamilyCreditSpecification": {
        "InstanceFamily": "t2",
        "CpuCredits": "standard"
    }
}
```

- Per i dettagli sull'API, consulta GetDefaultCreditSpecification AWS CLICommand Reference.

`get-ebs-default-kms-key-id`

Il seguente esempio di codice mostra come utilizzare`get-ebs-default-kms-key-id`.

AWS CLI

Come descrivere la chiave gestita dal cliente (CMK) predefinita per la crittografia EBS

L'`get-ebs-default-kms-key-id`esempio seguente descrive la crittografia CMK for EBS predefinita per il tuo AWS account.

```
aws ec2 get-ebs-default-kms-key-id
```

L'output mostra la CMK predefinita per la crittografia EBS, che è una CMK AWS gestita con l'alias. `alias/aws/ebs`

```
{
    "KmsKeyId": "alias/aws/ebs"
}
```

L'output seguente mostra una chiave gestita dal cliente (CMK) personalizzata per la crittografia EBS.

```
{
    "KmsKeyId": "arn:aws:kms:us-
west-2:123456789012:key/0ea3fef3-80a7-4778-9d8c-1c0c6EXAMPLE"
}
```

- Per i dettagli sull'API, consulta GetEbsDefaultKmsKeyIdCommand Reference.AWS CLI

`get-ebs-encryption-by-default`

Il seguente esempio di codice mostra come utilizzare`get-ebs-encryption-by-default`.

AWS CLI

Come descrivere se la crittografia EBS è abilitata per impostazione predefinita

L'`get-ebs-encryption-by-default`esempio seguente indica se la crittografia EBS per impostazione predefinita è abilitata per l' AWS account nell'area corrente.

```
aws ec2 get-ebs-encryption-by-default
```

L'output seguente indica che la crittografia EBS è disabilitata per impostazione predefinita.

```
{
    "EbsEncryptionByDefault": false
}
```

L'output seguente indica che la crittografia EBS è abilitata per impostazione predefinita.

```
{
    "EbsEncryptionByDefault": true
}
```

- Per i dettagli sull'API, consulta GetEbsEncryptionByDefault AWS CLICommand Reference.

`get-flow-logs-integration-template`

Il seguente esempio di codice mostra come utilizzare`get-flow-logs-integration-template`.

AWS CLI

Per creare un CloudFormation modello per automatizzare l'integrazione dei log di flusso VPC con Amazon Athena

`get-flow-logs-integration-template`Gli esempi seguenti creano un CloudFormation modello per automatizzare l'integrazione dei log di flusso VPC con Amazon Athena.

Linux:

```
aws ec2 get-flow-logs-integration-template \
    --flow-log-id fl-1234567890abcdef0 \
    --config-delivery-s3-destination-arn arn:aws:s3:::amzn-s3-demo-bucket \
    --integrate-services
 AthenaIntegrations='[{IntegrationResultS3DestinationArn=arn:aws:s3:::amzn-s3-demo-
bucket,PartitionLoadFrequency=none,PartitionStartDate=2021-07-21T00:40:00,PartitionEndDate=2
{IntegrationResultS3DestinationArn=arn:aws:s3:::amzn-s3-demo-
bucket,PartitionLoadFrequency=none,PartitionStartDate=2021-07-21T00:40:00,PartitionEndDate=2
```

Windows:

```
aws ec2 get-flow-logs-integration-template ^
    --flow-log-id fl-1234567890abcdef0 ^
    --config-delivery-s3-destination-arn arn:aws:s3:::amzn-s3-demo-bucket ^
    --integrate-
services AthenaIntegrations=[{IntegrationResultS3DestinationArn=arn:aws:s3:::amzn-
s3-demo-
bucket,PartitionLoadFrequency=none,PartitionStartDate=2021-07-21T00:40:00,PartitionEndDate=2
{IntegrationResultS3DestinationArn=arn:aws:s3:::amzn-s3-demo-
bucket,PartitionLoadFrequency=none,PartitionStartDate=2021-07-21T00:40:00,PartitionEndDate=2
```

Output:

```
{
    "Result": "https://amzn-s3-demo-bucket.s3.us-east-2.amazonaws.com/
VPCFlowLogsIntegrationTemplate_fl-1234567890abcdef0_Wed%20Jul
%2021%2000%3A57%3A56%20UTC%202021.yml"
}
```

Per informazioni sull'uso dei CloudFormation modelli, consulta Lavorare con i AWS CloudFormation modelli nella Guida per l'utente.AWS CloudFormation

Per informazioni sull'utilizzo di Amazon Athena e dei log di flusso, consulta Eseguire una query dei flussi di log tramite Amazon Athena nella Guida per l'utente di Amazon Virtual Private Cloud.

- Per i dettagli sull'API, consulta GetFlowLogsIntegrationTemplate AWS CLICommand Reference.

`get-groups-for-capacity-reservation`

Il seguente esempio di codice mostra come utilizzare`get-groups-for-capacity-reservation`.

AWS CLI

Come elencare i gruppi di risorse con una prenotazione della capacità

L'esempio `get-groups-for-capacity-reservation` seguente elenca i gruppi di risorse a cui è stata aggiunta la prenotazione della capacità specificata.

```
aws ec2 get-groups-for-capacity-reservation \
    --capacity-reservation-id cr-1234abcd56EXAMPLE
```

Output:

```
{
    "CapacityReservationsGroup": [
        {
            "GroupArn": "arn:aws:resource-groups:us-west-2:123456789012:group/my-
resource-group",
            "OwnerId": "123456789012"
        }
    ]
}
```

Per ulteriori informazioni, consulta la sezione Gruppi di prenotazione della capacità nella Amazon EC2 User Guide.

- Per i dettagli sull'API, consulta GetGroupsForCapacityReservation AWS CLICommand Reference.

`get-host-reservation-purchase-preview`

Il seguente esempio di codice mostra come utilizzare`get-host-reservation-purchase-preview`.

AWS CLI

Come visualizzare un'anteprima di acquisto per una prenotazione dell'host dedicato

Questo esempio fornisce un'anteprima dei costi di una specifica prenotazione dell'host dedicato specificato nell'account in uso.

Comando:

```
aws ec2 get-host-reservation-purchase-preview --offering-id hro-03f707bf363b6b324 --host-id-set h-013abcd2a00cbd123
```

Output:

```
{
  "TotalHourlyPrice": "1.499",
  "Purchase": [
      {
          "HourlyPrice": "1.499",
          "InstanceFamily": "m4",
          "PaymentOption": "NoUpfront",
          "HostIdSet": [
              "h-013abcd2a00cbd123"
          ],
          "UpfrontPrice": "0.000",
          "Duration": 31536000
      }
  ],
  "TotalUpfrontPrice": "0.000"
}
```

- Per i dettagli sull'API, consulta GetHostReservationPurchasePreview AWS CLICommand Reference.

`get-image-block-public-access-state`

Il seguente esempio di codice mostra come utilizzare`get-image-block-public-access-state`.

AWS CLI

Per ottenere lo stato di blocco dell'accesso pubblico AMIs nella regione specificata

L'`get-image-block-public-access-state`esempio seguente ottiene lo stato di blocco dell'accesso pubblico AMIs a livello di account nella regione specificata.

```
aws ec2 get-image-block-public-access-state \
    --region us-east-1
```

Output:

```
{
    "ImageBlockPublicAccessState": "block-new-sharing"
}
```

Per ulteriori informazioni, consulta Bloccare l'accesso pubblico al tuo AMIs nella Amazon EC2 User Guide.

- Per i dettagli sull'API, consulta GetImageBlockPublicAccessState AWS CLICommand Reference.

`get-instance-types-from-instance-requirements`

Il seguente esempio di codice mostra come utilizzare`get-instance-types-from-instance-requirements`.

AWS CLI

Come visualizzare in anteprima i tipi di istanza che corrispondono agli attributi specificati

L'esempio `get-instance-types-from-instance-requirements` seguente genera innanzitutto un elenco di tutti i possibili attributi che possono essere specificati utilizzando il parametro `--generate-cli-skeleton` e salva l'elenco in un file JSON. Quindi, il file JSON

viene utilizzato per personalizzare gli attributi per i quali visualizzare in anteprima i tipi di istanza corrispondenti.

Per generare tutti gli attributi possibili e salvare l'output direttamente in un file JSON, utilizza il seguente comando.

```
aws ec2 get-instance-types-from-instance-requirements \
    --region us-east-1 \
    --generate-cli-skeleton input > attributes.json
```

Output:

```
{
    "DryRun": true,
    "ArchitectureTypes": [
        "x86_64_mac"
    ],
    "VirtualizationTypes": [
        "paravirtual"
    ],
    "InstanceRequirements": {
        "VCpuCount": {
            "Min": 0,
            "Max": 0
        },
        "MemoryMiB": {
            "Min": 0,
            "Max": 0
        },
        "CpuManufacturers": [
            "intel"
        ],
        "MemoryGiBPerVCpu": {
            "Min": 0.0,
            "Max": 0.0
        },
        "ExcludedInstanceTypes": [
            ""
        ],
        "InstanceGenerations": [
            "current"
        ],
        "SpotMaxPricePercentageOverLowestPrice": 0,
```

```
        "OnDemandMaxPricePercentageOverLowestPrice": 0,
        "BareMetal": "included",
        "BurstablePerformance": "excluded",
        "RequireHibernateSupport": true,
        "NetworkInterfaceCount": {
            "Min": 0,
            "Max": 0
        },
        "LocalStorage": "required",
        "LocalStorageTypes": [
            "hdd"
        ],
        "TotalLocalStorageGB": {
            "Min": 0.0,
            "Max": 0.0
        },
        "BaselineEbsBandwidthMbps": {
            "Min": 0,
            "Max": 0
        },
        "AcceleratorTypes": [
            "inference"
        ],
        "AcceleratorCount": {
            "Min": 0,
            "Max": 0
        },
        "AcceleratorManufacturers": [
            "xilinx"
        ],
        "AcceleratorNames": [
            "t4"
        ],
        "AcceleratorTotalMemoryMiB": {
            "Min": 0,
            "Max": 0
        }
    },
    "MaxResults": 0,
    "NextToken": ""
}
```

Configurare il file JSON. È necessario fornire valori per `ArchitectureTypes`, `VirtualizationTypes`, `VCpuCount` e `MemoryMiB`. È possibile omettere gli altri attributi. Se omessi, vengono utilizzati i valori predefiniti. Per una descrizione di ogni attributo e dei relativi valori predefiniti, vedere get-instance-types-from -instance-requirements **<** - -requirements.html**>**. https://docs.aws.amazon.com/cli/ latest/reference/ec2/get instance-types-from-instance

Visualizza l'anteprima dei tipi di istanza che hanno gli attributi specificati in `attributes.json`. Specifica il nome e il percorso del file JSON utilizzando il parametro `--cli-input-json`. Nella seguente richiesta, l'output viene formattato come una tabella.

```
aws ec2 get-instance-types-from-instance-requirements \
    --cli-input-json file://attributes.json \
    --output table
```

Contenuto del file `attributes.json`:

```
{

    "ArchitectureTypes": [
        "x86_64"
    ],
    "VirtualizationTypes": [
        "hvm"
    ],
    "InstanceRequirements": {
        "VCpuCount": {
            "Min": 4,
            "Max": 6
        },
        "MemoryMiB": {
            "Min": 2048
        },
        "InstanceGenerations": [
            "current"
        ]
    }
}
```

Output:

```
-----------------------------------------
|GetInstanceTypesFromInstanceRequirements|
```

```
+---------------------------------------+
||             InstanceTypes             ||
|+-------------------------------------+|
||             InstanceType              ||
|+-------------------------------------+|
||  c4.xlarge                            ||
||  c5.xlarge                            ||
||  c5a.xlarge                           ||
||  c5ad.xlarge                          ||
||  c5d.xlarge                           ||
||  c5n.xlarge                           ||
||  d2.xlarge                            ||
...
```

Per ulteriori informazioni sulla selezione del tipo di istanza basata sugli attributi, consulta Come funziona la selezione del tipo di istanza basata sugli attributi nella Amazon User Guide. EC2

- Per i dettagli sull'API, consulta Command Reference. GetInstanceTypesFromInstanceRequirementsAWS CLI

`get-instance-uefi-data`

Il seguente esempio di codice mostra come utilizzare`get-instance-uefi-data`.

AWS CLI

Come recuperare i dati UEFI da un'istanza

L'esempio `get-instance-uefi-data` seguente recupera i dati UEFI da un'istanza. Se l'output è vuoto, l'istanza non contiene dati UEFI.

```
aws ec2 get-instance-uefi-data \
    --instance-id i-0123456789example
```

Output:

```
{
    "InstanceId": "i-0123456789example",
    "UefiData": "QU1aTlVFRkkf+uLXAAAAAHj5a7fZ9+3dBzxXb/.
    <snipped>
    AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAD4L/J/AODshho="
}
```

Per ulteriori informazioni, consulta UEFI Secure Boot nella Amazon EC2 User Guide.

- Per i dettagli sull'API, consulta GetInstanceUefiData AWS CLICommand Reference.

`get-ipam-address-history`

Il seguente esempio di codice mostra come utilizzare`get-ipam-address-history`.

AWS CLI

Come ottenere la cronologia di un CIDR

L'esempio `get-ipam-address-history` seguente recupera la cronologia di un CIDR.

Linux:

```
aws ec2 get-ipam-address-history \
   --cidr 10.0.0.0/16 \
   --ipam-scope-id ipam-scope-02fc38cd4c48e7d38 \
   --start-time 2021-12-08T01:00:00.000Z \
   --end-time 2021-12-10T01:00:00.000Z
```

Windows:

```
aws ec2 get-ipam-address-history ^
   --cidr 10.0.0.0/16 ^
   --ipam-scope-id ipam-scope-02fc38cd4c48e7d38 ^
   --start-time 2021-12-08T01:00:00.000Z ^
   --end-time 2021-12-10T01:00:00.000Z
```

Output:

```
{
    "HistoryRecords": [
        {
            "ResourceOwnerId": "123456789012",
            "ResourceRegion": "us-west-1",
            "ResourceType": "vpc",
            "ResourceId": "vpc-06cbefa9ee907e1c0",
            "ResourceCidr": "10.0.0.0/16",
            "ResourceName": "Demo",
            "ResourceComplianceStatus": "unmanaged",
```

```
            "ResourceOverlapStatus": "overlapping",
            "VpcId": "vpc-06cbefa9ee907e1c0",
            "SampledStartTime": "2021-12-08T19:54:57.675000+00:00"
        },
        {
            "ResourceOwnerId": "123456789012",
            "ResourceRegion": "us-east-2",
            "ResourceType": "vpc",
            "ResourceId": "vpc-042702f474812c9ad",
            "ResourceCidr": "10.0.0.0/16",
            "ResourceName": "test",
            "ResourceComplianceStatus": "unmanaged",
            "ResourceOverlapStatus": "overlapping",
            "VpcId": "vpc-042702f474812c9ad",
            "SampledStartTime": "2021-12-08T19:54:59.019000+00:00"
        },
        {
            "ResourceOwnerId": "123456789012",
            "ResourceRegion": "us-east-2",
            "ResourceType": "vpc",
            "ResourceId": "vpc-042b8a44f64267d67",
            "ResourceCidr": "10.0.0.0/16",
            "ResourceName": "tester",
            "ResourceComplianceStatus": "unmanaged",
            "ResourceOverlapStatus": "overlapping",
            "VpcId": "vpc-042b8a44f64267d67",
            "SampledStartTime": "2021-12-08T19:54:59.019000+00:00"
        }
    ]
}
```

Per ulteriori informazioni, consulta Visualizzare la cronologia degli indirizzi IP nella Guida per l'utente IPAM di Amazon VPC.

- Per i dettagli sull'API, consulta GetIpamAddressHistory AWS CLICommand Reference.

`get-ipam-discovered-accounts`

Il seguente esempio di codice mostra come utilizzare`get-ipam-discovered-accounts`.

AWS CLI

Come visualizzare gli account rilevati da un'IPAM

In questo scenario, sei un amministratore delegato IPAM che desidera visualizzare gli AWS account proprietari delle risorse rilevate dall'IPAM.

`--discovery-region` è la Regione operativa IPAM in cui visualizzare gli stati degli account monitorati. Ad esempio, se si dispone di tre Regioni operative IPAM, potrebbe essere necessario effettuare questa richiesta tre volte per visualizzare i timestamp specifici da individuare in ciascuna di tali particolari Regioni.

L'`get-ipam-discovered-accounts`esempio seguente elenca gli AWS account che possiedono le risorse rilevate dall'IPAM.

```
aws ec2 get-ipam-discovered-accounts \
   --ipam-resource-discovery-id ipam-res-disco-0365d2977fc1672fe \
   --discovery-region us-east-1
```

Output:

```
{
    "IpamDiscoveredAccounts": [
        {
            "AccountId": "149977607591",
            "DiscoveryRegion": "us-east-1",
            "LastAttemptedDiscoveryTime": "2024-02-09T19:04:31.379000+00:00",
            "LastSuccessfulDiscoveryTime": "2024-02-09T19:04:31.379000+00:00"
        }
    ]
}
```

Per ulteriori informazioni, consulta Come integrare IPAM con account esterni alla tua organizzazione nella Guida per l'utente IPAM di Amazon VPC.

- Per i dettagli sull'API, vedere GetIpamDiscoveredAccountsin AWS CLI Command Reference.

`get-ipam-discovered-public-addresses`

Il seguente esempio di codice mostra come utilizzare`get-ipam-discovered-public-addresses`.

AWS CLI

Come visualizzare gli indirizzi IP pubblici rilevati

In questo esempio, l'utente è un amministratore delegato IPAM che desidera visualizzare gli indirizzi IP delle risorse individuate dall'IPAM. È possibile ottenere l'ID di scoperta della risorsa con describe-ipam-resource-discoveries.

L'esempio `get-ipam-discovered-public-addresses` seguente mostra gli indirizzi IP pubblici rilevati per un'operazione di rilevamento delle risorse.

```
aws ec2 get-ipam-discovered-public-addresses \
    --ipam-resource-discovery-id ipam-res-disco-0f4ef577a9f37a162 \
    --address-region us-east-1 \
    --region us-east-1
```

Output:

```
{
    "IpamDiscoveredPublicAddresses": [
        {
           "IpamResourceDiscoveryId": "ipam-res-disco-0f4ef577a9f37a162",
            "AddressRegion": "us-east-1",
            "Address": "54.208.155.7",
            "AddressOwnerId": "320805250157",
            "AssociationStatus": "associated",
            "AddressType": "ec2-public-ip",
            "VpcId": "vpc-073b294916198ce49",
            "SubnetId": "subnet-0b6c8a8839e9a4f15",
            "NetworkInterfaceId": "eni-081c446b5284a5e06",
            "NetworkInterfaceDescription": "",
            "InstanceId": "i-07459a6fca5b35823",
            "Tags": {},
            "NetworkBorderGroup": "us-east-1c",
            "SecurityGroups": [
                {
                    "GroupName": "launch-wizard-2",
                    "GroupId": "sg-0a489dd6a65c244ce"
                }
            ],
            "SampleTime": "2024-04-05T15:13:59.228000+00:00"
        },
        {
            "IpamResourceDiscoveryId": "ipam-res-disco-0f4ef577a9f37a162",
            "AddressRegion": "us-east-1",
            "Address": "44.201.251.218",
            "AddressOwnerId": "470889052923",
```

```
            "AssociationStatus": "associated",
            "AddressType": "ec2-public-ip",
            "VpcId": "vpc-6c31a611",
            "SubnetId": "subnet-062f47608b99834b1",
            "NetworkInterfaceId": "eni-024845359c2c3ae9b",
            "NetworkInterfaceDescription": "",
            "InstanceId": "i-04ef786d9c4e03f41",
            "Tags": {},
            "NetworkBorderGroup": "us-east-1a",
            "SecurityGroups": [
                {
                    "GroupName": "launch-wizard-32",
                    "GroupId": "sg-0ed1a426e96a68374"
                }
            ],
            "SampleTime": "2024-04-05T15:13:59.145000+00:00"
        }
}
```

Per ulteriori informazioni, consulta Visualizzazione di informazioni sugli IP pubblici nella Guida per l'utente IPAM di Amazon VPC.

- Per i dettagli sull'API, consulta GetIpamDiscoveredPublicAddresses AWS CLICommand Reference.

`get-ipam-discovered-resource-cidrs`

Il seguente esempio di codice mostra come utilizzare`get-ipam-discovered-resource-cidrs`.

AWS CLI

Per visualizzare l'indirizzo IP CIDRs scoperto da un IPAM

In questo esempio, sei un amministratore delegato IPAM che desidera visualizzare i dettagli relativi all'indirizzo CIDRs IP delle risorse che l'IPAM sta scoprendo.

Per completare questa richiesta:

L'individuazione delle risorse scelta deve essere associata all'IPAM. Questa è la regione in cui `--resource-region` è stata creata la risorsa. AWS

L'esempio `get-ipam-discovered-resource-cidrs` seguente elenca gli indirizzi IP delle risorse rilevate dall'IPAM.

```
aws ec2 get-ipam-discovered-resource-cidrs \
   --ipam-resource-discovery-id ipam-res-disco-0365d2977fc1672fe \
   --resource-region us-east-1
```

Output:


```
{
    {
        "IpamDiscoveredResourceCidrs": [
        {
            "IpamResourceDiscoveryId": "ipam-res-disco-0365d2977fc1672fe",
            "ResourceRegion": "us-east-1",
            "ResourceId": "vpc-0c974c95ca7ceef4a",
            "ResourceOwnerId": "149977607591",
            "ResourceCidr": "172.31.0.0/16",
            "ResourceType": "vpc",
            "ResourceTags": [],
            "IpUsage": 0.375,
            "VpcId": "vpc-0c974c95ca7ceef4a",
            "SampleTime": "2024-02-09T19:15:16.529000+00:00"
        },
        {
            "IpamResourceDiscoveryId": "ipam-res-disco-0365d2977fc1672fe",
            "ResourceRegion": "us-east-1",
            "ResourceId": "subnet-07fe028119082a8c1",
            "ResourceOwnerId": "149977607591",
            "ResourceCidr": "172.31.0.0/20",
            "ResourceType": "subnet",
            "ResourceTags": [],
            "IpUsage": 0.0012,
            "VpcId": "vpc-0c974c95ca7ceef4a",
            "SampleTime": "2024-02-09T19:15:16.529000+00:00"
        },
        {
            "IpamResourceDiscoveryId": "ipam-res-disco-0365d2977fc1672fe",
            "ResourceRegion": "us-east-1",
            "ResourceId": "subnet-0a96893763984cc4e",
            "ResourceOwnerId": "149977607591",
            "ResourceCidr": "172.31.64.0/20",
            "ResourceType": "subnet",
            "ResourceTags": [],
            "IpUsage": 0.0012,
            "VpcId": "vpc-0c974c95ca7ceef4a",
```

```
            "SampleTime": "2024-02-09T19:15:16.529000+00:00"
        }
    }
}
```

Per ulteriori informazioni, consulta Monitoraggio dell'utilizzo del CIDR per risorsa nella Guida per l'utente IPAM di Amazon VPC.

- Per i dettagli sull'API, consulta Command GetIpamDiscoveredResourceCidrsReference AWS CLI .

`get-ipam-pool-allocations`

Il seguente esempio di codice mostra come utilizzare`get-ipam-pool-allocations`.

AWS CLI

Per ottenere l' CIDRs allocazione da un pool IPAM

L'`get-ipam-pool-allocations`esempio seguente ottiene l' CIDRs allocazione da un pool IPAM.

Linux:

```
aws ec2 get-ipam-pool-allocations \
    --ipam-pool-id ipam-pool-0533048da7d823723 \
    --filters Name=ipam-pool-allocation-id,Values=ipam-pool-
alloc-0e6186d73999e47389266a5d6991e6220
```

Windows:

```
aws ec2 get-ipam-pool-allocations ^
    --ipam-pool-id ipam-pool-0533048da7d823723 ^
    --filters Name=ipam-pool-allocation-id,Values=ipam-pool-
alloc-0e6186d73999e47389266a5d6991e6220
```

Output:

```
{
    "IpamPoolAllocations": [
        {
            "Cidr": "10.0.0.0/16",
```

```
            "IpamPoolAllocationId": "ipam-pool-
alloc-0e6186d73999e47389266a5d6991e6220",
            "ResourceType": "custom",
            "ResourceOwner": "123456789012"
        }
    ]
}
```

- Per i dettagli sull'API, vedere GetIpamPoolAllocationsin AWS CLI Command Reference.

`get-ipam-pool-cidrs`

Il seguente esempio di codice mostra come utilizzare`get-ipam-pool-cidrs`.

AWS CLI

Per far sì che il CIDRs provisioning venga inserito in un pool IPAM

L'`get-ipam-pool-cidrs`esempio seguente ottiene il CIDRs provisioning in un pool IPAM.

Linux:

```
aws ec2 get-ipam-pool-cidrs \
    --ipam-pool-id ipam-pool-0533048da7d823723 \
    --filters 'Name=cidr,Values=10.*'
```

Windows:

```
aws ec2 get-ipam-pool-cidrs ^
    --ipam-pool-id ipam-pool-0533048da7d823723 ^
    --filters Name=cidr,Values=10.*
```

Output:

```
{
    "IpamPoolCidr": {
        "Cidr": "10.0.0.0/24",
        "State": "provisioned"
    }
}
```

- Per i dettagli sull'API, vedere GetIpamPoolCidrsin AWS CLI Command Reference.

`get-ipam-resource-cidrs`

Il seguente esempio di codice mostra come utilizzare`get-ipam-resource-cidrs`.

AWS CLI

Per ottenere l' CIDRs allocazione a una risorsa

L'`get-ipam-resource-cidrs`esempio seguente ottiene l' CIDRs allocazione a una risorsa.

Linux:

```
aws ec2 get-ipam-resource-cidrs \
    --ipam-scope-id ipam-scope-02fc38cd4c48e7d38 \
    --filters Name=management-state,Values=unmanaged
```

Windows:

```
aws ec2 get-ipam-resource-cidrs ^
    --ipam-scope-id ipam-scope-02fc38cd4c48e7d38 ^
    --filters Name=management-state,Values=unmanaged
```

Output:

```
{
    "IpamResourceCidrs": [
        {
            "IpamId": "ipam-08440e7a3acde3908",
            "IpamScopeId": "ipam-scope-02fc38cd4c48e7d38",
            "ResourceRegion": "us-east-2",
            "ResourceOwnerId": "123456789012",
            "ResourceId": "vpc-621b8709",
            "ResourceName": "Default AWS VPC",
            "ResourceCidr": "172.33.0.0/16",
            "ResourceType": "vpc",
            "ResourceTags": [
                {
                    "Key": "Environment",
                    "Value": "Test"
                },
                {
                    "Key": "Name",
```

```
                    "Value": "Default AWS VPC"
                }
            ],
            "IpUsage": 0.0039,
            "ComplianceStatus": "unmanaged",
            "ManagementState": "unmanaged",
            "OverlapStatus": "nonoverlapping",
            "VpcId": "vpc-621b8709"
        }
    ]
}
```

Per ulteriori informazioni, consulta Monitoraggio dell'utilizzo del CIDR per risorsa nella Guida per l'utente IPAM di Amazon VPC.

- Per i dettagli sull'API, vedere GetIpamResourceCidrsin AWS CLI Command Reference.

`get-launch-template-data`

Il seguente esempio di codice mostra come utilizzare`get-launch-template-data`.

AWS CLI

Come ottenere i dati dell'istanza per un modello di avvio

Questo esempio ottiene i dati sull'istanza specificata e utilizza l'opzione `--query` per restituire il contenuto in `LaunchTemplateData`. È possibile utilizzare l'output come base per creare un nuovo modello di avvio o una nuova versione del modello di avvio.

Comando:

```
aws ec2 get-launch-template-data --instance-id i-0123d646e8048babc --query
 'LaunchTemplateData'
```

Output:

```
{
      "Monitoring": {},
      "ImageId": "ami-8c1be5f6",
      "BlockDeviceMappings": [
          {
```

```
                "DeviceName": "/dev/xvda",
                "Ebs": {
                    "DeleteOnTermination": true
                }
            }
        ],
        "EbsOptimized": false,
        "Placement": {
            "Tenancy": "default",
            "GroupName": "",
            "AvailabilityZone": "us-east-1a"
        },
        "InstanceType": "t2.micro",
        "NetworkInterfaces": [
            {
                "Description": "",
                "NetworkInterfaceId": "eni-35306abc",
                "PrivateIpAddresses": [
                    {
                        "Primary": true,
                        "PrivateIpAddress": "10.0.0.72"
                    }
                ],
                "SubnetId": "subnet-7b16de0c",
                "Groups": [
                    "sg-7c227019"
                ],
                "Ipv6Addresses": [
                    {
                        "Ipv6Address": "2001:db8:1234:1a00::123"
                    }
                ],
                "PrivateIpAddress": "10.0.0.72"
            }
        ]
}
```

- Per i dettagli sull'API, consulta GetLaunchTemplateData AWS CLICommand Reference.

`get-managed-prefix-list-associations`

Il seguente esempio di codice mostra come utilizzare`get-managed-prefix-list-associations`.

AWS CLI

Come ottenere le associazioni dell'elenco di prefissi

L'esempio `get-managed-prefix-list-associations` seguente recupera le risorse associate all'elenco di prefissi specificato.

```
aws ec2 get-managed-prefix-list-associations \
    --prefix-list-id pl-0123456abcabcabc1
```

Output:

```
{
    "PrefixListAssociations": [
        {
            "ResourceId": "sg-0abc123456abc12345",
            "ResourceOwner": "123456789012"
        }
    ]
}
```

Per ulteriori informazioni, consulta Elenchi di prefissi gestiti nella Guida per l'utente di Amazon VPC.

- Per i dettagli sull'API, consulta GetManagedPrefixListAssociations AWS CLICommand Reference.

`get-managed-prefix-list-entries`

Il seguente esempio di codice mostra come utilizzare`get-managed-prefix-list-entries`.

AWS CLI

Come ottenere le voci per un elenco di prefissi

Il comando `get-managed-prefix-list-entries` seguente recupera le voci per l'elenco di prefissi specificato.

```
aws ec2 get-managed-prefix-list-entries \
    --prefix-list-id pl-0123456abcabcabc1
```

Output:

```
{
    "Entries": [
        {
            "Cidr": "10.0.0.0/16",
            "Description": "vpc-a"
        },
        {
            "Cidr": "10.2.0.0/16",
            "Description": "vpc-b"
        }
    ]
}
```

Per ulteriori informazioni, consulta Elenchi di prefissi gestiti nella Guida per l'utente di Amazon VPC.

- Per i dettagli sull'API, consulta GetManagedPrefixListEntries AWS CLICommand Reference.

`get-network-insights-access-scope-analysis-findings`

Il seguente esempio di codice mostra come utilizzare`get-network-insights-access-scope-analysis-findings`.

AWS CLI

Come ottenere gli esiti dell'analisi dell'ambito di accesso alle informazioni di rete

L'`get-network-insights-access-scope-analysis-findings`esempio seguente ottiene i risultati dell'analisi dell'ambito selezionato nel tuo AWS account.

```
aws ec2 get-network-insights-access-scope-analysis-findings \
    --region us-east-1 \
    --network-insights-access-scope-analysis-id nis \
    --nis-123456789111
```

Output:

```
{
    "NetworkInsightsAccessScopeAnalysisId": "nisa-123456789222",
    "AnalysisFindings": [
        {
            "NetworkInsightsAccessScopeAnalysisId": "nisa-123456789222",
```

```
            "NetworkInsightsAccessScopeId": "nis-123456789111",
            "FindingComponents": [
                {
                    "SequenceNumber": 1,
                    "Component": {
                        "Id": "eni-02e3d42d5cceca67d",
                        "Arn": "arn:aws:ec2:us-east-1:936459623503:network-
interface/eni-02e3d32d9cceca17d"
                    },
                    "OutboundHeader": {
                        "DestinationAddresses": [
                            "0.0.0.0/5",
                            "11.0.0.0/8",
                            "12.0.0.0/6",
                            "128.0.0.0/3",
                            "16.0.0.0/4",
                            "160.0.0.0/5",
                            "168.0.0.0/6",
                            "172.0.0.0/12"
                            "8.0.0.0/7"
                        ],
                        "DestinationPortRanges": [
                            {
                                "From": 0,
                                "To": 65535
                            }
                        ],
                        "Protocol": "6",
                        "SourceAddresses": [
                            "10.0.2.253/32"
                        ],
                        "SourcePortRanges": [
                            {
                                "From": 0,
                                "To": 65535
                            }
                        ]
                    }, [etc]
                ]
            }
        }
    ]
}
```

Per ulteriori informazioni, consulta Guida introduttiva a Network Access Analyzer utilizzando la AWS CLI nella Guida Network Access Analyzer.

- Per i dettagli sull'API, consulta Command GetNetworkInsightsAccessScopeAnalysisFindingsReference AWS CLI .

`get-network-insights-access-scope-content`

Il seguente esempio di codice mostra come utilizzare`get-network-insights-access-scope-content`.

AWS CLI

Come ottenere il contenuto dell'ambito di accesso alle informazioni di rete

L'`get-network-insights-access-scope-content`esempio seguente ottiene il contenuto dell'ID di analisi dell'ambito selezionato nel tuo AWS account.

```
aws ec2 get-network-insights-access-scope-content \
    --region us-east-1 \
    --network-insights-access-scope-id nis-123456789222
```

Output:

```
{
    "NetworkInsightsAccessScopeContent": {
        "NetworkInsightsAccessScopeId": "nis-123456789222",
        "MatchPaths": [
            {
                "Source": {
                    "ResourceStatement": {
                        "ResourceTypes": [
                            "AWS::EC2::NetworkInterface"
                        ]
                    }
                },
                "Destination": {
                    "ResourceStatement": {
                        "ResourceTypes": [
                            "AWS::EC2::InternetGateway"
                        ]
                    }
```

```
                }
            }
        ]
    }
}
```

Per ulteriori informazioni, consulta Guida introduttiva a Network Access Analyzer utilizzando la AWS CLI nella Guida Network Access Analyzer.

- Per i dettagli sull'API, consulta Command GetNetworkInsightsAccessScopeContentReference AWS CLI .

`get-password-data`

Il seguente esempio di codice mostra come utilizzare`get-password-data`.

AWS CLI

Come ottenere la password crittografata

Questo esempio recupera la password crittografata.

Comando:

```
aws ec2 get-password-data --instance-id i-1234567890abcdef0
```

Output:

```
{
    "InstanceId": "i-1234567890abcdef0",
    "Timestamp": "2013-08-07T22:18:38.000Z",
    "PasswordData": "gSlJFq+VpcZXqy+iktxMF6NyxQ4qCrT4+gaOuNOenX1MmgXPTj7XEXAMPLE
UQ+YeFfb+L1U4C4AKv652Ux1iRB3CPTYP7WmU3TUnhsuBd+p6LVk7T2lKUml6OXbk6WPW1VYYm/TRPB1
e1DQ7PY4an/DgZT4mwcpRFigzhniQgDDeO1InvSDcwoUTwNs0Y1S8ouri2W4n5GNlriM3Q0AnNVelVz/
53TkDtxbNoU606M1gK9zUWSxqEgwvbV2j8c5rP0WCuaMWSFl4ziDu4bd7q+4RSyi8NUsVWnKZ4aEZffu
DPGzKrF5yLlf3etP2L4ZR6CvG7K1hx7VKOQVN32Dajw=="
}
```

Come ottenere la password decrittografata

Questo esempio recupera la password decrittografata.

Comando:

```
aws ec2 get-password-data --instance-id  i-1234567890abcdef0 --priv-launch-key C:
\Keys\MyKeyPair.pem
```

Output:

```
{
    "InstanceId": "i-1234567890abcdef0",
    "Timestamp": "2013-08-30T23:18:05.000Z",
    "PasswordData": "&ViJ652e*u"
}
```

- Per i dettagli sull'API, consulta GetPasswordData AWS CLICommand Reference.

`get-reserved-instances-exchange-quote`

Il seguente esempio di codice mostra come utilizzare`get-reserved-instances-exchange-quote`.

AWS CLI

Come ottenere un preventivo per lo scambio di un'istanza riservata modificabile

Questo esempio ottiene le informazioni di scambio per le istanze riservate modificabili specificate.

Comando:

```
aws ec2 get-reserved-instances-exchange-quote --reserved-
instance-ids 7b8750c3-397e-4da4-bbcb-a45ebexample --target-
configurations OfferingId=6fea5434-b379-434c-b07b-a7abexample
```

Output:

```
{
  "CurrencyCode": "USD",
  "ReservedInstanceValueSet": [
      {
          "ReservedInstanceId": "7b8750c3-397e-4da4-bbcb-a45ebexample",
          "ReservationValue": {
              "RemainingUpfrontValue": "0.000000",
              "HourlyPrice": "0.027800",
```

```
                "RemainingTotalValue": "730.556200"
            }
        }
    ],
    "PaymentDue": "424.983828",
    "TargetConfigurationValueSet": [
        {
            "TargetConfiguration": {
                "InstanceCount": 5,
                "OfferingId": "6fea5434-b379-434c-b07b-a7abexample"
            },
            "ReservationValue": {
                "RemainingUpfrontValue": "424.983828",
                "HourlyPrice": "0.016000",
                "RemainingTotalValue": "845.447828"
            }
        }
    ],
    "IsValidExchange": true,
    "OutputReservedInstancesWillExpireAt": "2020-10-01T13:03:39Z",
    "ReservedInstanceValueRollup": {
        "RemainingUpfrontValue": "0.000000",
        "HourlyPrice": "0.027800",
        "RemainingTotalValue": "730.556200"
    },
    "TargetConfigurationValueRollup": {
        "RemainingUpfrontValue": "424.983828",
        "HourlyPrice": "0.016000",
        "RemainingTotalValue": "845.447828"
    }
}
```

- Per i dettagli sull'API, consulta GetReservedInstancesExchangeQuote AWS CLICommand Reference.

`get-security-groups-for-vpc`

Il seguente esempio di codice mostra come utilizzare`get-security-groups-for-vpc`.

AWS CLI

Come visualizzare i gruppi di sicurezza che possono essere associati alle interfacce di rete in un VPC specificato.

L'esempio `get-security-groups-for-vpc` seguente mostra i gruppi di sicurezza che possono essere associati alle interfacce di rete nel VPC.

```
aws ec2 get-security-groups-for-vpc \
    --vpc-id vpc-6c31a611 \
    --region us-east-1
```

Output:

```
{
    "SecurityGroupForVpcs": [
        {
            "Description": "launch-wizard-36 created 2022-08-29T15:59:35.338Z",
            "GroupName": "launch-wizard-36",
            "OwnerId": "470889052923",
            "GroupId": "sg-007e0c3027ee885f5",
            "Tags": [],
            "PrimaryVpcId": "vpc-6c31a611"
        },
        {
            "Description": "launch-wizard-18 created 2024-01-19T20:22:27.527Z",
            "GroupName": "launch-wizard-18",
            "OwnerId": "470889052923",
            "GroupId": "sg-0147193bef51c9eef",
            "Tags": [],
            "PrimaryVpcId": "vpc-6c31a611"
        }
}
```

- Per i dettagli sull'API, consulta GetSecurityGroupsForVpc AWS CLICommand Reference.

`get-serial-console-access-status`

Il seguente esempio di codice mostra come utilizzare`get-serial-console-access-status`.

AWS CLI

Come visualizzare l'accesso account alla console seriale

L'esempio `get-serial-console-access-status` seguente determina se l'accesso alla console seriale è abilitato per l'account in uso.

```
aws ec2 get-serial-console-access-status
```

Output:

```
{
    "SerialConsoleAccessEnabled": true
}
```

Per ulteriori informazioni, consulta EC2 Serial Console nella Amazon EC2 User Guide.

- Per i dettagli sull'API, consulta GetSerialConsoleAccessStatus AWS CLICommand Reference.

`get-snapshot-block-public-access-state`

Il seguente esempio di codice mostra come utilizzare`get-snapshot-block-public-access-state`.

AWS CLI

Come ottenere lo stato corrente del blocco dell'accesso pubblico per gli snapshot

L'esempio `get-snapshot-block-public-access-state` seguente ottiene lo stato corrente del blocco dell'accesso pubblico per gli snapshot.

```
aws ec2 get-snapshot-block-public-access-state
```

Output:

```
{
    "State": "block-all-sharing"
}
```

Per ulteriori informazioni, consulta Blocco dell'accesso pubblico per gli snapshot nella Guida per l'utente di Amazon EBS.

- Per i dettagli sull'API, consulta GetSnapshotBlockPublicAccessState AWS CLICommand Reference.

`get-spot-placement-scores`

Il seguente esempio di codice mostra come utilizzare`get-spot-placement-scores`.

AWS CLI

Come calcolare il punteggio di posizionamento spot per i requisiti specificati

L'esempio `get-spot-placement-scores` seguente genera innanzitutto un elenco di tutti i parametri possibili che possono essere specificati per la configurazione del punteggio di posizionamento spot utilizzando il parametro `--generate-cli-skeleton` e salva l'elenco in un file JSON. Quindi, il file JSON viene utilizzato per configurare i requisiti da utilizzare per calcolare il punteggio di posizionamento spot.

Come generare tutti i possibili parametri che possono essere specificati per la configurazione del punteggio di posizionamento spot e salvare l'output direttamente in un file JSON.

```
aws ec2 get-spot-placement-scores \
    --region us-east-1 \
    --generate-cli-skeleton input > attributes.json
```

Output:

```
{
    "InstanceTypes": [
        ""
    ],
    "TargetCapacity": 0,
    "TargetCapacityUnitType": "vcpu",
    "SingleAvailabilityZone": true,
    "RegionNames": [
        ""
    ],
    "InstanceRequirementsWithMetadata": {
        "ArchitectureTypes": [
            "x86_64_mac"
        ],
        "VirtualizationTypes": [
            "hvm"
        ],
        "InstanceRequirements": {
            "VCpuCount": {
                "Min": 0,
                "Max": 0
            },
            "MemoryMiB": {
                "Min": 0,
```

```
                "Max": 0
            },
            "CpuManufacturers": [
                "amd"
            ],
            "MemoryGiBPerVCpu": {
                "Min": 0.0,
                "Max": 0.0
            },
            "ExcludedInstanceTypes": [
                ""
            ],
            "InstanceGenerations": [
                "previous"
            ],
            "SpotMaxPricePercentageOverLowestPrice": 0,
            "OnDemandMaxPricePercentageOverLowestPrice": 0,
            "BareMetal": "excluded",
            "BurstablePerformance": "excluded",
            "RequireHibernateSupport": true,
            "NetworkInterfaceCount": {
                "Min": 0,
                "Max": 0
            },
            "LocalStorage": "included",
            "LocalStorageTypes": [
                "hdd"
            ],
            "TotalLocalStorageGB": {
                "Min": 0.0,
                "Max": 0.0
            },
            "BaselineEbsBandwidthMbps": {
                "Min": 0,
                "Max": 0
            },
            "AcceleratorTypes": [
                "fpga"
            ],
            "AcceleratorCount": {
                "Min": 0,
                "Max": 0
            },
            "AcceleratorManufacturers": [
```

```
                "amd"
            ],
            "AcceleratorNames": [
                "vu9p"
            ],
            "AcceleratorTotalMemoryMiB": {
                "Min": 0,
                "Max": 0
            }
        }
    },
    "DryRun": true,
    "MaxResults": 0,
    "NextToken": ""
}
```

Configurare il file JSON. È necessario fornire un valore per `TargetCapacity`: Per una descrizione di ogni parametro e dei relativi valori predefiniti, consulta Calcolare il punteggio di posizionamento Spot (AWS CLI) <https://docs.aws.amazon.com/AWS EC2/latest/UserGuide/spotcalculate-sps-cli-placement-score.html# >.

Calcola il punteggio di posizionamento spot per i requisiti specificati in `attributes.json`. Specifica il nome e il percorso del file JSON utilizzando il parametro `--cli-input-json`.

```
aws ec2 get-spot-placement-scores \
    --region us-east-1 \
    --cli-input-json file://attributes.json
```

Viene restituito un output se `SingleAvailabilityZone` è impostato su `false` o omesso (se omesso, il valore predefinito è `false`). Viene restituito un elenco con punteggio di Regioni.

```
"Recommendation": [
    {
        "Region": "us-east-1",
        "Score": 7
    },
    {
        "Region": "us-west-1",
        "Score": 5
    },
   ...
```

Viene restituito un output se `SingleAvailabilityZone` è impostato su `true`. Viene restituito un elenco di zone con punteggio. SingleAvailability

```
"Recommendation": [
    {
        "Region": "us-east-1",
        "AvailabilityZoneId": "use1-az1"
        "Score": 8
    },
    {
        "Region": "us-east-1",
        "AvailabilityZoneId": "usw2-az3"
        "Score": 6
    },
   ...
```

Per ulteriori informazioni sul calcolo di un punteggio di posizionamento Spot e, ad esempio, sulle configurazioni, consulta Calculate a Spot placement score nella Amazon EC2 User Guide.

- Per i dettagli sull'API, consulta AWS CLI Command GetSpotPlacementScoresReference.

`get-subnet-cidr-reservations`

Il seguente esempio di codice mostra come utilizzare`get-subnet-cidr-reservations`.

AWS CLI

Come ottenere informazioni su una prenotazione CIDR della sottorete

L'esempio `get-subnet-cidr-reservations` seguente visualizza informazioni sulla prenotazione CIDR della sottorete specificata.

```
aws ec2 get-subnet-cidr-reservations \
    --subnet-id subnet-03c51e2e6cEXAMPLE
```

Output:

```
{
    "SubnetIpv4CidrReservations": [
        {
            "SubnetCidrReservationId": "scr-044f977c4eEXAMPLE",
            "SubnetId": "subnet-03c51e2e6cEXAMPLE",
```

```
            "Cidr": "10.1.0.16/28",
            "ReservationType": "prefix",
            "OwnerId": "123456789012"
        }
    ],
    "SubnetIpv6CidrReservations": []
}
```

Per ulteriori informazioni, consulta Prenotazioni della CIDR per la sottorete nella Guida per l'utente di Amazon VPC.

- Per i dettagli sull'API, consulta GetSubnetCidrReservations AWS CLICommand Reference.

`get-transit-gateway-attachment-propagations`

Il seguente esempio di codice mostra come utilizzare`get-transit-gateway-attachment-propagations`.

AWS CLI

Come elencare le tabelle di routing in cui l'allegato della risorsa specificata propaga le route

L'esempio `get-transit-gateway-attachment-propagations` seguente elenca le tabelle di routing in cui l'allegato della risorsa specificata propaga le route.

```
aws ec2 get-transit-gateway-attachment-propagations \
    --transit-gateway-attachment-id tgw-attach-09fbd47ddfEXAMPLE
```

Output:

```
{
    "TransitGatewayAttachmentPropagations": [
        {
            "TransitGatewayRouteTableId": "tgw-rtb-0882c61b97EXAMPLE",
            "State": "enabled"
        }
    ]
}
```

Per ulteriori informazioni, consulta Tabelle di routing del gateway di transito nella Guida di Transit Gateway.

- Per i dettagli sull'API, consulta GetTransitGatewayAttachmentPropagations AWS CLICommand Reference.

`get-transit-gateway-multicast-domain-associations`

Il seguente esempio di codice mostra come utilizzare`get-transit-gateway-multicast-domain-associations`.

AWS CLI

Come visualizzare le informazioni sulle associazioni per un dominio multicast (trasmissione uno a molti) del gateway di transito

L'esempio `get-transit-gateway-multicast-domain-associations` seguente restituisce le associazioni per il dominio multicast (trasmissione uno a molti) specificato.

```
aws ec2 get-transit-gateway-multicast-domain-associations \
    --transit-gateway-multicast-domain-id tgw-mcast-domain-0c4905cef7EXAMPLE
```

Output:

```
{
    "MulticastDomainAssociations": [
        {
            "TransitGatewayAttachmentId": "tgw-attach-028c1dd0f8EXAMPLE",
            "ResourceId": "vpc-01128d2c24EXAMPLE",
            "ResourceType": "vpc",
            "Subnet": {
                "SubnetId": "subnet-000de86e3bEXAMPLE",
                "State": "associated"
            }
        },
        {
            "TransitGatewayAttachmentId": "tgw-attach-070e571cd1EXAMPLE",
            "ResourceId": "vpc-7EXAMPLE",
            "ResourceType": "vpc",
            "Subnet": {
                "SubnetId": "subnet-4EXAMPLE",
                "State": "associated"
            }
        },
        {
```

```
            "TransitGatewayAttachmentId": "tgw-attach-070e571cd1EXAMPLE",
            "ResourceId": "vpc-7EXAMPLE",
            "ResourceType": "vpc",
            "Subnet": {
                "SubnetId": "subnet-5EXAMPLE",
                "State": "associated"
            }
        },
        {
            "TransitGatewayAttachmentId": "tgw-attach-070e571cd1EXAMPLE",
            "ResourceId": "vpc-7EXAMPLE",
            "ResourceType": "vpc",
            "Subnet": {
                "SubnetId": "subnet-aEXAMPLE",
                "State": "associated"
            }
        },
        {
            "TransitGatewayAttachmentId": "tgw-attach-070e571cd1EXAMPLE",
            "ResourceId": "vpc-7EXAMPLE",
            "ResourceType": "vpc",
            "Subnet": {
                "SubnetId": "subnet-fEXAMPLE",
                "State": "associated"
            }
        }
    ]
}
```

Per ulteriori informazioni, consulta Domini multicast (trasmissione uno a molti) nella Guida di Transit Gateway.

- Per i dettagli sull'API, consulta GetTransitGatewayMulticastDomainAssociations AWS CLICommand Reference.

`get-transit-gateway-prefix-list-references`

Il seguente esempio di codice mostra come utilizzare`get-transit-gateway-prefix-list-references`.

AWS CLI

Come ottenere i riferimenti all'elenco di prefissi nella tabella di routing del gateway di transito

L'esempio `get-transit-gateway-prefix-list-references` seguente ottiene i riferimenti all'elenco di prefissi per la tabella di routing del gateway di transito specificata e filtra in base all'ID di un elenco di prefissi specifico.

```
aws ec2 get-transit-gateway-prefix-list-references \
    --transit-gateway-route-table-id tgw-rtb-0123456789abcd123 \
    --filters Name=prefix-list-id,Values=pl-11111122222222333
```

Output:

```
{
    "TransitGatewayPrefixListReferences": [
        {
            "TransitGatewayRouteTableId": "tgw-rtb-0123456789abcd123",
            "PrefixListId": "pl-11111122222222333",
            "PrefixListOwnerId": "123456789012",
            "State": "available",
            "Blackhole": false,
            "TransitGatewayAttachment": {
                "TransitGatewayAttachmentId": "tgw-attach-aabbccddaabbccaab",
                "ResourceType": "vpc",
                "ResourceId": "vpc-112233445566aabbc"
            }
        }
    ]
}
```

Per ulteriori informazioni, consulta Riferimenti elenco dei prefissi nella Guida di Transit Gateway.

- Per i dettagli sull'API, consulta GetTransitGatewayPrefixListReferences AWS CLICommand Reference.

`get-transit-gateway-route-table-associations`

Il seguente esempio di codice mostra come utilizzare`get-transit-gateway-route-table-associations`.

AWS CLI

Come ottenere informazioni sulle associazioni per la tabella di routing del gateway di transito specificata

L'esempio `get-transit-gateway-route-table-associations` seguente visualizza informazioni sulle associazioni per la tabella di routing del gateway di transito specificata.

```
aws ec2 get-transit-gateway-route-table-associations \
    --transit-gateway-route-table-id tgw-rtb-0a823edbdeEXAMPLE
```

Output:

```
{
    "Associations": [
        {
            "TransitGatewayAttachmentId": "tgw-attach-09b52ccdb5EXAMPLE",
            "ResourceId": "vpc-4d7de228",
            "ResourceType": "vpc",
            "State": "associating"
        }
    ]
}
```

Per ulteriori informazioni, consulta Tabelle di routing del gateway di transito nella Guida di Transit Gateway.

- Per i dettagli sull'API, consulta GetTransitGatewayRouteTableAssociations AWS CLICommand Reference.

`get-transit-gateway-route-table-propagations`

Il seguente esempio di codice mostra come utilizzare`get-transit-gateway-route-table-propagations`.

AWS CLI

Come visualizzare informazioni sulle propagazioni delle tabelle di routing per la tabella di routing del gateway di transito specificato

L'esempio `get-transit-gateway-route-table-propagations` seguente restituisce le propagazioni della tabella di routing per la tabella di routing specificata.

```
aws ec2 get-transit-gateway-route-table-propagations \
    --transit-gateway-route-table-id tgw-rtb-002573ed1eEXAMPLE
```

Output:

```
{
    "TransitGatewayRouteTablePropagations": [
        {
            "TransitGatewayAttachmentId": "tgw-attach-01f8100bc7EXAMPLE",
            "ResourceId": "vpc-3EXAMPLE",
            "ResourceType": "vpc",
            "State": "enabled"
        },
        {
            "TransitGatewayAttachmentId": "tgw-attach-08e0bc912cEXAMPLE",
            "ResourceId": "11460968-4ac1-4fd3-bdb2-00599EXAMPLE",
            "ResourceType": "direct-connect-gateway",
            "State": "enabled"
        },
        {
            "TransitGatewayAttachmentId": "tgw-attach-0a89069f57EXAMPLE",
            "ResourceId": "8384da05-13ce-4a91-aada-5a1baEXAMPLE",
            "ResourceType": "direct-connect-gateway",
            "State": "enabled"
        }
    ]
}
```

Per ulteriori informazioni, consulta Tabelle di routing del gateway di transito nella Guida di Transit Gateway.

- Per i dettagli sull'API, consulta GetTransitGatewayRouteTablePropagations AWS CLICommand Reference.

`get-verified-access-endpoint-policy`

Il seguente esempio di codice mostra come utilizzare`get-verified-access-endpoint-policy`.

AWS CLI

Come ottenere la policy di accesso verificato di un endpoint

L'esempio `get-verified-access-endpoint-policy` seguente ottiene la policy di accesso verificato dell'endpoint specificato.

```
aws ec2 get-verified-access-endpoint-policy \
```

```
    --verified-access-endpoint-id vae-066fac616d4d546f2
```

Output:

```
{
    "PolicyEnabled": true,
    "PolicyDocument": "permit(principal,action,resource)\nwhen
 {\n    context.identity.groups.contains(\"finance\") &&\n
 context.identity.email_verified == true\n};"
}
```

Per ulteriori informazioni, consulta Policy di Accesso verificato nella Guida per l'utente di Accesso verificato da AWS .

- Per i dettagli sull'API, consulta GetVerifiedAccessEndpointPolicy AWS CLICommand Reference.

`get-verified-access-group-policy`

Il seguente esempio di codice mostra come utilizzare`get-verified-access-group-policy`.

AWS CLI

Come ottenere la policy di accesso verificato di un gruppo

L'esempio `get-verified-access-group-policy` seguente recupera la policy di accesso verificato del gruppo specificato.

```
aws ec2 get-verified-access-group-policy \
    --verified-access-group-id vagr-0dbe967baf14b7235
```

Output:

```
{
    "PolicyEnabled": true,
    "PolicyDocument": "permit(principal,action,resource)\nwhen
 {\n    context.identity.groups.contains(\"finance\") &&\n
 context.identity.email_verified == true\n};"
}
```

Per ulteriori informazioni, consulta Gruppi di Accesso verificato nella Guida per l'utente di Accesso verificato da AWS .

- Per i dettagli sull'API, consulta GetVerifiedAccessGroupPolicy AWS CLICommand Reference.

`get-vpn-connection-device-sample-configuration`

Il seguente esempio di codice mostra come utilizzare`get-vpn-connection-device-sample-configuration`.

AWS CLI

Come scaricare un file di configurazione di esempio

L'esempio `get-vpn-connection-device-sample-configuration` seguente scarica il file di configurazione di esempio specificato. Per elencare i dispositivi gateway con un file di configurazione di esempio, chiama il comando `get-vpn-connection-device-types`.

```
aws ec2 get-vpn-connection-device-sample-configuration \
    --vpn-connection-id vpn-123456789abc01234 \
    --vpn-connection-device-type-id 5fb390ba
```

Output:

```
{
    "VpnConnectionDeviceSampleConfiguration": "contents-of-the-sample-configuration-file"
}
```

Per ulteriori informazioni, consulta Scaricare il file di configurazione nella Guida per l'utente della AWS Site-to-Site VPN.

- Per i dettagli sull'API, consulta GetVpnConnectionDeviceSampleConfiguration AWS CLICommand Reference.

`get-vpn-connection-device-types`

Il seguente esempio di codice mostra come utilizzare`get-vpn-connection-device-types`.

AWS CLI

Come elencare i dispositivi gateway con un file di configurazione di esempio

L'esempio `get-vpn-connection-device-types` seguente elenca i dispositivi gateway di Palo Alto Networks che dispongono di file di configurazione di esempio.

```
aws ec2 get-vpn-connection-device-types \
    --query "VpnConnectionDeviceTypes[?Vendor==`Palo Alto Networks`]"
```

Output:

```
[
    {
        "VpnConnectionDeviceTypeId": "754a6372",
        "Vendor": "Palo Alto Networks",
        "Platform": "PA Series",
        "Software": "PANOS 4.1.2+"
    },
    {
        "VpnConnectionDeviceTypeId": "9612cbed",
        "Vendor": "Palo Alto Networks",
        "Platform": "PA Series",
        "Software": "PANOS 4.1.2+ (GUI)"
    },
    {
        "VpnConnectionDeviceTypeId": "5fb390ba",
        "Vendor": "Palo Alto Networks",
        "Platform": "PA Series",
        "Software": "PANOS 7.0+"
    }
]
```

Per ulteriori informazioni, consulta Scaricare il file di configurazione nella Guida per l'utente della AWS Site-to-Site VPN.

- Per i dettagli sull'API, consulta GetVpnConnectionDeviceTypes AWS CLICommand Reference.

`import-client-vpn-client-certificate-revocation-list`

Il seguente esempio di codice mostra come utilizzare`import-client-vpn-client-certificate-revocation-list`.

AWS CLI

Come importare un elenco di revoche di certificati client

L'esempio `import-client-vpn-client-certificate-revocation-list` seguente importa un elenco di revoche di certificati client nell'endpoint VPN Client specificando la posizione del file sul computer locale.

```
aws ec2 import-client-vpn-client-certificate-revocation-list \
    --certificate-revocation-list file:///path/to/crl.pem \
    --client-vpn-endpoint-id cvpn-endpoint-123456789123abcde
```

Output:

```
{
    "Return": true
}
```

Per ulteriori informazioni, consulta Elenchi di revoche di certificati client nella Guida per l'amministratore di VPN Client di AWS .

- Per i dettagli sull'API, consulta ImportClientVpnClientCertificateRevocationList AWS CLICommand Reference.

`import-image`

Il seguente esempio di codice mostra come utilizzare`import-image`.

AWS CLI

Come importare un file di immagine della VM come AMI

L'esempio `import-image` seguente importa l'OVA specificato.

```
aws ec2 import-image \
  --disk-containers Format=ova,UserBucket="{S3Bucket=my-import-bucket,S3Key=vms/my-server-vm.ova}"
```

Output:

```
{
    "ImportTaskId": "import-ami-1234567890abcdef0",
    "Progress": "2",
    "SnapshotDetails": [
        {
```

```
            "DiskImageSize": 0.0,
            "Format": "ova",
            "UserBucket": {
                "S3Bucket": "my-import-bucket",
                "S3Key": "vms/my-server-vm.ova"
            }
        }
    ],
    "Status": "active",
    "StatusMessage": "pending"
}
```

- Per i dettagli sull'API, consulta ImportImage AWS CLICommand Reference.

`import-key-pair`

Il seguente esempio di codice mostra come utilizzare`import-key-pair`.

AWS CLI

Come importare una chiave pubblica

Innanzitutto, genera una coppia di chiavi con lo strumento desiderato. Ad esempio, utilizza il comando ssh-keygen:

Comando:

```
ssh-keygen -t rsa -C "my-key" -f ~/.ssh/my-key
```

Output:

```
Generating public/private rsa key pair.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/ec2-user/.ssh/my-key.
Your public key has been saved in /home/ec2-user/.ssh/my-key.pub.
...
```

Questo comando di esempio importa la chiave pubblica specificata.

Comando:

```
aws ec2 import-key-pair --key-name "my-key" --public-key-material fileb://~/.ssh/my-key.pub
```

Output:

```
{
  "KeyName": "my-key",
  "KeyFingerprint": "1f:51:ae:28:bf:89:e9:d8:1f:25:5d:37:2d:7d:b8:ca"
}
```

- Per i dettagli sull'API, consulta ImportKeyPair AWS CLICommand Reference.

`import-snapshot`

Il seguente esempio di codice mostra come utilizzare`import-snapshot`.

AWS CLI

Come importare uno snapshot

L'esempio `import-snapshot` seguente importa il disco specificato come snapshot.

```
aws ec2 import-snapshot \
    --description "My server VMDK" \
    --disk-container Format=VMDK,UserBucket={'S3Bucket=my-import-bucket,S3Key=vms/my-server-vm.vmdk'}
```

Output:

```
{
    "Description": "My server VMDK",
    "ImportTaskId": "import-snap-1234567890abcdef0",
    "SnapshotTaskDetail": {
        "Description": "My server VMDK",
        "DiskImageSize": "0.0",
        "Format": "VMDK",
        "Progress": "3",
        "Status": "active",
        "StatusMessage": "pending"
        "UserBucket": {
            "S3Bucket": "my-import-bucket",
```

```
            "S3Key": "vms/my-server-vm.vmdk"
        }
    }
}
```

- Per i dettagli sull'API, consulta ImportSnapshot AWS CLICommand Reference.

`list-images-in-recycle-bin`

Il seguente esempio di codice mostra come utilizzare`list-images-in-recycle-bin`.

AWS CLI

Come elencare le immagini nel Cestino

L'esempio `list-images-in-recycle-bin` seguente elenca tutte le immagini attualmente mantenute nel Cestino.

```
aws ec2 list-images-in-recycle-bin
```

Output:

```
{
    "Images": [
        {
            "RecycleBinEnterTime": "2022-03-14T15:35:08.000Z",
            "Description": "Monthly AMI One",
            "RecycleBinExitTime": "2022-03-15T15:35:08.000Z",
            "Name": "AMI_01",
            "ImageId": "ami-0111222333444abcd"
        }
    ]
}
```

Per ulteriori informazioni, consulta Recover deleted AMIs from the Recycle Bin nella Amazon EBS User Guide.

- Per i dettagli sull'API, consulta AWS CLI Command ListImagesInRecycleBinReference.

`list-snapshots-in-recycle-bin`

Il seguente esempio di codice mostra come utilizzare`list-snapshots-in-recycle-bin`.

AWS CLI

Come visualizzare gli snapshot nel Cestino

L'esempio `list-snapshots-in-recycle-bin` seguente elenca le informazioni sugli snapshot nel Cestino, tra cui l'ID snapshot, una descrizione dello snapshot, l'ID del volume da cui è stato creato lo snapshot, la data e l'ora in cui lo snapshot è stato eliminato ed è entrato nel Cestino e la data e l'ora in cui scade il periodo di conservazione.

```
aws ec2 list-snapshots-in-recycle-bin \
    --snapshot-id snap-01234567890abcdef
```

Output:

```
{
    "SnapshotRecycleBinInfo": [
        {
            "Description": "Monthly data backup snapshot",
            "RecycleBinEnterTime": "2022-12-01T13:00:00.000Z",
            "RecycleBinExitTime": "2022-12-15T13:00:00.000Z",
            "VolumeId": "vol-abcdef09876543210",
            "SnapshotId": "snap-01234567890abcdef"
        }
    ]
}
```

Per ulteriori informazioni sul Cestino, consulta Ripristino degli snapshot dal Cestino nella Guida per l'utente di Amazon EBS.

- Per i dettagli sull'API, consulta ListSnapshotsInRecycleBin AWS CLICommand Reference.

`lock-snapshot`

Il seguente esempio di codice mostra come utilizzare`lock-snapshot`.

AWS CLI

Esempio 1: come bloccare uno snapshot in modalità di governance

L'esempio `lock-snapshot` seguente blocca lo snapshot specificato in modalità di governance.

```
aws ec2 lock-snapshot \
```

```
    --snapshot-id snap-0b5e733b4a8df6e0d \
    --lock-mode governance \
    --lock-duration 365
```

Output:

```
{
    "SnapshotId": "snap-0b5e733b4a8df6e0d",
    "LockState": "governance",
    "LockDuration": 365,
    "LockCreatedOn": "2024-05-05T00:56:06.208000+00:00",
    "LockExpiresOn": "2025-05-05T00:56:06.208000+00:00",
    "LockDurationStartTime": "2024-05-05T00:56:06.208000+00:00"
}
```

Per ulteriori informazioni, consulta Snapshot Lock nella Guida per l'utente di Amazon EBS.

Esempio 2: come bloccare uno snapshot in modalità di conformità

L'esempio `lock-snapshot` seguente blocca lo snapshot specificato in modalità di conformità.

```
aws ec2 lock-snapshot \
    --snapshot-id snap-0163a8524c5b9901f \
    --lock-mode compliance \
    --cool-off-period 24 \
    --lock-duration 365
```

Output:

```
{
    "SnapshotId": "snap-0b5e733b4a8df6e0d",
    "LockState": "compliance-cooloff",
    "LockDuration": 365,
    "CoolOffPeriod": 24,
    "CoolOffPeriodExpiresOn": "2024-05-06T01:02:20.527000+00:00",
    "LockCreatedOn": "2024-05-05T01:02:20.527000+00:00",
    "LockExpiresOn": "2025-05-05T01:02:20.527000+00:00",
    "LockDurationStartTime": "2024-05-05T01:02:20.527000+00:00"
}
```

Per ulteriori informazioni, consulta Snapshot Lock nella Guida per l'utente di Amazon EBS.

- Per i dettagli sull'API, consulta LockSnapshot AWS CLICommand Reference.

`modify-address-attribute`

Il seguente esempio di codice mostra come utilizzare`modify-address-attribute`.

AWS CLI

Come modificare l'attributo del nome di dominio associato a un indirizzo IP elastico

Gli esempi `modify-address-attribute` seguenti modificano l'attributo del nome di dominio di un indirizzo IP elastico.

Linux:

```
aws ec2 modify-address-attribute \
    --allocation-id eipalloc-abcdef01234567890 \
    --domain-name example.com
```

Windows:

```
aws ec2 modify-address-attribute ^
    --allocation-id eipalloc-abcdef01234567890 ^
    --domain-name example.com
```

Output:

```
{
    "Addresses": [
        {
            "PublicIp": "192.0.2.0",
            "AllocationId": "eipalloc-abcdef01234567890",
            "PtrRecord": "example.net."
            "PtrRecordUpdate": {
                "Value": "example.com.",
                "Status": "PENDING"
        }
    ]
}
```

Per monitorare la modifica in sospeso e per visualizzare gli attributi modificati di un indirizzo IP elastico, consulta la AWS CLI describe-addresses-attributeCommand Reference.

- Per i dettagli sull'API, consulta AWS CLI Command ModifyAddressAttributeReference.

`modify-availability-zone-group`

Il seguente esempio di codice mostra come utilizzare`modify-availability-zone-group`.

AWS CLI

Come abilitare un gruppo di zone

L'esempio `modify-availability-zone-group` seguente abilita il gruppo di zone specificato.

```
aws ec2 modify-availability-zone-group \
    --group-name us-west-2-lax-1 \
    --opt-in-status opted-in
```

Output:

```
{
    "Return": true
}
```

Per ulteriori informazioni, consulta Regioni e zone nella Amazon EC2 User Guide.

- Per i dettagli sull'API, consulta ModifyAvailabilityZoneGroup AWS CLICommand Reference.

`modify-capacity-reservation-fleet`

Il seguente esempio di codice mostra come utilizzare`modify-capacity-reservation-fleet`.

AWS CLI

Esempio 1: come modificare la capacità di destinazione totale e la data di un parco di prenotazione della capacità

L'esempio `modify-capacity-reservation-fleet` seguente modifica la capacità target totale del parco di prenotazione della capacità specificato. Quando modifichi la capacità target totale di un parco istanze di prenotazione della capacità, questo crea automaticamente nuove prenotazioni di capacità o modifica o annulla le prenotazioni di capacità esistenti nel parco istanze per soddisfare la nuova capacità target totale. Non è possibile tentare ulteriori modifiche a un parco istanze mentre si trova nello stato `modifying`.

```
aws ec2 modify-capacity-reservation-fleet \
```

```
    --capacity-reservation-fleet-id crf-01234567890abcedf \
    --total-target-capacity 160
```

Output:

```
{
    "Return": true
}
```

Esempio 2: come modificare la data di fine di un parco di prenotazione della capacità

L'esempio `modify-capacity-reservation-fleet` seguente modifica la data di fine del parco di prenotazione della capacità specificato. Quando modifichi la data di fine del parco istanze, le date di fine per tutte le singole prenotazioni di capacità vengono aggiornate di conseguenza. Non è possibile tentare ulteriori modifiche a un parco istanze mentre si trova nello stato `modifying`.

```
aws ec2 modify-capacity-reservation-fleet \
    --capacity-reservation-fleet-id crf-01234567890abcedf \
    --end-date 2022-07-04T23:59:59.000Z
```

Output:

```
{
    "Return": true
}
```

Per ulteriori informazioni sulle flotte di prenotazione della capacità, consulta Flotte di prenotazione della capacità nella Guida per gli EC2 utenti di Amazon.

- Per i dettagli sull'API, consulta AWS CLI Command ModifyCapacityReservationFleetReference.

`modify-capacity-reservation`

Il seguente esempio di codice mostra come utilizzare`modify-capacity-reservation`.

AWS CLI

Esempio 1: come modificare il numero di istanze riservate da una prenotazione della capacità esistente

L'esempio `modify-capacity-reservation` seguente modifica il numero di istanze per cui la prenotazione della capacità riserva la capacità.

```
aws ec2 modify-capacity-reservation \
    --capacity-reservation-id cr-1234abcd56EXAMPLE \
    --instance-count 5
```

Output:

```
{
    "Return": true
}
```

Per ulteriori informazioni, consulta Modificare una prenotazione di capacità nella Amazon EC2 User Guide.

Esempio 2: come modificare la data e l'ora di fine di una prenotazione della capacità esistente

L'esempio `modify-capacity-reservation` seguente modifica una prenotazione della capacità esistente in modo che termini alla data e all'ora specificate.

```
aws ec2 modify-capacity-reservation \
    --capacity-reservation-id cr-1234abcd56EXAMPLE \
    --end-date-type limited \
    --end-date 2019-08-31T23:59:59Z
```

Per ulteriori informazioni, consulta Modificare una prenotazione di capacità nella Amazon EC2 User Guide.

- Per i dettagli sull'API, consulta ModifyCapacityReservation AWS CLICommand Reference.

`modify-client-vpn-endpoint`

Il seguente esempio di codice mostra come utilizzare`modify-client-vpn-endpoint`.

AWS CLI

Come modificare un endpoint VPN Client

L'esempio `modify-client-vpn-endpoint` seguente abilita la registrazione della connessione client per l'endpoint VPN Client specificato.

```
aws ec2 modify-client-vpn-endpoint \
    --client-vpn-endpoint-id cvpn-endpoint-123456789123abcde \
    --connection-log-options Enabled=true,CloudwatchLogGroup=ClientVPNLogs
```

Output:

```
{
    "Return": true
}
```

Per ulteriori informazioni, consulta Endpoint Client VPN nella Guida per l'amministratore di VPN Client di AWS .

- Per i dettagli sull'API, consulta ModifyClientVpnEndpoint AWS CLICommand Reference.

`modify-default-credit-specification`

Il seguente esempio di codice mostra come utilizzare`modify-default-credit-specification`.

AWS CLI

Come modificare l'opzione di credito predefinita

L'esempio `modify-default-credit-specification` seguente modifica l'opzione di credito predefinita per le istanze T2.

```
aws ec2 modify-default-credit-specification \
    --instance-family t2 \
    --cpu-credits unlimited
```

Output:

```
{
    "InstanceFamilyCreditSpecification": {
        "InstanceFamily": "t2",
        "CpuCredits": "unlimited"
    }
}
```

- Per i dettagli sull'API, consulta ModifyDefaultCreditSpecification AWS CLICommand Reference.

`modify-ebs-default-kms-key-id`

Il seguente esempio di codice mostra come utilizzare`modify-ebs-default-kms-key-id`.

AWS CLI

Come impostare la chiave gestita dal cliente (CMK) predefinita per la crittografia EBS

L'`modify-ebs-default-kms-key-id`esempio seguente imposta la CMK specificata come CMK predefinita per la crittografia EBS per l' AWS account nella regione corrente.

```
aws ec2 modify-ebs-default-kms-key-id \
    --kms-key-id alias/my-cmk
```

Output:

```
{
    "KmsKeyId": "arn:aws:kms:us-
west-2:123456789012:key/0ea3fef3-80a7-4778-9d8c-1c0c6EXAMPLE"
}
```

- Per i dettagli sull'API, consulta Command ModifyEbsDefaultKmsKeyIdReference AWS CLI .

`modify-fleet`

Il seguente esempio di codice mostra come utilizzare`modify-fleet`.

AWS CLI

Per scalare una EC2 flotta

L'`modify-fleet`esempio seguente modifica la capacità target della EC2 flotta specificata. Se il valore specificato è maggiore della capacità corrente, EC2 Fleet avvia istanze aggiuntive. Se il valore specificato è inferiore alla capacità corrente, il EC2 parco istanze annulla tutte le richieste aperte e, se la politica di terminazione lo è`terminate`, termina tutte le EC2 istanze che superano la nuova capacità target.

```
aws ec2 modify-fleet \
    --fleet-ids fleet-12a34b55-67cd-8ef9-ba9b-9208dEXAMPLE \
    --target-capacity-specification TotalTargetCapacity=5
```

Output:

```
{
    "Return": true
}
```

Per ulteriori informazioni, consulta Manage an EC2 Fleet nella Amazon EC2 User Guide.

- Per i dettagli sull'API, consulta ModifyFleet AWS CLICommand Reference.

`modify-fpga-image-attribute`

Il seguente esempio di codice mostra come utilizzare`modify-fpga-image-attribute`.

AWS CLI

Come modificare gli attributi di un'immagine FPGA di Amazon (AFI)

Questo esempio aggiunge le autorizzazioni di caricamento per l'ID account `123456789012` per l'AFI specificata.

Comando:

```
aws ec2 modify-fpga-image-attribute --attribute loadPermission --fpga-image-
id afi-0d123e123bfc85abc --load-permission Add=[{UserId=123456789012}]
```

Output:

```
{
  "FpgaImageAttribute": {
      "FpgaImageId": "afi-0d123e123bfc85abc",
      "LoadPermissions": [
          {
              "UserId": "123456789012"
          }
      ]
  }
}
```

- Per i dettagli sull'API, consulta ModifyFpgaImageAttribute AWS CLICommand Reference.

`modify-hosts`

Il seguente esempio di codice mostra come utilizzare`modify-hosts`.

AWS CLI

Esempio 1: come abilitare il posizionamento automatico per un host dedicato

L'esempio `modify-hosts` seguente abilita il posizionamento automatico per un host dedicato in modo che accetti qualsiasi tipo di avvio di istanze non mirate corrispondenti alla relativa configurazione del tipo di istanza.

```
aws ec2 modify-hosts \
    --host-id h-06c2f189b4EXAMPLE \
    --auto-placement on
```

Output:

```
{
    "Successful": [
        "h-06c2f189b4EXAMPLE"
    ],
    "Unsuccessful": []
}
```

Per ulteriori informazioni, consulta Modificare l'impostazione di posizionamento automatico per un host dedicato nella Amazon EC2 User Guide.

Esempio 2: come abilitare il ripristino dell'host per un host dedicato

L'esempio `modify-hosts` seguente abilita il ripristino host per l'host dedicato specificato.

```
aws ec2 modify-hosts \
    --host-id h-06c2f189b4EXAMPLE \
    --host-recovery on
```

Output:

```
{
    "Successful": [
        "h-06c2f189b4EXAMPLE"
```

```
    ],
    "Unsuccessful": []
}
```

Per ulteriori informazioni, consulta Modificare l'impostazione di posizionamento automatico per un host dedicato nella Amazon EC2 User Guide.

- Per i dettagli sull'API, consulta ModifyHosts AWS CLICommand Reference.

`modify-id-format`

Il seguente esempio di codice mostra come utilizzare`modify-id-format`.

AWS CLI

Come abilitare il formato di ID più lungo per una risorsa

L'esempio `modify-id-format` seguente abilita il formato ID più lungo per il tipo di risorsa `instance`.

```
aws ec2 modify-id-format \
    --resource instance \
    --use-long-ids
```

Come disabilitare il formato di ID più lungo per una risorsa

L'esempio `modify-id-format` seguente disabilita il formato ID più lungo per il tipo di risorsa `instance`.

```
aws ec2 modify-id-format \
    --resource instance \
    --no-use-long-ids
```

L'esempio `modify-id-format` seguente abilita il formato ID più lungo per tutti i tipi di risorsa supportati che rientrano nel rispettivo periodo di adesione.

```
aws ec2 modify-id-format \
    --resource all-current \
    --use-long-ids
```

- Per i dettagli sull'API, consulta ModifyIdFormat AWS CLICommand Reference.

`modify-identity-id-format`

Il seguente esempio di codice mostra come utilizzare`modify-identity-id-format`.

AWS CLI

Per consentire a un ruolo IAM di utilizzare più a IDs lungo una risorsa

L'`modify-identity-id-format`esempio seguente consente al ruolo IAM `EC2Role` nel tuo AWS account di utilizzare il formato ID lungo per il tipo di `instance` risorsa.

```
aws ec2 modify-identity-id-format \
    --principal-arn arn:aws:iam::123456789012:role/EC2Role \
    --resource instance \
    --use-long-ids
```

Per consentire a un utente IAM di utilizzare più a IDs lungo una risorsa

L'`modify-identity-id-format`esempio seguente consente all'utente `AdminUser` IAM del tuo AWS account di utilizzare il formato ID più lungo per il tipo di `volume` risorsa.

```
aws ec2 modify-identity-id-format \
    --principal-arn arn:aws:iam::123456789012:user/AdminUser \
    --resource volume \
    --use-long-ids
```

L'`modify-identity-id-format`esempio seguente consente all'utente `AdminUser` IAM del tuo AWS account di utilizzare il formato ID più lungo per tutti i tipi di risorse supportati che rientrano nel periodo di attivazione.

```
aws ec2 modify-identity-id-format \
    --principal-arn arn:aws:iam::123456789012:user/AdminUser \
    --resource all-current \
    --use-long-ids
```

- Per i dettagli sull'API, consulta ModifyIdentityIdFormat AWS CLICommand Reference.

`modify-image-attribute`

Il seguente esempio di codice mostra come utilizzare`modify-image-attribute`.

AWS CLI

Esempio 1: come rendere pubblica un'AMI

L'esempio `modify-instance-attribute` seguente rende pubblica l'AMI specificata.

```
aws ec2 modify-image-attribute \
    --image-id ami-5731123e \
    --launch-permission "Add=[{Group=all}]"
```

Questo comando non produce alcun output.

Esempio 2: come rendere privata un'AMI

L'esempio `modify-instance-attribute` seguente rende privata l'AMI specificata.

```
aws ec2 modify-image-attribute \
    --image-id ami-5731123e \
    --launch-permission "Remove=[{Group=all}]"
```

Questo comando non produce alcun output.

Esempio 3: concedere l'autorizzazione di avvio a un AWS account

L'`modify-instance-attribute`esempio seguente concede le autorizzazioni di avvio all'account specificato AWS .

```
aws ec2 modify-image-attribute \
    --image-id ami-5731123e \
    --launch-permission "Add=[{UserId=123456789012}]"
```

Questo comando non produce alcun output.

Esempio 4: Per rimuovere l'autorizzazione di avvio da un account AWS

L'`modify-instance-attribute`esempio seguente rimuove le autorizzazioni di avvio dall' AWS account specificato.

```
aws ec2 modify-image-attribute \
    --image-id ami-5731123e \
```

```
    --launch-permission "Remove=[{UserId=123456789012}]"
```

- Per i dettagli sull'API, vedere ModifyImageAttributein AWS CLI Command Reference.

`modify-instance-attribute`

Il seguente esempio di codice mostra come utilizzare`modify-instance-attribute`.

AWS CLI

Esempio 1: come modificare il tipo di istanza

L'esempio `modify-instance-attribute` seguente modifica il tipo di istanza dell'istanza specificata. L'istanza deve essere nello stato `stopped`.

```
aws ec2 modify-instance-attribute \
    --instance-id i-1234567890abcdef0 \
    --instance-type "{\"Value\": \"m1.small\"}"
```

Questo comando non produce alcun output.

Esempio 2: come abilitare le reti avanzate per un'istanza

L'esempio `modify-instance-attribute` seguente abilita le reti avanzate per l'istanza specificata. L'istanza deve essere nello stato `stopped`.

```
aws ec2 modify-instance-attribute \
    --instance-id i-1234567890abcdef0 \
    --sriov-net-support simple
```

Questo comando non produce alcun output.

Esempio 3: Per modificare l' sourceDestCheck attributo

L'esempio `modify-instance-attribute` seguente imposta l'attributo `sourceDestCheck` dell'istanza specificata su `true`. L'istanza deve trovarsi in un VPC.

```
aws ec2 modify-instance-attribute --instance-id i-1234567890abcdef0 --source-dest-
check "{\"Value\": true}"
```

Questo comando non produce alcun output.

Esempio 4: modificare l' deleteOnTermination attributo del volume principale

L'esempio `modify-instance-attribute` seguente imposta l'attributo `deleteOnTermination` per il volume root dell'istanza supportata da Amazon EBS specificata su `false`. Per impostazione predefinita, questo attributo è `true` per il volume root.

Comando:

```
aws ec2 modify-instance-attribute \
  --instance-id i-1234567890abcdef0 \
  --block-device-mappings "[{\"DeviceName\": \"/dev/sda1\",\"Ebs\":
{\"DeleteOnTermination\":false}}]"
```

Questo comando non produce alcun output.

Esempio 5: come modificare i dati utente associati a un'istanza

L'`modify-instance-attribute`esempio seguente aggiunge il contenuto del file `UserData.txt` come UserData per l'istanza specificata.

Contenuto del file `UserData.txt` originale:

```
#!/bin/bash
yum update -y
service httpd start
chkconfig httpd on
```

I contenuti del file devono essere codificati in formato Base64. Il primo comando converte il file di testo in formato Base64 e lo salva come un nuovo file.

Versione Linux/macOS del comando:

```
base64 UserData.txt > UserData.base64.txt
```

Questo comando non produce alcun output.

Versione Windows del comando:

```
certutil -encode UserData.txt tmp.b64 && findstr /v /c:- tmp.b64 >
 UserData.base64.txt
```

Output:

```
Input Length = 67
Output Length = 152
CertUtil: -encode command completed successfully.
```

A questo punto è possibile fare riferimento al file nel comando della CLI che segue:

```
aws ec2 modify-instance-attribute \
    --instance-id=i-09b5a14dbca622e76 \
    --attribute userData --value file://UserData.base64.txt
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta User Data and the AWS CLI nella Guida per l'EC2 utente.

- Per i dettagli sull'API, consulta ModifyInstanceAttribute AWS CLICommand Reference.

`modify-instance-capacity-reservation-attributes`

Il seguente esempio di codice mostra come utilizzare`modify-instance-capacity-reservation-attributes`.

AWS CLI

Esempio 1: come modificare le impostazioni di targeting per la prenotazione della capacità di un'istanza

L'esempio `modify-instance-capacity-reservation-attributes` seguente modifica un'istanza per fare riferimento a una prenotazione della capacità specifica.

```
aws ec2 modify-instance-capacity-reservation-attributes \
    --instance-id i-EXAMPLE8765abcd4e \
    --capacity-reservation-specification
 'CapacityReservationTarget={CapacityReservationId= cr-1234abcd56EXAMPLE }'
```

Output:

```
{
    "Return": true
}
```

Per ulteriori informazioni, consulta Modificare le impostazioni di prenotazione della capacità della tua istanza nella Amazon EC2 User Guide.

Esempio 2: come modificare le impostazioni di targeting per la prenotazione della capacità di un'istanza

L'esempio `modify-instance-capacity-reservation-attributes` seguente modifica un'istanza interrotta che ha come obiettivo la prenotazione della capacità specificata per l'avvio in qualsiasi prenotazione della capacità che abbia attributi corrispondenti (tipo di istanza, piattaforma, zona di disponibilità) e che abbia criteri di corrispondenza dell'istanza aperti.

```
aws ec2 modify-instance-capacity-reservation-attributes \
    --instance-id i-EXAMPLE8765abcd4e \
    --capacity-reservation-specification 'CapacityReservationPreference=open'
```

Output:

```
{
    "Return": true
}
```

Per ulteriori informazioni, consulta Modificare le impostazioni di prenotazione della capacità della tua istanza nella Amazon EC2 User Guide.

- Per i dettagli sull'API, consulta ModifyInstanceCapacityReservationAttributes AWS CLICommand Reference.

`modify-instance-credit-specification`

Il seguente esempio di codice mostra come utilizzare`modify-instance-credit-specification`.

AWS CLI

Come modificare l'opzione di credito per l'utilizzo della CPU di un'istanza

Questo esempio modifica l'opzione di credito per l'utilizzo della CPU dell'istanza specificata nella Regione specificata impostandola su "unlimited" (Illimitato). Le opzioni di credito valide sono "standard" e "unlimited".

Comando:

```
aws ec2 modify-instance-credit-specification --instance-credit-
specification "InstanceId=i-1234567890abcdef0,CpuCredits=unlimited"
```

Output:

```
{
  "SuccessfulInstanceCreditSpecifications": [
    {
      "InstanceId": "i-1234567890abcdef0"
    }
  ],
  "UnsuccessfulInstanceCreditSpecifications": []
}
```

- Per i dettagli sull'API, consulta ModifyInstanceCreditSpecification AWS CLICommand Reference.

`modify-instance-event-start-time`

Il seguente esempio di codice mostra come utilizzare`modify-instance-event-start-time`.

AWS CLI

Come modificare l'ora di inizio dell'evento per un'istanza

Il comando `modify-instance-event-start-time` seguente mostra come modificare l'ora di inizio dell'evento per l'istanza specificata. Specifica l'ID evento utilizzando il parametro `--instance-event-id`. Specifica la nuova data e ora utilizzando il parametro `--not-before`.

```
aws ec2 modify-instance-event-start-time --instance-id i-1234567890abcdef0
 --instance-event-id instance-event-0abcdef1234567890 --not-
before 2019-03-25T10:00:00.000
```

Output:

```
"Event": {
    "InstanceEventId": "instance-event-0abcdef1234567890",
    "Code": "system-reboot",
    "Description": "scheduled reboot",
    "NotAfter": "2019-03-25T12:00:00.000Z",
    "NotBefore": "2019-03-25T10:00:00.000Z",
```

```
    "NotBeforeDeadline": "2019-04-22T21:00:00.000Z"
}
```

Per ulteriori informazioni, consulta Uso delle istanze per le quali è pianificato il riavvio nella Guida per l'utente di Amazon Elastic Compute Cloud.

- Per i dettagli sull'API, consulta ModifyInstanceEventStartTime AWS CLICommand Reference.

`modify-instance-event-window`

Il seguente esempio di codice mostra come utilizzare`modify-instance-event-window`.

AWS CLI

Esempio 1: come modificare l'intervallo temporale di una finestra di eventi

L'esempio `modify-instance-event-window` seguente modifica l'intervallo temporale di una finestra di eventi. Specifica il parametro `time-range` per modificare l'intervallo di tempo. Non è possibile specificare anche il parametro `cron-expression`.

```
aws ec2 modify-instance-event-window \
    --region us-east-1 \
    --instance-event-window-id iew-0abcdef1234567890
    --time-range StartWeekDay=monday,StartHour=2,EndWeekDay=wednesday,EndHour=8
```

Output:

```
{
    "InstanceEventWindow": {
        "InstanceEventWindowId": "iew-0abcdef1234567890",
        "TimeRanges": [
            {
                "StartWeekDay": "monday",
                "StartHour": 2,
                "EndWeekDay": "wednesday",
                "EndHour": 8
            }
        ],
        "Name": "myEventWindowName",
        "AssociationTarget": {
            "InstanceIds": [
                "i-0abcdef1234567890",
```

```
                "i-0be35f9acb8ba01f0"
            ],
            "Tags": [],
            "DedicatedHostIds": []
        },
        "State": "creating",
        "Tags": [
            {
                "Key": "K1",
                "Value": "V1"
            }
        ]
    }
}
```

Per i vincoli relativi alla finestra degli eventi, consulta Considerazioni nella sezione Eventi pianificati della Amazon EC2 User Guide.

Esempio 2: come modificare un insieme di intervalli temporali di una finestra di eventi

L'esempio `modify-instance-event-window` seguente modifica l'intervallo temporale di una finestra di eventi. Specifica il parametro `time-range` per modificare l'intervallo di tempo. Non è possibile specificare anche il parametro `cron-expression`.

```
aws ec2 modify-instance-event-window \
    --region us-east-1 \
    --instance-event-window-id iew-0abcdef1234567890 \
    --time-range '[{"StartWeekDay": "monday", "StartHour": 2, "EndWeekDay":
 "wednesday", "EndHour": 8},
        {"StartWeekDay": "thursday", "StartHour": 2, "EndWeekDay": "friday",
 "EndHour": 8}]'
```

Output:

```
{
    "InstanceEventWindow": {
        "InstanceEventWindowId": "iew-0abcdef1234567890",
        "TimeRanges": [
            {
                "StartWeekDay": "monday",
                "StartHour": 2,
                "EndWeekDay": "wednesday",
```

```
                "EndHour": 8
            },
            {
                "StartWeekDay": "thursday",
                "StartHour": 2,
                "EndWeekDay": "friday",
                "EndHour": 8
            }
        ],
        "Name": "myEventWindowName",
        "AssociationTarget": {
            "InstanceIds": [
                "i-0abcdef1234567890",
                "i-0be35f9acb8ba01f0"
            ],
            "Tags": [],
            "DedicatedHostIds": []
        },
        "State": "creating",
        "Tags": [
            {
                "Key": "K1",
                "Value": "V1"
            }
        ]
    }
}
```

Per i vincoli relativi alla finestra degli eventi, consulta Considerazioni nella sezione Eventi pianificati della Amazon EC2 User Guide.

Esempio 3: come modificare l'espressione cron di una finestra di eventi

L'esempio `modify-instance-event-window` seguente modifica l'espressione cron di una finestra di eventi. Specifica il parametro `cron-expression` per modificare l'espressione cron. Non è possibile specificare anche il parametro `time-range`.

```
aws ec2 modify-instance-event-window \
    --region us-east-1 \
    --instance-event-window-id iew-0abcdef1234567890 \
    --cron-expression "* 21-23 * * 2,3"
```

Output:

```
{
    "InstanceEventWindow": {
        "InstanceEventWindowId": "iew-0abcdef1234567890",
        "Name": "myEventWindowName",
        "CronExpression": "* 21-23 * * 2,3",
        "AssociationTarget": {
            "InstanceIds": [
                "i-0abcdef1234567890",
                "i-0be35f9acb8ba01f0"
            ],
            "Tags": [],
            "DedicatedHostIds": []
        },
        "State": "creating",
        "Tags": [
            {
                "Key": "K1",
                "Value": "V1"
            }
        ]
    }
}
```

Per i vincoli relativi alla finestra degli eventi, consulta Considerazioni nella sezione Eventi pianificati della Amazon EC2 User Guide.

- Per i dettagli sull'API, consulta Command ModifyInstanceEventWindowReference AWS CLI .

`modify-instance-maintenance-options`

Il seguente esempio di codice mostra come utilizzare`modify-instance-maintenance-options`.

AWS CLI

Esempio 1: come disabilitare il comportamento di ripristino di un'istanza

L'esempio `modify-instance-maintenance-options` seguente disabilita il recupero automatico semplificato per un'istanza in esecuzione o arrestata.

```
aws ec2 modify-instance-maintenance-options \
    --instance-id i-0abcdef1234567890 \
    --auto-recovery disabled
```

Output:

```
{
    "InstanceId": "i-0abcdef1234567890",
    "AutoRecovery": "disabled"
}
```

Per ulteriori informazioni, consulta Configurare il ripristino automatico semplificato nella Amazon EC2 User Guide.

Esempio 2: come impostare il comportamento di ripristino di un'istanza come predefinito

L'esempio `modify-instance-maintenance-options` seguente imposta il comportamento di ripristino automatico come predefinito, il che consente il ripristino automatico semplificato per i tipi di istanza supportati.

```
aws ec2 modify-instance-maintenance-options \
    --instance-id i-0abcdef1234567890 \
    --auto-recovery default
```

Output:

```
{
    "InstanceId": "i-0abcdef1234567890",
    "AutoRecovery": "default"
}
```

Per ulteriori informazioni, consulta Configurare il ripristino automatico semplificato nella Amazon EC2 User Guide.

- Per i dettagli sull'API, consulta ModifyInstanceMaintenanceOptions AWS CLICommand Reference.

`modify-instance-metadata-options`

Il seguente esempio di codice mostra come utilizzare`modify-instance-metadata-options`.

AWS CLI

Esempio 1: abilitare IMDSv2

L'`modify-instance-metadata-options`esempio seguente configura l'uso di IMDSv2 sull'istanza specificata.

```
aws ec2 modify-instance-metadata-options \
    --instance-id i-1234567898abcdef0 \
    --http-tokens required \
    --http-endpoint enabled
```

Output:

```
{
    "InstanceId": "i-1234567898abcdef0",
    "InstanceMetadataOptions": {
        "State": "pending",
        "HttpTokens": "required",
        "HttpPutResponseHopLimit": 1,
        "HttpEndpoint": "enabled"
    }
}
```

Per ulteriori informazioni, consulta i metadati dell'istanza nella Amazon EC2 User Guide.

Esempio 2: come disabilitare i metadati dell'istanza

L'esempio `modify-instance-metadata-options` seguente disabilita l'uso di tutte le versioni dei metadati dell'istanza sull'istanza specificata.

```
aws ec2 modify-instance-metadata-options \
    --instance-id i-1234567898abcdef0 \
    --http-endpoint disabled
```

Output:

```
{
    "InstanceId": "i-1234567898abcdef0",
    "InstanceMetadataOptions": {
        "State": "pending",
        "HttpTokens": "required",
        "HttpPutResponseHopLimit": 1,
        "HttpEndpoint": "disabled"
    }
```

```
}
```

Per ulteriori informazioni, consulta i metadati dell'istanza nella Amazon EC2 User Guide.

Esempio 3: per abilitare l' IPv6 endpoint dei metadati dell'istanza per la tua istanza

L'`modify-instance-metadata-options`esempio seguente mostra come attivare l' IPv6 endpoint per il servizio di metadati dell'istanza. Per impostazione predefinita, l' IPv6 endpoint è disabilitato. Questo è vero anche se hai avviato un'istanza in una sottorete IPv6 solo. L' IPv6 endpoint per IMDS è accessibile solo sulle istanze create sul sistema Nitro.

```
aws ec2 modify-instance-metadata-options \
    --instance-id i-1234567898abcdef0 \
    --http-protocol-ipv6 enabled \
    --http-endpoint enabled
```

Output:

```
{
    "InstanceId": "i-1234567898abcdef0",
    "InstanceMetadataOptions": {
        "State": "pending",
        "HttpTokens": "required",
        "HttpPutResponseHopLimit": 1,
        "HttpEndpoint": "enabled",
        HttpProtocolIpv6": "enabled"
    }
}
```

Per ulteriori informazioni, consulta i metadati dell'istanza nella Amazon EC2 User Guide.

- Per i dettagli sull'API, consulta ModifyInstanceMetadataOptions AWS CLICommand Reference.

`modify-instance-placement`

Il seguente esempio di codice mostra come utilizzare`modify-instance-placement`.

AWS CLI

Esempio 1: come rimuovere l'affinità di un'istanza con un host dedicato

L'esempio `modify-instance-placement` seguente rimuove l'affinità di un'istanza con un host dedicato e consente di avviarla su qualsiasi host dedicato disponibile nell'account che supporti il relativo tipo di istanza.

```
aws ec2 modify-instance-placement \
    --instance-id i-0e6ddf6187EXAMPLE \
    --affinity default
```

Output:

```
{
    "Return": true
}
```

Esempio 2: come stabilire l'affinità tra un'istanza e l'host dedicato specificato

L'esempio `modify-instance-placement` seguente stabilisce la relazione di avvio tra un'istanza e un host dedicato. L'istanza può essere eseguita solo sull'host dedicato specificato.

```
aws ec2 modify-instance-placement \
    --instance-id i-0e6ddf6187EXAMPLE \
    --affinity host \
    --host-id i-0e6ddf6187EXAMPLE
```

Output:

```
{
    "Return": true
}
```

Esempio 3: come spostare un'istanza in un gruppo di posizionamento

L'esempio `modify-instance-placement` seguente sposta un'istanza in un gruppo di posizionamento, interrompe l'istanza, modifica il posizionamento dell'istanza e quindi riavvia l'istanza.

```
aws ec2 stop-instances \
    --instance-ids i-0123a456700123456

aws ec2 modify-instance-placement \
    --instance-id i-0123a456700123456 \
```

```
    --group-name MySpreadGroup

aws ec2 start-instances \
    --instance-ids i-0123a456700123456
```

Esempio 4: come rimuovere un'istanza da un gruppo di posizionamento

L'esempio `modify-instance-placement` seguente rimuove un'istanza da un gruppo di posizionamento arrestando l'istanza, modificando il posizionamento dell'istanza e quindi riavviando l'istanza. L'esempio seguente specifica una stringa vuota ("") per il nome del gruppo di posizionamento per indicare che l'istanza non deve essere collocata in un gruppo di posizionamento.

Arresta l'istanza:

```
aws ec2 stop-instances \
    --instance-ids i-0123a456700123456
```

Modifica il posizionamento (prompt dei comandi di Windows):

```
aws ec2 modify-instance-placement \
    --instance-id i-0123a456700123456 \
    --group-name ""
```

Modifica la posizione (Windows PowerShell, Linux e macOS):

```
aws ec2 modify-instance-placement `
    --instance-id i-0123a456700123456 `
    --group-name ''
```

Riavvia l'istanza:

```
aws ec2 start-instances \
    --instance-ids i-0123a456700123456
```

Output:

```
{
    "Return": true
}
```

Per ulteriori informazioni, consulta Modify Dedicated Host tenancy and affinity nella Amazon EC2 User Guide.

- Per i dettagli sull'API, consulta AWS CLI Command ModifyInstancePlacementReference.

`modify-ipam-pool`

Il seguente esempio di codice mostra come utilizzare`modify-ipam-pool`.

AWS CLI

Come modificare un pool IPAM

L'esempio `modify-ipam-pool` seguente modifica un pool IPAM.

Linux:

```
aws ec2 modify-ipam-pool \
    --ipam-pool-id ipam-pool-0533048da7d823723 \
    --add-allocation-resource-tags "Key=Owner,Value=Build Team" \
    --clear-allocation-default-netmask-length \
    --allocation-min-netmask-length 14
```

Windows:

```
aws ec2 modify-ipam-pool ^
    --ipam-pool-id ipam-pool-0533048da7d823723 ^
    --add-allocation-resource-tags "Key=Owner,Value=Build Team" ^
    --clear-allocation-default-netmask-length ^
    --allocation-min-netmask-length 14
```

Output:

```
{
    "IpamPool": {
        "OwnerId": "123456789012",
        "IpamPoolId": "ipam-pool-0533048da7d823723",
        "IpamPoolArn": "arn:aws:ec2::123456789012:ipam-pool/ipam-
pool-0533048da7d823723",
        "IpamScopeArn": "arn:aws:ec2::123456789012:ipam-scope/ipam-
scope-02fc38cd4c48e7d38",
```

```
        "IpamScopeType": "private",
        "IpamArn": "arn:aws:ec2::123456789012:ipam/ipam-08440e7a3acde3908",
        "IpamRegion": "us-east-1",
        "Locale": "None",
        "PoolDepth": 1,
        "State": "modify-complete",
        "AutoImport": true,
        "AddressFamily": "ipv4",
        "AllocationMinNetmaskLength": 14,
        "AllocationMaxNetmaskLength": 26,
        "AllocationResourceTags": [
            {
                "Key": "Environment",
                "Value": "Preprod"
            },
            {
                "Key": "Owner",
                "Value": "Build Team"
            }
        ]
    }
}
```

Per ulteriori informazioni, consulta Modifica di un pool nella Guida per l'utente IPAM di Amazon VPC.

- Per i dettagli sull'API, consulta ModifyIpamPool AWS CLICommand Reference.

`modify-ipam-resource-cidr`

Il seguente esempio di codice mostra come utilizzare`modify-ipam-resource-cidr`.

AWS CLI

Come modificare il CIDR assegnato a una risorsa

L'esempio `modify-ipam-resource-cidr` seguente modifica una risorsa CIDR.

Linux:

```
aws ec2 modify-ipam-resource-cidr \
    --current-ipam-scope-id ipam-scope-02fc38cd4c48e7d38 \
    --destination-ipam-scope-id ipam-scope-0da34c61fd189a141 \
```

```
    --resource-id vpc-010e1791024eb0af9 \
    --resource-cidr 10.0.1.0/24 \
    --resource-region us-east-1 \
    --monitored
```

Windows:

```
aws ec2 modify-ipam-resource-cidr ^
    --current-ipam-scope-id ipam-scope-02fc38cd4c48e7d38 ^
    --destination-ipam-scope-id ipam-scope-0da34c61fd189a141 ^
    --resource-id vpc-010e1791024eb0af9 ^
    --resource-cidr 10.0.1.0/24 ^
    --resource-region us-east-1 ^
    --monitored
```

Output:

```
{
    "IpamResourceCidr": {
        "IpamId": "ipam-08440e7a3acde3908",
        "IpamScopeId": "ipam-scope-0da34c61fd189a141",
        "IpamPoolId": "ipam-pool-0533048da7d823723",
        "ResourceRegion": "us-east-1",
        "ResourceOwnerId": "123456789012",
        "ResourceId": "vpc-010e1791024eb0af9",
        "ResourceCidr": "10.0.1.0/24",
        "ResourceType": "vpc",
        "ResourceTags": [
            {
                "Key": "Environment",
                "Value": "Preprod"
            },
            {
                "Key": "Owner",
                "Value": "Build Team"
            }
        ],
        "IpUsage": 0.0,
        "ComplianceStatus": "noncompliant",
        "ManagementState": "managed",
        "OverlapStatus": "overlapping",
        "VpcId": "vpc-010e1791024eb0af9"
    }
```

```
}
```

Per ulteriori informazioni sullo spostamento delle risorse, consulta Move resource CIDRs between scope nella Amazon VPC IPAM User Guide.

Per ulteriori informazioni sulla modifica degli stati di monitoraggio, consulta Modifica dello stato di monitoraggio della risorsa CIDRs nella Guida per l'utente IPAM di Amazon VPC.

- Per i dettagli sull'API, consulta AWS CLI Command ModifyIpamResourceCidrReference.

`modify-ipam-resource-discovery`

Il seguente esempio di codice mostra come utilizzare`modify-ipam-resource-discovery`.

AWS CLI

Come modificare le regioni operative di un rilevamento delle risorse

In questo esempio, l'utente è un amministratore delegato IPAM che desidera modificare le Regioni operative di un rilevamento di risorse.

Per completare questa richiesta:

Non è possibile modificare un rilevamento di risorse predefinito ed è necessario essere il proprietario del rilevamento delle risorse. È necessario l'ID di rilevamento delle risorse, che è possibile utilizzare. describe-ipam-resource-discoveries

L'`modify-ipam-resource-discovery`esempio seguente modifica un rilevamento di risorse non predefinito nel tuo account. AWS

```
aws ec2 modify-ipam-resource-discovery \
    --ipam-resource-discovery-id ipam-res-disco-0f4ef577a9f37a162 \
    --add-operating-regions RegionName='us-west-1' \
    --remove-operating-regions RegionName='us-east-2' \
    --region us-east-1
```

Output:

```
{
    "IpamResourceDiscovery": {
        "OwnerId": "149977607591",
        "IpamResourceDiscoveryId": "ipam-res-disco-0365d2977fc1672fe",
```

```
        "IpamResourceDiscoveryArn": "arn:aws:ec2::149977607591:ipam-resource-
discovery/ipam-res-disco-0365d2977fc1672fe",
        "IpamResourceDiscoveryRegion": "us-east-1",
        "Description": "Example",
        "OperatingRegions": [
            {
                "RegionName": "us-east-1"
            },
            {
                "RegionName": "us-west-1"
            }
        ],
        "IsDefault": false,
        "State": "modify-in-progress"
    }
}
```

Per ulteriori informazioni, consulta Lavorare con il rilevamento delle risorse nella Guida per l'utente IPAM di Amazon VPC.

- Per i dettagli sull'API, consulta AWS CLI Command ModifyIpamResourceDiscoveryReference.

`modify-ipam-scope`

Il seguente esempio di codice mostra come utilizzare`modify-ipam-scope`.

AWS CLI

Come modificare la descrizione di un ambito

In questo scenario, l'utente è un amministratore delegato IPAM che desidera modificare la descrizione di un ambito IPAM.

Per completare questa richiesta, è necessario l'ID dell'ambito, che è possibile utilizzare describe-ipam-scopes.

L'esempio `modify-ipam-scope` seguente mostra come aggiornare la descrizione dell'ambito.

```
aws ec2 modify-ipam-scope \
    --ipam-scope-id ipam-scope-0d3539a30b57dcdd1 \
    --description example \
    --region us-east-1
```

Output:

```
{
"IpamScope": {
        "OwnerId": "320805250157",
        "IpamScopeId": "ipam-scope-0d3539a30b57dcdd1",
        "IpamScopeArn": "arn:aws:ec2::320805250157:ipam-scope/ipam-
scope-0d3539a30b57dcdd1",
        "IpamArn": "arn:aws:ec2::320805250157:ipam/ipam-005f921c17ebd5107",
        "IpamRegion": "us-east-1",
        "IpamScopeType": "public",
        "IsDefault": true,
        "Description": "example",
        "PoolCount": 1,
        "State": "modify-in-progress"
    }
}
```

Per ulteriori informazioni sugli ambiti, consulta Funzionamento di IPAM nella Guida per l'utente IPAM di Amazon VPC.

- Per i dettagli sull'API, consulta ModifyIpamScope AWS CLICommand Reference.

`modify-ipam`

Il seguente esempio di codice mostra come utilizzare`modify-ipam`.

AWS CLI

Come modificare un IPAM

L'esempio `modify-ipam` seguente modifica un'IPAM aggiungendo una Regione operativa.

Linux:

```
aws ec2 modify-ipam \
    --ipam-id ipam-08440e7a3acde3908 \
    --add-operating-regions RegionName=us-west-2
```

Windows:

```
aws ec2 modify-ipam ^
```

```
    --ipam-id ipam-08440e7a3acde3908 ^
    --add-operating-regions RegionName=us-west-2
```

Output:

```
{
    "Ipam": {
        "OwnerId": "123456789012",
        "IpamId": "ipam-08440e7a3acde3908",
        "IpamArn": "arn:aws:ec2::123456789012:ipam/ipam-08440e7a3acde3908",
        "IpamRegion": "us-east-1",
        "PublicDefaultScopeId": "ipam-scope-0b9eed026396dbc16",
        "PrivateDefaultScopeId": "ipam-scope-02fc38cd4c48e7d38",
        "ScopeCount": 3,
        "OperatingRegions": [
            {
                "RegionName": "us-east-1"
            },
            {
                "RegionName": "us-east-2"
            },
            {
                "RegionName": "us-west-1"
            },
            {
                "RegionName": "us-west-2"
            }
        ],
        "State": "modify-in-progress"
    }
}
```

- Per i dettagli sull'API, consulta ModifyIpam AWS CLICommand Reference.

`modify-launch-template`

Il seguente esempio di codice mostra come utilizzare`modify-launch-template`.

AWS CLI

Come modificare la versione del modello di avvio predefinita

Questo esempio specifica la versione 2 del modello di avvio specificato come versione predefinita.

Comando:

```
aws ec2 modify-launch-template --launch-template-id lt-0abcd290751193123 --default-version 2
```

Output:

```
{
  "LaunchTemplate": {
      "LatestVersionNumber": 2,
      "LaunchTemplateId": "lt-0abcd290751193123",
      "LaunchTemplateName": "WebServers",
      "DefaultVersionNumber": 2,
      "CreatedBy": "arn:aws:iam::123456789012:root",
      "CreateTime": "2017-12-01T13:35:46.000Z"
  }
}
```

- Per i dettagli sull'API, consulta ModifyLaunchTemplate AWS CLICommand Reference.

`modify-managed-prefix-list`

Il seguente esempio di codice mostra come utilizzare`modify-managed-prefix-list`.

AWS CLI

Come modificare un elenco di prefissi

L'esempio `modify-managed-prefix-list` seguente aggiunge una voce all'elenco di prefissi specificato.

```
aws ec2 modify-managed-prefix-list \
    --prefix-list-id pl-0123456abcabcabc1 \
    --add-entries Cidr=10.1.0.0/16,Description=vpc-c \
    --current-version 1
```

Output:

```
{
    "PrefixList": {
        "PrefixListId": "pl-0123456abcabcabc1",
```

```
        "AddressFamily": "IPv4",
        "State": "modify-in-progress",
        "PrefixListArn": "arn:aws:ec2:us-west-2:123456789012:prefix-list/
pl-0123456abcabcabc1",
        "PrefixListName": "vpc-cidrs",
        "MaxEntries": 10,
        "Version": 1,
        "OwnerId": "123456789012"
    }
}
```

Per ulteriori informazioni, consulta Elenchi di prefissi gestiti nella Guida per l'utente di Amazon VPC.

- Per i dettagli sull'API, consulta ModifyManagedPrefixList AWS CLICommand Reference.

`modify-network-interface-attribute`

Il seguente esempio di codice mostra come utilizzare`modify-network-interface-attribute`.

AWS CLI

Come modificare l'attributo attachment di un'interfaccia di rete

Questo comando di esempio modifica l'attributo `attachment` dell'interfaccia di rete specificata.

Comando:

```
aws ec2 modify-network-interface-attribute --network-interface-id eni-686ea200 --
attachment AttachmentId=eni-attach-43348162,DeleteOnTermination=false
```

Come modificare l'attributo description di un'interfaccia di rete

Questo comando di esempio modifica l'attributo `description` dell'interfaccia di rete specificata.

Comando:

```
aws ec2 modify-network-interface-attribute --network-interface-id eni-686ea200 --
description "My description"
```

Come modificare l'attributo groupSet di un'interfaccia di rete

Questo comando di esempio modifica l'attributo `groupSet` dell'interfaccia di rete specificata.

Comando:

```
aws ec2 modify-network-interface-attribute --network-interface-id eni-686ea200 --
groups sg-903004f8 sg-1a2b3c4d
```

Per modificare l' sourceDestCheck attributo di un'interfaccia di rete

Questo comando di esempio modifica l'attributo `sourceDestCheck` dell'interfaccia di rete specificata.

Comando:

```
aws ec2 modify-network-interface-attribute --network-interface-id eni-686ea200 --no-
source-dest-check
```

- Per i dettagli sull'API, vedere ModifyNetworkInterfaceAttributein AWS CLI Command Reference.

`modify-private-dns-name-options`

Il seguente esempio di codice mostra come utilizzare`modify-private-dns-name-options`.

AWS CLI

Come modificare le opzioni per i nomi host delle istanze

L'esempio `modify-private-dns-name-options` seguente disabilita l'opzione per rispondere alle query DNS, per nomi host delle istanze con record DNS A.

```
aws ec2 modify-private-dns-name-options \
    --instance-id i-1234567890abcdef0 \
    --no-enable-resource-name-dns-a-record
```

Output:

```
{
    "Return": true
}
```

Per ulteriori informazioni, consulta i tipi di hostname delle EC2 istanze Amazon nella Amazon EC2 User Guide.

- Per i dettagli sull'API, consulta ModifyPrivateDnsNameOptions AWS CLICommand Reference.

`modify-reserved-instances`

Il seguente esempio di codice mostra come utilizzare`modify-reserved-instances`.

AWS CLI

Come modificare le istanze riservate

Questo comando di esempio sposta un'istanza riservata in un'altra zona di disponibilità nella stessa Regione.

Comando:

```
aws ec2 modify-reserved-instances --reserved-instances-ids b847fa93-e282-4f55-b59a-1342f5bd7c02 --target-configurations AvailabilityZone=us-west-1c,Platform=EC2-Classic,InstanceCount=10
```

Output:

```
{
  "ReservedInstancesModificationId": "rimod-d3ed4335-b1d3-4de6-ab31-0f13aaf46687"
}
```

Come modificare la piattaforma di rete delle istanze riservate

Questo comando di esempio converte EC2 -Classic Reserved Instances in -VPC. EC2

Comando:

```
aws ec2 modify-reserved-instances --reserved-instances-ids f127bd27-edb7-44c9-a0eb-0d7e09259af0 --target-configurations AvailabilityZone=us-west-1c,Platform=EC2-VPC,InstanceCount=5
```

Output:

```
{
  "ReservedInstancesModificationId": "rimod-82fa9020-668f-4fb6-945d-61537009d291"
}
```

Per ulteriori informazioni, consulta Modifying Your Reserved Instances nella Amazon EC2 User Guide.

Come modificare le dimensioni delle istanze riservate

Questo comando di esempio modifica un'istanza riservata con 10 istanze m1.small in us-west-1c in modo che 8 Linux/UNIX istanze m1.small diventino 2 istanze m1.large e le restanti 2 m1.small diventino 1 istanza m1.medium nella stessa zona di disponibilità. Comando:

```
aws ec2 modify-reserved-instances --reserved-instances-
ids 1ba8e2e3-3556-4264-949e-63ee671405a9 --target-
configurations AvailabilityZone=us-west-1c,Platform=EC2-
Classic,InstanceCount=2,InstanceType=m1.large AvailabilityZone=us-
west-1c,Platform=EC2-Classic,InstanceCount=1,InstanceType=m1.medium
```

Output:

```
{
    "ReservedInstancesModificationId": "rimod-acc5f240-080d-4717-b3e3-1c6b11fa00b6"
}
```

Per ulteriori informazioni, consulta Modifica della dimensione dell'istanza delle prenotazioni nella Amazon EC2 User Guide.

- Per i dettagli sull'API, consulta ModifyReservedInstances AWS CLICommand Reference.

`modify-security-group-rules`

Il seguente esempio di codice mostra come utilizzare`modify-security-group-rules`.

AWS CLI

Modificare le regole di un gruppo di sicurezza per aggiornare la descrizione della regola, il protocollo IP e l'intervallo di CidrIpv 4 indirizzi

L'`modify-security-group-rules`esempio seguente aggiorna la descrizione, il protocollo IP e l'intervallo IPV4 CIDR di una regola del gruppo di sicurezza specificato. Utilizza il parametro `security-group-rules` per immettere gli aggiornamenti per le regole del gruppo di sicurezza specificato. `-1` specifica tutti i protocolli.

```
aws ec2 modify-security-group-rules \
```

```
    --group-id sg-1234567890abcdef0 \
    --security-group-rules SecurityGroupRuleId=sgr-
abcdef01234567890,SecurityGroupRule='{Description=test,IpProtocol=-1,CidrIpv4=0.0.0.0/0}'
```

Output:

```
{
    "Return": true
}
```

Per ulteriori informazioni sulle regole dei gruppi di sicurezza, consulta le regole dei gruppi di sicurezza nella Amazon EC2 User Guide.

- Per i dettagli sull'API, consulta ModifySecurityGroupRules AWS CLICommand Reference.

`modify-snapshot-attribute`

Il seguente esempio di codice mostra come utilizzare`modify-snapshot-attribute`.

AWS CLI

Esempio 1: come modificare un attributo snapshot

L'esempio `modify-snapshot-attribute` seguente aggiorna l'attributo `createVolumePermission` per lo snapshot specificato, rimuovendo le autorizzazioni di volume per l'utente specificato.

```
aws ec2 modify-snapshot-attribute \
    --snapshot-id snap-1234567890abcdef0 \
    --attribute createVolumePermission \
    --operation-type remove \
    --user-ids 123456789012
```

Esempio 2: come rendere pubblico uno snapshot

L'esempio `modify-snapshot-attribute` seguente rende pubblico lo snapshot specificato.

```
aws ec2 modify-snapshot-attribute \
    --snapshot-id snap-1234567890abcdef0 \
    --attribute createVolumePermission \
    --operation-type add \
```

```
    --group-names all
```

- Per i dettagli sull'API, consulta ModifySnapshotAttribute AWS CLICommand Reference.

`modify-snapshot-tier`

Il seguente esempio di codice mostra come utilizzare`modify-snapshot-tier`.

AWS CLI

Come archiviare uno snapshot

L'esempio `modify-snapshot-tier` seguente archivia lo snapshot specificato. Il parametro di risposta `TieringStartTime` indica la data e l'ora in cui è stato avviato il processo di archiviazione, in formato ora UTC (YYYYY-MM-DTHH:MM:SSZ).

```
aws ec2 modify-snapshot-tier \
    --snapshot-id snap-01234567890abcedf \
    --storage-tier archive
```

Output:

```
{
    "SnapshotId": "snap-01234567890abcedf",
    "TieringStartTime": "2021-09-15T16:44:37.574Z"
}
```

Per ulteriori informazioni sull'archiviazione degli snapshot, consulta Archiviazione degli snapshot Amazon EBS nella Guida per l'utente di Amazon EBS.

- Per i dettagli sull'API, consulta ModifySnapshotTier AWS CLICommand Reference.

`modify-spot-fleet-request`

Il seguente esempio di codice mostra come utilizzare`modify-spot-fleet-request`.

AWS CLI

Come modificare una richiesta di parco istanze spot

Questo comando di esempio aggiorna la capacità di destinazione della richiesta del parco istanze spot specificata.

Comando:

```
aws ec2 modify-spot-fleet-request --target-capacity 20 --spot-fleet-request-
id sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE
```

Output:

```
{
    "Return": true
}
```

Questo comando di esempio diminuisce la capacità di destinazione del parco istanze spot specificata senza terminare le istanze spot.

Comando:

```
aws ec2 modify-spot-fleet-request --target-capacity 10 --excess-capacity-
termination-policy NoTermination --spot-fleet-request-ids sfr-73fbd2ce-
aa30-494c-8788-1cee4EXAMPLE
```

Output:

```
{
    "Return": true
}
```

- Per i dettagli sull'API, consulta ModifySpotFleetRequest AWS CLICommand Reference.

`modify-subnet-attribute`

Il seguente esempio di codice mostra come utilizzare`modify-subnet-attribute`.

AWS CLI

Per modificare il comportamento di IPv4 indirizzamento pubblico di una sottorete

Questo esempio modifica subnet-1a2b3c4d per specificare che a tutte le istanze avviate in questa sottorete viene assegnato un indirizzo pubblico. IPv4 Se il comando va a buon fine, non viene restituito alcun output.

Comando:

```
aws ec2 modify-subnet-attribute --subnet-id subnet-1a2b3c4d --map-public-ip-on-
launch
```

Per modificare il comportamento di IPv6 indirizzamento di una sottorete

Questo esempio modifica subnet-1a2b3c4d per specificare che a tutte le istanze avviate in questa sottorete viene assegnato un indirizzo compreso nell'intervallo della sottorete. IPv6

Comando:

```
aws ec2 modify-subnet-attribute --subnet-id subnet-1a2b3c4d --assign-ipv6-address-
on-creation
```

Per ulteriori informazioni, consulta Assegnazione degli indirizzi IP nel VPC nella Guida per l'utente di AWS Virtual Private Cloud.

- Per i dettagli sull'API, ModifySubnetAttributevedere AWS CLI in Command Reference.

`modify-traffic-mirror-filter-network-services`

Il seguente esempio di codice mostra come utilizzare`modify-traffic-mirror-filter-network-services`.

AWS CLI

Come aggiungere servizi di rete a un filtro di mirroring del traffico

L'esempio `modify-traffic-mirror-filter-network-services` seguente aggiunge i servizi di rete Amazon DNS al filtro specificato.

```
aws ec2 modify-traffic-mirror-filter-network-services \
    --traffic-mirror-filter-id tmf-04812ff784EXAMPLE \
    --add-network-service amazon-dns
```

Output:

```
{
    "TrafficMirrorFilter": {
        "Tags": [
            {
```

```
                "Key": "Name",
                "Value": "Production"
            }
        ],
        "EgressFilterRules": [],
        "NetworkServices": [
            "amazon-dns"
        ],
        "TrafficMirrorFilterId": "tmf-04812ff784EXAMPLE",
        "IngressFilterRules": [
            {
                "SourceCidrBlock": "0.0.0.0/0",
                "RuleNumber": 1,
                "DestinationCidrBlock": "0.0.0.0/0",
                "Description": "TCP Rule",
                "Protocol": 6,
                "TrafficDirection": "ingress",
                "TrafficMirrorFilterId": "tmf-04812ff784EXAMPLE",
                "RuleAction": "accept",
                "TrafficMirrorFilterRuleId": "tmf-04812ff784EXAMPLE"
            }
        ]
    }
}
```

Per ulteriori informazioni, consulta Modificare i servizi di rete per i filtri di mirroring del traffico nella Guida di Mirroring del traffico AWS .

- Per i dettagli sull'API, consulta ModifyTrafficMirrorFilterNetworkServices AWS CLICommand Reference.

`modify-traffic-mirror-filter-rule`

Il seguente esempio di codice mostra come utilizzare`modify-traffic-mirror-filter-rule`.

AWS CLI

Come modificare una regola di filtro di mirroring del traffico

L'esempio `modify-traffic-mirror-filter-rule` seguente modifica la descrizione della regola di filtro di mirroring del traffico specificata.

```
aws ec2 modify-traffic-mirror-filter-rule \
```

```
    --traffic-mirror-filter-rule-id tmfr-0ca76e0e08EXAMPLE \
    --description "TCP Rule"
```

Output:

```
{
    "TrafficMirrorFilterRule": {
        "TrafficMirrorFilterRuleId": "tmfr-0ca76e0e08EXAMPLE",
        "TrafficMirrorFilterId": "tmf-0293f26e86EXAMPLE",
        "TrafficDirection": "ingress",
        "RuleNumber": 100,
        "RuleAction": "accept",
        "Protocol": 6,
        "DestinationCidrBlock": "10.0.0.0/24",
        "SourceCidrBlock": "10.0.0.0/24",
        "Description": "TCP Rule"
    }
}
```

Per ulteriori informazioni, consulta Modificare le regole di filtro del mirroring del traffico nella Guida di Mirroring del traffico AWS .

- Per i dettagli sull'API, consulta ModifyTrafficMirrorFilterRule AWS CLICommand Reference.

`modify-traffic-mirror-session`

Il seguente esempio di codice mostra come utilizzare`modify-traffic-mirror-session`.

AWS CLI

Come modificare una sessione di mirroring del traffico

L'esempio `modify-traffic-mirror-session` seguente modifica la descrizione della sessione di mirroring del traffico e il numero di pacchetti di cui eseguire il mirroring.

```
aws ec2 modify-traffic-mirror-session \
    --description "Change packet length" \
    --traffic-mirror-session-id tms-08a33b1214EXAMPLE \
    --remove-fields "packet-length"
```

Output:

```
{
    "TrafficMirrorSession": {
        "TrafficMirrorSessionId": "tms-08a33b1214EXAMPLE",
        "TrafficMirrorTargetId": "tmt-07f75d8feeEXAMPLE",
        "TrafficMirrorFilterId": "tmf-04812ff784EXAMPLE",
        "NetworkInterfaceId": "eni-070203f901EXAMPLE",
        "OwnerId": "111122223333",
        "SessionNumber": 1,
        "VirtualNetworkId": 7159709,
        "Description": "Change packet length",
        "Tags": []
    }
}
```

Per ulteriori informazioni, consulta Modificare la sessione di mirroring del traffico nella Guida di Mirroring del traffico.

- Per i dettagli sull'API, consulta ModifyTrafficMirrorSession AWS CLICommand Reference.

`modify-transit-gateway-prefix-list-reference`

Il seguente esempio di codice mostra come utilizzare`modify-transit-gateway-prefix-list-reference`.

AWS CLI

Come modificare un riferimento a un elenco di prefissi

L'esempio `modify-transit-gateway-prefix-list-reference` seguente modifica il riferimento all'elenco di prefissi nella tabella di routing specificata cambiando il collegamento a cui viene instradato il traffico.

```
aws ec2 modify-transit-gateway-prefix-list-reference \
    --transit-gateway-route-table-id tgw-rtb-0123456789abcd123 \
    --prefix-list-id pl-1111122222222333 \
    --transit-gateway-attachment-id tgw-attach-aabbccddaabbccaab
```

Output:

```
{
    "TransitGatewayPrefixListReference": {
```

```
        "TransitGatewayRouteTableId": "tgw-rtb-0123456789abcd123",
        "PrefixListId": "pl-11111122222222333",
        "PrefixListOwnerId": "123456789012",
        "State": "modifying",
        "Blackhole": false,
        "TransitGatewayAttachment": {
            "TransitGatewayAttachmentId": "tgw-attach-aabbccddaabbccaab",
            "ResourceType": "vpc",
            "ResourceId": "vpc-112233445566aabbc"
        }
    }
}
```

Per ulteriori informazioni, consulta Riferimenti elenco dei prefissi nella Guida di Transit Gateway.

- Per i dettagli sull'API, consulta ModifyTransitGatewayPrefixListReference AWS CLICommand Reference.

`modify-transit-gateway-vpc-attachment`

Il seguente esempio di codice mostra come utilizzare`modify-transit-gateway-vpc-attachment`.

AWS CLI

Come modificare un collegamento VPC del gateway di transito

L'esempio `modify-transit-gateway-vpc-attachment` seguente aggiunge una sottorete al collegamento VPC del gateway di transito specificato.

```
aws ec2 modify-transit-gateway-vpc-attachment \
    --transit-gateway-attachment-id tgw-attach-09fbd47ddfEXAMPLE \
    --add-subnet-ids subnet-0e51f45802EXAMPLE
```

Output:

```
{
    "TransitGatewayVpcAttachment": {
        "TransitGatewayAttachmentId": "tgw-attach-09fbd47ddfEXAMPLE",
        "TransitGatewayId": "tgw-0560315ccfEXAMPLE",
        "VpcId": "vpc-5eccc927",
        "VpcOwnerId": "111122223333",
```

```
        "State": "modifying",
        "SubnetIds": [
            "subnet-0e51f45802EXAMPLE",
            "subnet-1EXAMPLE"
        ],
        "CreationTime": "2019-08-08T16:47:38.000Z",
        "Options": {
            "DnsSupport": "enable",
            "Ipv6Support": "disable"
        }
    }
}
```

Per ulteriori informazioni, consulta Collegamenti del gateway di transito alla VPN a un VPC nella Guida di Transit Gateway.

- Per i dettagli sull'API, consulta ModifyTransitGatewayVpcAttachment AWS CLICommand Reference.

`modify-transit-gateway`

Il seguente esempio di codice mostra come utilizzare`modify-transit-gateway`.

AWS CLI

Come modificare un gateway di transito

L'esempio `modify-transit-gateway` seguente modifica il gateway di transito specificato abilitando il supporto ECMP per gli allegati VPN.

```
aws ec2 modify-transit-gateway \
    --transit-gateway-id tgw-111111222222aaaaa \
    --options VpnEcmpSupport=enable
```

Output:

```
{
    "TransitGateway": {
        "TransitGatewayId": "tgw-111111222222aaaaa",
        "TransitGatewayArn": "64512",
        "State": "modifying",
        "OwnerId": "123456789012",
```

```
        "CreationTime": "2020-04-30T08:41:37.000Z",
        "Options": {
            "AmazonSideAsn": 64512,
            "AutoAcceptSharedAttachments": "disable",
            "DefaultRouteTableAssociation": "enable",
            "AssociationDefaultRouteTableId": "tgw-rtb-0123456789abcd123",
            "DefaultRouteTablePropagation": "enable",
            "PropagationDefaultRouteTableId": "tgw-rtb-0123456789abcd123",
            "VpnEcmpSupport": "enable",
            "DnsSupport": "enable"
        }
    }
}
```

Per ulteriori informazioni, consulta Gateway di transito nella Guida di Transit Gateway.

- Per i dettagli sull'API, consulta ModifyTransitGateway AWS CLICommand Reference.

`modify-verified-access-endpoint-policy`

Il seguente esempio di codice mostra come utilizzare`modify-verified-access-endpoint-policy`.

AWS CLI

Come configurare la policy di accesso verificato per un endpoint

L'esempio `modify-verified-access-endpoint-policy` seguente aggiunge la policy di accesso verificato specificata all'endpoint di accesso verificato specificato.

```
aws ec2 modify-verified-access-endpoint-policy \
    --verified-access-endpoint-id vae-066fac616d4d546f2 \
    --policy-enabled \
    --policy-document file://policy.txt
```

Contenuto di `policy.txt`:

```
permit(principal,action,resource)
when {
    context.identity.groups.contains("finance") &&
    context.identity.email.verified == true
};
```

Output:

```
{
    "PolicyEnabled": true,
    "PolicyDocument": "permit(principal,action,resource)\nwhen
 {\n    context.identity.groups.contains(\"finance\") &&\n
 context.identity.email_verified == true\n};"
}
```

Per ulteriori informazioni, consulta Policy di Accesso verificato nella Guida per l'utente di Accesso verificato da AWS .

- Per i dettagli sull'API, consulta ModifyVerifiedAccessEndpointPolicy AWS CLICommand Reference.

`modify-verified-access-endpoint`

Il seguente esempio di codice mostra come utilizzare`modify-verified-access-endpoint`.

AWS CLI

Come modificare la configurazione di un endpoint di accesso verificato

L'esempio `modify-verified-access-endpoint` seguente aggiunge la descrizione specificata all'endpoint di accesso verificato specificato.

```
aws ec2 modify-verified-access-endpoint \
    --verified-access-endpoint-id vae-066fac616d4d546f2 \
    --description 'Testing Verified Access'
```

Output:

```
{
    "VerifiedAccessEndpoint": {
        "VerifiedAccessInstanceId": "vai-0ce000c0b7643abea",
        "VerifiedAccessGroupId": "vagr-0dbe967baf14b7235",
        "VerifiedAccessEndpointId": "vae-066fac616d4d546f2",
        "ApplicationDomain": "example.com",
        "EndpointType": "network-interface",
        "AttachmentType": "vpc",
        "DomainCertificateArn": "arn:aws:acm:us-east-2:123456789012:certificate/
eb065ea0-26f9-4e75-a6ce-0a1a7EXAMPLE",
```

```
        "EndpointDomain": "my-ava-
app.edge-00c3372d53b1540bb.vai-0ce000c0b7643abea.prod.verified-access.us-
east-2.amazonaws.com",
        "SecurityGroupIds": [
            "sg-004915970c4c8f13a"
        ],
        "NetworkInterfaceOptions": {
            "NetworkInterfaceId": "eni-0aec70418c8d87a0f",
            "Protocol": "https",
            "Port": 443
        },
        "Status": {
            "Code": "updating"
        },
        "Description": "Testing Verified Access",
        "CreationTime": "2023-08-25T20:54:43",
        "LastUpdatedTime": "2023-08-25T22:46:32"
    }
}
```

Per ulteriori informazioni, consulta Endpoint di Accesso verificato nella Guida per l'utente di Accesso verificato da AWS .

- Per i dettagli sull'API, consulta ModifyVerifiedAccessEndpoint AWS CLICommand Reference.

`modify-verified-access-group-policy`

Il seguente esempio di codice mostra come utilizzare`modify-verified-access-group-policy`.

AWS CLI

Come configurare una policy di accesso verificato per un gruppo

L'esempio `modify-verified-access-group-policy` seguente aggiunge la policy di accesso verificato specificata al gruppo di accesso verificato specificato.

```
aws ec2 modify-verified-access-group-policy \
    --verified-access-group-id vagr-0dbe967baf14b7235 \
    --policy-enabled \
    --policy-document file://policy.txt
```

Contenuto di `policy.txt`:

```
permit(principal,action,resource)
when {
    context.identity.groups.contains("finance") &&
    context.identity.email.verified == true
};
```

Output:

```
{
    "PolicyEnabled": true,
    "PolicyDocument": "permit(principal,action,resource)\nwhen
 {\n    context.identity.groups.contains(\"finance\") &&\n
 context.identity.email_verified == true\n};"
}
```

Per ulteriori informazioni, consulta Gruppi di Accesso verificato nella Guida per l'utente di Accesso verificato da AWS .

- Per i dettagli sull'API, consulta ModifyVerifiedAccessGroupPolicy AWS CLICommand Reference.

`modify-verified-access-group`

Il seguente esempio di codice mostra come utilizzare`modify-verified-access-group`.

AWS CLI

Come modificare la configurazione di un gruppo di accesso verificato

L'esempio `modify-verified-access-group` seguente aggiunge la descrizione specificata al gruppo di accesso verificato specificato.

```
aws ec2 modify-verified-access-group \
    --verified-access-group-id vagr-0dbe967baf14b7235 \
    --description "Testing Verified Access"
```

Output:

```
{
    "VerifiedAccessGroup": {
        "VerifiedAccessGroupId": "vagr-0dbe967baf14b7235",
```

```
        "VerifiedAccessInstanceId": "vai-0ce000c0b7643abea",
        "Description": "Testing Verified Access",
        "Owner": "123456789012",
        "VerifiedAccessGroupArn": "arn:aws:ec2:us-east-2:123456789012:verified-
access-group/vagr-0dbe967baf14b7235",
        "CreationTime": "2023-08-25T19:55:19",
        "LastUpdatedTime": "2023-08-25T22:17:25"
    }
}
```

Per ulteriori informazioni, consulta Gruppi di Accesso verificato nella Guida per l'utente di Accesso verificato da AWS .

- Per i dettagli sull'API, consulta ModifyVerifiedAccessGroup AWS CLICommand Reference.

`modify-verified-access-instance-logging-configuration`

Il seguente esempio di codice mostra come utilizzare`modify-verified-access-instance-logging-configuration`.

AWS CLI

Come abilitare la registrazione per un'istanza di accesso verificato

L'esempio `modify-verified-access-instance-logging-configuration` seguente abilita la registrazione dell'accesso per l'istanza di accesso verificato specificata. I log verranno consegnati al gruppo di CloudWatch log Logs specificato.

```
aws ec2 modify-verified-access-instance-logging-configuration \
    --verified-access-instance-id vai-0ce000c0b7643abea \
    --access-logs CloudWatchLogs={Enabled=true,LogGroup=my-log-group}
```

Output:

```
{
    "LoggingConfiguration": {
        "VerifiedAccessInstanceId": "vai-0ce000c0b7643abea",
        "AccessLogs": {
            "S3": {
                "Enabled": false
            },
            "CloudWatchLogs": {
```

```
                "Enabled": true,
                "DeliveryStatus": {
                    "Code": "success"
                },
                "LogGroup": "my-log-group"
            },
            "KinesisDataFirehose": {
                "Enabled": false
            },
            "LogVersion": "ocsf-1.0.0-rc.2",
            "IncludeTrustContext": false
        }
    }
}
```

Per ulteriori informazioni, consulta Log di accesso verificato nella Guida per l'utente di Accesso verificato da AWS .

- Per i dettagli sull'API, vedere ModifyVerifiedAccessInstanceLoggingConfigurationin AWS CLI Command Reference.

`modify-verified-access-instance`

Il seguente esempio di codice mostra come utilizzare`modify-verified-access-instance`.

AWS CLI

Come modificare la configurazione di un'istanza di accesso verificato

L'esempio `modify-verified-access-instance` seguente aggiunge la descrizione specificata all'istanza di accesso verificato specificata.

```
aws ec2 modify-verified-access-instance \
    --verified-access-instance-id vai-0ce000c0b7643abea \
    --description "Testing Verified Access"
```

Output:

```
{
    "VerifiedAccessInstance": {
        "VerifiedAccessInstanceId": "vai-0ce000c0b7643abea",
        "Description": "Testing Verified Access",
```

```
        "VerifiedAccessTrustProviders": [
            {
                "VerifiedAccessTrustProviderId": "vatp-0bb32de759a3e19e7",
                "TrustProviderType": "user",
                "UserTrustProviderType": "iam-identity-center"
            }
        ],
        "CreationTime": "2023-08-25T18:27:56",
        "LastUpdatedTime": "2023-08-25T22:41:04"
    }
}
```

Per ulteriori informazioni, consulta Istanze di Accesso verificato nella Guida per l'utente di Accesso verificato da AWS .

- Per i dettagli sull'API, consulta ModifyVerifiedAccessInstance AWS CLICommand Reference.

`modify-verified-access-trust-provider`

Il seguente esempio di codice mostra come utilizzare`modify-verified-access-trust-provider`.

AWS CLI

Come modificare la configurazione di un provider di attendibilità di accesso verificato

L'esempio `modify-verified-access-trust-provider` seguente aggiunge la descrizione specificata al provider di attendibilità di accesso verificato.

```
aws ec2 modify-verified-access-trust-provider \
    --verified-access-trust-provider-id vatp-0bb32de759a3e19e7 \
    --description "Testing Verified Access"
```

Output:

```
{
    "VerifiedAccessTrustProvider": {
        "VerifiedAccessTrustProviderId": "vatp-0bb32de759a3e19e7",
        "Description": "Testing Verified Access",
        "TrustProviderType": "user",
        "UserTrustProviderType": "iam-identity-center",
        "PolicyReferenceName": "idc",
```

```
        "CreationTime": "2023-08-25T19:00:38",
        "LastUpdatedTime": "2023-08-25T19:18:21"
    }
}
```

Per ulteriori informazioni, consulta Provider di attendibilità per Accesso verificato nella Guida per l'utente di Accesso verificato da AWS .

- Per i dettagli sull'API, consulta ModifyVerifiedAccessTrustProvider AWS CLICommand Reference.

`modify-volume-attribute`

Il seguente esempio di codice mostra come utilizzare`modify-volume-attribute`.

AWS CLI

Come modificare un attributo del volume

Questo esempio imposta l'attributo `autoEnableIo` del volume con l'ID `vol-1234567890abcdef0` su `true`. Se il comando va a buon fine, non viene restituito alcun output.

Comando:

```
aws ec2 modify-volume-attribute --volume-id vol-1234567890abcdef0 --auto-enable-io
```

- Per i dettagli sull'API, consulta ModifyVolumeAttribute AWS CLICommand Reference.

`modify-volume`

Il seguente esempio di codice mostra come utilizzare`modify-volume`.

AWS CLI

Esempio 1: come modificare un volume cambiando le dimensioni

L'esempio `modify-volume` seguente modifica le dimensioni del volume specificato in 150 GB.

Comando:

```
aws ec2 modify-volume --size 150 --volume-id vol-1234567890abcdef0
```

Output:

```
{
    "VolumeModification": {
        "TargetSize": 150,
        "TargetVolumeType": "io1",
        "ModificationState": "modifying",
        "VolumeId": " vol-1234567890abcdef0",
        "TargetIops": 100,
        "StartTime": "2019-05-17T11:27:19.000Z",
        "Progress": 0,
        "OriginalVolumeType": "io1",
        "OriginalIops": 100,
        "OriginalSize": 100
    }
}
```

Esempio 2: come modificare un volume cambiando tipo, dimensioni e valore IOPS

L'esempio `modify-volume` seguente modifica il tipo di volume in SSD capacità di IOPS allocata, imposta la velocità IOPS target su 10000 e imposta le dimensioni del volume su 350 GB.

```
aws ec2 modify-volume \
    --volume-type io1 \
    --iops 10000 \
    --size 350 \
    --volume-id vol-1234567890abcdef0
```

Output:

```
{
    "VolumeModification": {
        "TargetSize": 350,
        "TargetVolumeType": "io1",
        "ModificationState": "modifying",
        "VolumeId": "vol-0721c1a9d08c93bf6",
        "TargetIops": 10000,
        "StartTime": "2019-05-17T11:38:57.000Z",
        "Progress": 0,
        "OriginalVolumeType": "gp2",
        "OriginalIops": 150,
        "OriginalSize": 50
```

```
    }
}
```

- Per i dettagli sull'API, consulta ModifyVolume AWS CLICommand Reference.

`modify-vpc-attribute`

Il seguente esempio di codice mostra come utilizzare`modify-vpc-attribute`.

AWS CLI

Per modificare l' enableDnsSupport attributo

Questo esempio modifica l'attributo `enableDnsSupport`. Questo attributo indica se la risoluzione DNS è abilitata per il VPC. Se questo attributo è `true`, il server Amazon DNS risolve i nomi host DNS per le istanze negli indirizzi IP corrispondenti, ma solo in quel caso. Se il comando va a buon fine, non viene restituito alcun output.

Comando:

```
aws ec2 modify-vpc-attribute --vpc-id vpc-a01106c2 --enable-dns-support "{\"Value
\":false}"
```

Per modificare l' enableDnsHostnames attributo

Questo esempio modifica l'attributo `enableDnsHostnames`. Questo attributo indica se le istanze avviate nel VPC ricevono nomi host DNS. Se questo attributo è `true`, le istanze nel VPC ottengono nomi host DNS, altrimenti no. Se il comando va a buon fine, non viene restituito alcun output.

Comando:

```
aws ec2 modify-vpc-attribute --vpc-id vpc-a01106c2 --enable-dns-hostnames "{\"Value
\":false}"
```

- Per i dettagli sull'API, consulta ModifyVpcAttribute AWS CLICommand Reference.

`modify-vpc-endpoint-connection-notification`

Il seguente esempio di codice mostra come utilizzare`modify-vpc-endpoint-connection-notification`.

AWS CLI

Come modificare una notifica della connessione endpoint

Questo esempio modifica l'argomento SNS per la notifica della connessione endpoint specificata.

Comando:

```
aws ec2 modify-vpc-endpoint-connection-notification --connection-notification-
id vpce-nfn-008776de7e03f5abc --connection-events Accept Reject --connection-
notification-arn arn:aws:sns:us-east-2:123456789012:mytopic
```

Output:

```
{
   "ReturnValue": true
}
```

- Per i dettagli sull'API, consulta ModifyVpcEndpointConnectionNotification AWS CLICommand Reference.

`modify-vpc-endpoint-service-configuration`

Il seguente esempio di codice mostra come utilizzare`modify-vpc-endpoint-service-configuration`.

AWS CLI

Come modificare una configurazione di servizio endpoint

Questo esempio modifica il requisito di accettazione per il servizio endpoint specificato.

Comando:

```
aws ec2 modify-vpc-endpoint-service-configuration --service-id vpce-
svc-09222513e6e77dc86 --no-acceptance-required
```

Output:

```
{
```

```
    "ReturnValue": true
}
```

- Per i dettagli sull'API, consulta ModifyVpcEndpointServiceConfiguration AWS CLICommand Reference.

`modify-vpc-endpoint-service-payer-responsibility`

Il seguente esempio di codice mostra come utilizzare`modify-vpc-endpoint-service-payer-responsibility`.

AWS CLI

Come modificare la responsabilità del pagante

L'esempio `modify-vpc-endpoint-service-payer-responsibility` seguente modifica la responsabilità del pagante del servizio endpoint specificato.

```
aws ec2 modify-vpc-endpoint-service-payer-responsibility \
    --service-id vpce-svc-071afff70666e61e0 \
    --payer-responsibility ServiceOwner
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta ModifyVpcEndpointServicePayerResponsibility AWS CLICommand Reference.

`modify-vpc-endpoint-service-permissions`

Il seguente esempio di codice mostra come utilizzare`modify-vpc-endpoint-service-permissions`.

AWS CLI

Come modificare le autorizzazioni del servizio endpoint

Questo esempio aggiunge l'autorizzazione per la connessione di un AWS account al servizio endpoint specificato.

Comando:

```
aws ec2 modify-vpc-endpoint-service-permissions --service-id vpce-
svc-03d5ebb7d9579a2b3 --add-allowed-principals '["arn:aws:iam::123456789012:root"]'
```

Output:

```
{
   "ReturnValue": true
}
```

Questo esempio aggiunge l'autorizzazione per un utente IAM specifico (`admin`) per connettersi al servizio endpoint specificato.

Comando:

```
aws ec2 modify-vpc-endpoint-service-permissions --service-id vpce-
svc-03d5ebb7d9579a2b3 --add-allowed-principals '["arn:aws:iam::123456789012:user/
admin"]'
```

- Per i dettagli sull'API, vedere ModifyVpcEndpointServicePermissionsin AWS CLI Command Reference.

`modify-vpc-endpoint`

Il seguente esempio di codice mostra come utilizzare`modify-vpc-endpoint`.

AWS CLI

Come modificare un endpoint del gateway

Questo esempio modifica l'endpoint del gateway `vpce-1a2b3c4d` associando la tabella di routing `rtb-aaa222bb` all'endpoint e reimpostando il documento di policy.

Comando:

```
aws ec2 modify-vpc-endpoint --vpc-endpoint-id vpce-1a2b3c4d --add-route-table-
ids rtb-aaa222bb --reset-policy
```

Output:

```
{
```

```
  "Return": true
}
```

Come modificare un endpoint di interfaccia

Questo esempio modifica l'endpoint di interfaccia `vpce-0fe5b17a0707d6fa5` aggiungendo una sottorete `subnet-d6fcaa8d` all'endpoint.

Comando:

```
aws ec2 modify-vpc-endpoint --vpc-endpoint-id vpce-0fe5b17a0707d6fa5 --add-subnet-
id subnet-d6fcaa8d
```

Output:

```
{
  "Return": true
}
```

- Per i dettagli sull'API, consulta ModifyVpcEndpoint AWS CLICommand Reference.

`modify-vpc-peering-connection-options`

Il seguente esempio di codice mostra come utilizzare`modify-vpc-peering-connection-options`.

AWS CLI

Per abilitare la comunicazione tramite una connessione peering VPC dalla connessione locale ClassicLink

In questo esempio, per la connessione peering`pcx-aaaabbb`, il proprietario del VPC richiedente modifica le opzioni di connessione peering VPC per consentire a una connessione ClassicLink locale di comunicare con il VPC peer.

Comando:

```
aws ec2 modify-vpc-peering-connection-options --vpc-peering-
connection-id pcx-aaaabbbb --requester-peering-connection-
options AllowEgressFromLocalClassicLinkToRemoteVpc=true
```

Output:

```
{
  "RequesterPeeringConnectionOptions": {
      "AllowEgressFromLocalClassicLinkToRemoteVpc": true
  }
}
```

Per abilitare la comunicazione tramite una connessione peering VPC dal VPC locale a una connessione remota ClassicLink

In questo esempio, il proprietario del VPC accettante modifica le opzioni di connessione peering VPC per consentire al VPC locale di comunicare con la connessione nel VPC peer. ClassicLink

Comando:

```
aws ec2 modify-vpc-peering-connection-options --vpc-peering-
connection-id pcx-aaaabbbb --accepter-peering-connection-
options AllowEgressFromLocalVpcToRemoteClassicLink=true
```

Output:

```
{
  "AccepterPeeringConnectionOptions": {
    "AllowEgressFromLocalVpcToRemoteClassicLink": true
  }
}
```

Come abilitare il supporto della risoluzione DNS per una connessione peering VPC

In questo esempio, il proprietario del VPC richiedente modifica le opzioni di connessione peering VPC per `pcx-aaaabbbb` per consentire al VPC di risolvere nomi host DNS pubblici in indirizzi IP privati quando viene interrogato da istanze nel VPC peer.

Comando:

```
aws ec2 modify-vpc-peering-connection-options --vpc-peering-connection-id pcx-
aaaabbbb --requester-peering-connection-options AllowDnsResolutionFromRemoteVpc=true
```

Output:

```
{
  "RequesterPeeringConnectionOptions": {
      "AllowDnsResolutionFromRemoteVpc": true
  }
}
```

- Per i dettagli sull'API, vedere in Command Reference. ModifyVpcPeeringConnectionOptionsAWS CLI

`modify-vpc-tenancy`

Il seguente esempio di codice mostra come utilizzare`modify-vpc-tenancy`.

AWS CLI

Come modificare la tenancy di un VPC

Questo esempio modifica la tenancy del VPC `vpc-1a2b3c4d` in `default`.

Comando:

```
aws ec2 modify-vpc-tenancy --vpc-id vpc-1a2b3c4d --instance-tenancy default
```

Output:

```
{
  "Return": true
}
```

- Per i dettagli sull'API, consulta ModifyVpcTenancy AWS CLICommand Reference.

`modify-vpn-connection-options`

Il seguente esempio di codice mostra come utilizzare`modify-vpn-connection-options`.

AWS CLI

Come modificare le opzioni di connessione VPN

L'`modify-vpn-connection-options`esempio seguente modifica il IPv4 CIDR locale sul lato gateway del cliente della connessione VPN specificata.

```
aws ec2 modify-vpn-connection-options \
    --vpn-connection-id vpn-1122334455aabbccd \
    --local-ipv4-network-cidr 10.0.0.0/16
```

Output:

```
{
    "VpnConnections": [
        {
            "CustomerGatewayConfiguration": "...configuration information...",
            "CustomerGatewayId": "cgw-01234567abcde1234",
            "Category": "VPN",
            "State": "modifying",
            "Type": "ipsec.1",
            "VpnConnectionId": "vpn-1122334455aabbccd",
            "TransitGatewayId": "tgw-00112233445566aab",
            "Options": {
                "EnableAcceleration": false,
                "StaticRoutesOnly": true,
                "LocalIpv4NetworkCidr": "10.0.0.0/16",
                "RemoteIpv4NetworkCidr": "0.0.0.0/0",
                "TunnelInsideIpVersion": "ipv4"
            },
            "Routes": [],
            "Tags": [
                {
                    "Key": "Name",
                    "Value": "CanadaVPN"
                }
            ],
            "VgwTelemetry": [
                {
                    "AcceptedRouteCount": 0,
                    "LastStatusChange": "2020-07-29T10:35:11.000Z",
                    "OutsideIpAddress": "203.0.113.3",
                    "Status": "DOWN",
                    "StatusMessage": ""
                },
                {
                    "AcceptedRouteCount": 0,
                    "LastStatusChange": "2020-09-02T09:09:33.000Z",
                    "OutsideIpAddress": "203.0.113.5",
                    "Status": "UP",
```

```
                    "StatusMessage": ""
                }
            ]
        }
    ]
}
```

Per ulteriori informazioni, consulta Modifica delle opzioni di connessione Site-to-Site VPN nella Guida per l'utente AWS Site-to-Site VPN.

- Per i dettagli sull'API, consulta ModifyVpnConnectionOptions AWS CLICommand Reference.

`modify-vpn-connection`

Il seguente esempio di codice mostra come utilizzare`modify-vpn-connection`.

AWS CLI

Come modificare una connessione VPN

L'esempio `modify-vpn-connection` seguente modifica il gateway di destinazione per la connessione VPN `vpn-12345678901234567` nel gateway privato virtuale `vgw-11223344556677889`:

```
aws ec2 modify-vpn-connection \
    --vpn-connection-id vpn-12345678901234567 \
    --vpn-gateway-id vgw-11223344556677889
```

Output:

```
{
    "VpnConnection": {
        "CustomerGatewayConfiguration": "...configuration information...",
        "CustomerGatewayId": "cgw-aabbccddee1122334",
        "Category": "VPN",
        "State": "modifying",
        "Type": "ipsec.1",
        "VpnConnectionId": "vpn-12345678901234567",
        "VpnGatewayId": "vgw-11223344556677889",
        "Options": {
            "StaticRoutesOnly": false
        },
```

```
        "VgwTelemetry": [
            {
                "AcceptedRouteCount": 0,
                "LastStatusChange": "2019-07-17T07:34:00.000Z",
                "OutsideIpAddress": "18.210.3.222",
                "Status": "DOWN",
                "StatusMessage": "IPSEC IS DOWN"
            },
            {
                "AcceptedRouteCount": 0,
                "LastStatusChange": "2019-07-20T21:20:16.000Z",
                "OutsideIpAddress": "34.193.129.33",
                "Status": "DOWN",
                "StatusMessage": "IPSEC IS DOWN"
            }
        ]
    }
}
```

- Per i dettagli sull'API, consulta ModifyVpnConnection AWS CLICommand Reference.

`modify-vpn-tunnel-certificate`

Il seguente esempio di codice mostra come utilizzare`modify-vpn-tunnel-certificate`.

AWS CLI

Come ruotare un certificato del tunnel VPN

L'esempio `modify-vpn-tunnel-certificate` seguente ruota il certificato del tunnel specificato per una connessione VPN

```
aws ec2 modify-vpn-tunnel-certificate \
    --vpn-tunnel-outside-ip-address 203.0.113.17 \
    --vpn-connection-id vpn-12345678901234567
```

Output:

```
{
    "VpnConnection": {
        "CustomerGatewayConfiguration": ...configuration information...,
        "CustomerGatewayId": "cgw-aabbccddee1122334",
```

```
        "Category": "VPN",
        "State": "modifying",
        "Type": "ipsec.1",
        "VpnConnectionId": "vpn-12345678901234567",
        "VpnGatewayId": "vgw-11223344556677889",
        "Options": {
            "StaticRoutesOnly": false
        },
        "VgwTelemetry": [
            {
                "AcceptedRouteCount": 0,
                "LastStatusChange": "2019-09-11T17:27:14.000Z",
                "OutsideIpAddress": "203.0.113.17",
                "Status": "DOWN",
                "StatusMessage": "IPSEC IS DOWN",
                "CertificateArn": "arn:aws:acm:us-east-1:123456789101:certificate/
c544d8ce-20b8-4fff-98b0-example"
            },
            {
                "AcceptedRouteCount": 0,
                "LastStatusChange": "2019-09-11T17:26:47.000Z",
                "OutsideIpAddress": "203.0.114.18",
                "Status": "DOWN",
                "StatusMessage": "IPSEC IS DOWN",
                "CertificateArn": "arn:aws:acm:us-
east-1:123456789101:certificate/5ab64566-761b-4ad3-b259-example"
            }
        ]
    }
}
```

- Per i dettagli sull'API, consulta ModifyVpnTunnelCertificate AWS CLICommand Reference.

`modify-vpn-tunnel-options`

Il seguente esempio di codice mostra come utilizzare`modify-vpn-tunnel-options`.

AWS CLI

Come modificare le opzioni tunnel per una connessione VPN

L'esempio `modify-vpn-tunnel-options` seguente aggiorna i gruppi Diffie-Hellman consentiti per il tunnel e la connessione VPN specificati.

```
aws ec2 modify-vpn-tunnel-options \
    --vpn-connection-id vpn-12345678901234567 \
    --vpn-tunnel-outside-ip-address 203.0.113.17 \
    --tunnel-options Phase1DHGroupNumbers=[{Value=14},{Value=15},{Value=16},
{Value=17},{Value=18}],Phase2DHGroupNumbers=[{Value=14},{Value=15},{Value=16},
{Value=17},{Value=18}]
```

Output:

```
{
    "VpnConnection": {
        "CustomerGatewayConfiguration": "...configuration information...",
        "CustomerGatewayId": "cgw-aabbccddee1122334",
        "Category": "VPN",
        "State": "available",
        "Type": "ipsec.1",
        "VpnConnectionId": "vpn-12345678901234567",
        "VpnGatewayId": "vgw-11223344556677889",
        "Options": {
            "StaticRoutesOnly": false,
            "TunnelOptions": [
                {
                    "OutsideIpAddress": "203.0.113.17",
                    "Phase1DHGroupNumbers": [
                        {
                            "Value": 14
                        },
                        {
                            "Value": 15
                        },
                        {
                            "Value": 16
                        },
                        {
                            "Value": 17
                        },
                        {
                            "Value": 18
                        }
                    ],
                    "Phase2DHGroupNumbers": [
                        {
                            "Value": 14
```

```
                        },
                        {
                            "Value": 15
                        },
                        {
                            "Value": 16
                        },
                        {
                            "Value": 17
                        },
                        {
                            "Value": 18
                        }
                    ]
                },
                {
                    "OutsideIpAddress": "203.0.114.19"
                }
            ]
        },
        "VgwTelemetry": [
            {
                "AcceptedRouteCount": 0,
                "LastStatusChange": "2019-09-10T21:56:54.000Z",
                "OutsideIpAddress": "203.0.113.17",
                "Status": "DOWN",
                "StatusMessage": "IPSEC IS DOWN"
            },
            {
                "AcceptedRouteCount": 0,
                "LastStatusChange": "2019-09-10T21:56:43.000Z",
                "OutsideIpAddress": "203.0.114.19",
                "Status": "DOWN",
                "StatusMessage": "IPSEC IS DOWN"
            }
        ]
    }
}
```

- Per i dettagli sull'API, consulta ModifyVpnTunnelOptions AWS CLICommand Reference.

`monitor-instances`

Il seguente esempio di codice mostra come utilizzare`monitor-instances`.

AWS CLI

Come abilitare il monitoraggio dettagliato per un'istanza

Questo comando di esempio abilita il monitoraggio dettagliato per l'istanza specificata.

Comando:

```
aws ec2 monitor-instances --instance-ids i-1234567890abcdef0
```

Output:

```
{
  "InstanceMonitorings": [
      {
          "InstanceId": "i-1234567890abcdef0",
          "Monitoring": {
              "State": "pending"
          }
      }
  ]
}
```

- Per i dettagli sull'API, consulta MonitorInstances AWS CLICommand Reference.

`move-address-to-vpc`

Il seguente esempio di codice mostra come utilizzare`move-address-to-vpc`.

AWS CLI

Per spostare un indirizzo su EC2 -VPC

Questo esempio sposta l'indirizzo IP elastico 54.123.4.56 sulla piattaforma -VPC. EC2

Comando:

```
aws ec2 move-address-to-vpc --public-ip 54.123.4.56
```

Output:

```
{
  "Status": "MoveInProgress"
}
```

- Per i dettagli sull'API, consulta Command Reference. MoveAddressToVpcAWS CLI

`move-byoip-cidr-to-ipam`

Il seguente esempio di codice mostra come utilizzare`move-byoip-cidr-to-ipam`.

AWS CLI

Come trasferire un CIDR BYOIP in IPAM

L'esempio `move-byoip-cidr-to-ipam` seguente trasferisce un CIDR BYOIP in IPAM.

Linux:

```
aws ec2 move-byoip-cidr-to-ipam \
    --region us-west-2 \
    --ipam-pool-id ipam-pool-0a03d430ca3f5c035 \
    --ipam-pool-owner 111111111111 \
    --cidr 130.137.249.0/24
```

Windows:

```
aws ec2 move-byoip-cidr-to-ipam ^
    --region us-west-2 ^
    --ipam-pool-id ipam-pool-0a03d430ca3f5c035 ^
    --ipam-pool-owner 111111111111 ^
    --cidr 130.137.249.0/24
```

Output:

```
{
    "ByoipCidr": {
        "Cidr": "130.137.249.0/24",
        "State": "pending-transfer"
    }
}
```

Per ulteriori informazioni, consulta Tutorial: trasferimento di un IPv4 CIDR BYOIP esistente a IPAM nella Amazon VPC IPAM User Guide.

- Per i dettagli sull'API, consulta Command Reference. MoveByoipCidrToIpamAWS CLI

`network-insights-access-scope`

Il seguente esempio di codice mostra come utilizzare`network-insights-access-scope`.

AWS CLI

Come creare gli ambiti di accesso alle informazioni di rete

L'`create-network-insights-access-scope`esempio seguente crea un ambito di accesso a Network Insights nel tuo AWS account.

```
aws ec2 create-network-insights-access-scope \
    --cli-input-json file://access-scope-file.json
```

Contenuto di `access-scope-file.json`:

```
{
    {
        "MatchPaths": [
            {
                "Source": {
                    "ResourceStatement": {
                         "Resources": [
                            "vpc-abcd12e3"
                        ]
                    }
                }
            }
        ],
        "ExcludePaths": [
            {
                "Source": {
                    "ResourceStatement": {
                        "ResourceTypes": [
                            "AWS::EC2::InternetGateway"
                        ]
                    }
                }
```

```
            }
        ]
    }
}
```

Output:

```
{
    "NetworkInsightsAccessScopeAnalysisId": "nisa-123456789111"
    }{
    "NetworkInsightsAccessScope": {
        "NetworkInsightsAccessScopeId": "nis-123456789222",
        "NetworkInsightsAccessScopeArn": "arn:aws:ec2:us-
east-1:123456789222:network-insights-access-scope/nis-123456789222",
        "CreatedDate": "2022-01-25T19:20:28.796000+00:00",
        "UpdatedDate": "2022-01-25T19:20:28.797000+00:00"
    },
    "NetworkInsightsAccessScopeContent": {
        "NetworkInsightsAccessScopeId": "nis-04c0c0fbca737c404",
        "MatchPaths": [
            {
                "Source": {
                    "ResourceStatement": {
                        "Resources": [
                            "vpc-abcd12e3"
                        ]
                    }
                }
            }
        ],
        "ExcludePaths": [
            {
                "Source": {
                    "ResourceStatement": {
                        "ResourceTypes": [
                            "AWS::EC2::InternetGateway"
                        ]
                    }
                }
            }
        ]
    }
}
```

Per ulteriori informazioni, consulta Guida introduttiva a Network Access Analyzer utilizzando la AWS CLI nella Guida Network Access Analyzer.

- Per i dettagli sull'API, consulta Command NetworkInsightsAccessScopeReference AWS CLI .

`provision-byoip-cidr`

Il seguente esempio di codice mostra come utilizzare`provision-byoip-cidr`.

AWS CLI

Come effettuare il provisioning di un intervallo di indirizzi

L'`provision-byoip-cidr`esempio seguente fornisce un intervallo di indirizzi IP pubblici da utilizzare con AWS.

```
aws ec2 provision-byoip-cidr \
    --cidr 203.0.113.25/24 \
    --cidr-authorization-context Message="$text_message",Signature="$signed_message"
```

Output:

```
{
    "ByoipCidr": {
        "Cidr": "203.0.113.25/24",
        "State": "pending-provision"
    }
}
```

Per ulteriori informazioni sulla creazione delle stringhe dei messaggi per il contesto di autorizzazione, consulta Bring Your Own IP Addresses nella Amazon EC2 User Guide.

- Per i dettagli sull'API, consulta ProvisionByoipCidr AWS CLICommand Reference.

`provision-ipam-pool-cidr`

Il seguente esempio di codice mostra come utilizzare`provision-ipam-pool-cidr`.

AWS CLI

Come effettuare il provisioning di un CIDR a un pool IPAM

L'esempio `provision-ipam-pool-cidr` seguente effettua il provisioning di un CIDR a un pool IPAM.

Linux:

```
aws ec2 provision-ipam-pool-cidr \
    --ipam-pool-id ipam-pool-0533048da7d823723 \
    --cidr 10.0.0.0/24
```

Windows:

```
aws ec2 provision-ipam-pool-cidr ^
    --ipam-pool-id ipam-pool-0533048da7d823723 ^
    --cidr 10.0.0.0/24
```

Output:

```
{
    "IpamPoolCidr": {
        "Cidr": "10.0.0.0/24",
        "State": "pending-provision"
    }
}
```

Per ulteriori informazioni, consulta Provision CIDRs to a pool nella Amazon VPC IPAM User Guide.

- Per i dettagli sull'API, consulta AWS CLI Command ProvisionIpamPoolCidrReference.

`purchase-host-reservation`

Il seguente esempio di codice mostra come utilizzare`purchase-host-reservation`.

AWS CLI

Come acquistare una prenotazione dell'host dedicato

Questo esempio acquista l'offerta di prenotazione dell'host dedicato per l'host dedicato specificato nell'account.

Comando:

```
aws ec2 purchase-host-reservation --offering-id hro-03f707bf363b6b324 --host-id-
set h-013abcd2a00cbd123
```

Output:

```
{
  "TotalHourlyPrice": "1.499",
  "Purchase": [
      {
          "HourlyPrice": "1.499",
          "InstanceFamily": "m4",
          "PaymentOption": "NoUpfront",
          "HostIdSet": [
              "h-013abcd2a00cbd123"
          ],
          "HostReservationId": "hr-0d418a3a4ffc669ae",
          "UpfrontPrice": "0.000",
          "Duration": 31536000
      }
  ],
  "TotalUpfrontPrice": "0.000"
}
```

- Per i dettagli sull'API, consulta PurchaseHostReservation AWS CLICommand Reference.

`purchase-reserved-instances-offering`

Il seguente esempio di codice mostra come utilizzare`purchase-reserved-instances-offering`.

AWS CLI

Come acquistare un'offerta di istanza riservata

Questo comando di esempio illustra l'acquisto di un'offerta di istanza riservata, specificando l'ID dell'offerta e il numero di istanze.

Comando:

```
aws ec2 purchase-reserved-instances-offering --reserved-instances-offering-
id ec06327e-dd07-46ee-9398-75b5fexample --instance-count 3
```

Output:

```
{
  "ReservedInstancesId": "af9f760e-6f91-4559-85f7-4980eexample"
}
```

- Per i dettagli sull'API, consulta PurchaseReservedInstancesOffering AWS CLICommand Reference.

`purchase-scheduled-instances`

Il seguente esempio di codice mostra come utilizzare`purchase-scheduled-instances`.

AWS CLI

Come acquistare un'istanza pianificata

Questo esempio acquista un'istanza pianificata.

Comando:

```
aws ec2 purchase-scheduled-instances --purchase-requests file://purchase-request.json
```

Purchase-request.json:

```
[
    {
        "PurchaseToken": "eyJ2IjoiMSIsInMiOjEsImMiOi...",
        "InstanceCount": 1
    }
]
```

Output:

```
{
  "ScheduledInstanceSet": [
      {
          "AvailabilityZone": "us-west-2b",
          "ScheduledInstanceId": "sci-1234-1234-1234-1234-123456789012",
          "HourlyPrice": "0.095",
```

```
            "CreateDate": "2016-01-25T21:43:38.612Z",
            "Recurrence": {
                "OccurrenceDaySet": [
                    1
                ],
                "Interval": 1,
                "Frequency": "Weekly",
                "OccurrenceRelativeToEnd": false,
                "OccurrenceUnit": ""
            },
            "Platform": "Linux/UNIX",
            "TermEndDate": "2017-01-31T09:00:00Z",
            "InstanceCount": 1,
            "SlotDurationInHours": 32,
            "TermStartDate": "2016-01-31T09:00:00Z",
            "NetworkPlatform": "EC2-VPC",
            "TotalScheduledInstanceHours": 1696,
            "NextSlotStartTime": "2016-01-31T09:00:00Z",
            "InstanceType": "c4.large"
        }
    ]
}
```

- Per i dettagli sull'API, consulta PurchaseScheduledInstances AWS CLICommand Reference.

`reboot-instances`

Il seguente esempio di codice mostra come utilizzare`reboot-instances`.

AWS CLI

Per riavviare un'istanza Amazon EC2

Questo esempio riavvia l'istanza specificata. Se il comando va a buon fine, non viene restituito alcun output.

Comando:

```
aws ec2 reboot-instances --instance-ids i-1234567890abcdef5
```

Per ulteriori informazioni, consulta Riavvio dell'istanza nella Guida per l'utente di Amazon Elastic Compute Cloud.

- Per i dettagli sull'API, consulta RebootInstances AWS CLICommand Reference.

`register-image`

Il seguente esempio di codice mostra come utilizzare`register-image`.

AWS CLI

Esempio 1: come registrare un'AMI utilizzando un file manifesto

L'esempio `register-image` seguente registra un'AMI utilizzando il file manifesto specificato in Amazon S3.

```
aws ec2 register-image \
    --name my-image \
    --image-location amzn-s3-demo-bucket/myimage/image.manifest.xml
```

Output:

```
{
    "ImageId": "ami-1234567890EXAMPLE"
}
```

Per ulteriori informazioni, consulta Amazon Machine Images (AMI) nella Amazon EC2 User Guide.

Esempio 2: come registrare un'AMI utilizzando uno snapshot di un dispositivo root

L'esempio `register-image` seguente registra un'AMI utilizzando lo snapshot specificato di un volume root EBS come dispositivo `/dev/xvda`. La mappatura dei dispositivi a blocchi include anche un volume EBS vuoto da 100 GiB come dispositivo `/dev/xvdf`.

```
aws ec2 register-image \
    --name my-image \
    --root-device-name /dev/xvda \
    --block-device-mappings DeviceName=/dev/
xvda,Ebs={SnapshotId=snap-0db2cf683925d191f} DeviceName=/dev/
xvdf,Ebs={VolumeSize=100}
```

Output:

```
{
```

```
    "ImageId": "ami-1a2b3c4d5eEXAMPLE"
}
```

Per ulteriori informazioni, consulta Amazon Machine Images (AMI) nella Amazon EC2 User Guide.

- Per i dettagli sull'API, consulta RegisterImage AWS CLICommand Reference.

`register-instance-event-notification-attributes`

Il seguente esempio di codice mostra come utilizzare`register-instance-event-notification-attributes`.

AWS CLI

Esempio 1: come includere tutti i tag nelle notifiche di eventi

L'esempio `register-instance-event-notification-attributes` seguente include tutti i tag nelle notifiche di eventi.

```
aws ec2 register-instance-event-notification-attributes \
    --instance-tag-attribute IncludeAllTagsOfInstance=true
```

Output:

```
{
    "InstanceTagAttribute": {
        "InstanceTagKeys": [],
        "IncludeAllTagsOfInstance": true
    }
}
```

Per ulteriori informazioni, consulta Eventi pianificati per le tue istanze nella Amazon EC2 User Guide.

Esempio 2: come includere tag specifici nelle notifiche di eventi

L'esempio `register-instance-event-notification-attributes` seguente include i tag specificati nelle notifiche di eventi. Non è possibile specificare tag se `IncludeAllTagsOfInstance` è `true`.

```
aws ec2 register-instance-event-notification-attributes \
```

```
    --instance-tag-attribute InstanceTagKeys="tag-key1","tag-key2"
```

Output:

```
{
    "InstanceTagAttribute": {
        "InstanceTagKeys": [
            "tag-key1",
            "tag-key2"
        ],
        "IncludeAllTagsOfInstance": false
    }
}
```

Per ulteriori informazioni, consulta Eventi pianificati per le tue istanze nella Amazon EC2 User Guide.

- Per i dettagli sull'API, consulta RegisterInstanceEventNotificationAttributes AWS CLICommand Reference.

`register-transit-gateway-multicase-group-sources`

Il seguente esempio di codice mostra come utilizzare`register-transit-gateway-multicase-group-sources`.

AWS CLI

Come registrare un'origine con un gruppo multicast (trasmissione uno a molti) del gateway di transito.

L'esempio `register-transit-gateway-multicast-group-sources` seguente registra l'origine del gruppo di interfacce di rete specificato con un gruppo multicast (trasmissione uno a molti).

```
aws ec2 register-transit-gateway-multicast-group-sources \
    --transit-gateway-multicast-domain-id tgw-mcast-domain-0c4905cef79d6e597 \
    --group-ip-address 224.0.1.0 \
    --network-interface-ids eni-07f290fc3c090cbae
```

Output:

```
{
    "RegisteredMulticastGroupSources": {
        "TransitGatewayMulticastDomainId": "tgw-mcast-domain-0c4905cef79d6e597",
        "RegisteredNetworkInterfaceIds": [
            "eni-07f290fc3c090cbae"
        ],
        "GroupIpAddress": "224.0.1.0"
    }
}
```

Per ulteriori informazioni, consulta Registrare le origini con un gruppo multicast (trasmissione uno a molti) nella Guida per l'utente di AWS Transit Gateway.

- Per i dettagli sull'API, consulta RegisterTransitGatewayMulticaseGroupSources AWS CLICommand Reference.

`register-transit-gateway-multicast-group-members`

Il seguente esempio di codice mostra come utilizzare`register-transit-gateway-multicast-group-members`.

AWS CLI

Come visualizzare le informazioni sulle associazioni per un dominio multicast (trasmissione uno a molti) del gateway di transito

L'esempio `register-transit-gateway-multicast-group-members` seguente restituisce le associazioni per il dominio multicast (trasmissione uno a molti) specificato.

```
aws ec2 register-transit-gateway-multicast-group-members \
    --transit-gateway-multicast-domain-id tgw-mcast-domain-0c4905cef79d6e597 \
    --group-ip-address 224.0.1.0 \
    --network-interface-ids eni-0e246d32695012e81
```

Output:

```
{
    "RegisteredMulticastGroupMembers": {
        "TransitGatewayMulticastDomainId": "tgw-mcast-domain-0c4905cef79d6e597",
        "RegisteredNetworkInterfaceIds": [
```

```
            "eni-0e246d32695012e81"
        ],
        "GroupIpAddress": "224.0.1.0"
    }
}
```

Per ulteriori informazioni, consulta Domini multicast (trasmissione uno a molti) nella Guida per l'utente di Transit Gateway.

- Per i dettagli sull'API, consulta RegisterTransitGatewayMulticastGroupMembers AWS CLICommand Reference.

`register-transit-gateway-multicast-group-sources`

Il seguente esempio di codice mostra come utilizzare`register-transit-gateway-multicast-group-sources`.

AWS CLI

Come registrare un'origine con un gruppo multicast (trasmissione uno a molti) del gateway di transito.

L'esempio `register-transit-gateway-multicast-group-sources` seguente registra l'origine del gruppo di interfacce di rete specificato con un gruppo multicast (trasmissione uno a molti).

```
aws ec2 register-transit-gateway-multicast-group-sources \
    --transit-gateway-multicast-domain-id tgw-mcast-domain-0c4905cef79d6e597 \
    --group-ip-address 224.0.1.0 \
    --network-interface-ids eni-07f290fc3c090cbae
```

Output:

```
{
    "RegisteredMulticastGroupSources": {
        "TransitGatewayMulticastDomainId": "tgw-mcast-domain-0c4905cef79d6e597",
        "RegisteredNetworkInterfaceIds": [
            "eni-07f290fc3c090cbae"
        ],
        "GroupIpAddress": "224.0.1.0"
    }
```

```
}
```

Per ulteriori informazioni, consulta Domini multicast (trasmissione uno a molti) nella Guida di Transit Gateway.

- Per i dettagli sull'API, consulta RegisterTransitGatewayMulticastGroupSources AWS CLICommand Reference.

`reject-transit-gateway-peering-attachment`

Il seguente esempio di codice mostra come utilizzare`reject-transit-gateway-peering-attachment`.

AWS CLI

Come rifiutare un collegamento peering del gateway di transito

L'esempio `reject-transit-gateway-peering-attachment` seguente rifiuta la richiesta di accessorio di peering del gateway di transito specificato. Il parametro `--region` specifica la Regione in cui si trova il gateway di transito accettante.

```
aws ec2 reject-transit-gateway-peering-attachment \
    --transit-gateway-attachment-id tgw-attach-4455667788aabbccd \
    --region us-east-2
```

Output:

```
{
    "TransitGatewayPeeringAttachment": {
        "TransitGatewayAttachmentId": "tgw-attach-4455667788aabbccd",
        "RequesterTgwInfo": {
            "TransitGatewayId": "tgw-123abc05e04123abc",
            "OwnerId": "123456789012",
            "Region": "us-west-2"
        },
        "AccepterTgwInfo": {
            "TransitGatewayId": "tgw-11223344aabbcc112",
            "OwnerId": "123456789012",
            "Region": "us-east-2"
        },
        "State": "rejecting",
        "CreationTime": "2019-12-09T11:50:31.000Z"
```

```
    }
}
```

Per ulteriori informazioni, consulta Accessori di peering del gateway di transito nella Guida di Transit Gateway.

- Per i dettagli sull'API, consulta RejectTransitGatewayPeeringAttachment AWS CLICommand Reference.

`reject-transit-gateway-vpc-attachment`

Il seguente esempio di codice mostra come utilizzare`reject-transit-gateway-vpc-attachment`.

AWS CLI

Come rifiutare un collegamento VPC del gateway di transito

L'esempio `reject-transit-gateway-vpc-attachment` seguente rifiuta il collegamento VPC del gateway di transito specificato.

```
aws ec2 reject-transit-gateway-vpc-attachment \
    --transit-gateway-attachment-id tgw-attach-0a34fe6b4fEXAMPLE
```

Output:

```
{
    "TransitGatewayVpcAttachment": {
        "TransitGatewayAttachmentId": "tgw-attach-0a34fe6b4fEXAMPLE",
        "TransitGatewayId": "tgw-0262a0e521EXAMPLE",
        "VpcId": "vpc-07e8ffd50fEXAMPLE",
        "VpcOwnerId": "111122223333",
        "State": "pending",
        "SubnetIds": [
            "subnet-0752213d59EXAMPLE"
        ],
        "CreationTime": "2019-07-10T17:33:46.000Z",
        "Options": {
            "DnsSupport": "enable",
            "Ipv6Support": "disable"
        }
    }
```

```
}
```

Per ulteriori informazioni, consulta Collegamenti del gateway di transito alla VPN a un VPC nella Guida di Transit Gateway.

- Per i dettagli sull'API, consulta RejectTransitGatewayVpcAttachment AWS CLICommand Reference.

`reject-transit-gateway-vpc-attachments`

Il seguente esempio di codice mostra come utilizzare`reject-transit-gateway-vpc-attachments`.

AWS CLI

Come rifiutare un collegamento VPC del gateway di transito

L'esempio `reject-transit-gateway-vpc-attachment` seguente rifiuta il collegamento VPC del gateway di transito specificato.

```
aws ec2 reject-transit-gateway-vpc-attachment \
    --transit-gateway-attachment-id tgw-attach-0a34fe6b4fEXAMPLE
```

Output:

```
{
    "TransitGatewayVpcAttachment": {
        "TransitGatewayAttachmentId": "tgw-attach-0a34fe6b4fEXAMPLE",
        "TransitGatewayId": "tgw-0262a0e521EXAMPLE",
        "VpcId": "vpc-07e8ffd50fEXAMPLE",
        "VpcOwnerId": "111122223333",
        "State": "pending",
        "SubnetIds": [
            "subnet-0752213d59EXAMPLE"
        ],
        "CreationTime": "2019-07-10T17:33:46.000Z",
        "Options": {
            "DnsSupport": "enable",
            "Ipv6Support": "disable"
        }
    }
}
```

Per ulteriori informazioni, consulta Collegamenti del gateway di transito alla VPN a un VPC nella Guida di Transit Gateway.

- Per i dettagli sull'API, consulta RejectTransitGatewayVpcAttachments AWS CLICommand Reference.

`reject-vpc-endpoint-connections`

Il seguente esempio di codice mostra come utilizzare`reject-vpc-endpoint-connections`.

AWS CLI

Come rifiutare le richieste di connessione endpoint dell'interfaccia

Questo esempio rifiuta la richiesta di connessione endpoint specificata per il servizio endpoint specificato.

Comando:

```
aws ec2 reject-vpc-endpoint-connections --service-id vpce-svc-03d5ebb7d9579a2b3 --
vpc-endpoint-ids vpce-0c1308d7312217abc
```

Output:

```
{
  "Unsuccessful": []
}
```

- Per i dettagli sull'API, consulta RejectVpcEndpointConnections AWS CLICommand Reference.

`reject-vpc-peering-connection`

Il seguente esempio di codice mostra come utilizzare`reject-vpc-peering-connection`.

AWS CLI

Come rifiutare una connessione peering VPC

Questo esempio rifiuta la richiesta di connessione peering VPC specificata.

Comando:

```
aws ec2 reject-vpc-peering-connection --vpc-peering-connection-id pcx-1a2b3c4d
```

Output:

```
{
    "Return": true
}
```

- Per i dettagli sull'API, consulta RejectVpcPeeringConnection AWS CLICommand Reference.

`release-address`

Il seguente esempio di codice mostra come utilizzare`release-address`.

AWS CLI

Per rilasciare un indirizzo IP elastico per EC2 -Classic

Questo esempio rilascia un indirizzo IP elastico da utilizzare con le istanze in EC2 -Classic. Se il comando va a buon fine, non viene restituito alcun output.

Comando:

```
aws ec2 release-address --public-ip 198.51.100.0
```

Per rilasciare un indirizzo IP elastico per EC2 -VPC

Nell'esempio seguente viene rilasciato un indirizzo IP elastico per l'utilizzo con le istanze in un VPC. Se il comando va a buon fine, non viene restituito alcun output.

Comando:

```
aws ec2 release-address --allocation-id eipalloc-64d5890a
```

- Per i dettagli sull'API, consulta AWS CLI Command ReleaseAddressReference.

`release-hosts`

Il seguente esempio di codice mostra come utilizzare`release-hosts`.

AWS CLI

Come rilasciare un host dedicato dall'account

Come rilasciare un host dedicato dall'account. Le istanze presenti sull'host devono essere arrestate o terminate prima che l'host possa essere rilasciato.

Comando:

```
aws ec2 release-hosts --host-id=h-0029d6e3cacf1b3da
```

Output:

```
{
    "Successful":  [
        "h-0029d6e3cacf1b3da"
         ],
  "Unsuccessful": []

 }
```

- Per i dettagli sull'API, consulta ReleaseHosts AWS CLICommand Reference.

`release-ipam-pool-allocation`

Il seguente esempio di codice mostra come utilizzare`release-ipam-pool-allocation`.

AWS CLI

Come rilasciare un'allocazione del pool IPAM

In questo esempio, l'utente è un amministratore delegato IPAM che ha cercato di eliminare un pool IPAM, ma ha ricevuto un errore che indica che non è possibile eliminare il pool mentre il pool contiene allocazioni. Questo comando viene utilizzato per rilasciare un'allocazione del pool.

Tenere presente quanto segue:

È possibile utilizzare questo comando solo per le allocazioni personalizzate. Per rimuovere un'allocazione per una risorsa senza eliminarla, impostane lo stato monitorato su false utilizzando modify-ipam-resource-cidr.Per completare questa richiesta, avrai bisogno dell'ID del pool IPAM,

che puoi utilizzare. describe-ipam-pools È inoltre necessario l'ID di allocazione, che è possibile ottenere con get-ipam-pool-allocations. Se non si desidera rimuovere le allocazioni una per una, è possibile utilizzare `--cascade option` quando si elimina un pool IPAM per rilasciare automaticamente tutte le allocazioni nel pool prima di eliminarlo. Esistono diversi prerequisiti prima di eseguire questo comando. Per ulteriori informazioni, consulta Rilasciare un'assegnazione nella Guida per l'utente IPAM di Amazon VPC. La `--region` in cui si esegue questo comando deve essere la l'impostazione locale del pool IPAM in cui si trova l'allocazione.

L'esempio `release-ipam-pool-allocation` seguente rilascia un'allocazione del pool IPAM.

```
aws ec2 release-ipam-pool-allocation \
    --ipam-pool-id ipam-pool-07bdd12d7c94e4693 \
    --cidr 10.0.0.0/23 \
    --ipam-pool-allocation-id ipam-pool-alloc-0e66a1f730da54791b99465b79e7d1e89 \
    --region us-west-1
```

Output:

```
{
    "Success": true
}
```

Dopo aver rilasciato un'allocazione, potresti voler eseguirla. delete-ipam-pool

- Per i dettagli sull'API, consulta ReleaseIpamPoolAllocation AWS CLICommand Reference.

`replace-iam-instance-profile-association`

Il seguente esempio di codice mostra come utilizzare`replace-iam-instance-profile-association`.

AWS CLI

Come sostituire un profilo dell'istanza IAM per un'istanza

In questo esempio il profilo dell'istanza IAM rappresentato dall'associazione `iip-assoc-060bae234aac2e7fa` viene sostituito con il profilo dell'istanza IAM denominato `AdminRole`.

```
aws ec2 replace-iam-instance-profile-association \
```

```
    --iam-instance-profile Name=AdminRole \
    --association-id iip-assoc-060bae234aac2e7fa
```

Output:

```
{
    "IamInstanceProfileAssociation": {
        "InstanceId": "i-087711ddaf98f9489",
        "State": "associating",
        "AssociationId": "iip-assoc-0b215292fab192820",
        "IamInstanceProfile": {
            "Id": "AIPAJLNLDX3AMYZNWYYAY",
            "Arn": "arn:aws:iam::123456789012:instance-profile/AdminRole"
        }
    }
}
```

- Per i dettagli sull'API, consulta ReplaceIamInstanceProfileAssociation AWS CLICommand Reference.

`replace-network-acl-association`

Il seguente esempio di codice mostra come utilizzare`replace-network-acl-association`.

AWS CLI

Come sostituire la lista di controllo degli accessi alla rete (ACL) associata a una sottorete

Questo esempio associa la lista di controllo degli accessi alla rete (ACL) specificata alla sottorete dell'associazione lista di controllo degli accessi alla rete (ACL) specificata.

Comando:

```
aws ec2 replace-network-acl-association --association-id aclassoc-e5b95c8c --
network-acl-id acl-5fb85d36
```

Output:

```
{
    "NewAssociationId": "aclassoc-3999875b"
```

```
}
```

- Per i dettagli sull'API, consulta ReplaceNetworkAclAssociation AWS CLICommand Reference.

`replace-network-acl-entry`

Il seguente esempio di codice mostra come utilizzare`replace-network-acl-entry`.

AWS CLI

Come sostituire una voce della lista di controllo degli accessi alla rete (ACL)

Questo esempio sostituisce una voce per la lista di controllo degli accessi alla rete (ACL) specificata. La nuova regola 100 consente il traffico in entrata da 203.0.113.12/24 sulla porta UDP 53 (DNS) in qualsiasi sottorete associata.

Comando:

```
aws ec2 replace-network-acl-entry --network-acl-id acl-5fb85d36 --ingress --rule-number 100 --protocol udp --port-range From=53,To=53 --cidr-block 203.0.113.12/24 --rule-action allow
```

- Per i dettagli sull'API, consulta ReplaceNetworkAclEntry AWS CLICommand Reference.

`replace-route-table-association`

Il seguente esempio di codice mostra come utilizzare`replace-route-table-association`.

AWS CLI

Come sostituire una tabella di routing associata a una sottorete

Questo esempio associa la tabella di routing specificata alla sottorete per l'associazione di tabelle di routing specificata.

Comando:

```
aws ec2 replace-route-table-association --association-id rtbassoc-781d0d1a --route-table-id rtb-22574640
```

Output:

```
{
    "NewAssociationId": "rtbassoc-3a1f0f58"
}
```

- Per i dettagli sull'API, consulta ReplaceRouteTableAssociation AWS CLICommand Reference.

`replace-route`

Il seguente esempio di codice mostra come utilizzare`replace-route`.

AWS CLI

Come sostituire una route

Questo esempio sostituisce la route specificata per la tabella di routing specificata. La nuova route corrisponde al CIDR specificato e invia il traffico al gateway privato virtuale specificato. Se il comando va a buon fine, non viene restituito alcun output.

Comando:

```
aws ec2 replace-route --route-table-id rtb-22574640 --destination-cidr-
block 10.0.0.0/16 --gateway-id vgw-9a4cacf3
```

- Per i dettagli sull'API, consulta ReplaceRoute AWS CLICommand Reference.

`replace-transit-gateway-route`

Il seguente esempio di codice mostra come utilizzare`replace-transit-gateway-route`.

AWS CLI

Come sostituire la route specificata nella tabella di routing del gateway di transito specificata

L'esempio `replace-transit-gateway-route` seguente sostituisce la route nella tabella di routing del gateway di transito specificata.

```
aws ec2 replace-transit-gateway-route \
    --destination-cidr-block 10.0.2.0/24 \
    --transit-gateway-attachment-id tgw-attach-09b52ccdb5EXAMPLE \
    --transit-gateway-route-table-id tgw-rtb-0a823edbdeEXAMPLE
```

Output:

```
{
    "Route": {
        "DestinationCidrBlock": "10.0.2.0/24",
        "TransitGatewayAttachments": [
            {
                "ResourceId": "vpc-4EXAMPLE",
                "TransitGatewayAttachmentId": "tgw-attach-09b52ccdb5EXAMPLE",
                "ResourceType": "vpc"
            }
        ],
        "Type": "static",
        "State": "active"
    }
}
```

Per ulteriori informazioni, consulta Tabelle di routing del gateway di transito nella Guida di Transit Gateway.

- Per i dettagli sull'API, consulta ReplaceTransitGatewayRoute AWS CLICommand Reference.

`report-instance-status`

Il seguente esempio di codice mostra come utilizzare`report-instance-status`.

AWS CLI

Come segnalare il feedback sullo stato di un'istanza

Questo comando di esempio restituisce il feedback sullo stato per l'istanza specificata.

Comando:

```
aws ec2 report-instance-status --instances i-1234567890abcdef0 --status impaired --
reason-codes unresponsive
```

- Per i dettagli sull'API, consulta ReportInstanceStatus AWS CLICommand Reference.

`request-spot-fleet`

Il seguente esempio di codice mostra come utilizzare`request-spot-fleet`.

AWS CLI

Come richiedere un parco istanze spot nella sottorete al prezzo più basso

Questo comando di esempio crea una richiesta di parco istanze spot con due specifiche di avvio che differiscono solo a livello di sottorete. Il parco istanze spot avvia le istanze nella sottorete specificata con il prezzo più basso. Se le istanze vengono avviate in un VPC predefinito, esse ricevono un indirizzo IP pubblico per impostazione predefinita. Se le istanze vengono avviate in un VPC non predefinito, non ricevono un indirizzo IP pubblico per impostazione predefinita.

Nota che in una richiesta di parco istanze spot non è possibile specificare sottoreti diverse nella stessa zona di disponibilità.

Comando:

```
aws ec2 request-spot-fleet --spot-fleet-request-config file://config.json
```

Config.json:

```
{
  "SpotPrice": "0.04",
  "TargetCapacity": 2,
  "IamFleetRole": "arn:aws:iam::123456789012:role/my-spot-fleet-role",
  "LaunchSpecifications": [
      {
          "ImageId": "ami-1a2b3c4d",
          "KeyName": "my-key-pair",
          "SecurityGroups": [
              {
                  "GroupId": "sg-1a2b3c4d"
              }
          ],
          "InstanceType": "m3.medium",
          "SubnetId": "subnet-1a2b3c4d, subnet-3c4d5e6f",
          "IamInstanceProfile": {
              "Arn": "arn:aws:iam::123456789012:instance-profile/my-iam-role"
          }
      }
  ]
}
```

Output:

```
{
  "SpotFleetRequestId": "sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE"
}
```

Come richiedere un parco istanze spot nella zona di disponibilità al prezzo più basso

Questo comando di esempio crea una richiesta di parco istanze spot con due specifiche di avvio che differiscono solo a livello di zona di disponibilità. Il parco istanze spot avvia le istanze nella zona di disponibilità specificata con il prezzo più basso. Se il tuo account supporta solo EC2 - VPC, Amazon EC2 avvia le istanze Spot nella sottorete predefinita della zona di disponibilità. Se il tuo account supporta EC2 -Classic, Amazon EC2 avvia le istanze in EC2 -Classic nella zona di disponibilità.

Comando:

```
aws ec2 request-spot-fleet --spot-fleet-request-config file://config.json
```

Config.json:

```
{
  "SpotPrice": "0.04",
  "TargetCapacity": 2,
  "IamFleetRole": "arn:aws:iam::123456789012:role/my-spot-fleet-role",
  "LaunchSpecifications": [
      {
          "ImageId": "ami-1a2b3c4d",
          "KeyName": "my-key-pair",
          "SecurityGroups": [
              {
                  "GroupId": "sg-1a2b3c4d"
              }
          ],
          "InstanceType": "m3.medium",
          "Placement": {
              "AvailabilityZone": "us-west-2a, us-west-2b"
          },
          "IamInstanceProfile": {
              "Arn": "arn:aws:iam::123456789012:instance-profile/my-iam-role"
          }
      }
  ]
```

```
}
```

Come avviare istanze spot in una sottorete e assegnare loro indirizzi IP pubblici

Questo comando di esempio assegna indirizzi pubblici alle istanze avviate in un VPC non predefinito. Nota che quando si specifica un'interfaccia di rete, bisogna includere l'ID della sottorete e l'ID del gruppo di sicurezza utilizzando l'interfaccia di rete.

Comando:

```
aws ec2 request-spot-fleet --spot-fleet-request-config file://config.json
```

Config.json:

```
{
  "SpotPrice": "0.04",
  "TargetCapacity": 2,
  "IamFleetRole": "arn:aws:iam::123456789012:role/my-spot-fleet-role",
  "LaunchSpecifications": [
      {
          "ImageId": "ami-1a2b3c4d",
          "KeyName": "my-key-pair",
          "InstanceType": "m3.medium",
          "NetworkInterfaces": [
              {
                  "DeviceIndex": 0,
                  "SubnetId": "subnet-1a2b3c4d",
                  "Groups": [ "sg-1a2b3c4d" ],
                  "AssociatePublicIpAddress": true
              }
          ],
          "IamInstanceProfile": {
              "Arn": "arn:aws:iam::880185128111:instance-profile/my-iam-role"
          }
      }
  ]
}
```

Come richiedere un parco istanze spot utilizzando la strategia di allocazione diversificata

Questo comando di esempio crea una richiesta di parco istanze spot che avvia 30 istanze utilizzando la strategia di allocazione diversificata. Le specifiche di avvio differiscono in base al

tipo di istanza. Il parco istanze spot distribuisce le istanze tra le specifiche di avvio, in modo che ci siano 10 istanze di ogni tipo.

Comando:

```
aws ec2 request-spot-fleet --spot-fleet-request-config file://config.json
```

Config.json:

```
{
  "SpotPrice": "0.70",
  "TargetCapacity": 30,
  "AllocationStrategy": "diversified",
  "IamFleetRole": "arn:aws:iam::123456789012:role/my-spot-fleet-role",
  "LaunchSpecifications": [
      {
          "ImageId": "ami-1a2b3c4d",
          "InstanceType": "c4.2xlarge",
          "SubnetId": "subnet-1a2b3c4d"
      },
      {
          "ImageId": "ami-1a2b3c4d",
          "InstanceType": "m3.2xlarge",
          "SubnetId": "subnet-1a2b3c4d"
      },
      {
          "ImageId": "ami-1a2b3c4d",
          "InstanceType": "r3.2xlarge",
          "SubnetId": "subnet-1a2b3c4d"
      }
  ]
}
```

Per ulteriori informazioni, consulta Richieste di parchi istanze spot nella Guida per l'utente di Amazon Elastic Compute Cloud.

- Per i dettagli sull'API, consulta RequestSpotFleetCommand Reference.AWS CLI

`request-spot-instances`

Il seguente esempio di codice mostra come utilizzare`request-spot-instances`.

AWS CLI

Come richiedere istanze spot

Questo comando di esempio crea una richiesta una tantum di istanze spot per cinque istanze nella zona di disponibilità specificata. Se il tuo account supporta solo EC2 -VPC, Amazon EC2 avvia le istanze nella sottorete predefinita della zona di disponibilità specificata. Se il tuo account supporta EC2 -Classic, Amazon EC2 avvia le istanze in EC2 -Classic nella zona di disponibilità specificata.

Comando:

```
aws ec2 request-spot-instances --spot-price "0.03" --instance-count 5 --type "one-time" --launch-specification file://specification.json
```

Specification.json:

```
{
  "ImageId": "ami-1a2b3c4d",
  "KeyName": "my-key-pair",
  "SecurityGroupIds": [ "sg-1a2b3c4d" ],
  "InstanceType": "m3.medium",
  "Placement": {
    "AvailabilityZone": "us-west-2a"
  },
  "IamInstanceProfile": {
      "Arn": "arn:aws:iam::123456789012:instance-profile/my-iam-role"
  }
}
```

Output:

```
{
  "SpotInstanceRequests": [
      {
          "Status": {
              "UpdateTime": "2014-03-25T20:54:21.000Z",
              "Code": "pending-evaluation",
              "Message": "Your Spot request has been submitted for review, and is
 pending evaluation."
          },
```

```
            "ProductDescription": "Linux/UNIX",
            "SpotInstanceRequestId": "sir-df6f405d",
            "State": "open",
            "LaunchSpecification": {
                "Placement": {
                    "AvailabilityZone": "us-west-2a"
                },
                "ImageId": "ami-1a2b3c4d",
                "KeyName": "my-key-pair",
                "SecurityGroups": [
                    {
                        "GroupName": "my-security-group",
                        "GroupId": "sg-1a2b3c4d"
                    }
                ],
                "Monitoring": {
                    "Enabled": false
                },
                "IamInstanceProfile": {
                    "Arn": "arn:aws:iam::123456789012:instance-profile/my-iam-role"
                },
                "InstanceType": "m3.medium"
            },
            "Type": "one-time",
            "CreateTime": "2014-03-25T20:54:20.000Z",
            "SpotPrice": "0.050000"
        },
        ...
    ]
}
```

Questo comando di esempio crea una richiesta una tantum di istanze spot per cinque istanze nella sottorete specificata. Amazon EC2 avvia le istanze nella sottorete specificata. Se il VPC corrente non è un VPC predefinito, le istanze non ricevono un indirizzo IP pubblico per impostazione predefinita.

Comando:

```
aws ec2 request-spot-instances --spot-price "0.050" --instance-count 5 --type "one-time" --launch-specification file://specification.json
```

Specification.json:

```
{
  "ImageId": "ami-1a2b3c4d",
  "SecurityGroupIds": [ "sg-1a2b3c4d" ],
  "InstanceType": "m3.medium",
  "SubnetId": "subnet-1a2b3c4d",
  "IamInstanceProfile": {
      "Arn": "arn:aws:iam::123456789012:instance-profile/my-iam-role"
  }
}
```

Output:

```
{
  "SpotInstanceRequests": [
      {
          "Status": {
             "UpdateTime": "2014-03-25T22:21:58.000Z",
             "Code": "pending-evaluation",
             "Message": "Your Spot request has been submitted for review, and is
 pending evaluation."
          },
          "ProductDescription": "Linux/UNIX",
          "SpotInstanceRequestId": "sir-df6f405d",
          "State": "open",
          "LaunchSpecification": {
             "Placement": {
                 "AvailabilityZone": "us-west-2a"
             }
             "ImageId": "ami-1a2b3c4d"
             "SecurityGroups": [
                 {
                     "GroupName": "my-security-group",
                     "GroupID": "sg-1a2b3c4d"
                 }
             ]
             "SubnetId": "subnet-1a2b3c4d",
             "Monitoring": {
                 "Enabled": false
             },
             "IamInstanceProfile": {
                 "Arn": "arn:aws:iam::123456789012:instance-profile/my-iam-role"
             },
             "InstanceType": "m3.medium",
```

```
            },
            "Type": "one-time",
            "CreateTime": "2014-03-25T22:21:58.000Z",
            "SpotPrice": "0.050000"
        },
        ...
    ]
}
```

Questo esempio assegna un indirizzo IP pubblico alle istanze spot avviate in un VPC non predefinito. Nota che quando si specifica un'interfaccia di rete, bisogna includere l'ID della sottorete e l'ID del gruppo di sicurezza utilizzando l'interfaccia di rete.

Comando:

```
aws ec2 request-spot-instances --spot-price "0.050" --instance-count 1 --type "one-time" --launch-specification file://specification.json
```

Specification.json:

```
{
  "ImageId": "ami-1a2b3c4d",
  "KeyName": "my-key-pair",
  "InstanceType": "m3.medium",
  "NetworkInterfaces": [
    {
      "DeviceIndex": 0,
      "SubnetId": "subnet-1a2b3c4d",
      "Groups": [ "sg-1a2b3c4d" ],
      "AssociatePublicIpAddress": true
    }
  ],
  "IamInstanceProfile": {
      "Arn": "arn:aws:iam::123456789012:instance-profile/my-iam-role"
  }
}
```

- Per i dettagli sull'API, consulta Command RequestSpotInstancesReference AWS CLI .

`reset-address-attribute`

Il seguente esempio di codice mostra come utilizzare`reset-address-attribute`.

AWS CLI

Come ripristinare l'attributo del nome di dominio associato a un indirizzo IP elastico

Gli esempi `reset-address-attribute` seguenti ripristinano l'attributo del nome di dominio di un indirizzo IP elastico.

Linux:

```
aws ec2 reset-address-attribute \
    --allocation-id eipalloc-abcdef01234567890 \
    --attribute domain-name
```

Windows:

```
aws ec2 reset-address-attribute ^
    --allocation-id eipalloc-abcdef01234567890 ^
    --attribute domain-name
```

Output:

```
{
    "Addresses": [
        {
            "PublicIp": "192.0.2.0",
            "AllocationId": "eipalloc-abcdef01234567890",
            "PtrRecord": "example.com."
            "PtrRecordUpdate": {
                "Value": "example.net.",
                "Status": "PENDING"
        }
    ]
}
```

Per monitorare la modifica in sospeso, consulta il riferimento describe-addresses-attributeai comandi della AWS CLI.

- Per i dettagli sull'API, consulta AWS CLI Command ResetAddressAttributeReference.

`reset-ebs-default-kms-key-id`

Il seguente esempio di codice mostra come utilizzare`reset-ebs-default-kms-key-id`.

AWS CLI

Come reimpostare la chiave gestita dal cliente (CMK) predefinita per la crittografia EBS

L'`reset-ebs-default-kms-key-id`esempio seguente reimposta la crittografia CMK predefinita per EBS per il tuo AWS account nella regione corrente.

```
aws ec2 reset-ebs-default-kms-key-id
```

Output:

```
{
   "KmsKeyId": "arn:aws:kms:us-west-2:123456789012:key/8c5b2c63-b9bc-45a3-
a87a-5513eEXAMPLE"
}
```

- Per i dettagli sull'API, consulta Command ResetEbsDefaultKmsKeyIdReference AWS CLI .

`reset-fpga-image-attribute`

Il seguente esempio di codice mostra come utilizzare`reset-fpga-image-attribute`.

AWS CLI

Come reimpostare gli attributi di un'immagine FPGA di Amazon (AFI)

Questo esempio ripristina le autorizzazioni di caricamento per l'AFI specificato.

Comando:

```
aws ec2 reset-fpga-image-attribute --fpga-image-id afi-0d123e123bfc85abc --
attribute loadPermission
```

Output:

```
{
  "Return": true
}
```

- Per i dettagli sull'API, consulta ResetFpgaImageAttribute AWS CLICommand Reference.

`reset-image-attribute`

Il seguente esempio di codice mostra come utilizzare`reset-image-attribute`.

AWS CLI

Come reimpostare l'attributo launchPermission

Questo esempio ripristina il valore predefinito dell'attributo `launchPermission` per l'AMI specificata. Per impostazione predefinita, AMIs sono private. Se il comando va a buon fine, non viene restituito alcun output.

Comando:

```
aws ec2 reset-image-attribute --image-id ami-5731123e --attribute launchPermission
```

- Per i dettagli sull'API, consulta ResetImageAttribute AWS CLICommand Reference.

`reset-instance-attribute`

Il seguente esempio di codice mostra come utilizzare`reset-instance-attribute`.

AWS CLI

Per reimpostare l' sourceDestCheck attributo

Questo esempio ripristina l'attributo `sourceDestCheck` dell'istanza specificata. L'istanza deve trovarsi in un VPC. Se il comando va a buon fine, non viene restituito alcun output.

Comando:

```
aws ec2 reset-instance-attribute --instance-id i-1234567890abcdef0 --
attribute sourceDestCheck
```

Come reimpostare l'attributo kernel

Questo esempio ripristina l'attributo `kernel` dell'istanza specificata. L'istanza deve essere nello stato `stopped`. Se il comando va a buon fine, non viene restituito alcun output.

Comando:

```
aws ec2 reset-instance-attribute --instance-id i-1234567890abcdef0 --
attribute kernel
```

Come reimpostare l'attributo ramdisk

Questo esempio ripristina l'attributo `ramdisk` dell'istanza specificata. L'istanza deve essere nello stato `stopped`. Se il comando va a buon fine, non viene restituito alcun output.

Comando:

```
aws ec2 reset-instance-attribute --instance-id i-1234567890abcdef0 --
attribute ramdisk
```

- Per i dettagli sull'API, consulta ResetInstanceAttribute AWS CLICommand Reference.

`reset-network-interface-attribute`

Il seguente esempio di codice mostra come utilizzare`reset-network-interface-attribute`.

AWS CLI

Come reimpostare un attributo dell'interfaccia di rete

L'`reset-network-interface-attribute`esempio seguente reimposta il valore dell'attributo source/destination `true` checking su.

```
aws ec2 reset-network-interface-attribute \
    --network-interface-id eni-686ea200 \
    --source-dest-check
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, vedere ResetNetworkInterfaceAttributein AWS CLI Command Reference.

`reset-snapshot-attribute`

Il seguente esempio di codice mostra come utilizzare`reset-snapshot-attribute`.

AWS CLI

Come reimpostare un attributo dello snapshot

Questo esempio ripristina le autorizzazioni di creazione del volume per lo snapshot `snap-1234567890abcdef0`. Se il comando va a buon fine, non viene restituito alcun output.

Comando:

```
aws ec2 reset-snapshot-attribute --snapshot-id snap-1234567890abcdef0 --
attribute createVolumePermission
```

- Per i dettagli sull'API, consulta ResetSnapshotAttribute AWS CLICommand Reference.

`restore-address-to-classic`

Il seguente esempio di codice mostra come utilizzare`restore-address-to-classic`.

AWS CLI

Per ripristinare un indirizzo su EC2 -Classic

Questo esempio ripristina l'indirizzo IP elastico 198.51.100.0 sulla piattaforma -Classic. EC2

Comando:

```
aws ec2 restore-address-to-classic --public-ip 198.51.100.0
```

Output:

```
{
  "Status": "MoveInProgress",
  "PublicIp": "198.51.100.0"
}
```

- Per i dettagli sull'API, consulta Command Reference. RestoreAddressToClassicAWS CLI

`restore-image-from-recycle-bin`

Il seguente esempio di codice mostra come utilizzare`restore-image-from-recycle-bin`.

AWS CLI

Come ripristinare un'immagine dal Cestino

Il comando `restore-image-from-recycle-bin` seguente ripristina l'AMI ami-0111222333444abcd dal Cestino.

```
aws ec2 restore-image-from-recycle-bin \
    --image-id ami-0111222333444abcd
```

Output:

```
{
    "Return": true
}
```

Per ulteriori informazioni, consulta Recover deleted AMIs from the Recycle Bin nella Amazon EBS User Guide.

- Per i dettagli sull'API, consulta AWS CLI Command RestoreImageFromRecycleBinReference.

`restore-managed-prefix-list-version`

Il seguente esempio di codice mostra come utilizzare`restore-managed-prefix-list-version`.

AWS CLI

US-west-2** Per ripristinare una versione dell'elenco di prefissi**

Il comando `restore-managed-prefix-list-version` seguente ripristina le voci della versione 1 dell'elenco di prefissi specificato.

```
aws ec2 restore-managed-prefix-list-version \
    --prefix-list-id pl-0123456abcabcabc1 \
    --current-version 2 \
    --previous-version 1
```

Output:

```
{
    "PrefixList": {
        "PrefixListId": "pl-0123456abcabcabc1",
        "AddressFamily": "IPv4",
        "State": "restore-in-progress",
        "PrefixListArn": "arn:aws:ec2:us-west-2:123456789012:prefix-list/
pl-0123456abcabcabc1",
```

```
        "PrefixListName": "vpc-cidrs",
        "MaxEntries": 10,
        "Version": 2,
        "OwnerId": "123456789012"
    }
}
```

Per ulteriori informazioni, consulta Elenchi di prefissi gestiti nella Guida per l'utente di Amazon VPC.

- Per i dettagli sull'API, consulta RestoreManagedPrefixListVersion AWS CLICommand Reference.

`restore-snapshot-from-recycle-bin`

Il seguente esempio di codice mostra come utilizzare`restore-snapshot-from-recycle-bin`.

AWS CLI

Come ripristinare gli snapshot dal Cestino

Il comando `restore-snapshot-from-recycle-bin` seguente ripristina uno snapshot dal Cestino. Quando si ripristina uno snapshot dal Cestino di riciclaggio, lo snapshot diventa immediatamente disponibile per l'uso e viene rimosso dal Cestino. Dopo averlo ripristinato, potrà essere utilizzato nello stesso modo in cui qualsiasi altro snapshot viene utilizzato nel proprio account.

```
aws ec2 restore-snapshot-from-recycle-bin \
    --snapshot-id snap-01234567890abcdef
```

Questo comando non produce alcun output.

Per ulteriori informazioni sul Cestino, consulta Ripristino degli snapshot dal Cestino nella Guida per l'utente di Amazon EBS.

- Per i dettagli sull'API, consulta RestoreSnapshotFromRecycleBin AWS CLICommand Reference.

`restore-snapshot-tier`

Il seguente esempio di codice mostra come utilizzare`restore-snapshot-tier`.

AWS CLI

Esempio 1: come ripristinare definitivamente uno snapshot archiviato

Il comando `restore-snapshot-tier` seguente ripristina definitivamente lo snapshot specificato. Specifica `--snapshot-id` e includi l'opzione `permanent-restore`.

```
aws ec2 restore-snapshot-tier \
    --snapshot-id snap-01234567890abcedf \
    --permanent-restore
```

Output:

```
{
    "SnapshotId": "snap-01234567890abcedf",
    "IsPermanentRestore": true
}
```

Per ulteriori informazioni sull'archiviazione degli snapshot, consulta Archiviazione degli snapshot Amazon EBS nella Guida per l'utente di Amazon EBS.

Esempio 2: come ripristinare temporaneamente uno snapshot archiviato

L'esempio `restore-snapshot-tier` seguente ripristina temporaneamente lo snapshot specificato. Omettere l'opzione `--permanent-restore`. Specifica `--snapshot-id` e, per `temporary-restore-days`, specifica il numero di giorni durante i quali ripristinare lo snapshot. `temporary-restore-days` deve essere specificato in giorni. L'intervallo consentito è `1 - 180`. Se non si specifica un valore, sarà impostato su 1 giorno.

```
aws ec2 restore-snapshot-tier \
    --snapshot-id snap-01234567890abcedf \
    --temporary-restore-days 5
```

Output:

```
{
    "SnapshotId": "snap-01234567890abcedf",
    "RestoreDuration": 5,
    "IsPermanentRestore": false
}
```

Per ulteriori informazioni sull'archiviazione degli snapshot, consulta Archiviazione degli snapshot Amazon EBS nella Guida per l'utente di Amazon EBS.

Esempio 3: come modificare il periodo di ripristino

L'esempio `restore-snapshot-tier` seguente modifica il periodo di ripristino per lo snapshot specificato su `10` giorni.

```
aws ec2 restore-snapshot-tier \
    --snapshot-id snap-01234567890abcedf
    --temporary-restore-days 10
```

Output:

```
{
    "SnapshotId": "snap-01234567890abcedf",
    "RestoreDuration": 10,
    "IsPermanentRestore": false
}
```

Per ulteriori informazioni sull'archiviazione degli snapshot, consulta Archiviazione degli snapshot Amazon EBS nella Guida per l'utente di Amazon EBS.

Esempio 4: come modificare il tipo di ripristino

Il comando `restore-snapshot-tier` seguente modifica il tipo di ripristino per lo snapshot specificato da temporaneo a permanente.

```
aws ec2 restore-snapshot-tier \
    --snapshot-id snap-01234567890abcedf
    --permanent-restore
```

Output:

```
{
    "SnapshotId": "snap-01234567890abcedf",
    "IsPermanentRestore": true
}
```

Per ulteriori informazioni sull'archiviazione degli snapshot, consulta Archiviazione degli snapshot Amazon EBS nella Guida per l'utente di Amazon EBS.

- Per i dettagli sull'API, consulta RestoreSnapshotTier AWS CLICommand Reference.

`revoke-client-vpn-ingress`

Il seguente esempio di codice mostra come utilizzare`revoke-client-vpn-ingress`.

AWS CLI

Come revocare una regola di autorizzazione per un endpoint VPN Client

L'esempio `revoke-client-vpn-ingress` seguente revoca una regola per l'accesso a Internet (`0.0.0.0/0`) per tutti i gruppi.

```
aws ec2 revoke-client-vpn-ingress \
    --client-vpn-endpoint-id cvpn-endpoint-123456789123abcde \
    --target-network-cidr 0.0.0.0/0 --revoke-all-groups
```

Output:

```
{
    "Status": {
        "Code": "revoking"
    }
}
```

Per ulteriori informazioni, consulta Regole di autorizzazione nella Guida per l'amministratore di VPN Client di AWS .

- Per i dettagli sull'API, consulta RevokeClientVpnIngress AWS CLICommand Reference.

`revoke-security-group-egress`

Il seguente esempio di codice mostra come utilizzare`revoke-security-group-egress`.

AWS CLI

Esempio 1: come rimuovere la regola che consente il traffico in uscita verso un intervallo di indirizzi specifico

L'esempio `revoke-security-group-egress` seguente rimuove la regola che concede l'accesso agli intervalli di indirizzi specificati sulla porta TCP 80.

```
aws ec2 revoke-security-group-egress \
    --group-id sg-026c12253ce15eff7 \
    --ip-
permissions [{IpProtocol=tcp,FromPort=80,ToPort=80,IpRanges=[{CidrIp=10.0.0.0/16}]
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Security groups nella Amazon EC2 User Guide.

Esempio 2: come rimuovere una regola che consente il traffico in uscita verso un gruppo di sicurezza specifico

L'esempio `revoke-security-group-egress` seguente rimuove la regola che concede l'accesso al gruppo di sicurezza specificato sulla porta TCP 80.

```
aws ec2 revoke-security-group-egress \
    --group-id sg-026c12253ce15eff7 \
    --ip-permissions '[{"IpProtocol": "tcp", "FromPort": 443, "ToPort":
 443,"UserIdGroupPairs": [{"GroupId": "sg-06df23a01ff2df86d"}]}]'
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Security groups nella Amazon EC2 User Guide.

- Per i dettagli sull'API, consulta RevokeSecurityGroupEgress AWS CLICommand Reference.

`revoke-security-group-ingress`

Il seguente esempio di codice mostra come utilizzare`revoke-security-group-ingress`.

AWS CLI

Esempio 1: come rimuovere una regola da un gruppo di sicurezza

L'esempio `revoke-security-group-ingress` seguente rimuove l'accesso alla porta TCP 22 per l'intervallo di indirizzi `203.0.113.0/24` dal gruppo di sicurezza specificato per un VPC predefinito.

```
aws ec2 revoke-security-group-ingress \
    --group-name mySecurityGroup
    --protocol tcp \
    --port 22 \
```

```
    --cidr 203.0.113.0/24
```

Questo comando non produce alcun output se ha esito positivo.

Per ulteriori informazioni, consulta la sezione Gruppi di sicurezza nella Amazon EC2 User Guide.

Esempio 2: come rimuovere una regola utilizzando il set di autorizzazioni IP

L'esempio `revoke-security-group-ingress` seguente utilizza il parametro `ip-permissions` per aggiungere una regola in entrata che consenta il messaggio ICMP `Destination Unreachable: Fragmentation Needed and Don't Fragment was Set` (tipo 3, codice 4).

```
aws ec2 revoke-security-group-ingress \
    --group-id sg-026c12253ce15eff7 \
    --ip-
permissions IpProtocol=icmp,FromPort=3,ToPort=4,IpRanges=[{CidrIp=0.0.0.0/0}]
```

Questo comando non produce alcun output se ha esito positivo.

Per ulteriori informazioni, consulta la sezione Gruppi di sicurezza nella Amazon EC2 User Guide.

- Per i dettagli sull'API, consulta RevokeSecurityGroupIngress AWS CLICommand Reference.

`run-instances`

Il seguente esempio di codice mostra come utilizzare`run-instances`.

AWS CLI

Esempio 1: come avviare un'istanza in una sottorete predefinita

Nell'esempio di `run-instances` seguente viene avviata una singola istanza di tipo `t2.micro` nella sottorete predefinita per la regione attuale e viene associata alla sottorete predefinita per il VPC predefinito per la regione. La coppia di chiavi è opzionale se non si desidera collegare l'istanza tramite SSH (Linux) o RDP (Windows).

```
aws ec2 run-instances \
    --image-id ami-0abcdef1234567890 \
    --instance-type t2.micro \
    --key-name MyKeyPair
```

Output:


```
{
    "Instances": [
        {
            "AmiLaunchIndex": 0,
            "ImageId": "ami-0abcdef1234567890",
            "InstanceId": "i-1231231230abcdef0",
            "InstanceType": "t2.micro",
            "KeyName": "MyKeyPair",
            "LaunchTime": "2018-05-10T08:05:20.000Z",
            "Monitoring": {
                "State": "disabled"
            },
            "Placement": {
                "AvailabilityZone": "us-east-2a",
                "GroupName": "",
                "Tenancy": "default"
            },
            "PrivateDnsName": "ip-10-0-0-157.us-east-2.compute.internal",
            "PrivateIpAddress": "10.0.0.157",
            "ProductCodes": [],
            "PublicDnsName": "",
            "State": {
                "Code": 0,
                "Name": "pending"
            },
            "StateTransitionReason": "",
            "SubnetId": "subnet-04a636d18e83cfacb",
            "VpcId": "vpc-1234567890abcdef0",
            "Architecture": "x86_64",
            "BlockDeviceMappings": [],
            "ClientToken": "",
            "EbsOptimized": false,
            "Hypervisor": "xen",
            "NetworkInterfaces": [
                {
                    "Attachment": {
                        "AttachTime": "2018-05-10T08:05:20.000Z",
                        "AttachmentId": "eni-attach-0e325c07e928a0405",
                        "DeleteOnTermination": true,
                        "DeviceIndex": 0,
                        "Status": "attaching"
                    },
```

```
                    "Description": "",
                    "Groups": [
                        {
                            "GroupName": "MySecurityGroup",
                            "GroupId": "sg-0598c7d356eba48d7"
                        }
                    ],
                    "Ipv6Addresses": [],
                    "MacAddress": "0a:ab:58:e0:67:e2",
                    "NetworkInterfaceId": "eni-0c0a29997760baee7",
                    "OwnerId": "123456789012",
                    "PrivateDnsName": "ip-10-0-0-157.us-east-2.compute.internal",
                    "PrivateIpAddress": "10.0.0.157",
                    "PrivateIpAddresses": [
                        {
                            "Primary": true,
                            "PrivateDnsName": "ip-10-0-0-157.us-
east-2.compute.internal",
                            "PrivateIpAddress": "10.0.0.157"
                        }
                    ],
                    "SourceDestCheck": true,
                    "Status": "in-use",
                    "SubnetId": "subnet-04a636d18e83cfacb",
                    "VpcId": "vpc-1234567890abcdef0",
                    "InterfaceType": "interface"
                }
            ],
            "RootDeviceName": "/dev/xvda",
            "RootDeviceType": "ebs",
            "SecurityGroups": [
                {
                    "GroupName": "MySecurityGroup",
                    "GroupId": "sg-0598c7d356eba48d7"
                }
            ],
            "SourceDestCheck": true,
            "StateReason": {
                "Code": "pending",
                "Message": "pending"
            },
            "Tags": [],
            "VirtualizationType": "hvm",
            "CpuOptions": {
```

```
                "CoreCount": 1,
                "ThreadsPerCore": 1
            },
            "CapacityReservationSpecification": {
                "CapacityReservationPreference": "open"
            },
            "MetadataOptions": {
                "State": "pending",
                "HttpTokens": "optional",
                "HttpPutResponseHopLimit": 1,
                "HttpEndpoint": "enabled"
            }
        }
    ],
    "OwnerId": "123456789012",
    "ReservationId": "r-02a3f596d91211712"
}
```

Esempio 2: come avviare un'istanza in una sottorete non predefinita e aggiungere un indirizzo IP pubblico

Nell'esempio di `run-instances` seguente viene richiesto un indirizzo IP pubblico per un'istanza avviata in una sottorete non predefinita. L'istanza è associata al gruppo di sicurezza specificato.

```
aws ec2 run-instances \
    --image-id ami-0abcdef1234567890 \
    --instance-type t2.micro \
    --subnet-id subnet-08fc749671b2d077c \
    --security-group-ids sg-0b0384b66d7d692f9 \
    --associate-public-ip-address \
    --key-name MyKeyPair
```

Per un esempio dell'output di `run-instances`, consulta l'Esempio 1.

Esempio 3: come avviare un'istanza con volumi aggiuntivi

Nell'esempio di `run-instances` seguente viene utilizzata una mappatura dei dispositivi a blocchi, specificata in mapping.json, per collegare volumi aggiuntivi al momento del lancio. Una mappatura dei dispositivi a blocchi può specificare volumi EBS, oppure sia volumi EBS e volumi di archivio dell'istanza.

```
aws ec2 run-instances \
```

```
    --image-id ami-0abcdef1234567890 \
    --instance-type t2.micro \
    --subnet-id subnet-08fc749671b2d077c \
    --security-group-ids sg-0b0384b66d7d692f9 \
    --key-name MyKeyPair \
    --block-device-mappings file://mapping.json
```

Contenuto di `mapping.json`. In questo esempio viene aggiunto `/dev/sdh`, un volume EBS vuoto della dimensione di 100 GiB.

```
[
    {
        "DeviceName": "/dev/sdh",
        "Ebs": {
            "VolumeSize": 100
        }
    }
]
```

Contenuto di `mapping.json`. In questo esempio viene aggiunto `ephemeral1`, un volume di archivio dell'istanza.

```
[
    {
        "DeviceName": "/dev/sdc",
        "VirtualName": "ephemeral1"
    }
]
```

Per un esempio dell'output di `run-instances`, consulta l'Esempio 1.

Per ulteriori informazioni sulle mappature dei dispositivi a blocchi, consulta la sezione Block device mapping nella Amazon EC2 User Guide.

Esempio 4: come avviare un'istanza e aggiungere tag al momento della creazione

Nell'esempio di `run-instances` seguente viene aggiunto un tag con una chiave `webserver` e un valore `production` all'istanza. Il comando avvia applica inoltre un tag con una chiave `cost-center` e un valore `cc123` a qualsiasi volume EBS creato, in questo caso il volume root.

```
aws ec2 run-instances \
```

```
    --image-id ami-0abcdef1234567890 \
    --instance-type t2.micro \
    --count 1 \
    --subnet-id subnet-08fc749671b2d077c \
    --key-name MyKeyPair \
    --security-group-ids sg-0b0384b66d7d692f9 \
    --tag-specifications
 'ResourceType=instance,Tags=[{Key=webserver,Value=production}]'
 'ResourceType=volume,Tags=[{Key=cost-center,Value=cc123}]'
```

Per un esempio dell'output di `run-instances`, consulta l'Esempio 1.

Esempio 5: come avviare un'istanza con dati utente

Nell'esempio di `run-instances` seguente i dati utente vengono trasferiti in un file denominato `my_script.txt` che contiene uno script di configurazione per l'istanza. Lo script viene eseguito al momento dell'avvio.

```
aws ec2 run-instances \
    --image-id ami-0abcdef1234567890 \
    --instance-type t2.micro \
    --count 1 \
    --subnet-id subnet-08fc749671b2d077c \
    --key-name MyKeyPair \
    --security-group-ids sg-0b0384b66d7d692f9 \
    --user-data file://my_script.txt
```

Per un esempio dell'output di `run-instances`, consulta l'Esempio 1.

Per ulteriori informazioni sui dati utente dell'istanza, consulta Working with instance user data nella Amazon EC2 User Guide.

Esempio 6: come avviare un'istanza a prestazioni espandibili

Nell'esempio di `run-instances` seguente viene avviata un'istanza t2.micro con l'opzione di credito `unlimited`. All'avvio di un'istanza T2, se non si specifica `--credit-specification`, l'opzione di credito predefinita `standard`. All'avvio di un'istanza T3, l'opzione di credito predefinita è `unlimited`.

```
aws ec2 run-instances \
    --image-id ami-0abcdef1234567890 \
```

```
    --instance-type t2.micro \
    --count 1 \
    --subnet-id subnet-08fc749671b2d077c \
    --key-name MyKeyPair \
    --security-group-ids sg-0b0384b66d7d692f9 \
    --credit-specification CpuCredits=unlimited
```

Per un esempio dell'output di `run-instances`, vedi l'Esempio 1.

Per ulteriori informazioni sulle istanze con prestazioni espandibili, consulta le istanze Burstable Performance nella Amazon User Guide. EC2

- Per i dettagli sull'API, consulta RunInstancesCommand Reference.AWS CLI

`run-scheduled-instances`

Il seguente esempio di codice mostra come utilizzare`run-scheduled-instances`.

AWS CLI

Come avviare un'istanza pianificata

Questo esempio avvia l'istanza pianificata specificata in un VPC.

Comando:

```
aws ec2 run-scheduled-instances --scheduled-instance-
id sci-1234-1234-1234-1234-123456789012 --instance-count 1 --launch-
specification file://launch-specification.json
```

Launch-specification.json:

```
{
  "ImageId": "ami-12345678",
  "KeyName": "my-key-pair",
  "InstanceType": "c4.large",
  "NetworkInterfaces": [
    {
        "DeviceIndex": 0,
        "SubnetId": "subnet-12345678",
        "AssociatePublicIpAddress": true,
        "Groups": ["sg-12345678"]
```

```
    }
  ],
  "IamInstanceProfile": {
      "Name": "my-iam-role"
  }
}
```

Output:

```
{
  "InstanceIdSet": [
      "i-1234567890abcdef0"
  ]
}
```

Questo esempio avvia l'istanza pianificata specificata in EC2 -Classic.

Comando:

```
aws ec2 run-scheduled-instances --scheduled-instance-
id sci-1234-1234-1234-1234-123456789012 --instance-count 1 --launch-
specification file://launch-specification.json
```

Launch-specification.json:

```
{
  "ImageId": "ami-12345678",
  "KeyName": "my-key-pair",
  "SecurityGroupIds": ["sg-12345678"],
  "InstanceType": "c4.large",
  "Placement": {
    "AvailabilityZone": "us-west-2b"
  }
  "IamInstanceProfile": {
      "Name": "my-iam-role"
  }
}
```

Output:

```
{
```

```
  "InstanceIdSet": [
      "i-1234567890abcdef0"
  ]
}
```

- Per i dettagli sull'API, vedere RunScheduledInstancesin AWS CLI Command Reference.

`search-local-gateway-routes`

Il seguente esempio di codice mostra come utilizzare`search-local-gateway-routes`.

AWS CLI

Come ricercare le route in una tabella di routing del gateway locale

L'esempio `search-local-gateway-routes` seguente ricerca route statiche nella tabella di routing specificata per il gateway locale.

```
aws ec2 search-local-gateway-routes \
    --local-gateway-route-table-id lgw-rtb-059615ef7dEXAMPLE \
    --filters "Name=type,Values=static"
```

Output:

```
{
    "Route": {
        "DestinationCidrBlock": "0.0.0.0/0",
        "LocalGatewayVirtualInterfaceGroupId": "lgw-vif-grp-07145b276bEXAMPLE",
        "Type": "static",
        "State": "deleted",
        "LocalGatewayRouteTableId": "lgw-rtb-059615ef7EXAMPLE"
    }
}
```

- Per i dettagli sull'API, consulta SearchLocalGatewayRoutes AWS CLICommand Reference.

`search-transit-gateway-multicast-groups`

Il seguente esempio di codice mostra come utilizzare`search-transit-gateway-multicast-groups`.

AWS CLI

Come cercare uno o più gruppi multicast (trasmissione uno a molti) di gateway di transito e restituire le informazioni sull'appartenenza al gruppo

L'esempio `search-transit-gateway-multicast-groups` seguente restituisce l'appartenenza al gruppo del gruppo multicast (trasmissione uno a molti) specificato.

```
aws ec2 search-transit-gateway-multicast-groups \
    --transit-gateway-multicast-domain-id tgw-mcast-domain-000fb24d04EXAMPLE
```

Output:

```
{
    "MulticastGroups": [
        {
            "GroupIpAddress": "224.0.1.0",
            "TransitGatewayAttachmentId": "tgw-attach-0372e72386EXAMPLE",
            "SubnetId": "subnet-0187aff814EXAMPLE",
            "ResourceId": "vpc-0065acced4EXAMPLE",
            "ResourceType": "vpc",
            "NetworkInterfaceId": "eni-03847706f6EXAMPLE",
            "GroupMember": false,
            "GroupSource": true,
            "SourceType": "static"
        }
    ]
}
```

Per ulteriori informazioni, consulta Multicast (trasmissione uno a molti) sui gateway di transito nella Guida di Transit Gateway.

- Per i dettagli sull'API, consulta SearchTransitGatewayMulticastGroups AWS CLICommand Reference.

`search-transit-gateway-routes`

Il seguente esempio di codice mostra come utilizzare`search-transit-gateway-routes`.

AWS CLI

Come ricercare le route nella tabella di routing specificata del gateway di transito

L'esempio `search-transit-gateway-routes` seguente restituisce tutte le route che sono di tipo `static` nella tabella di routing specificata.

```
aws ec2 search-transit-gateway-routes \
    --transit-gateway-route-table-id tgw-rtb-0a823edbdeEXAMPLE \
    --filters "Name=type,Values=static"
```

Output:

```
{
    "Routes": [
        {
            "DestinationCidrBlock": "10.0.2.0/24",
            "TransitGatewayAttachments": [
                {
                    "ResourceId": "vpc-4EXAMPLE",
                    "TransitGatewayAttachmentId": "tgw-attach-09b52ccdb5EXAMPLE",
                    "ResourceType": "vpc"
                }
            ],
            "Type": "static",
            "State": "active"
        },
        {
            "DestinationCidrBlock": "10.1.0.0/24",
            "TransitGatewayAttachments": [
                {
                    "ResourceId": "vpc-4EXAMPLE",
                    "TransitGatewayAttachmentId": "tgw-attach-09b52ccdb5EXAMPLE",
                    "ResourceType": "vpc"
                }
            ],
            "Type": "static",
            "State": "active"
        }
    ],
    "AdditionalRoutesAvailable": false
}
```

Per ulteriori informazioni, consulta Tabelle di routing del gateway di transito nella Guida di Transit Gateway.

- Per i dettagli sull'API, consulta SearchTransitGatewayRoutes AWS CLICommand Reference.

`send-diagnostic-interrupt`

Il seguente esempio di codice mostra come utilizzare`send-diagnostic-interrupt`.

AWS CLI

Come inviare un'interruzione della diagnostica

L'esempio `send-diagnostic-interrupt` seguente invia un'interruzione della diagnostica all'istanza specificata.

```
aws ec2 send-diagnostic-interrupt \
    --instance-id i-1234567890abcdef0
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta SendDiagnosticInterrupt AWS CLICommand Reference.

`start-instances`

Il seguente esempio di codice mostra come utilizzare`start-instances`.

AWS CLI

Per avviare un' EC2 istanza Amazon

In questo esempio viene avviata l'istanza supportata da Amazon EBS specificata.

Comando:

```
aws ec2 start-instances --instance-ids i-1234567890abcdef0
```

Output:

```
{
    "StartingInstances": [
        {
            "InstanceId": "i-1234567890abcdef0",
            "CurrentState": {
                "Code": 0,
```

```
                "Name": "pending"
            },
            "PreviousState": {
                "Code": 80,
                "Name": "stopped"
            }
        }
    ]
}
```

Per ulteriori informazioni, consulta Arrestare e avviare un'istanza nella Guida per l'utente di Amazon Elastic Compute Cloud.

- Per i dettagli sull'API, consulta StartInstances AWS CLICommand Reference.

`start-network-insights-access-scope-analysis`

Il seguente esempio di codice mostra come utilizzare`start-network-insights-access-scope-analysis`.

AWS CLI

Come avviare un'analisi dell'ambito di accesso alle informazioni di rete

L'`start-network-insights-access-scope-analysis`esempio seguente avvia l'analisi dell'ambito nel tuo AWS account.

```
aws ec2 start-network-insights-access-scope-analysis \
    --region us-east-1 \
    --network-insights-access-scope-id nis-123456789111
```

Output:

```
{
    "NetworkInsightsAccessScopeAnalysis": {
        "NetworkInsightsAccessScopeAnalysisId": "nisa-123456789222",
        "NetworkInsightsAccessScopeAnalysisArn": "arn:aws:ec2:us-
east-1:123456789012:network-insights-access-scope-analysis/nisa-123456789222",
        "NetworkInsightsAccessScopeId": "nis-123456789111",
        "Status": "running",
        "StartDate": "2022-01-26T00:47:06.814000+00:00"
```

```
    }
}
```

Per ulteriori informazioni, consulta Guida introduttiva a Network Access Analyzer utilizzando la AWS CLI nella Guida Network Access Analyzer.

- Per i dettagli sull'API, consulta Command StartNetworkInsightsAccessScopeAnalysisReference AWS CLI .

`start-network-insights-analysis`

Il seguente esempio di codice mostra come utilizzare`start-network-insights-analysis`.

AWS CLI

Come analizzare un percorso

L'esempio `start-network-insights-analysis` seguente analizza il percorso tra l'origine e la destinazione. Per visualizzare i risultati dell'analisi del percorso, utilizza il comando `describe-network-insights-analyses`.

```
aws ec2 start-network-insights-analysis \
    --network-insights-path-id nip-0b26f224f1d131fa8
```

Output:

```
{
    "NetworkInsightsAnalysis": {
        "NetworkInsightsAnalysisId": "nia-02207aa13eb480c7a",
        "NetworkInsightsAnalysisArn": "arn:aws:ec2:us-east-1:123456789012:network-
insights-analysis/nia-02207aa13eb480c7a",
        "NetworkInsightsPathId": "nip-0b26f224f1d131fa8",
        "StartDate": "2021-01-20T22:58:37.495Z",
        "Status": "running"
    }
}
```

Per ulteriori informazioni, consulta Guida introduttiva all'uso della AWS CLI nella Reachability Analyzer Guide.

- Per i dettagli sull'API, consulta Command StartNetworkInsightsAnalysisReference AWS CLI .

`start-vpc-endpoint-service-private-dns-verification`

Il seguente esempio di codice mostra come utilizzare`start-vpc-endpoint-service-private-dns-verification`.

AWS CLI

Come avviare il processo di verifica DNS

L'esempio `start-vpc-endpoint-service-private-dns-verification` seguente avvia il processo di verifica DNS per il servizio endpoint specificato.

```
aws ec2 start-vpc-endpoint-service-private-dns-verification \
    --service-id vpce-svc-071afff70666e61e0
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Gestire i nomi DNS nella Guida per l'AWS PrivateLink utente.

- Per i dettagli sull'API, consulta StartVpcEndpointServicePrivateDnsVerification AWS CLICommand Reference.

`stop-instances`

Il seguente esempio di codice mostra come utilizzare`stop-instances`.

AWS CLI

Esempio 1: interrompere un' EC2 istanza Amazon

Nell'esempio di `stop-instances` seguente viene interrotta l'istanza supportata da Amazon EBS specificata.

```
aws ec2 stop-instances \
    --instance-ids i-1234567890abcdef0
```

Output:

```
{
    "StoppingInstances": [
        {
```

```
            "InstanceId": "i-1234567890abcdef0",
            "CurrentState": {
                "Code": 64,
                "Name": "stopping"
            },
            "PreviousState": {
                "Code": 16,
                "Name": "running"
            }
        }
    ]
}
```

Per ulteriori informazioni, consulta Arrestare e avviare un'istanza nella Guida per l'utente di Amazon Elastic Compute Cloud.

Esempio 2: ibernare un'istanza Amazon EC2

Nell'esempio di `stop-instances` seguente viene ibernata un'istanza supportata da Amazon EBS se tale istanza è abilitata per l'ibernazione e soddisfa i prerequisiti di ibernazione. Dopo l'ibernazione dell'istanza, questa viene arrestata.

```
aws ec2 stop-instances \
    --instance-ids i-1234567890abcdef0 \
    --hibernate
```

Output:

```
{
    "StoppingInstances": [
        {
            "CurrentState": {
                "Code": 64,
                "Name": "stopping"
            },
            "InstanceId": "i-1234567890abcdef0",
            "PreviousState": {
                "Code": 16,
                "Name": "running"
            }
        }
    ]
```

```
}
```

Per ulteriori informazioni, consulta Ibernazione di un'istanza Linux on demand nella Guida per l'utente di Amazon Elastic Cloud Compute.

- Per i dettagli sull'API, consulta AWS CLI Command StopInstancesReference.

`terminate-client-vpn-connections`

Il seguente esempio di codice mostra come utilizzare`terminate-client-vpn-connections`.

AWS CLI

Come terminare una connessione a un endpoint VPN Client

L'esempio `terminate-client-vpn-connections` seguente termina la connessione specificata all'endpoint VPN Client.

```
aws ec2 terminate-client-vpn-connections \
    --client-vpn-endpoint-id vpn-endpoint-123456789123abcde \
    --connection-id cvpn-connection-04edd76f5201e0cb8
```

Output:

```
{
    "ClientVpnEndpointId": "vpn-endpoint-123456789123abcde",
    "ConnectionStatuses": [
        {
            "ConnectionId": "cvpn-connection-04edd76f5201e0cb8",
            "PreviousStatus": {
                "Code": "active"
            },
            "CurrentStatus": {
                "Code": "terminating"
            }
        }
    ]
}
```

Per ulteriori informazioni, consulta Connessioni client nella Guida per l'amministratore di VPN Client di AWS .

- Per i dettagli sull'API, consulta TerminateClientVpnConnections AWS CLICommand Reference.

`terminate-instances`

Il seguente esempio di codice mostra come utilizzare`terminate-instances`.

AWS CLI

Per terminare un'istanza Amazon EC2

Questo esempio termina l'istanza specificata.

Comando:

```
aws ec2 terminate-instances --instance-ids i-1234567890abcdef0
```

Output:

```
{
    "TerminatingInstances": [
        {
            "InstanceId": "i-1234567890abcdef0",
            "CurrentState": {
                "Code": 32,
                "Name": "shutting-down"
            },
            "PreviousState": {
                "Code": 16,
                "Name": "running"
            }
        }
    ]
}
```

Per ulteriori informazioni, consulta Using Amazon EC2 Instances nella AWS Command Line Interface User Guide.

- Per i dettagli sull'API, consulta TerminateInstances AWS CLICommand Reference.

`unassign-ipv6-addresses`

Il seguente esempio di codice mostra come utilizzare`unassign-ipv6-addresses`.

AWS CLI

Per annullare l'assegnazione di un IPv6 indirizzo da un'interfaccia di rete

Questo esempio annulla l'assegnazione dell' IPv6 indirizzo specificato dall'interfaccia di rete specificata.

Comando:

```
aws ec2 unassign-ipv6-addresses --ipv6-
addresses 2001:db8:1234:1a00:3304:8879:34cf:4071 --network-interface-id eni-23c49b68
```

Output:

```
{
  "NetworkInterfaceId": "eni-23c49b68",
  "UnassignedIpv6Addresses": [
      "2001:db8:1234:1a00:3304:8879:34cf:4071"
  ]
}
```

- Per i dettagli sull'API, vedere UnassignIpv6Addresses in AWS CLI Command Reference.

`unassign-private-ip-addresses`

Il seguente esempio di codice mostra come utilizzare`unassign-private-ip-addresses`.

AWS CLI

Come annullare l'assegnazione a un'interfaccia di rete di un indirizzo IP privato secondario

Questo esempio annulla l'assegnazione dell'indirizzo IP privato specificato all'interfaccia di rete specificata. Se il comando va a buon fine, non viene restituito alcun output.

Comando:

```
aws ec2 unassign-private-ip-addresses --network-interface-id eni-e5aa89a3 --private-
ip-addresses 10.0.0.82
```

- Per i dettagli sull'API, consulta UnassignPrivateIpAddresses AWS CLICommand Reference.

`unassign-private-nat-gateway-address`

Il seguente esempio di codice mostra come utilizzare`unassign-private-nat-gateway-address`.

AWS CLI

Come annullare l'assegnazione di un indirizzo IP al gateway NAT privato

L'esempio `unassign-private-nat-gateway-address` seguente annulla l'assegnazione dell'indirizzo IP specificato al gateway NAT privato specificato.

```
aws ec2 unassign-private-nat-gateway-address \
    --nat-gateway-id nat-1234567890abcdef0 \
    --private-ip-addresses 10.0.20.197
```

Output:

```
{
    "NatGatewayId": "nat-0ee3edd182361f662",
    "NatGatewayAddresses": [
        {
            "NetworkInterfaceId": "eni-0065a61b324d1897a",
            "PrivateIp": "10.0.20.197",
            "IsPrimary": false,
            "Status": "unassigning"
        }
    ]
}
```

Per ulteriori informazioni, consulta Gateway NAT nella Guida per l'utente di Amazon VPC.

- Per i dettagli sull'API, consulta UnassignPrivateNatGatewayAddress AWS CLICommand Reference.

`unlock-snapshot`

Il seguente esempio di codice mostra come utilizzare`unlock-snapshot`.

AWS CLI

Come sbloccare uno snapshot

L'esempio `unlock-snapshot` seguente sblocca lo snapshot specificato.

```
aws ec2 unlock-snapshot \
    --snapshot-id snap-0b5e733b4a8df6e0d
```

Output:

```
{
    "SnapshotId": "snap-0b5e733b4a8df6e0d"
}
```

Per ulteriori informazioni, consulta Snapshot Lock nella Guida per l'utente di Amazon EBS.

- Per i dettagli sull'API, consulta UnlockSnapshot AWS CLICommand Reference.

`unmonitor-instances`

Il seguente esempio di codice mostra come utilizzareunmonitor-instances.

AWS CLI

Come disabilitare il monitoraggio dettagliato per un'istanza

Questo comando di esempio disabilita il monitoraggio dettagliato per l'istanza specificata.

Comando:

```
aws ec2 unmonitor-instances --instance-ids i-1234567890abcdef0
```

Output:

```
{
  "InstanceMonitorings": [
      {
          "InstanceId": "i-1234567890abcdef0",
          "Monitoring": {
              "State": "disabling"
          }
      }
  ]
```

```
}
```

- Per i dettagli sull'API, consulta UnmonitorInstances AWS CLICommand Reference.

`update-security-group-rule-descriptions-egress`

Il seguente esempio di codice mostra come utilizzare`update-security-group-rule-descriptions-egress`.

AWS CLI

Come aggiornare la descrizione di una regola in uscita del gruppo di sicurezza

L'`update-security-group-rule-descriptions-egress`esempio seguente aggiorna la descrizione della regola del gruppo di sicurezza per la porta e l'intervallo di IPv4 indirizzi specificati. La descrizione "`Outbound HTTP access to server 2`" sostituisce qualsiasi descrizione esistente per la regola.

```
aws ec2 update-security-group-rule-descriptions-egress \
    --group-id sg-02f0d35a850ba727f \
    --ip-permissions
 IpProtocol=tcp,FromPort=80,ToPort=80,IpRanges=[{CidrIp=203.0.113.0/24,Description="Outbound
 HTTP access to server 2"}]
```

Output:

```
{
    "Return": true
}
```

Per ulteriori informazioni, consulta le regole dei gruppi di sicurezza nella Amazon EC2 User Guide.

- Per i dettagli sull'API, consulta UpdateSecurityGroupRuleDescriptionsEgress AWS CLICommand Reference.

`update-security-group-rule-descriptions-ingress`

Il seguente esempio di codice mostra come utilizzare`update-security-group-rule-descriptions-ingress`.

AWS CLI

Esempio 1: come aggiornare la descrizione di una regola del gruppo di sicurezza in entrata con un'origine CIDR

L'`update-security-group-rule-descriptions-ingress`esempio seguente aggiorna la descrizione della regola del gruppo di sicurezza per la porta e l'intervallo di IPv4 indirizzi specificati. La descrizione "`SSH access from ABC office`" sostituisce qualsiasi descrizione esistente per la regola.

```
aws ec2 update-security-group-rule-descriptions-ingress \
    --group-id sg-02f0d35a850ba727f \
    --ip-permissions
 IpProtocol=tcp,FromPort=22,ToPort=22,IpRanges='[{CidrIp=203.0.113.0/16,Description="SSH
 access from corpnet"}]'
```

Output:

```
{
    "Return": true
}
```

Per ulteriori informazioni, consulta le regole dei gruppi di sicurezza nella Amazon EC2 User Guide.

Esempio 2: come aggiornare la descrizione di una regola del gruppo di sicurezza in entrata con un'origine dell'elenco di prefissi

L'esempio `update-security-group-rule-descriptions-ingress` seguente aggiorna la descrizione della regola del gruppo di sicurezza per la porta e l'elenco di prefissi specificati. La descrizione "`SSH access from ABC office`" sostituisce qualsiasi descrizione esistente per la regola.

```
aws ec2 update-security-group-rule-descriptions-ingress \
    --group-id sg-02f0d35a850ba727f \
    --ip-permissions
 IpProtocol=tcp,FromPort=22,ToPort=22,PrefixListIds='[{PrefixListId=pl-12345678,Description=
 access from corpnet"}]'
```

Output:

```
{
    "Return": true
}
```

Per ulteriori informazioni, consulta le regole dei gruppi di sicurezza nella Amazon EC2 User Guide.

- Per i dettagli sull'API, consulta UpdateSecurityGroupRuleDescriptionsIngress AWS CLICommand Reference.

`withdraw-byoip-cidr`

Il seguente esempio di codice mostra come utilizzare`withdraw-byoip-cidr`.

AWS CLI

Come smettere di pubblicizzare un intervallo di indirizzi

L'esempio `withdraw-byoip-cidr` seguente smette di pubblicizzare l'intervallo di indirizzi specificato.

```
aws ec2 withdraw-byoip-cidr
    --cidr 203.0.113.25/24
```

Output:

```
{
    "ByoipCidr": {
        "Cidr": "203.0.113.25/24",
        "StatusMessage": "ipv4pool-ec2-1234567890abcdef0",
        "State": "advertised"
    }
}
```

- Per i dettagli sull'API, consulta WithdrawByoipCidr AWS CLICommand Reference.

Esempi di Amazon EC2 Instance Connect utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando Amazon EC2 Instance Connect. AWS Command Line Interface

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`send-ssh-public-key`

Il seguente esempio di codice mostra come usare`send-ssh-public-key`.

AWS CLI

Come inviare una chiave pubblica SSH a un'istanza

L'esempio `send-ssh-public-key` seguente invia la chiave pubblica SSH specificata all'istanza specificata. La chiave viene utilizzata per autenticare l'utente specificato.

```
aws ec2-instance-connect send-ssh-public-key \
    --instance-id i-1234567890abcdef0 \
    --instance-os-user ec2-user \
    --availability-zone us-east-2b \
    --ssh-public-key file://path/my-rsa-key.pub
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta SendSshPublicKey AWS CLICommand Reference.

Esempi di utilizzo di Amazon ECR AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando Amazon ECR. AWS Command Line Interface

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`batch-check-layer-availability`

Il seguente esempio di codice mostra come usare`batch-check-layer-availability`.

AWS CLI

Come verificare la disponibilità di un livello

L'esempio `batch-check-layer-availability` seguente verifica la disponibilità di un livello con il digest `sha256:6171c7451a50945f8ddd72f7732cc04d7a0d1f48138a426b2e64387fdeb834ed` nel repository `cluster-autoscaler`.

```
aws ecr batch-check-layer-availability \
    --repository-name cluster-autoscaler \
    --layer-
digests sha256:6171c7451a50945f8ddd72f7732cc04d7a0d1f48138a426b2e64387fdeb834ed
```

Output:

```
{
    "layers": [
        {
            "layerDigest":
 "sha256:6171c7451a50945f8ddd72f7732cc04d7a0d1f48138a426b2e64387fdeb834ed",
            "layerAvailability": "AVAILABLE",
            "layerSize": 2777,
            "mediaType": "application/vnd.docker.container.image.v1+json"
        }
    ],
    "failures": []
}
```

- Per i dettagli sull'API, consulta BatchCheckLayerAvailability AWS CLICommand Reference.

`batch-delete-image`

Il seguente esempio di codice mostra come utilizzare`batch-delete-image`.

AWS CLI

Esempio 1: come eliminare un'immagine

L'esempio `batch-delete-image` seguente elimina un'immagine con tag `precise` nel repository specificato nel registro predefinito per un account.

```
aws ecr batch-delete-image \
    --repository-name ubuntu \
    --image-ids imageTag=precise
```

Output:

```
{
    "failures": [],
    "imageIds": [
        {
            "imageTag": "precise",
            "imageDigest":
 "sha256:19665f1e6d1e504117a1743c0a3d3753086354a38375961f2e665416ef4b1b2f"
        }
    ]
}
```

Esempio 2: come eliminare più immagini

L'esempio `batch-delete-image` seguente elimina tutte le immagini con tag `prod` e `team1` nel repository specificato.

```
aws ecr batch-delete-image \
    --repository-name MyRepository \
    --image-ids imageTag=prod imageTag=team1
```

Output:

```
{
    "imageIds": [
        {
            "imageDigest": "sha256:123456789012",
            "imageTag": "prod"
        },
        {
            "imageDigest": "sha256:567890121234",
            "imageTag": "team1"
        }
    ],
    "failures": []
}
```

Per ulteriori informazioni, consulta Eliminazione di un'immagine nella Guida per l'utente di Amazon ECR.

- Per i dettagli sull'API, consulta BatchDeleteImage AWS CLICommand Reference.

`batch-get-image`

Il seguente esempio di codice mostra come utilizzare`batch-get-image`.

AWS CLI

Esempio 1: come ottenere un'immagine

L'esempio `batch-get-image` seguente ottiene un'immagine con tag `v1.13.6` nel repository denominato `cluster-autoscaler` nel registro predefinito per un account.

```
aws ecr batch-get-image \
    --repository-name cluster-autoscaler \
    --image-ids imageTag=v1.13.6
```

Output:

```
{
    "images": [
        {
            "registryId": "012345678910",
            "repositoryName": "cluster-autoscaler",
```

```
            "imageId": {
                "imageDigest":
 "sha256:4a1c6567c38904384ebc64e35b7eeddd8451110c299e3368d2210066487d97e5",
                "imageTag": "v1.13.6"
            },
            "imageManifest": "{\n   \"schemaVersion\": 2,\n
 \"mediaType\": \"application/vnd.docker.distribution.manifest.v2+json
\",\n   \"config\": {\n      \"mediaType\": \"application/
vnd.docker.container.image.v1+json\",\n      \"size\": 2777,\n      \"digest
\": \"sha256:6171c7451a50945f8ddd72f7732cc04d7a0d1f48138a426b2e64387fdeb834ed
\"\n   },\n   \"layers\": [\n      {\n         \"mediaType
\": \"application/vnd.docker.image.rootfs.diff.tar.gzip
\",\n         \"size\": 17743696,\n         \"digest\":
 \"sha256:39fafc05754f195f134ca11ecdb1c9a691ab0848c697fffeb5a85f900caaf6e1\"\n
      },\n      {\n         \"mediaType\": \"application/
vnd.docker.image.rootfs.diff.tar.gzip\",\n         \"size\": 2565026,\n
 \"digest\":
 \"sha256:8c8a779d3a537b767ae1091fe6e00c2590afd16767aa6096d1b318d75494819f
\"\n      },\n      {\n         \"mediaType\": \"application/
vnd.docker.image.rootfs.diff.tar.gzip\",\n         \"size\": 28005981,\n
 \"digest\":
 \"sha256:c44ba47496991c9982ee493b47fd25c252caabf2b4ae7dd679c9a27b6a3c8fb7\"\n
      },\n      {\n         \"mediaType\": \"application/
vnd.docker.image.rootfs.diff.tar.gzip\",\n         \"size\": 775,\n         \"digest
\": \"sha256:e2c388b44226544363ca007be7b896bcce1baebea04da23cbd165eac30be650f\"\n
   }\n   ]\n}"
        }
    ],
    "failures": []
}
```

Esempio 2: come ottenere più immagini

L'esempio `batch-get-image` seguente mostra i dettagli di tutte le immagini con tag `prod` e `team1` nel repository specificato.

```
aws ecr batch-get-image \
    --repository-name MyRepository \
    --image-ids imageTag=prod imageTag=team1
```

Output:

```
{
```

```
    "images": [
        {
            "registryId": "123456789012",
            "repositoryName": "MyRepository",
            "imageId": {
                "imageDigest": "sha256:123456789012",
                "imageTag": "prod"
            },
            "imageManifest": "manifestExample1"
        },
        {
            "registryId": "567890121234",
            "repositoryName": "MyRepository",
            "imageId": {
                "imageDigest": "sha256:123456789012",
                "imageTag": "team1"
            },
            "imageManifest": "manifestExample2"
        }
    ],
    "failures": []
}
```

Per ulteriori informazioni, consulta Immagini nella Guida per l'utente di Amazon ECR.

- Per i dettagli sull'API, consulta BatchGetImage AWS CLICommand Reference.

`complete-layer-upload`

Il seguente esempio di codice mostra come utilizzare`complete-layer-upload`.

AWS CLI

Come completare il caricamento di un livello immagine

L'esempio `complete-layer-upload` seguente completa il caricamento di un livello immagine nel repository `layer-test`.

```
aws ecr complete-layer-upload \
    --repository-name layer-test \
    --upload-id 6cb64b8a-9378-0e33-2ab1-b780fab8a9e9 \
    --layer-digests 6cb64b8a-9378-0e33-2ab1-
b780fab8a9e9:48074e6d3a68b39aad8ccc002cdad912d4148c0f92b3729323e
```

Output:

```
{
    "uploadId": "6cb64b8a-9378-0e33-2ab1-b780fab8a9e9",
    "layerDigest":
 "sha256:9a77f85878aa1906f2020a0ecdf7a7e962d57e882250acd773383224b3fe9a02",
    "repositoryName": "layer-test",
    "registryId": "130757420319"
}
```

- Per i dettagli sull'API, consulta CompleteLayerUpload AWS CLICommand Reference.

`create-repository`

Il seguente esempio di codice mostra come utilizzare`create-repository`.

AWS CLI

Esempio 1: come creare un repository

L'esempio `create-repository` seguente crea un repository all'interno del namespace specificato nel registro predefinito per un account.

```
aws ecr create-repository \
    --repository-name project-a/sample-repo
```

Output:

```
{
    "repository": {
        "registryId": "123456789012",
        "repositoryName": "project-a/sample-repo",
        "repositoryArn": "arn:aws:ecr:us-west-2:123456789012:repository/project-a/
sample-repo"
    }
}
```

Per ulteriori informazioni, consulta Creazione di un repository nella Guida per l'utente di Amazon ECR.

Esempio 2: come creare un repository configurato con l'immutabilità dei tag delle immagini

L'esempio `create-repository` seguente crea un repository configurato per l'immutabilità dei tag nel registro predefinito per un account.

```
aws ecr create-repository \
    --repository-name project-a/sample-repo \
    --image-tag-mutability IMMUTABLE
```

Output:

```
{
    "repository": {
        "registryId": "123456789012",
        "repositoryName": "project-a/sample-repo",
        "repositoryArn": "arn:aws:ecr:us-west-2:123456789012:repository/project-a/
sample-repo",
        "imageTagMutability": "IMMUTABLE"
    }
}
```

Per ulteriori informazioni, consulta Mutabilità del tag immagine nella Guida per l'utente di Amazon ECR.

Esempio 3: come creare un repository configurato con una configurazione di scansione

L'esempio `create-repository` seguente crea un repository configurato per l'esecuzione di una scansione delle vulnerabilità all'esecuzione del push delle immagini nel registro predefinito per un account.

```
aws ecr create-repository \
    --repository-name project-a/sample-repo \
    --image-scanning-configuration scanOnPush=true
```

Output:

```
{
    "repository": {
        "registryId": "123456789012",
        "repositoryName": "project-a/sample-repo",
        "repositoryArn": "arn:aws:ecr:us-west-2:123456789012:repository/project-a/
sample-repo",
        "imageScanningConfiguration": {
```

```
            "scanOnPush": true
        }
    }
}
```

Per ulteriori informazioni, consulta Scansione delle immagini nella Guida per l'utente di Amazon ECR.

- Per i dettagli sull'API, consulta CreateRepository AWS CLICommand Reference.

`delete-lifecycle-policy`

Il seguente esempio di codice mostra come utilizzare`delete-lifecycle-policy`.

AWS CLI

Come eliminare la policy del ciclo di vita per un repository

L'esempio `delete-lifecycle-policy` seguente elimina la policy del ciclo di vita per il repository `hello-world`.

```
aws ecr delete-lifecycle-policy \
    --repository-name hello-world
```

Output:

```
{
    "registryId": "012345678910",
    "repositoryName": "hello-world",
    "lifecyclePolicyText": "{\"rules\":[{\"rulePriority\":1,\"description\":\"Remove
 untagged images.\",\"selection\":{\"tagStatus\":\"untagged\",\"countType\":
\"sinceImagePushed\",\"countUnit\":\"days\",\"countNumber\":10},\"action\":{\"type
\":\"expire\"}}]}",
    "lastEvaluatedAt": 0.0
}
```

- Per i dettagli sull'API, consulta DeleteLifecyclePolicy AWS CLICommand Reference.

`delete-repository-policy`

Il seguente esempio di codice mostra come utilizzare`delete-repository-policy`.

AWS CLI

Come eliminare la policy del repository specificato

L'esempio `delete-repository-policy` seguente elimina la policy di repository per il repository `cluster-autoscaler`.

```
aws ecr delete-repository-policy \
    --repository-name cluster-autoscaler
```

Output:

```
{
    "registryId": "012345678910",
    "repositoryName": "cluster-autoscaler",
    "policyText": "{\n  \"Version\" : \"2008-10-17\",\n  \"Statement\" : [ {\n
 \"Sid\" : \"allow public pull\",\n    \"Effect\" : \"Allow\",\n    \"Principal\" :
 \"*\",\n    \"Action\" : [ \"ecr:BatchCheckLayerAvailability\", \"ecr:BatchGetImage
\", \"ecr:GetDownloadUrlForLayer\" ]\n  } ]\n}"
}
```

- Per i dettagli sull'API, consulta DeleteRepositoryPolicy AWS CLICommand Reference.

`delete-repository`

Il seguente esempio di codice mostra come utilizzare`delete-repository`.

AWS CLI

Come eliminare un repository

L'esempio `delete-repository` seguente forza l'eliminazione del repository specificato nel registro predefinito per un account. Il flag `--force` è obbligatorio se il repository contiene immagini.

```
aws ecr delete-repository \
    --repository-name ubuntu \
    --force
```

Output:

```
{
    "repository": {
        "registryId": "123456789012",
        "repositoryName": "ubuntu",
        "repositoryArn": "arn:aws:ecr:us-west-2:123456789012:repository/ubuntu"
    }
}
```

Per ulteriori informazioni, consulta Eliminazione di un repository nella Guida per l'utente di Amazon ECR.

- Per i dettagli sull'API, consulta DeleteRepository AWS CLICommand Reference.

`describe-image-scan-findings`

Il seguente esempio di codice mostra come utilizzare`describe-image-scan-findings`.

AWS CLI

Come descrivere gli esiti della scansione di un'immagine

L'esempio `describe-image-scan-findings` seguente restituisce gli esiti della scansione di un'immagine usando il digest dell'immagine nel repository specificato nel registro predefinito per un account.

```
aws ecr describe-image-scan-findings \
    --repository-name sample-repo \
    --image-
id imageDigest=sha256:74b2c688c700ec95a93e478cdb959737c148df3fbf5ea706abe0318726e885e6
```

Output:

```
{
    "imageScanFindings": {
      "findings": [
          {
              "name": "CVE-2019-5188",
              "description": "A code execution vulnerability exists in the directory
 rehashing functionality of E2fsprogs e2fsck 1.45.4. A specially crafted ext4
 directory can cause an out-of-bounds write on the stack, resulting in code
 execution. An attacker can corrupt a partition to trigger this vulnerability.",
```

```
                "uri": "http://people.ubuntu.com/~ubuntu-security/cve/CVE-2019-5188",
                "severity": "MEDIUM",
                "attributes": [
                    {
                        "key": "package_version",
                        "value": "1.44.1-1ubuntu1.1"
                    },
                    {
                        "key": "package_name",
                        "value": "e2fsprogs"
                    },
                    {
                        "key": "CVSS2_VECTOR",
                        "value": "AV:L/AC:L/Au:N/C:P/I:P/A:P"
                    },
                    {
                        "key": "CVSS2_SCORE",
                        "value": "4.6"
                    }
                ]
            }
        ],
        "imageScanCompletedAt": 1579839105.0,
        "vulnerabilitySourceUpdatedAt": 1579811117.0,
        "findingSeverityCounts": {
            "MEDIUM": 1
        }
    },
    "registryId": "123456789012",
    "repositoryName": "sample-repo",
    "imageId": {
        "imageDigest":
 "sha256:74b2c688c700ec95a93e478cdb959737c148df3fbf5ea706abe0318726e885e6"
    },
    "imageScanStatus": {
        "status": "COMPLETE",
        "description": "The scan was completed successfully."
    }
}
```


Per ulteriori informazioni, consulta Scansione delle immagini nella Guida per l'utente di Amazon ECR.

- Per i dettagli sull'API, consulta DescribeImageScanFindings AWS CLICommand Reference.

`describe-images`

Il seguente esempio di codice mostra come utilizzare`describe-images`.

AWS CLI

Come descrivere un'immagine in un repository

L'esempio `describe-images` seguente mostra i dettagli di un'immagine nel repository `cluster-autoscaler` con il tag `v1.13.6`.

```
aws ecr describe-images \
    --repository-name cluster-autoscaler \
    --image-ids imageTag=v1.13.6
```

Output:

```
{
    "imageDetails": [
        {
            "registryId": "012345678910",
            "repositoryName": "cluster-autoscaler",
            "imageDigest":
 "sha256:4a1c6567c38904384ebc64e35b7eeddd8451110c299e3368d2210066487d97e5",
            "imageTags": [
                "v1.13.6"
            ],
            "imageSizeInBytes": 48318255,
            "imagePushedAt": 1565128275.0
        }
    ]
}
```

- Per i dettagli sull'API, consulta DescribeImages AWS CLICommand Reference.

`describe-repositories`

Il seguente esempio di codice mostra come utilizzare`describe-repositories`.

AWS CLI

Come descrivere i repository in un registro

Questo esempio descrive i repository nel registro predefinito di un account.

Comando:

```
aws ecr describe-repositories
```

Output:

```
{
    "repositories": [
        {
            "registryId": "012345678910",
            "repositoryName": "ubuntu",
            "repositoryArn": "arn:aws:ecr:us-west-2:012345678910:repository/ubuntu"
        },
        {
            "registryId": "012345678910",
            "repositoryName": "test",
            "repositoryArn": "arn:aws:ecr:us-west-2:012345678910:repository/test"
        }
    ]
}
```

- Per i dettagli sull'API, consulta DescribeRepositories AWS CLICommand Reference.

`get-authorization-token`

Il seguente esempio di codice mostra come utilizzare`get-authorization-token`.

AWS CLI

Come ottenere un token di autorizzazione per il registro predefinito

L'esempio `get-authorization-token` seguente ottiene un token di autorizzazione per il registro predefinito.

```
aws ecr get-authorization-token
```

Output:

```
{
    "authorizationData": [
```

```
        {
            "authorizationToken": "QVdTOkN...",
            "expiresAt": 1448875853.241,
            "proxyEndpoint": "https://123456789012.dkr.ecr.us-west-2.amazonaws.com"
        }
    ]
}
```

- Per i dettagli sull'API, consulta GetAuthorizationToken AWS CLICommand Reference.

`get-download-url-for-layer`

Il seguente esempio di codice mostra come utilizzare`get-download-url-for-layer`.

AWS CLI

Come ottenere l'URL di download di un livello

L'esempio `get-download-url-for-layer` seguente mostra l'URL di download di un livello con il digest `sha256:6171c7451a50945f8ddd72f7732cc04d7a0d1f48138a426b2e64387fdeb834ed` nel repository `cluster-autoscaler`.

```
aws ecr get-download-url-for-layer \
    --repository-name cluster-autoscaler  \
    --layer-
digest sha256:6171c7451a50945f8ddd72f7732cc04d7a0d1f48138a426b2e64387fdeb834ed
```

Output:

```
{
    "downloadUrl": "https://prod-us-west-2-starport-layer-bucket.s3.us-
west-2.amazonaws.com/e501-012345678910-9cb60dc0-7284-5643-3987-
da6dac0465f0/04620aac-66a5-4167-8232-55ee7ef6d565?X-Amz-Algorithm=AWS4-HMAC-
SHA256&X-Amz-Date=20190814T220617Z&X-Amz-SignedHeaders=host&X-Amz-Expires=3600&X-
Amz-Credential=AKIA32P3D2JDNMVAJLGF%2F20190814%2Fus-west-2%2Fs3%2Faws4_request&X-
Amz-Signature=9161345894947a1672467a0da7a1550f2f7157318312fe4941b59976239c3337",
    "layerDigest":
 "sha256:6171c7451a50945f8ddd72f7732cc04d7a0d1f48138a426b2e64387fdeb834ed"
}
```

- Per i dettagli sull'API, consulta GetDownloadUrlForLayer AWS CLICommand Reference.

`get-lifecycle-policy-preview`

Il seguente esempio di codice mostra come utilizzare`get-lifecycle-policy-preview`.

AWS CLI

Come recuperare i dettagli di un'anteprima della policy del ciclo di vita

L'esempio `get-lifecycle-policy-preview` seguente recupera il risultato di un'anteprima della policy del ciclo di vita per il repository specificato nel registro predefinito per un account.

Comando:

```
aws ecr get-lifecycle-policy-preview \
    --repository-name "project-a/amazon-ecs-sample"
```

Output:

```
{
    "registryId": "012345678910",
    "repositoryName": "project-a/amazon-ecs-sample",
    "lifecyclePolicyText": "{\n    \"rules\": [\n        {\n
 \"rulePriority\": 1,\n            \"description\": \"Expire images older than 14
 days\",\n            \"selection\": {\n                \"tagStatus\": \"untagged\",
\n                \"countType\": \"sinceImagePushed\",\n                \"countUnit
\": \"days\",\n                \"countNumber\": 14\n            },\n
 \"action\": {\n                \"type\": \"expire\"\n            }\n        }\n
  ]\n}\n",
    "status": "COMPLETE",
    "previewResults": [],
    "summary": {
        "expiringImageTotalCount": 0
    }
}
```

Per ulteriori informazioni, consulta Policy del ciclo di vita nella Guida per l'utente di Amazon ECR.

- Per i dettagli sull'API, consulta GetLifecyclePolicyPreview AWS CLICommand Reference.

`get-lifecycle-policy`

Il seguente esempio di codice mostra come utilizzare`get-lifecycle-policy`.

AWS CLI

Come recuperare una policy del ciclo di vita

L'esempio `get-lifecycle-policy` seguente mostra i dettagli della policy del ciclo di vita per il repository specificato nel registro predefinito per un account.

```
aws ecr get-lifecycle-policy \
    --repository-name "project-a/amazon-ecs-sample"
```

Output:

```
{
     "registryId": "123456789012",
     "repositoryName": "project-a/amazon-ecs-sample",
     "lifecyclePolicyText": "{\"rules\":[{\"rulePriority\":1,\"description\":
\"Expire images older than 14 days\",\"selection\":{\"tagStatus\":\"untagged\",
\"countType\":\"sinceImagePushed\",\"countUnit\":\"days\",\"countNumber\":14},
\"action\":{\"type\":\"expire\"}}]}",
     "lastEvaluatedAt": 1504295007.0
}
```

Per ulteriori informazioni, consulta Policy del ciclo di vita nella Guida per l'utente di Amazon ECR.

- Per i dettagli sull'API, consulta GetLifecyclePolicy AWS CLICommand Reference.

`get-login-password`

Il seguente esempio di codice mostra come utilizzare`get-login-password`.

AWS CLI

Come recuperare una password per l'autenticazione in un registro

L'esempio `get-login-password` seguente mostra una password da utilizzare con un client container a tua scelta per l'autenticazione in qualsiasi registro Amazon ECR a cui il tuo principale IAM ha accesso.

```
aws ecr get-login-password
```

Output:

```
<password>
```

Per utilizzare con la CLI Docker, reindirizza l'output del comando `get-login-password` al comando `docker login`. Quando recuperi la password, assicurati di specificare la stessa Regione in cui si trova il Registro Amazon ECR.

```
aws ecr get-login-password \
    --region <region>
 \
| docker login \
    --username AWS \
    --password-stdin <aws_account_id>.dkr.ecr.<region>.amazonaws.com
```

Per ulteriori informazioni, consulta Autenticazione dei registri nella Guida per l'utente di Amazon ECR.

- Per i dettagli sull'API, consulta GetLoginPassword AWS CLICommand Reference.

`get-login`

Il seguente esempio di codice mostra come utilizzare`get-login`.

AWS CLI

Come recuperare un comando Docker di login nel registro predefinito

Questo esempio stampa un comando da utilizzare per accedere al registro Amazon ECR predefinito.

Comando:

```
aws ecr get-login
```

Output:

```
docker login -u AWS -p <password> -e none https://
<aws_account_id>.dkr.ecr.<region>.amazonaws.com
```

Come accedere al registro di un altro account

Questo esempio stampa un comando da utilizzare per accedere al registro Amazon ECR predefinito.

Comando:

```
aws ecr get-login --registry-ids 012345678910 023456789012
```

Output:

```
docker login -u <username> -p <token-1> -e none <endpoint-1>
docker login -u <username> -p <token-2> -e none <endpoint-2>
```

- Per i dettagli sull'API, consulta GetLogin AWS CLICommand Reference.

`get-repository-policy`

Il seguente esempio di codice mostra come utilizzare`get-repository-policy`.

AWS CLI

Come recuperare la policy del repository specificato

L'esempio `get-repository-policy` seguente mostra i dettagli della policy di repository per il repository `cluster-autoscaler`.

```
aws ecr get-repository-policy \
    --repository-name cluster-autoscaler
```

Output:

```
{
    "registryId": "012345678910",
    "repositoryName": "cluster-autoscaler",
    "policyText": "{\n  \"Version\" : \"2008-10-17\",\n  \"Statement\" : [ {\n
 \"Sid\" : \"allow public pull\",\n    \"Effect\" : \"Allow\",\n    \"Principal\" :
 \"*\",\n    \"Action\" : [ \"ecr:BatchCheckLayerAvailability\", \"ecr:BatchGetImage
\", \"ecr:GetDownloadUrlForLayer\" ]\n  } ]\n}"
}
```

- Per i dettagli sull'API, consulta GetRepositoryPolicy AWS CLICommand Reference.

`initiate-layer-upload`

Il seguente esempio di codice mostra come utilizzare`initiate-layer-upload`.

AWS CLI

Come avviare il caricamento di un livello immagine

L'esempio `initiate-layer-upload` seguente avvia il caricamento di un livello immagine nel repository `layer-test`.

```
aws ecr initiate-layer-upload \
    --repository-name layer-test
```

Output:

```
{
    "partSize": 10485760,
    "uploadId": "6cb64b8a-9378-0e33-2ab1-b780fab8a9e9"
}
```

- Per i dettagli sull'API, consulta InitiateLayerUpload AWS CLICommand Reference.

`list-images`

Il seguente esempio di codice mostra come utilizzare`list-images`.

AWS CLI

Come elencare le immagini in un repository

L'esempio `list-images` seguente mostra un elenco delle immagini nel repository `cluster-autoscaler`.

```
aws ecr list-images \
    --repository-name cluster-autoscaler
```

Output:

```
{
```

```
    "imageIds": [
        {
            "imageDigest":
 "sha256:99c6fb4377e9a420a1eb3b410a951c9f464eff3b7dbc76c65e434e39b94b6570",
            "imageTag": "v1.13.8"
        },
        {
            "imageDigest":
 "sha256:99c6fb4377e9a420a1eb3b410a951c9f464eff3b7dbc76c65e434e39b94b6570",
            "imageTag": "v1.13.7"
        },
        {
            "imageDigest":
 "sha256:4a1c6567c38904384ebc64e35b7eeddd8451110c299e3368d2210066487d97e5",
            "imageTag": "v1.13.6"
        }
    ]
}
```

- Per i dettagli sull'API, consulta ListImages AWS CLICommand Reference.

`list-tags-for-resource`

Il seguente esempio di codice mostra come utilizzare`list-tags-for-resource`.

AWS CLI

Come elencare i tag per il repository

L'esempio `list-tags-for-resource` seguente mostra un elenco dei tag associati al repository `hello-world`.

```
aws ecr list-tags-for-resource \
    --resource-arn arn:aws:ecr:us-west-2:012345678910:repository/hello-world
```

Output:

```
{
    "tags": [
        {
            "Key": "Stage",
```

```
            "Value": "Integ"
        }
    ]
}
```

- Per i dettagli sull'API, consulta ListTagsForResource AWS CLICommand Reference.

`put-image-scanning-configuration`

Il seguente esempio di codice mostra come utilizzare`put-image-scanning-configuration`.

AWS CLI

Come aggiornare la configurazione della scansione delle immagini per un repository

L'esempio `put-image-scanning-configuration` seguente aggiorna la configurazione della scansione delle immagini per il repository specificato.

```
aws ecr put-image-scanning-configuration \
    --repository-name sample-repo \
    --image-scanning-configuration scanOnPush=true
```

Output:

```
{
   "registryId": "012345678910",
   "repositoryName": "sample-repo",
   "imageScanningConfiguration": {
     "scanOnPush": true
   }
}
```

Per ulteriori informazioni, consulta Scansione delle immagini nella Guida per l'utente di Amazon ECR.

- Per i dettagli sull'API, consulta PutImageScanningConfiguration AWS CLICommand Reference.

`put-image-tag-mutability`

Il seguente esempio di codice mostra come utilizzare`put-image-tag-mutability`.

AWS CLI

Come aggiornare l'impostazione di immutabilità dei tag immagine per un repository

L'esempio `put-image-tag-mutability` seguente configura il repository specificato per l'immutabilità dei tag. Questo impedisce la sovrascrittura di tutti i tag immagine all'interno del repository.

```
aws ecr put-image-tag-mutability \
    --repository-name hello-repository \
    --image-tag-mutability IMMUTABLE
```

Output:

```
{
   "registryId": "012345678910",
   "repositoryName": "sample-repo",
   "imageTagMutability": "IMMUTABLE"
}
```

Per ulteriori informazioni, consulta Mutabilità del tag immagine nella Guida per l'utente di Amazon ECR.

- Per i dettagli sull'API, consulta PutImageTagMutability AWS CLICommand Reference.

`put-image`

Il seguente esempio di codice mostra come utilizzare`put-image`.

AWS CLI

Come taggare nuovamente un'immagine con il relativo manifesto

L'esempio `put-image` seguente crea un nuovo tag nel repository `hello-world` con un manifesto di immagini esistente.

```
aws ecr put-image \
    --repository-name hello-world \
    --image-tag 2019.08 \
    --image-manifest file://hello-world.manifest.json
```

Contenuto di `hello-world.manifest.json`:

```
{
    "schemaVersion": 2,
    "mediaType": "application/vnd.docker.distribution.manifest.v2+json",
    "config": {
        "mediaType": "application/vnd.docker.container.image.v1+json",
        "size": 5695,
        "digest":
 "sha256:cea5fe7701b7db3dd1c372f3cea6f43cdda444fcc488f530829145e426d8b980"
    },
    "layers": [
        {
            "mediaType": "application/vnd.docker.image.rootfs.diff.tar.gzip",
            "size": 39096921,
            "digest":
 "sha256:d8868e50ac4c7104d2200d42f432b661b2da8c1e417ccfae217e6a1e04bb9295"
        },
        {
            "mediaType": "application/vnd.docker.image.rootfs.diff.tar.gzip",
            "size": 57938,
            "digest":
 "sha256:83251ac64627fc331584f6c498b3aba5badc01574e2c70b2499af3af16630eed"
        },
        {
            "mediaType": "application/vnd.docker.image.rootfs.diff.tar.gzip",
            "size": 423,
            "digest":
 "sha256:589bba2f1b36ae56f0152c246e2541c5aa604b058febfcf2be32e9a304fec610"
        },
        {
            "mediaType": "application/vnd.docker.image.rootfs.diff.tar.gzip",
            "size": 680,
            "digest":
 "sha256:d62ecaceda3964b735cdd2af613d6bb136a52c1da0838b2ff4b4dab4212bcb1c"
        },
        {
            "mediaType": "application/vnd.docker.image.rootfs.diff.tar.gzip",
            "size": 162,
            "digest":
 "sha256:6d93b41cfc6bf0d2522b7cf61588de4cd045065b36c52bd3aec2ba0622b2b22b"
        },
        {
            "mediaType": "application/vnd.docker.image.rootfs.diff.tar.gzip",
```

```
            "size": 28268840,
            "digest":
 "sha256:6986b4d4c07932c680b3587f2eac8b0e013568c003cc23b04044628a5c5e599f"
        },
        {
            "mediaType": "application/vnd.docker.image.rootfs.diff.tar.gzip",
            "size": 35369152,
            "digest":
 "sha256:8c5ec60f10102dc8da0649d866c7c2f706e459d0bdc25c83ad2de86f4996c276"
        },
        {
            "mediaType": "application/vnd.docker.image.rootfs.diff.tar.gzip",
            "size": 155,
            "digest":
 "sha256:cde50b1c594539c5f67cbede9aef95c9ae321ccfb857f7b251b45b84198adc85"
        },
        {
            "mediaType": "application/vnd.docker.image.rootfs.diff.tar.gzip",
            "size": 28737,
            "digest":
 "sha256:2e102807ab72a73fc9abf53e8c50e421bdc337a0a8afcb242176edeec65977e4"
        },
        {
            "mediaType": "application/vnd.docker.image.rootfs.diff.tar.gzip",
            "size": 190,
            "digest":
 "sha256:fc379bbd5ed37808772bef016553a297356c59b8f134659e6ee4ecb563c2f5a7"
        },
        {
            "mediaType": "application/vnd.docker.image.rootfs.diff.tar.gzip",
            "size": 28748,
            "digest":
 "sha256:021db240dfccf5a1aff19507d17c0177e5888e518acf295b52204b1825e8b7ee"
        }
    ]
}
```

Output:

```
{
    "image": {
        "registryId": "130757420319",
        "repositoryName": "hello-world",
```

```
        "imageId": {
            "imageDigest":
 "sha256:8ece96b74f87652876199d83bd107d0435a196133af383ac54cb82b6cc5283ae",
            "imageTag": "2019.08"
        },
        "imageManifest": "{\n  \"schemaVersion\": 2,\n  \"mediaType
\": \"application/vnd.docker.distribution.manifest.v2+json
\",\n  \"config\": {\n    \"mediaType\": \"application/
vnd.docker.container.image.v1+json\",\n    \"size\": 5695,\n    \"digest\":
 \"sha256:cea5fe7701b7db3dd1c372f3cea6f43cdda444fcc488f530829145e426d8b980\"\n
  },\n  \"layers\": [\n    {\n      \"mediaType\": \"application/
vnd.docker.image.rootfs.diff.tar.gzip\",\n      \"size\": 39096921,\n      \"digest
\": \"sha256:d8868e50ac4c7104d2200d42f432b661b2da8c1e417ccfae217e6a1e04bb9295\"\n
  },\n    {\n      \"mediaType\": \"application/
vnd.docker.image.rootfs.diff.tar.gzip\",\n      \"size\": 57938,\n      \"digest
\": \"sha256:83251ac64627fc331584f6c498b3aba5badc01574e2c70b2499af3af16630eed
\"\n    },\n    {\n      \"mediaType\": \"application/
vnd.docker.image.rootfs.diff.tar.gzip\",\n      \"size\": 423,\n      \"digest\":
 \"sha256:589bba2f1b36ae56f0152c246e2541c5aa604b058febfcf2be32e9a304fec610\"\n    },
\n    {\n      \"mediaType\": \"application/vnd.docker.image.rootfs.diff.tar.gzip\",
\n      \"size\": 680,\n      \"digest\":
 \"sha256:d62ecaceda3964b735cdd2af613d6bb136a52c1da0838b2ff4b4dab4212bcb1c
\"\n    },\n    {\n      \"mediaType\": \"application/
vnd.docker.image.rootfs.diff.tar.gzip\",\n      \"size\": 162,\n      \"digest
\": \"sha256:6d93b41cfc6bf0d2522b7cf61588de4cd045065b36c52bd3aec2ba0622b2b22b
\"\n    },\n    {\n      \"mediaType\": \"application/
vnd.docker.image.rootfs.diff.tar.gzip\",\n      \"size\": 28268840,\n      \"digest
\": \"sha256:6986b4d4c07932c680b3587f2eac8b0e013568c003cc23b04044628a5c5e599f
\"\n    },\n    {\n      \"mediaType\": \"application/
vnd.docker.image.rootfs.diff.tar.gzip\",\n      \"size\": 35369152,\n      \"digest
\": \"sha256:8c5ec60f10102dc8da0649d866c7c2f706e459d0bdc25c83ad2de86f4996c276\"\n
  },\n    {\n      \"mediaType\": \"application/
vnd.docker.image.rootfs.diff.tar.gzip\",\n      \"size\": 155,\n      \"digest\":
 \"sha256:cde50b1c594539c5f67cbede9aef95c9ae321ccfb857f7b251b45b84198adc85\"\n    },
\n    {\n      \"mediaType\": \"application/vnd.docker.image.rootfs.diff.tar.gzip\",
\n      \"size\": 28737,\n      \"digest\":
 \"sha256:2e102807ab72a73fc9abf53e8c50e421bdc337a0a8afcb242176edeec65977e4\"\n    },
\n    {\n      \"mediaType\": \"application/vnd.docker.image.rootfs.diff.tar.gzip\",
\n      \"size\": 190,\n      \"digest\":
 \"sha256:fc379bbd5ed37808772bef016553a297356c59b8f134659e6ee4ecb563c2f5a7\"\n    },
\n    {\n      \"mediaType\": \"application/vnd.docker.image.rootfs.diff.tar.gzip\",
\n      \"size\": 28748,\n      \"digest\":
 \"sha256:021db240dfccf5a1aff19507d17c0177e5888e518acf295b52204b1825e8b7ee\"\n
  }\n  ]\n}\n"
```

```
    }
}
```

- Per i dettagli sull'API, consulta PutImage AWS CLICommand Reference.

`put-lifecycle-policy`

Il seguente esempio di codice mostra come utilizzare`put-lifecycle-policy`.

AWS CLI

Come creare una policy del ciclo di vita

L'esempio `put-lifecycle-policy` seguente crea una policy del ciclo di vita per il repository specificato nel registro predefinito per un account.

```
aws ecr put-lifecycle-policy \
    --repository-name "project-a/amazon-ecs-sample" \
    --lifecycle-policy-text "file://policy.json"
```

Contenuto di `policy.json`:

```
{
   "rules": [
       {
           "rulePriority": 1,
           "description": "Expire images older than 14 days",
           "selection": {
               "tagStatus": "untagged",
               "countType": "sinceImagePushed",
               "countUnit": "days",
               "countNumber": 14
           },
           "action": {
               "type": "expire"
           }
       }
   ]
}
```

Output:

```
{
   "registryId": "<aws_account_id>",
   "repositoryName": "project-a/amazon-ecs-sample",
   "lifecyclePolicyText": "{\"rules\":[{\"rulePriority\":1,\"description\":\"Expire
 images older than 14 days\",\"selection\":{\"tagStatus\":\"untagged\",\"countType
\":\"sinceImagePushed\",\"countUnit\":\"days\",\"countNumber\":14},\"action\":
{\"type\":\"expire\"}}]}"
}
```

Per ulteriori informazioni, consulta Policy del ciclo di vita nella Guida per l'utente di Amazon ECR.

- Per i dettagli sull'API, consulta PutLifeCyclePolicy AWS CLICommand Reference.

`set-repository-policy`

Il seguente esempio di codice mostra come utilizzare`set-repository-policy`.

AWS CLI

Come impostare la policy per un repository

L'esempio `set-repository-policy` seguente associa una policy di repository contenuta in un file al repository `cluster-autoscaler`.

```
aws ecr set-repository-policy \
    --repository-name cluster-autoscaler \
    --policy-text file://my-policy.json
```

Contenuto di `my-policy.json`:

```
{
    "Version":"2012-10-17",
    "Statement" : [
        {
            "Sid" : "allow public pull",
            "Effect" : "Allow",
            "Principal" : "*",
            "Action" : [
                "ecr:BatchCheckLayerAvailability",
                "ecr:BatchGetImage",
                "ecr:GetDownloadUrlForLayer"
```

```
            ]
        }
    ]
}
```

Output:

```
{
    "registryId": "012345678910",
    "repositoryName": "cluster-autoscaler",
    "policyText": "{\n  \"Version\" : \"2008-10-17\",\n  \"Statement\" : [ {\n
 \"Sid\" : \"allow public pull\",\n    \"Effect\" : \"Allow\",\n    \"Principal\" :
 \"*\",\n    \"Action\" : [ \"ecr:BatchCheckLayerAvailability\", \"ecr:BatchGetImage
\", \"ecr:GetDownloadUrlForLayer\" ]\n  } ]\n}"
}
```

- Per i dettagli sull'API, consulta SetRepositoryPolicy AWS CLICommand Reference.

`start-image-scan`

Il seguente esempio di codice mostra come utilizzare`start-image-scan`.

AWS CLI

Come avviare una scansione delle vulnerabilità di un'immagine

L'esempio `start-image-scan` seguente avvia una scansione di un'immagine specificata dal digest nel repository specificato.

```
aws ecr start-image-scan \
    --repository-name sample-repo \
    --image-
id imageDigest=sha256:74b2c688c700ec95a93e478cdb959737c148df3fbf5ea706abe0318726e885e6
```

Output:

```
{
   "registryId": "012345678910",
   "repositoryName": "sample-repo",
   "imageId": {
```

```
        "imageDigest":
 "sha256:74b2c688c700ec95a93e478cdb959737c148df3fbf5ea706abe0318726e885e6"
    },
    "imageScanStatus": {
        "status": "IN_PROGRESS"
    }
}
```

Per ulteriori informazioni, consulta Scansione delle immagini nella Guida per l'utente di Amazon ECR.

- Per i dettagli sull'API, consulta StartImageScan AWS CLICommand Reference.

`start-lifecycle-policy-preview`

Il seguente esempio di codice mostra come utilizzare`start-lifecycle-policy-preview`.

AWS CLI

Come creare un'anteprima di una policy del ciclo di vita

L'esempio `start-lifecycle-policy-preview` seguente crea un'anteprima della policy del ciclo di vita definita in un file JSON per il repository specificato.

```
aws ecr start-lifecycle-policy-preview \
    --repository-name "project-a/amazon-ecs-sample" \
    --lifecycle-policy-text "file://policy.json"
```

Contenuto di `policy.json`:

```
{
    "rules": [
        {
            "rulePriority": 1,
            "description": "Expire images older than 14 days",
            "selection": {
                "tagStatus": "untagged",
                "countType": "sinceImagePushed",
                "countUnit": "days",
                "countNumber": 14
            },
            "action": {
```

```
                "type": "expire"
            }
        }
    ]
}
```

Output:

```
{
    "registryId": "012345678910",
    "repositoryName": "project-a/amazon-ecs-sample",
    "lifecyclePolicyText": "{\n    \"rules\": [\n        {\n
 \"rulePriority\": 1,\n            \"description\": \"Expire images older than 14
 days\",\n            \"selection\": {\n                \"tagStatus\": \"untagged\",
\n                \"countType\": \"sinceImagePushed\",\n                \"countUnit
\": \"days\",\n                \"countNumber\": 14\n            },\n
 \"action\": {\n                \"type\": \"expire\"\n            }\n        }\n
  ]\n}\n",
    "status": "IN_PROGRESS"
}
```

- Per i dettagli sull'API, consulta StartLifecyclePolicyPreview AWS CLICommand Reference.

`tag-resource`

Il seguente esempio di codice mostra come utilizzare`tag-resource`.

AWS CLI

Come aggiungere un tag a un repository

L'esempio `tag-resource` seguente imposta un tag con chiave `Stage` e valore `Integ` sul repository `hello-world`.

```
aws ecr tag-resource \
    --resource-arn arn:aws:ecr:us-west-2:012345678910:repository/hello-world \
    --tags Key=Stage,Value=Integ
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta TagResource AWS CLICommand Reference.

`untag-resource`

Il seguente esempio di codice mostra come utilizzare`untag-resource`.

AWS CLI

Come rimuovere un tag da un repository

L'esempio `untag-resource` seguente rimuove il tag con la chiave `Stage` dal repository `hello-world`.

```
aws ecr untag-resource \
    --resource-arn arn:aws:ecr:us-west-2:012345678910:repository/hello-world \
    --tag-keys Stage
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta UntagResource AWS CLICommand Reference.

`upload-layer-part`

Il seguente esempio di codice mostra come utilizzare`upload-layer-part`.

AWS CLI

Come caricare una parte del livello

L'esempio `upload-layer-part` seguente carica una parte di livello immagine nel repository `layer-test`.

```
aws ecr upload-layer-part \
    --repository-name layer-test \
    --upload-id 6cb64b8a-9378-0e33-2ab1-b780fab8a9e9 \
    --part-first-byte 0 \
    --part-last-byte 8323314 \
    --layer-part-blob file:///var/lib/docker/image/overlay2/layerdb/sha256/
ff986b10a018b48074e6d3a68b39aad8ccc002cdad912d4148c0f92b3729323e/layer.b64
```

Output:

```
{
    "uploadId": "6cb64b8a-9378-0e33-2ab1-b780fab8a9e9",
```

```
    "registryId": "012345678910",
    "lastByteReceived": 8323314,
    "repositoryName": "layer-test"
}
```

- Per i dettagli sull'API, consulta UploadLayerPart AWS CLICommand Reference.

Amazon ECR Public: esempi di utilizzo AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando Amazon ECR Public. AWS Command Line Interface

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`batch-delete-image`

Il seguente esempio di codice mostra come usare`batch-delete-image`.

AWS CLI

Esempio 1: se si elimina un'immagine utilizzando gli ID digest dell'immagine, verrà eliminata l'immagine assieme a tutti i relativi tag all'interno di un repository in un registro pubblico

L'esempio `batch-delete-image` seguente elimina un'immagine specificandone il digest:

```
aws ecr-public batch-delete-image \
    --repository-name project-a/nginx-web-app \
    --image-
ids imageDigest=sha256:b1f9deb5fe3711a3278379ebbcaefbc5d70a2263135db86bd27a0dae150546c2
```

Output:

```
{
"imageIds": [
    {
        "imageDigest":
 "sha256:b1f9deb5fe3711a3278379ebbcaefbc5d70a2263135db86bd27a0dae150546c2",
        "imageTag": "latest"
    }
],
"failures": []
}
```

Per ulteriori informazioni, consulta Eliminazione di un'immagine in un repository pubblico nella Guida per l'utente di Amazon ECR Public.

Esempio 2: come eliminare qualsiasi immagine specificando il tag associato all'immagine da eliminare dal repository.

L'esempio `batch-delete-image` seguente elimina un'immagine specificando il tag associato all'archivio di immagini denominato `project-a/nginx-web-app` in un registro pubblico. Se è presente un solo tag ed esegui questo comando, l'immagine verrà rimossa. In caso di più tag associati alla stessa immagine, specificane uno e dal repository sarà rimosso solo il tag e non l'immagine.

```
aws ecr-public batch-delete-image \
    --repository-name project-a/nginx-web-app \
    --image-ids imageTag=_temp
```

Output:

```
{
    "imageIds": [
        {
            "imageDigest":
 "sha256:f7a86a0760e2f8d7eff07e515fc87bf4bac45c35376c06f9a280f15ecad6d7e0",
            "imageTag": "_temp"
        }
    ],
    "failures": []
}
```

Per ulteriori informazioni, consulta Eliminazione di un'immagine in un repository pubblico nella Guida per l'utente di Amazon ECR Public.

Esempio 3: come eliminare più immagini, puoi specificare più tag immagine o i digest delle immagini nella richiesta per un repository in un registro pubblico.

L'`batch-delete-image`esempio seguente elimina più immagini da un archivio denominato project-a/ nginx-web-app specificando più tag di immagine o digest di immagini nella richiesta.

```
aws ecr-public batch-delete-image \
    --repository-name project-a/nginx-web-app \
    --image-ids imageTag=temp2.0
  imageDigest=sha256:47ba980bc055353d9c0af89b1894f68faa43ca93856917b8406316be86f01278
```

Output:

```
{
     "imageIds": [
         {
             "imageDigest":
 "sha256:47ba980bc055353d9c0af89b1894f68faa43ca93856917b8406316be86f01278"
         },
         {
             "imageDigest":
 "sha256:f7a86a0760e2f8d7eff07e515fc87bf4bac45c35376c06f9a280f15ecad6d7e0",
             "imageTag": "temp2.0"
         }
     ],
     "failures": []
 }
```

Per ulteriori informazioni, consulta Eliminazione di un'immagine in un repository pubblico nella Guida per l'utente di Amazon ECR Public.

Esempio 4: Per eliminare un'immagine in cross AWS Account utilizzando registry-id e imagedigest ids, l'immagine e tutti i relativi tag vengono eliminati all'interno di un archivio in un registro pubblico

L'`batch-delete-image`esempio seguente elimina un'immagine specificando l'image digest nel cross Account. AWS :

```
aws ecr-public batch-delete-image \
    --registry-id 123456789098 \
```

```
    --repository-name project-a/nginx-web-app \
    --image-
ids imageDigest=sha256:b1f9deb5fe3711a3278379ebbcaefbc5d70a2263135db86bd27a0dae150546c2
 \
    --region us-east-1
```

Output:

```
{
    "imageIds": [
        {
            "imageDigest":
 "sha256:b1f9deb5fe3711a3278379ebbcaefbc5d70a2263135db86bd27a0dae150546c2",
            "imageTag": "temp2.0"
        }
    ],
    "failures": []
}
```

Per ulteriori informazioni, consulta Eliminazione di un'immagine in un repository pubblico nella Guida per l'utente di Amazon ECR Public.

- Per i dettagli sull'API, consulta BatchDeleteImage AWS CLICommand Reference.

`create-repository`

Il seguente esempio di codice mostra come utilizzare`create-repository`.

AWS CLI

Esempio 1: come creare un repository in un registro pubblico

L'`create-repository`esempio seguente crea un repository denominato project-a/ nginx-web-app in un registro pubblico.

```
aws ecr-public create-repository \
    --repository-name project-a/nginx-web-app
```

Output:

```
{
    "repository": {
```

```
        "repositoryArn": "arn:aws:ecr-public::123456789012:repository/project-a/
nginx-web-app",
        "registryId": "123456789012",
        "repositoryName": "project-a/nginx-web-app",
        "repositoryUri": "public.ecr.aws/public-registry-custom-alias/project-a/
nginx-web-app",
        "createdAt": "2024-07-01T21:08:55.131000+00:00"
    },
    "catalogData": {}
}
```

Per ulteriori informazioni, consulta Creazione di un repository pubblico nella Guida per l'utente di Amazon ECR Public.

Esempio 2: come creare un repository in un registro pubblico con una breve descrizione del contenuto del repository, del sistema e dell'architettura operativa con cui le immagini del repository sono compatibili

L'`create-repository`esempio seguente crea un archivio denominato project-a/ nginx-web-app in un registro pubblico con una breve descrizione del contenuto del repository, del sistema e dell'architettura operativa con cui le immagini del repository sono compatibili.

```
aws ecr-public create-repository \
    --repository-name project-a/nginx-web-app \
    --catalog-data 'description=My project-a ECR Public
 Repository,architectures=ARM,ARM 64,x86,x86-64,operatingSystems=Linux'
```

Output:

```
{
    "repository": {
        "repositoryArn": "arn:aws:ecr-public::123456789012:repository/project-a/
nginx-web-app",
        "registryId": "123456789012",
        "repositoryName": "project-a/nginx-web-app",
        "repositoryUri": "public.ecr.aws/public-registry-custom-alias/project-a/
nginx-web-app",
        "createdAt": "2024-07-01T21:23:20.455000+00:00"
    },
    "catalogData": {
        "description": "My project-a ECR Public Repository",
        "architectures": [
```

```
            "ARM",
            "ARM 64",
            "x86",
            "x86-64"
        ],
        "operatingSystems": [
            "Linux"
        ]
    }
}
```

Per ulteriori informazioni, consulta Creazione di un repository pubblico nella Guida per l'utente di Amazon ECR Public.

Esempio 3: creare un repository in un registro pubblico, insieme alle informazioni su aboutText, UsageText e sui tag logoImageBlob

L'`create-repository`esempio seguente crea un archivio denominato project-a/ nginx-web-app in un registro pubblico, insieme a informazioni su aboutText, UsageText e sui tag. logoImageBlob

```
aws ecr-public create-repository \
    --cli-input-json file://myfile.json
```

Contenuto di `myfile.json`:

```
{
    "repositoryName": "project-a/nginx-web-app",
    "catalogData": {
        "description": "My project-a ECR Public Repository",
        "architectures": [
            "ARM",
            "ARM 64",
            "x86",
            "x86-64"
        ],
        "operatingSystems": [
            "Linux"
        ],
        "logoImageBlob": "iVBORw0KGgoA<<truncated-for-better-reading>>ErkJggg==",
        "aboutText": "## Quick reference\n\nMaintained by: [the Amazon Linux Team]
(https://github.com/aws/amazon-linux-docker-images)\n\nWhere to get help: [the
 Docker Community Forums](https://forums.docker.com/), [the Docker Community Slack]
(https://dockr.ly/slack), or [Stack Overflow](https://stackoverflow.com/search?
```

tab=newest&q=docker)\n\n## Supported tags and respective `dockerfile` links\n\n*
 [`2.0.20200722.0`, `2`, `latest`](https://github.com/amazonlinux/container-images/
blob/03d54f8c4d522bf712cffd6c8f9aafba0a875e78/Dockerfile)\n* [`2.0.20200722.0-
with-sources`, `2-with-sources`, `with-sources`](https://github.com/
amazonlinux/container-images/blob/1e7349845e029a2e6afe6dc473ef17d052e3546f/
Dockerfile)\n* [`2018.03.0.20200602.1`, `2018.03`, `1`](https://github.com/
amazonlinux/container-images/blob/f10932e08c75457eeb372bf1cc47ea2a4b8e98c8/
Dockerfile)\n* [`2018.03.0.20200602.1-with-sources`, `2018.03-with-sources`,
 `1-with-sources`](https://github.com/amazonlinux/container-images/
blob/8c9ee491689d901aa72719be0ec12087a5fa8faf/Dockerfile)\n\n## What is Amazon
 Linux?\n\nAmazon Linux is provided by Amazon Web Services (AWS). It is designed
 to provide a stable, secure, and high-performance execution environment for
 applications running on Amazon EC2. The full distribution includes packages that
 enable easy integration with AWS, including launch configuration tools and many
 popular AWS libraries and tools. AWS provides ongoing security and maintenance
 updates to all instances running Amazon Linux.\n\nThe Amazon Linux container image
 contains a minimal set of packages. To install additional packages, [use `yum`]
(https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/managing-software.html).\n\nAWS
 provides two versions of Amazon Linux: [Amazon Linux 2](https://aws.amazon.com/
amazon-linux-2/) and [Amazon Linux AMI](https://aws.amazon.com/amazon-linux-ami/).
\n\nFor information on security updates for Amazon Linux, please refer to [Amazon
 Linux 2 Security Advisories](https://alas.aws.amazon.com/alas2.html) and [Amazon
 Linux AMI Security Advisories](https://alas.aws.amazon.com/). Note that Docker
 Hub's vulnerability scanning for Amazon Linux is currently based on RPM versions,
 which does not reflect the state of backported patches for vulnerabilities.\n
\n## Where can I run Amazon Linux container images?\n\nYou can run Amazon Linux
 container images in any Docker based environment. Examples include, your laptop,
 in Amazon EC2 instances, and Amazon ECS clusters.\n\n## License\n\nAmazon Linux is
 available under the [GNU General Public License, version 2.0](https://github.com/
aws/amazon-linux-docker-images/blob/master/LICENSE). Individual software packages
 are available under their own licenses; run `rpm -qi [package name]` or check
 `/usr/share/doc/[package name]-*` and `/usr/share/licenses/[package name]-*` for
 details.\n\nAs with all Docker images, these likely also contain other software
 which may be under other licenses (such as Bash, etc from the base distribution,
 along with any direct or indirect dependencies of the primary software being
 contained).\n\nSome additional license information which was able to be auto-
detected might be found in [the `repo-info` repository's `amazonlinux/` directory]
(https://github.com/docker-library/repo-info/tree/master/repos/amazonlinux).\n\n##
 Security\n\nFor information on security updates for Amazon Linux, please refer
 to [Amazon Linux 2 Security Advisories](https://alas.aws.amazon.com/alas2.html)
 and [Amazon Linux AMI Security Advisories](https://alas.aws.amazon.com/). Note
 that Docker Hub's vulnerability scanning for Amazon Linux is currently based
 on RPM versions, which does not reflect the state of backported patches for
 vulnerabilities.",

```
        "usageText": "## Supported architectures\n\namd64, arm64v8\n\n## Where
 can I run Amazon Linux container images?\n\nYou can run Amazon Linux container
 images in any Docker based environment. Examples include, your laptop, in Amazon
 EC2 instances, and ECS clusters.\n\n## How do I install a software package from
 Extras repository in Amazon Linux 2?\n\nAvailable packages can be listed with the
 `amazon-linux-extras` command. Packages can be installed with the `amazon-linux-
extras install <package>` command. Example: `amazon-linux-extras install rust1`\n
\n## Will updates be available for Amazon Linux containers?\n\nSimilar to the Amazon
 Linux images for Amazon EC2 and on-premises use, Amazon Linux container images will
 get ongoing updates from Amazon in the form of security updates, bug fix updates,
 and other enhancements. Security bulletins for Amazon Linux are available at
 https://alas.aws.amazon.com/\n\n## Will AWS Support the current version of Amazon
 Linux going forward?\n\nYes; in order to avoid any disruption to your existing
 applications and to facilitate migration to Amazon Linux 2, AWS will provide
 regular security updates for Amazon Linux 2018.03 AMI and container image for 2
 years after the final LTS build is announced. You can also use all your existing
 support channels such as AWS Support and Amazon Linux Discussion Forum to continue
 to submit support requests."
    },
    "tags": [
        {
            "Key": "Name",
            "Value": "project-a/nginx-web-app"
        },
        {
            "Key": "Environment",
            "Value": "Prod"
        }
    ]
}
```

Output:

```
{
    "repository": {
        "repositoryArn": "arn:aws:ecr-public::123456789012:repository/project-a/
nginx-web-app",
        "registryId": "123456789012",
        "repositoryName": "project-a/nginx-web-app",
        "repositoryUri": "public.ecr.aws/public-registry-custom-alias/project-a/
nginx-web-app",
        "createdAt": "2024-07-01T21:53:05.749000+00:00"
    },
```

```
    "catalogData": {
        "description": "My project-a ECR Public Repository",
        "architectures": [
            "ARM",
            "ARM 64",
            "x86",
            "x86-64"
        ],
        "operatingSystems": [
            "Linux"
        ],
        "logoUrl": "https://d3g9o9u8re44ak.cloudfront.net/
logo/23861450-4b9b-403c-9a4c-7aa0ef140bb8/2f9bf5a7-a32f-45b4-b5cd-c5770a35e6d7.png",
        "aboutText": "## Quick reference\n\nMaintained by: [the Amazon Linux Team]
(https://github.com/aws/amazon-linux-docker-images)\n\nWhere to get help: [the
 Docker Community Forums](https://forums.docker.com/), [the Docker Community Slack]
(https://dockr.ly/slack), or [Stack Overflow](https://stackoverflow.com/search?
tab=newest&q=docker)\n\n## Supported tags and respective `dockerfile` links\n\n*
 [`2.0.20200722.0`, `2`, `latest`](https://github.com/amazonlinux/container-images/
blob/03d54f8c4d522bf712cffd6c8f9aafba0a875e78/Dockerfile)\n* [`2.0.20200722.0-
with-sources`, `2-with-sources`, `with-sources`](https://github.com/
amazonlinux/container-images/blob/1e7349845e029a2e6afe6dc473ef17d052e3546f/
Dockerfile)\n* [`2018.03.0.20200602.1`, `2018.03`, `1`](https://github.com/
amazonlinux/container-images/blob/f10932e08c75457eeb372bf1cc47ea2a4b8e98c8/
Dockerfile)\n* [`2018.03.0.20200602.1-with-sources`, `2018.03-with-sources`,
 `1-with-sources`](https://github.com/amazonlinux/container-images/
blob/8c9ee491689d901aa72719be0ec12087a5fa8faf/Dockerfile)\n\n## What is Amazon
 Linux?\n\nAmazon Linux is provided by Amazon Web Services (AWS). It is designed
 to provide a stable, secure, and high-performance execution environment for
 applications running on Amazon EC2. The full distribution includes packages that
 enable easy integration with AWS, including launch configuration tools and many
 popular AWS libraries and tools. AWS provides ongoing security and maintenance
 updates to all instances running Amazon Linux.\n\nThe Amazon Linux container image
 contains a minimal set of packages. To install additional packages, [use `yum`]
(https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/managing-software.html).\n\nAWS
 provides two versions of Amazon Linux: [Amazon Linux 2](https://aws.amazon.com/
amazon-linux-2/) and [Amazon Linux AMI](https://aws.amazon.com/amazon-linux-ami/).
\n\nFor information on security updates for Amazon Linux, please refer to [Amazon
 Linux 2 Security Advisories](https://alas.aws.amazon.com/alas2.html) and [Amazon
 Linux AMI Security Advisories](https://alas.aws.amazon.com/). Note that Docker
 Hub's vulnerability scanning for Amazon Linux is currently based on RPM versions,
 which does not reflect the state of backported patches for vulnerabilities.\n
\n## Where can I run Amazon Linux container images?\n\nYou can run Amazon Linux
 container images in any Docker based environment. Examples include, your laptop,
```

```
 in Amazon EC2 instances, and Amazon ECS clusters.\n\n## License\n\nAmazon Linux is
 available under the [GNU General Public License, version 2.0](https://github.com/
aws/amazon-linux-docker-images/blob/master/LICENSE). Individual software packages
 are available under their own licenses; run `rpm -qi [package name]` or check
 `/usr/share/doc/[package name]-*` and `/usr/share/licenses/[package name]-*` for
 details.\n\nAs with all Docker images, these likely also contain other software
 which may be under other licenses (such as Bash, etc from the base distribution,
 along with any direct or indirect dependencies of the primary software being
 contained).\n\nSome additional license information which was able to be auto-
detected might be found in [the `repo-info` repository's `amazonlinux/` directory]
(https://github.com/docker-library/repo-info/tree/master/repos/amazonlinux).\n\n##
 Security\n\nFor information on security updates for Amazon Linux, please refer
 to [Amazon Linux 2 Security Advisories](https://alas.aws.amazon.com/alas2.html)
 and [Amazon Linux AMI Security Advisories](https://alas.aws.amazon.com/). Note
 that Docker Hub's vulnerability scanning for Amazon Linux is currently based
 on RPM versions, which does not reflect the state of backported patches for
 vulnerabilities.",
        "usageText": "## Supported architectures\n\namd64, arm64v8\n\n## Where
 can I run Amazon Linux container images?\n\nYou can run Amazon Linux container
 images in any Docker based environment. Examples include, your laptop, in Amazon
 EC2 instances, and ECS clusters.\n\n## How do I install a software package from
 Extras repository in Amazon Linux 2?\n\nAvailable packages can be listed with the
 `amazon-linux-extras` command. Packages can be installed with the `amazon-linux-
extras install <package>` command. Example: `amazon-linux-extras install rust1`\n
\n## Will updates be available for Amazon Linux containers?\n\nSimilar to the Amazon
 Linux images for Amazon EC2 and on-premises use, Amazon Linux container images will
 get ongoing updates from Amazon in the form of security updates, bug fix updates,
 and other enhancements. Security bulletins for Amazon Linux are available at
 https://alas.aws.amazon.com/\n\n## Will AWS Support the current version of Amazon
 Linux going forward?\n\nYes; in order to avoid any disruption to your existing
 applications and to facilitate migration to Amazon Linux 2, AWS will provide
 regular security updates for Amazon Linux 2018.03 AMI and container image for 2
 years after the final LTS build is announced. You can also use all your existing
 support channels such as AWS Support and Amazon Linux Discussion Forum to continue
 to submit support requests."
    }
}
```

Per ulteriori informazioni, consulta Creazione di un repository pubblico nella Guida per l'utente di Amazon ECR Public e Dati di catalogo di un repository nella Guida per l'utente di Amazon ECR Public.

- Per i dettagli sull'API, vedere in Command Reference. CreateRepositoryAWS CLI

`delete-repository-policy`

Il seguente esempio di codice mostra come utilizzare`delete-repository-policy`.

AWS CLI

Come eliminare una policy di repository in un registro pubblico

L'`delete-repository-policy`esempio seguente elimina la politica di archiviazione per l'archivio ECR Public nel tuo account. AWS

```
aws ecr-public delete-repository-policy \
     --repository-name project-a/nginx-web-app \
     --region us-east-1
```

Output:

```
{
    "registryId": "123456789012",
    "repositoryName": "project-a/nginx-web-app",
    "policyText": "{\n  \"Version\" : \"2008-10-17\",\n  \"Statement\" : [ {\n
    \"Sid\" : \"AllowPush\",\n    \"Effect\" : \"Allow\",\n    \"Principal
\" : {\n      \"AWS\" : [ \"arn:aws:iam:"123456789012":user/eksuser1\",
 \"arn:aws:iam:"123456789012":user/admin\" ]\n    },\n    \"Action\" :
 [ \"ecr-public:BatchCheckLayerAvailability\", \"ecr-public:PutImage\",
 \"ecr-public:InitiateLayerUpload\", \"ecr-public:UploadLayerPart\", \"ecr-
public:CompleteLayerUpload\" ]\n  } ]\n}"
}
```

Per ulteriori informazioni, consulta Eliminazione di una dichiarazione di policy per i repository nella Guida per l'utente di Amazon ECR Public.

- Per i dettagli sull'API, consulta Command DeleteRepositoryPolicyReference AWS CLI .

`delete-repository`

Il seguente esempio di codice mostra come utilizzare`delete-repository`.

AWS CLI

Come eliminare un repository in un registro pubblico

L'esempio `delete-repository` seguente elimina un repository denominato `project-a/nginx-web-app` dal registro pubblico.

```
aws ecr-public delete-repository \
    --repository-name project-a/nginx-web-app
```

Output:

```
{
    "repository": {
        "repositoryArn": "arn:aws:ecr-public::123456789012:repository/project-a/
nginx-web-app",
        "registryId": "123456789012",
        "repositoryName": "project-a/nginx-web-app",
        "repositoryUri": "public.ecr.aws/public-registry-custom-alias/project-a/
nginx-web-app",
        "createdAt": "2024-07-01T22:14:50.103000+00:00"
    }
}
```

Per ulteriori informazioni, consulta Eliminazione di un repository pubblico nella Guida per l'utente di Amazon ECR Public.

- Per i dettagli sull'API, consulta DeleteRepository AWS CLICommand Reference.

`describe-image-tags`

Il seguente esempio di codice mostra come utilizzare`describe-image-tags`.

AWS CLI

Esempio 1: come descrivere i tag delle immagini in un repository pubblico

L'esempio `describe-image-tags` seguente descrive gli oggetti imagetags nel repository di esempio `project-a/nginx-web-app`.

```
aws ecr-public describe-image-tags \
    --repository-name project-a/nginx-web-app \
    --region us-east-1
```

Output:

```
{
    "imageTagDetails": [
        {
            "imageTag": "latest",
            "createdAt": "2024-07-10T22:29:00-05:00",
            "imageDetail": {
                "imageDigest":
 "sha256:b1f9deb5fe3711a3278379ebbcaefbc5d70a2263135db86bd27a0dae150546c2",
                "imageSizeInBytes": 121956548,
                "imagePushedAt": "2024-07-10T22:29:00-05:00",
                "imageManifestMediaType": "application/
vnd.docker.distribution.manifest.v2+json",
                "artifactMediaType": "application/
vnd.docker.container.image.v1+json"
            }
        }
    ]
}
```

- Per i dettagli sull'API, consulta DescribeImageTags AWS CLICommand Reference.

`describe-images`

Il seguente esempio di codice mostra come utilizzare`describe-images`.

AWS CLI

Esempio 1: come descrivere le immagini in un repository del registro pubblico

L'esempio `describe-images` seguente descrive gli oggetti imagesDetails in un archivio denominato `project-a/nginx-web-app` in un registro pubblico.

```
aws ecr-public describe-images \
    --repository-name project-a/nginx-web-app \
    --region us-east-1
```

Output:

```
{
    "imageDetails": [
        {
            "registryId": "123456789012",
```

```
            "repositoryName": "project-a/nginx-web-app",
            "imageDigest":
 "sha256:0d8c93e72e82fa070d49565c00af32abbe8ddfd7f75e39f4306771ae0628c7e8",
            "imageTags": [
                "temp1.0"
            ],
            "imageSizeInBytes": 123184716,
            "imagePushedAt": "2024-07-23T11:32:49-05:00",
            "imageManifestMediaType": "application/
vnd.docker.distribution.manifest.v2+json",
            "artifactMediaType": "application/vnd.docker.container.image.v1+json"
        },
        {
            "registryId": "123456789012",
            "repositoryName": "project-a/nginx-web-app",
            "imageDigest":
 "sha256:b1f9deb5fe3711a3278379ebbcaefbc5d70a2263135db86bd27a0dae150546c2",
            "imageTags": [
                "temp2.0"
            ],
            "imageSizeInBytes": 121956548,
            "imagePushedAt": "2024-07-23T11:39:38-05:00",
            "imageManifestMediaType": "application/
vnd.docker.distribution.manifest.v2+json",
            "artifactMediaType": "application/vnd.docker.container.image.v1+json"
        },
        {
            "registryId": "123456789012",
            "repositoryName": "project-a/nginx-web-app",
            "imageDigest":
 "sha256:f7a86a0760e2f8d7eff07e515fc87bf4bac45c35376c06f9a280f15ecad6d7e0",
            "imageTags": [
                "temp3.0",
                "latest"
            ],
            "imageSizeInBytes": 232108879,
            "imagePushedAt": "2024-07-22T00:54:34-05:00",
            "imageManifestMediaType": "application/
vnd.docker.distribution.manifest.v2+json",
            "artifactMediaType": "application/vnd.docker.container.image.v1+json"
        }
    ]
}
```

Per ulteriori informazioni, consulta Descrivere un'immagine in un repository pubblico nella Guida per l'utente di Amazon ECR Public.

Esempio 2: Per descrivere le immagini dal repository ordinando ImageTags & imagePushedAt

L'`describe-images`esempio seguente descrive le immagini all'interno del repository denominato nginx-web-app project-a/ in un registro pubblico.

```
aws ecr-public describe-images \
    --repository-name project-a/nginx-web-app \
    --query 'sort_by(imageDetails,& imagePushedAt)[*].imageTags[*]' \
    --output text
```

Output:

```
temp3.0 latest
temp1.0
temp2.0
```

Esempio 3: come descrivere le immagini dal repository per generare gli ultimi 2 tag di immagine inseriti nel repository

L'esempio `describe-images` seguente ottiene i dettagli degli oggetti imagetags dal repository denominato `project-a/nginx-web-app` in un registro pubblico ed esegue una query sul risultato per visualizzare solo i primi due record.

```
aws ecr-public describe-images \
    --repository-name project-a/nginx-web-app  \
    --query 'sort_by(imageDetails,& imagePushedAt)[*].imageTags[*] | [0:2]' \
    --output text
```

Output:

```
temp3.0 latest
temp1.0
```

- Per i dettagli sull'API, vedere DescribeImagesin Command Reference.AWS CLI

`describe-registries`

Il seguente esempio di codice mostra come utilizzare`describe-registries`.

AWS CLI

Come descrivere tutti i registri di un registro pubblico

L'esempio `describe-registries` seguente descrive tutti i registri nell'account.

```
aws ecr-public describe-registries
```

Output:

```
{
 "registries": [
     {
         "registryId": "123456789012",
         "registryArn": "arn:aws:ecr-public::123456789012:registry/123456789012",
         "registryUri": "public.ecr.aws/publicregistrycustomalias",
         "verified": false,
         "aliases": [
             {
                 "name": "publicregistrycustomalias",
                 "status": "ACTIVE",
                 "primaryRegistryAlias": true,
                 "defaultRegistryAlias": true
             }
         ]
     }
     ]
 }
```

- Per i dettagli sull'API, consulta DescribeRegistries AWS CLICommand Reference.

`describe-repository`

Il seguente esempio di codice mostra come utilizzare`describe-repository`.

AWS CLI

Esempio 1: come descrivere un repository in un registro pubblico

L'esempio `describe-repositories` seguente descrive un archivio denominato `project-a/nginx-web-app` in un registro pubblico.

```
aws ecr-public describe-repositories \
    --repository-name project-a/nginx-web-app
```

Output:

```
{
    "repositories": [
        {
            "repositoryArn": "arn:aws:ecr-public::123456789012:repository/project-a/
nginx-web-app",
            "registryId": "123456789012",
            "repositoryName": "project-a/nginx-web-app",
            "repositoryUri": "public.ecr.aws/public-registry-custom-alias/project-a/
nginx-web-app",
            "createdAt": "2024-07-07T00:07:56.526000-05:00"
        }
    ]
}
```

Esempio 2: come descrivere tutti i repository in un registro pubblico in una tabella

L'esempio `describe-repositories` seguente descrive tutti gli archivi in un registro pubblico e quindi restituisce i nomi dei repository in un formato tabellare.

```
aws ecr-public describe-repositories \
    --region us-east-1 \
    --output table \
    --query "repositories[*].repositoryName"
```

Output:

```
----------------------------
|   DescribeRepositories    |
+---------------------------+
|  project-a/nginx-web-app  |
|  nginx                    |
|  myfirstrepo1             |
|  helm-test-chart          |
|  test-ecr-public          |
|  nginx-web-app            |
|  sample-repo              |
```

```
+---------------------------+
```

- Per i dettagli sull'API, consulta DescribeRepository AWS CLICommand Reference.

`get-authorization-token`

Il seguente esempio di codice mostra come utilizzare`get-authorization-token`.

AWS CLI

Esempio 1: come recuperare un token di autorizzazione per qualsiasi registro pubblico Amazon ECR a cui il principale IAM ha accesso

L'`get-authorization-token`esempio seguente ottiene un token di autorizzazione con la AWS CLI e lo imposta su una variabile di ambiente.

```
aws ecr-public get-authorization-token \
    --region us-east-1
```

Output:

```
{
    "authorizationData": {
        "authorizationToken":
 "QVdTOmV5SndZWGxzYjJKJFHDSFKJHERWUY65IOU36TRYEGFNSDLRIUOTUYTHJKLDFGOcmFUQk9OSFV2UVV4a0x6Sm1
        "expiresAt": "2024-07-25T21:37:26.301000-04:00"
    }
}
```

Per ulteriori informazioni, consulta Registri Amazon ECR Public nella Guida per l'utente di Amazon ECR Public.

Esempio 2: come recuperare un token di autorizzazione per qualsiasi registro pubblico Amazon ECR a cui il principale IAM ha accesso

L'`get-authorization-token`esempio seguente ottiene un token di autorizzazione con la AWS CLI e lo imposta su una variabile di ambiente.

```
aws ecr-public get-authorization-token \
    --region us-east-1 \
```

```
    --output=text \
    --query 'authorizationData.authorizationToken'
```

Output:

```
QVdTOmV5SndZWGxzYjJKJFHDSFKJHERWUY65IOU36TRYEGFNSDLRIUOTUYTHJKLDFGOcmFUQk9OSFV2UVV4a0x6Sm1ZV
```

Per ulteriori informazioni, consulta Registri Amazon ECR Public nella Guida per l'utente di Amazon ECR Public.

- Per i dettagli sull'API, consulta GetAuthorizationToken AWS CLICommand Reference.

`get-login-password`

Il seguente esempio di codice mostra come utilizzare`get-login-password`.

AWS CLI

Esempio 1: come autenticare il docker in un registro pubblico Amazon ECR

L'`get-login-password`esempio seguente recupera e visualizza un token di autenticazione utilizzando l' GetAuthorizationToken API che puoi utilizzare per autenticarti in un registro pubblico Amazon ECR.

```
aws ecr-public get-login-password \
    --region us-
east-1
| docker login \
    --username AWS \
    --password-stdin public.ecr.aws
```

Questo comando non produce alcun output nel terminale, ma invia invece l'output a Docker.

Per ulteriori informazioni, consulta Autenticazione sul registro pubblico nella Guida per l'utente di Amazon ECR Public.

Esempio 2: come autenticare il docker nel registro pubblico Amazon ECR personalizzato

L'`get-login-password`esempio seguente recupera e visualizza un token di autenticazione utilizzando l' GetAuthorizationToken API che puoi utilizzare per autenticarti nel tuo registro pubblico Amazon ECR personalizzato.

```
 aws ecr-public get-login-password \
    --region us-east-1 \
| docker login \
    --username AWS \
    --password-stdin public.ecr.aws/<your-public-registry-custom-alias>
```

Questo comando non produce alcun output nel terminale, ma invia invece l'output a Docker.

Per ulteriori informazioni, consulta Autenticazione sul proprio Amazon ECR Public nella Guida per l'utente di Amazon ECR Public.

- Per i dettagli sull'API, consulta Command GetLoginPasswordReference AWS CLI .

`get-registry-catalog-data`

Il seguente esempio di codice mostra come utilizzare`get-registry-catalog-data`.

AWS CLI

Come recuperare i metadati del catalogo per un registro ECR pubblico

L'esempio `get-registry-catalog-data` seguente recupera i metadati del catalogo per un registro ECR pubblico.

```
aws ecr-public get-registry-catalog-data \
    --region us-east-1
```

Output:

```
{
    "registryCatalogData": {
        "displayName": "YourCustomPublicRepositoryalias"
    }
}
```

- Per i dettagli sull'API, consulta GetRegistryCatalogData AWS CLICommand Reference.

`get-repository-catalog-data`

Il seguente esempio di codice mostra come utilizzare`get-repository-catalog-data`.

AWS CLI

Come recuperare i metadati del catalogo per un registro pubblico

L'esempio `get-repository-catalog-data` seguente elenca i metadati del catalogo per il repository `project-a/nginx-web-app` in un registro pubblico.

```
aws ecr-public get-repository-catalog-data \
    --repository-name project-a/nginx-web-app \
    --region us-east-1
```

Output:

```
{
    "catalogData": {
        "description": "My project-a ECR Public Repository",
        "architectures": [
            "ARM",
            "ARM 64",
            "x86",
            "x86-64"
        ],
        "operatingSystems": [
            "Linux"
        ],
        "logoUrl": "https://d3g9o9u8re44ak.cloudfront.net/logo/491d3846-8f33-4d8b-
a10c-c2ce271e6c0d/4f09d87c-2569-4916-a932-5c296bf6f88a.png",
        "aboutText": "## Quick reference\n\nMaintained <truncated>",
        "usageText": "## Supported architectures\n\namd64, arm64v8\n\n##
 <truncated>"
    }
}
```

Per ulteriori informazioni, consulta Dati del catalogo del repository nella Guida per l'utente di Amazon ECR Public.

- Per i dettagli sull'API, consulta GetRepositoryCatalogData AWS CLICommand Reference.

`get-repository-policy`

Il seguente esempio di codice mostra come utilizzare`get-repository-policy`.

AWS CLI

Come ottenere una policy di repository associata al repository

L'esempio `get-repository-policy` seguente ottiene una policy di repository associata al repository.

```
aws ecr-public get-repository-policy \
    --repository-name project-a/nginx-web-app \
    --region us-east-1
```

Output:

```
{
    "registryId": "123456789012",
    "repositoryName": "project-a/nginx-web-app",
    "policyText": "{\n  \"Version\" : \"2008-10-17\",\n  \"Statement\" : [ {\n
    \"Sid\" : \"AllowPush\",\n    \"Effect\" : \"Allow\",\n    \"Principal
\" : {\n      \"AWS\" : [ \"arn:aws:iam::123456789012:user/eksuser1\",
 \"arn:aws:iam::123456789012:user/admin\" ]\n    },\n    \"Action\" :
 [ \"ecr-public:BatchCheckLayerAvailability\", \"ecr-public:PutImage\",
 \"ecr-public:InitiateLayerUpload\", \"ecr-public:UploadLayerPart\", \"ecr-
public:CompleteLayerUpload\" ]\n  } ]\n}"
}
```

Per ulteriori informazioni, consulta Use GetRepositoryPolicy with an AWS SDK o CLI nella Amazon ECR Public User Guide.

- Per i dettagli sull'API, consulta Command GetRepositoryPolicyReference AWS CLI .

`list-tags-for-resource`

Il seguente esempio di codice mostra come utilizzare`list-tags-for-resource`.

AWS CLI

Come elencare i tag di un repository pubblico in un registro pubblico

L'esempio `list-tags-for-resource` seguente elenca i tag per una risorsa denominata `project-a/nginx-web-app` in un registro pubblico.

```
aws ecr-public list-tags-for-resource \
```

```
    --resource-arn arn:aws:ecr-public::123456789012:repository/project-a/nginx-web-
app \
    --region us-east-1
```

Output:

```
{
    "tags": [
        {
            "Key": "Environment",
            "Value": "Prod"
        },
        {
            "Key": "stack",
            "Value": "dev1"
        },
        {
            "Key": "Name",
            "Value": "project-a/nginx-web-app"
        }
    ]
}
```

Per ulteriori informazioni, consulta Elenco di tag per un repository pubblico nella Guida per l'utente di Amazon ECR Public.

- Per i dettagli sull'API, consulta ListTagsForResource AWS CLICommand Reference.

`put-registry-catalog-data`

Il seguente esempio di codice mostra come utilizzare`put-registry-catalog-data`.

AWS CLI

Come creare o aggiornare i metadati del catalogo per un registro ECR pubblico

L'esempio `put-registry-catalog-data` seguente crea o aggiorna i metadati del catalogo per un registro ECR pubblico. Solo gli account che dispongono del badge dell'account verificato possono avere un nome visualizzato nel registro.

```
aws ecr-public put-registry-catalog-data \
    --region us-east-1 \
```

```
    --display-name <YourCustomPublicRepositoryalias>
```

Output:

```
{
    "registryCatalogData": {
        "displayName": "YourCustomPublicRepositoryalias"
    }
}
```

- Per i dettagli sull'API, consulta PutRegistryCatalogData AWS CLICommand Reference.

`put-repository-catalog-data`

Il seguente esempio di codice mostra come utilizzare`put-repository-catalog-data`.

AWS CLI

Come creare o aggiornare i dati del catalogo per un repository in un registro pubblico

L'`put-repository-catalog-data`esempio seguente crea o aggiorna i dati del catalogo per il repository denominato project-a/ nginx-web-app in un registro pubblico, insieme a aboutText, UsageText e alle informazioni sui logoImageBlob tag.

```
aws ecr-public put-repository-catalog-data \
    --repository-name project-a/nginx-web-app \
    --cli-input-json file://repository-catalog-data.json \
    --region us-east-1
```

Contenuto di `repository-catalog-data.json`:

```
{
    "repositoryName": "project-a/nginx-web-app",
    "catalogData": {
        "description": "My project-a ECR Public Repository",
        "architectures": [
            "ARM",
            "ARM 64",
            "x86",
            "x86-64"
        ],
```

```
        "operatingSystems": [
            "Linux"
        ],
        "logoImageBlob": "iVBORw0KGgoA<<truncated-for-better-reading>>ErkJggg==",
        "aboutText": "## Quick reference.",
        "usageText": "## Supported architectures are as follows."
    }
}
```

Output:

```
{
    "catalogData": {
        "description": "My project-a ECR Public Repository",
        "architectures": [
            "ARM",
            "ARM 64",
            "x86",
            "x86-64"
        ],
        "operatingSystems": [
            "Linux"
        ],
        "logoUrl": "https://d3g9o9u8re44ak.cloudfront.net/logo/df86cf58-ee60-4061-
b804-0be24d97ccb1/4a9ed9b2-69e4-4ede-b924-461462d20ef0.png",
        "aboutText": "## Quick reference.",
        "usageText": "## Supported architectures are as follows."
    }
}
```

Per ulteriori informazioni, consulta Dati del catalogo del repository nella Guida per l'utente di Amazon ECR Public.

- Per PutRepositoryCatalogDatai AWS CLI dettagli sull'API, vedere in Command Reference.

`set-repository-policy`

Il seguente esempio di codice mostra come utilizzare`set-repository-policy`.

AWS CLI

Esempio 1: come impostare una policy di repository per consentire un pull sul repository

L'esempio `set-repository-policy` seguente applica una policy di repository ECR Public al repository specificato per controllare le autorizzazioni di accesso.

```
aws ecr-public set-repository-policy \
    --repository-name project-a/nginx-web-app \
    --policy-text file://my-repository-policy.json
```

Contenuto di `my-repository-policy.json`:

```
{
    "Version":"2012-10-17",
    "Statement" : [
        {
            "Sid" : "allow public pull",
            "Effect" : "Allow",
            "Principal" : "*",
            "Action" : [
                "ecr:BatchCheckLayerAvailability",
                "ecr:BatchGetImage",
                "ecr:GetDownloadUrlForLayer"
            ]
        }
    ]
}
```

Output:

```
{
    "registryId": "12345678901",
    "repositoryName": "project-a/nginx-web-app",
    "policyText": "{\n  \"Version\" : \"2008-10-17\",\n  \"Statement\" : [ {\n
 \"Sid\" : \"allow public pull\",\n    \"Effect\" : \"Allow\",\n    \"Principal\" :
 \"*\",\n    \"Action\" : [ \"ecr:BatchCheckLayerAvailability\", \"ecr:BatchGetImage
\", \"ecr:GetDownloadUrlForLayer\" ]\n  } ]\n}"
}
```

Per ulteriori informazioni, consulta Impostazione di un'istruzione di una policy di un repository nella Guida per l'utente di Amazon ECR Public.

Esempio 2: come impostare una policy di repository per consentire a un utente IAM all'interno dell'account corrente di eseguire il push delle immagini

L'`set-repository-policy`esempio seguente consente a un utente IAM del tuo account di inviare le immagini utilizzate a un repository ECR del tuo AWS account utilizzando il file di input `file://my-repository-policy.json` denominato testo della policy.

```
aws ecr-public set-repository-policy \
    --repository-name project-a/nginx-web-app \
    --policy-text file://my-repository-policy.json
```

Contenuto di `my-repository-policy.json`:

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "AllowPush",
            "Effect": "Allow",
            "Principal": {
                "AWS": [
                    "arn:aws:iam::account-id:user/push-pull-user-1",
                    "arn:aws:iam::account-id:user/push-pull-user-2"
                ]
            },
            "Action": [
                "ecr-public:BatchCheckLayerAvailability",
                "ecr-public:PutImage",
                "ecr-public:InitiateLayerUpload",
                "ecr-public:UploadLayerPart",
                "ecr-public:CompleteLayerUpload"
            ]
        }
    ]
}
```

Output:

```
{
    "registryId": "12345678901",
    "repositoryName": "project-a/nginx-web-app",
    "policyText": "{\n  \"Version\" : \"2008-10-17\",\n  \"Statement\" :
 [ {\n    \"Sid\" : \"AllowPush\",\n    \"Effect\" : \"Allow\",\n
 \"Principal\" : {\n      \"AWS\" : [ \"arn:aws:iam::12345678901:user/admin
\", \"arn:aws:iam::12345678901:user/eksuser1\" ]\n    },\n    \"Action\" :
```

```
 [ \"ecr-public:BatchCheckLayerAvailability\", \"ecr-public:PutImage\",
 \"ecr-public:InitiateLayerUpload\", \"ecr-public:UploadLayerPart\", \"ecr-
public:CompleteLayerUpload\" ]\n  } ]\n}"
}
```

Per ulteriori informazioni, consulta Impostazione di un'istruzione di una policy di un repository nella Guida per l'utente di Amazon ECR Public.

Esempio 3: come impostare una policy di repository per consentire a un utente IAM all'interno di un account diverso di eseguire il push delle immagini

L'`set-repository-policy`esempio seguente consente a un account specifico di inviare immagini utilizzando il file di input cli://my-repository-policy.json nel tuo account. AWS

```
aws ecr-public set-repository-policy \
    --repository-name project-a/nginx-web-app \
    --policy-text file://my-repository-policy.json
```

Contenuto di `my-repository-policy.json`:

```
 {
     "Version":"2012-10-17",
     "Statement": [
         {
             "Sid": "AllowCrossAccountPush",
             "Effect": "Allow",
             "Principal": {
                 "AWS": "arn:aws:iam::other-or-same-account-id:role/RoleName"
             },
             "Action": [
                 "ecr-public:BatchCheckLayerAvailability",
                 "ecr-public:PutImage",
                 "ecr-public:InitiateLayerUpload",
                 "ecr-public:UploadLayerPart",
                 "ecr-public:CompleteLayerUpload"
             ]
         }
     ]
}
```

Output:

```
{
    "registryId": "12345678901",
    "repositoryName": "project-a/nginx-web-app",
    "policyText": "{\n  \"Version\" : \"2008-10-17\",\n  \"Statement\" : [ {\n
 \"Sid\" : \"AllowCrossAccountPush\",\n    \"Effect\" : \"Allow\",\n    \"Principal
\" : {\n      \"AWS\" : \"arn:aws:iam::12345678901:role/RoleName\"\n    },\n
 \"Action\" : [ \"ecr-public:BatchCheckLayerAvailability\", \"ecr-public:PutImage
\", \"ecr-public:InitiateLayerUpload\", \"ecr-public:UploadLayerPart\", \"ecr-
public:CompleteLayerUpload\" ]\n  } ]\n}"
}
```

Per ulteriori informazioni, consulta Esempi di policy di repository pubblici nella Guida per l'utente di Amazon ECR Public.

- Per i dettagli sull'API, consulta Command SetRepositoryPolicyReference AWS CLI .

`tag-resource`

Il seguente esempio di codice mostra come utilizzare`tag-resource`.

AWS CLI

Esempio 1: come taggare un repository pubblico esistente in un registro pubblico

L'esempio `tag-resource` seguente tagga un repository denominato `project-a/nginx-web-app` in un registro pubblico.

```
aws ecr-public tag-resource \
    --resource-arn arn:aws:ecr-public::123456789012:repository/project-a/nginx-web-
app \
    --tags Key=stack,Value=dev \
    --region us-east-1
```

Per ulteriori informazioni, consulta Utilizzo di tag per un repository pubblico nella Guida per l'utente di Amazon ECR Public.

Esempio 2: come taggare un repository pubblico esistente con più tag in un registro pubblico.

L'esempio `tag-resource` seguente tagga un repository esistente con più tag.

```
aws ecr-public tag-resource \
```

```
    --resource-arn arn:aws:ecr-public::890517186334:repository/project-a/nginx-web-
app  \
    --tags Key=key1,Value=value1 Key=key2,Value=value2 Key=key3,Value=value3 \
    --region us-east-1
```

Per ulteriori informazioni, consulta Utilizzo di tag per un repository pubblico nella Guida per l'utente di Amazon ECR Public.

- Per i dettagli sull'API, consulta TagResource AWS CLICommand Reference.

`untag-resource`

Il seguente esempio di codice mostra come utilizzare`untag-resource`.

AWS CLI

Esempio 1: come rimuovere i tag da un repository pubblico esistente in un registro pubblico

L'esempio `untag-resource` seguente tagga un repository denominato `project-a/nginx-web-app` in un registro pubblico.

```
aws ecr-public untag-resource \
    --resource-arn arn:aws:ecr-public::123456789012:repository/project-a/nginx-web-
app \
    --tag-keys stack \
    --region us-east-1
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Utilizzo di tag per un repository pubblico nella Guida per l'utente di Amazon ECR Public.

- Per i dettagli sull'API, consulta UntagResource AWS CLICommand Reference.

Esempi di utilizzo di Amazon ECS AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando Amazon ECS. AWS Command Line Interface

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`capacity-provider-update`

Il seguente esempio di codice mostra come usare`capacity-provider-update`.

AWS CLI

Come aggiornare il provider di capacità in un cluster ECS

L'esempio `update-capacity-provider` seguente modifica i parametri del provider di capacità in un cluster ECS.

```
aws ecs update-capacity-provider \
    --name Infra-ECS-Cluster-ECS-project-update-cluster-d6bb6d5b-
EC2CapacityProvider-3fIpdkLywwFt \
    --auto-scaling-group-
provider "managedScaling={status=DISABLED,targetCapacity=50,minimumScalingStepSize=2,maximum
```

Output:

```
{
    "capacityProvider": {
        "capacityProviderArn": "arn:aws:ecs:us-west-2:123456789012:capacity-
provider/Infra-ECS-Cluster-ECS-project-update-cluster-d6bb6d5b-
EC2CapacityProvider-3fIpdkLywwFt",
        "name": "Infra-ECS-Cluster-ECS-project-update-cluster-d6bb6d5b-
EC2CapacityProvider-3fIpdkLywwFt",
        "status": "ACTIVE",
        "autoScalingGroupProvider": {
            "autoScalingGroupArn": "arn:aws:autoscaling:us-
west-2:123456789012:autoScalingGroup:424941d1-b43f-4a17-
adbb-08b6a6e397e1:autoScalingGroupName/Infra-ECS-Cluster-ECS-project-update-cluster-
d6bb6d5b-ECSAutoScalingGroup-f44jrQHS2nRB",
            "managedScaling": {
                "status": "ENABLED",
```

```
                "targetCapacity": 100,
                "minimumScalingStepSize": 1,
                "maximumScalingStepSize": 10000,
                "instanceWarmupPeriod": 300
            },
            "managedTerminationProtection": "DISABLED",
            "managedDraining": "ENABLED"
        },
        "updateStatus": "UPDATE_IN_PROGRESS",
        "tags": []
    }
}
```

Per ulteriori informazioni su Capacity Provider, consulta i provider di capacità di Amazon ECS per il tipo di EC2 lancio nella Amazon ECS Developer Guide.

- Per i dettagli sulle API, consulta AWS CLI Command CapacityProviderUpdateReference.

`create-capacity-provider`

Il seguente esempio di codice mostra come utilizzare`create-capacity-provider`.

AWS CLI

Come creare un provider di capacità

L' create-capacity-provideresempio seguente crea un provider di capacità che utilizza un gruppo Auto Scaling denominato MyASG, ha la scalabilità gestita e la protezione gestita dalla terminazione abilitata. Questa configurazione viene utilizzata per il dimensionamento automatico del cluster Amazon ECS.

```
aws ecs create-capacity-provider \
    --name "MyCapacityProvider" \
    --auto-scaling-group-provider "autoScalingGroupArn=arn:aws:autoscaling:us-
east-1:123456789012:autoScalingGroup:57ffcb94-11f0-4d6d-
bf60-3bac5EXAMPLE:autoScalingGroupName/
MyASG,managedScaling={status=ENABLED,targetCapacity=100},managedTerminationProtection=ENABLE
```

Output:

```
{
    "capacityProvider": {
```

```
    "capacityProviderArn": "arn:aws:ecs:us-east-1:123456789012:capacity-provider/
MyCapacityProvider",
    "name": "MyCapacityProvider",
    "status": "ACTIVE",
    "autoScalingGroupProvider": {
        "autoScalingGroupArn": "arn:aws:autoscaling:us-
east-1:132456789012:autoScalingGroup:57ffcb94-11f0-4d6d-
bf60-3bac5EXAMPLE:autoScalingGroupName/MyASG",
        "managedScaling": {
            "status": "ENABLED",
            "targetCapacity": 100,
            "minimumScalingStepSize": 1,
            "maximumScalingStepSize": 10000,
            "instanceWarmupPeriod": 300
        },
        "managedTerminationProtection": "ENABLED"
    },
    "tags": []
}
```

Per ulteriori informazioni, consulta Dimensionamento automatico del cluster Amazon ECS nella Guida per gli sviluppatori di Amazon ECS.

- Per i dettagli sull'API, consulta Command Reference. CreateCapacityProviderAWS CLI

`create-cluster`

Il seguente esempio di codice mostra come utilizzare`create-cluster`.

AWS CLI

Esempio 1: come creare un nuovo cluster

L'`create-cluster`esempio seguente crea un cluster denominato `MyCluster` e abilita CloudWatch Container Insights con una migliore osservabilità.

```
aws ecs create-cluster \
    --cluster-name MyCluster \
    --settings name=containerInsights,value=enhanced
```

Output:

```
{
```

```
    "cluster": {
        "clusterArn": "arn:aws:ecs:us-west-2:123456789012:cluster/MyCluster",
        "clusterName": "MyCluster",
        "status": "ACTIVE",
        "registeredContainerInstancesCount": 0,
        "pendingTasksCount": 0,
        "runningTasksCount": 0,
        "activeServicesCount": 0,
        "statistics": [],
        "settings": [
            {
                "name": "containerInsights",
                "value": "enhanced"
            }
        ],
        "tags": []
    }
}
```

Per ulteriori informazioni, consulta Creazione di un cluster nella Guida per gli sviluppatori di Amazon ECS.

Esempio 2: come creare un nuovo cluster utilizzando provider di capacità

L'esempio `create-cluster` seguente crea un cluster a cui associa due provider di capacità esistenti. Il comando `create-capacity-provider` viene utilizzato per creare un provider di capacità. La definizione di una strategia predefinita per i provider di capacità è facoltativa, ma consigliata. In questo esempio, viene creato un cluster denominato `MyCluster` a cui vengono associati i provider di capacità `MyCapacityProvider1` e `MyCapacityProvider2`. Viene specificata una strategia predefinita dei provider di capacità che distribuisce le attività in modo uniforme su entrambi i provider di capacità.

```
aws ecs create-cluster \
    --cluster-name MyCluster \
    --capacity-providers MyCapacityProvider1 MyCapacityProvider2 \
    --default-capacity-provider-
strategy capacityProvider=MyCapacityProvider1,weight=1 capacityProvider=MyCapacityProvider2,
```

Output:

```
{
    "cluster": {
```

```
        "clusterArn": "arn:aws:ecs:us-west-2:123456789012:cluster/MyCluster",
        "clusterName": "MyCluster",
        "status": "PROVISIONING",
        "registeredContainerInstancesCount": 0,
        "pendingTasksCount": 0,
        "runningTasksCount": 0,
        "activeServicesCount": 0,
        "statistics": [],
        "settings": [
            {
                "name": "containerInsights",
                "value": "enabled"
            }
        ],
        "capacityProviders": [
            "MyCapacityProvider1",
            "MyCapacityProvider2"
        ],
        "defaultCapacityProviderStrategy": [
            {
                "capacityProvider": "MyCapacityProvider1",
                "weight": 1,
                "base": 0
            },
            {
                "capacityProvider": "MyCapacityProvider2",
                "weight": 1,
                "base": 0
            }
        ],
        "attachments": [
           {
               "id": "0fb0c8f4-6edd-4de1-9b09-17e470ee1918",
               "type": "asp",
               "status": "PRECREATED",
               "details": [
                   {
                       "name": "capacityProviderName",
                       "value": "MyCapacityProvider1"
                   },
                   {
                       "name": "scalingPlanName",
                       "value": "ECSManagedAutoScalingPlan-a1b2c3d4-5678-90ab-cdef-
EXAMPLE11111"
```

```
                    }
                ]
            },
            {
                "id": "ae592060-2382-4663-9476-b015c685593c",
                "type": "asp",
                "status": "PRECREATED",
                "details": [
                    {
                        "name": "capacityProviderName",
                        "value": "MyCapacityProvider2"
                    },
                    {
                        "name": "scalingPlanName",
                        "value": "ECSManagedAutoScalingPlan-a1b2c3d4-5678-90ab-cdef-
EXAMPLE22222"
                    }
                ]
            }
        ],
        "attachmentsStatus": "UPDATE_IN_PROGRESS"
    }
}
```

Per ulteriori informazioni, consulta Provider di capacità del cluster nella Guida per gli sviluppatori di Amazon ECS.

Esempio 3: come creare un nuovo cluster con più tag

L'esempio `create-cluster` seguente crea un cluster con più tag. Per ulteriori informazioni sull'aggiunta di tag utilizzando la sintassi abbreviata, consulta Uso della sintassi abbreviata con l'interfaccia a AWS riga di comando nella Guida per l'utente della CLI.AWS

```
aws ecs create-cluster \
    --cluster-name MyCluster \
    --tags key=key1,value=value1 key=key2,value=value2
```

Output:

```
{
    "cluster": {
        "clusterArn": "arn:aws:ecs:us-west-2:123456789012:cluster/MyCluster",
```

```
        "clusterName": "MyCluster",
        "status": "ACTIVE",
        "registeredContainerInstancesCount": 0,
        "pendingTasksCount": 0,
        "runningTasksCount": 0,
        "activeServicesCount": 0,
        "statistics": [],
        "tags": [
            {
                "key": "key1",
                "value": "value1"
            },
            {
                "key": "key2",
                "value": "value2"
            }
        ]
    }
}
```

Per ulteriori informazioni, consulta Creazione di un cluster nella Guida per gli sviluppatori di Amazon ECS.

- Per i dettagli sull'API, consulta Command Reference. CreateClusterAWS CLI

`create-service`

Il seguente esempio di codice mostra come utilizzare`create-service`.

AWS CLI

Esempio 1: come creare un servizio con un'attività Fargate

L'esempio `create-service` seguente crea un servizio utilizzando un'attività Fargate.

```
aws ecs create-service \
    --cluster MyCluster \
    --service-name MyService \
    --task-definition sample-fargate:1 \
    --desired-count 2 \
    --launch-type FARGATE \
    --platform-version LATEST \
```

```
    --network-configuration
 'awsvpcConfiguration={subnets=[subnet-12344321],securityGroups=[sg-12344321],assignPublicIp
 \
    --tags key=key1,value=value1 key=key2,value=value2 key=key3,value=value3
```

Output:

```
{
    "service": {
        "serviceArn": "arn:aws:ecs:us-west-2:123456789012:service/MyCluster/
MyService",
        "serviceName": "MyService",
          "clusterArn": "arn:aws:ecs:us-west-2:123456789012:cluster/MyCluster",
        "loadBalancers": [],
        "serviceRegistries": [],
        "status": "ACTIVE",
        "desiredCount": 2,
        "runningCount": 0,
        "pendingCount": 0,
        "launchType": "FARGATE",
        "platformVersion": "LATEST",
        "taskDefinition": "arn:aws:ecs:us-west-2:123456789012:task-definition/
sample-fargate:1",
        "deploymentConfiguration": {
            "maximumPercent": 200,
            "minimumHealthyPercent": 100
        },
        "deployments": [
            {
                "id": "ecs-svc/1234567890123456789",
                "status": "PRIMARY",
                "taskDefinition": "arn:aws:ecs:us-west-2:123456789012:task-
definition/sample-fargate:1",
                "desiredCount": 2,
                "pendingCount": 0,
                "runningCount": 0,
                "createdAt": 1557119253.821,
                "updatedAt": 1557119253.821,
                "launchType": "FARGATE",
                "platformVersion": "1.3.0",
                "networkConfiguration": {
                    "awsvpcConfiguration": {
                        "subnets": [
```

```
                            "subnet-12344321"
                        ],
                        "securityGroups": [
                            "sg-12344321"
                        ],
                        "assignPublicIp": "ENABLED"
                    }
                }
            }
        ],
        "roleArn": "arn:aws:iam::123456789012:role/aws-service-role/
ecs.amazonaws.com/AWSServiceRoleForECS",
        "events": [],
        "createdAt": 1557119253.821,
        "placementConstraints": [],
        "placementStrategy": [],
        "networkConfiguration": {
            "awsvpcConfiguration": {
                "subnets": [
                    "subnet-12344321"
                ],
                "securityGroups": [
                    "sg-12344321"
                ],
                "assignPublicIp": "ENABLED"
            }
        },
        "schedulingStrategy": "REPLICA",
        "tags": [
            {
                "key": "key1",
                "value": "value1"
            },
            {
                "key": "key2",
                "value": "value2"
            },
            {
                "key": "key3",
                "value": "value3"
            }
        ],
        "enableECSManagedTags": false,
        "propagateTags": "NONE"
```

```
    }
}
```

Per ulteriori informazioni, consulta Creazione di un servizio nella Guida per gli sviluppatori di Amazon ECS.

Esempio 2: creare un servizio utilizzando il tipo di EC2 avvio

L'`create-service`esempio seguente mostra come creare un servizio chiamato `ecs-simple-service` con un'attività che utilizza il tipo di EC2 avvio. Il servizio utilizza la definizione dell'attività `sleep360` e conserva un'istanza dell'attività.

```
aws ecs create-service \
    --cluster MyCluster \
    --service-name ecs-simple-service \
    --task-definition sleep360:2 \
    --desired-count 1
```

Output:

```
{
    "service": {
        "serviceArn": "arn:aws:ecs:us-west-2:123456789012:service/MyCluster/ecs-
simple-service",
        "serviceName": "ecs-simple-service",
        "clusterArn": "arn:aws:ecs:us-west-2:123456789012:cluster/MyCluster",
        "loadBalancers": [],
        "serviceRegistries": [],
        "status": "ACTIVE",
        "desiredCount": 1,
        "runningCount": 0,
        "pendingCount": 0,
        "launchType": "EC2",
        "taskDefinition": "arn:aws:ecs:us-west-2:123456789012:task-definition/
sleep360:2",
        "deploymentConfiguration": {
            "maximumPercent": 200,
            "minimumHealthyPercent": 100
        },
        "deployments": [
            {
                "id": "ecs-svc/1234567890123456789",
```

```
                "status": "PRIMARY",
                "taskDefinition": "arn:aws:ecs:us-west-2:123456789012:task-
definition/sleep360:2",
                "desiredCount": 1,
                "pendingCount": 0,
                "runningCount": 0,
                "createdAt": 1557206498.798,
                "updatedAt": 1557206498.798,
                "launchType": "EC2"
            }
        ],
        "events": [],
        "createdAt": 1557206498.798,
        "placementConstraints": [],
        "placementStrategy": [],
        "schedulingStrategy": "REPLICA",
        "enableECSManagedTags": false,
        "propagateTags": "NONE"
    }
}
```

Per ulteriori informazioni, consulta Creazione di un servizio nella Guida per gli sviluppatori di Amazon ECS.

Esempio 3: come creare un servizio che utilizza un controller di implementazione esterno

L'esempio `create-service` seguente crea un servizio che utilizza un controller di implementazione esterno.

```
aws ecs create-service \
    --cluster MyCluster \
    --service-name MyService \
    --deployment-controller type=EXTERNAL \
    --desired-count 1
```

Output:

```
{
    "service": {
        "serviceArn": "arn:aws:ecs:us-west-2:123456789012:service/MyCluster/
MyService",
        "serviceName": "MyService",
        "clusterArn": "arn:aws:ecs:us-west-2:123456789012:cluster/MyCluster",
```

```
        "loadBalancers": [],
        "serviceRegistries": [],
        "status": "ACTIVE",
        "desiredCount": 1,
        "runningCount": 0,
        "pendingCount": 0,
        "launchType": "EC2",
        "deploymentConfiguration": {
            "maximumPercent": 200,
            "minimumHealthyPercent": 100
        },
        "taskSets": [],
        "deployments": [],
        "roleArn": "arn:aws:iam::123456789012:role/aws-service-role/
ecs.amazonaws.com/AWSServiceRoleForECS",
        "events": [],
        "createdAt": 1557128207.101,
        "placementConstraints": [],
        "placementStrategy": [],
        "schedulingStrategy": "REPLICA",
        "deploymentController": {
            "type": "EXTERNAL"
        },
        "enableECSManagedTags": false,
        "propagateTags": "NONE"
    }
}
```

Per ulteriori informazioni, consulta Creazione di un servizio nella Guida per gli sviluppatori di Amazon ECS.

Esempio 4: come creare un nuovo servizio dietro un bilanciatore del carico

L'esempio `create-service` seguente crea un servizio dietro un bilanciatore del carico. Un bilanciatore del carico deve essere configurato nella stessa Regione dell'istanza di container. Questo esempio utilizza l'opzione `--cli-input-json` e un file di input JSON denominato `ecs-simple-service-elb.json` con il seguente contenuto.

```
aws ecs create-service \
    --cluster MyCluster \
    --service-name ecs-simple-service-elb \
    --cli-input-json file://ecs-simple-service-elb.json
```

Contenuto di `ecs-simple-service-elb.json`:

```
 {
    "serviceName": "ecs-simple-service-elb",
    "taskDefinition": "ecs-demo",
    "loadBalancers": [
        {
            "loadBalancerName": "EC2Contai-EcsElast-123456789012",
            "containerName": "simple-demo",
            "containerPort": 80
        }
    ],
    "desiredCount": 10,
    "role": "ecsServiceRole"
}
```

Output:

```
{
    "service": {
        "status": "ACTIVE",
        "taskDefinition": "arn:aws:ecs:us-west-2:123456789012:task-definition/ecs-
demo:1",
        "pendingCount": 0,
        "loadBalancers": [
            {
                "containerName": "ecs-demo",
                "containerPort": 80,
                "loadBalancerName": "EC2Contai-EcsElast-123456789012"
            }
        ],
        "roleArn": "arn:aws:iam::123456789012:role/ecsServiceRole",
        "desiredCount": 10,
        "serviceName": "ecs-simple-service-elb",
        "clusterArn": "arn:aws:ecs:us-west-2:123456789012:cluster/MyCluster",
        "serviceArn": "arn:aws:ecs:us-west-2:123456789012:service/ecs-simple-
service-elb",
        "deployments": [
            {
                "status": "PRIMARY",
                "pendingCount": 0,
                "createdAt": 1428100239.123,
                "desiredCount": 10,
```

```
                "taskDefinition": "arn:aws:ecs:us-west-2:123456789012:task-
definition/ecs-demo:1",
                "updatedAt": 1428100239.123,
                "id": "ecs-svc/1234567890123456789",
                "runningCount": 0
            }
        ],
        "events": [],
        "runningCount": 0
    }
}
```

Per ulteriori informazioni, consulta Utilizzare il bilanciamento del carico per distribuire il traffico del servizio Amazon ECS nella Guida per gli sviluppatori di Amazon ECS.

Esempio 5: come configurare i volumi Amazon EBS durante la creazione del servizio

L'esempio `create-service` seguente configura i volumi Amazon EBS per ogni attività gestita dal servizio. È necessario disporre di un ruolo dell'infrastruttura Amazon ECS configurato con associata la policy gestita da `AmazonECSInfrastructureRolePolicyForVolumes`. È necessario specificare una definizione di attività con lo stesso nome di volume della richiesta `create-service`. Questo esempio utilizza l'opzione `--cli-input-json` e un file di input JSON denominato `ecs-simple-service-ebs.json` con il seguente contenuto.

```
aws ecs create-service \
    --cli-input-json file://ecs-simple-service-ebs.json
```

Contenuto di `ecs-simple-service-ebs.json`:

```
{
    "cluster": "mycluster",
    "taskDefinition": "mytaskdef",
    "serviceName": "ecs-simple-service-ebs",
    "desiredCount": 2,
    "launchType": "FARGATE",
    "networkConfiguration":{
        "awsvpcConfiguration":{
            "assignPublicIp": "ENABLED",
            "securityGroups": ["sg-12344321"],
            "subnets":["subnet-12344321"]
        }
    },
```

```
    "volumeConfigurations": [
        {
            "name": "myEbsVolume",
            "managedEBSVolume": {
                "roleArn":"arn:aws:iam::123456789012:role/ecsInfrastructureRole",
                "volumeType": "gp3",
                "sizeInGiB": 100,
                "iops": 3000,
                "throughput": 125,
                "filesystemType": "ext4"
            }
        }
   ]
}
```

Output:

```
{
    "service": {
        "serviceArn": "arn:aws:ecs:us-west-2:123456789012:service/mycluster/ecs-
simple-service-ebs",
        "serviceName": "ecs-simple-service-ebs",
        "clusterArn": "arn:aws:ecs:us-west-2:123456789012:cluster/mycluster",
        "loadBalancers": [],
        "serviceRegistries": [],
        "status": "ACTIVE",
        "desiredCount": 2,
        "runningCount": 0,
        "pendingCount": 0,
        "launchType": "EC2",
        "taskDefinition": "arn:aws:ecs:us-west-2:123456789012:task-definition/
mytaskdef:3",
        "deploymentConfiguration": {
            "deploymentCircuitBreaker": {
                "enable": false,
                "rollback": false
            },
            "maximumPercent": 200,
            "minimumHealthyPercent": 100
        },
        "deployments": [
            {
                "id": "ecs-svc/7851020056849183687",
```

```
                "status": "PRIMARY",
                "taskDefinition": "arn:aws:ecs:us-west-2:123456789012:task-
definition/mytaskdef:3",
                "desiredCount": 0,
                "pendingCount": 0,
                "runningCount": 0,
                "failedTasks": 0,
                "createdAt": "2025-01-21T11:32:38.034000-06:00",
                "updatedAt": "2025-01-21T11:32:38.034000-06:00",
                "launchType": "EC2",
                "networkConfiguration": {
                    "awsvpcConfiguration": {
                        "subnets": [
                            "subnet-12344321"
                        ],
                        "securityGroups": [
                            "sg-12344321"
                        ],
                        "assignPublicIp": "DISABLED"
                    }
                },
                "rolloutState": "IN_PROGRESS",
                "rolloutStateReason": "ECS deployment ecs-svc/7851020056849183687 in
 progress.",
                "volumeConfigurations": [
                    {
                        "name": "myEBSVolume",
                        "managedEBSVolume": {
                            "volumeType": "gp3",
                            "sizeInGiB": 100,
                            "iops": 3000,
                            "throughput": 125,
                            "roleArn": "arn:aws:iam::123456789012:role/
ecsInfrastructureRole",
                            "filesystemType": "ext4"
                        }
                    }
                ]
            }
        ],
        "roleArn": "arn:aws:iam::123456789012:role/aws-service-role/
ecs.amazonaws.com/AWSServiceRoleForECS",
        "events": [],
        "createdAt": "2025-01-21T11:32:38.034000-06:00",
```

```
        "placementConstraints": [],
        "placementStrategy": [],
        "networkConfiguration": {
            "awsvpcConfiguration": {
                "subnets": [
                    "subnet-12344321"
                ],
                "securityGroups": [
                    "sg-12344321"
                ],
                "assignPublicIp": "DISABLED"
            }
        },
        "healthCheckGracePeriodSeconds": 0,
        "schedulingStrategy": "REPLICA",
        "deploymentController": {
            "type": "ECS"
        },
        "createdBy": "arn:aws:iam::123456789012:user/AIDACKCEVSQ6C2EXAMPLE",
        "enableECSManagedTags": false,
        "propagateTags": "NONE",
        "enableExecuteCommand": false,
        "availabilityZoneRebalancing": "DISABLED"
    }
}
```

Per ulteriori informazioni, consulta Utilizzare i volumi Amazon EBS con Amazon ECS nella Guida per gli sviluppatori di Amazon ECS.

- Per i dettagli sull'API, consulta CreateService AWS CLICommand Reference.

`create-task-set`

Il seguente esempio di codice mostra come utilizzare`create-task-set`.

AWS CLI

Come creare un set di attività

L'esempio `create-task-set` seguente crea un set di attività in un servizio che utilizza un controller di implementazione esterno.

```
aws ecs create-task-set \
```

```
    --cluster MyCluster \
    --service MyService \
    --task-definition MyTaskDefinition:2 \
    --network-
configuration "awsvpcConfiguration={subnets=[subnet-12344321],securityGroups=[sg-12344321]}"
```

Output:

```
{
    "taskSet": {
        "id": "ecs-svc/1234567890123456789",
        "taskSetArn": "arn:aws:ecs:us-west-2:123456789012:task-set/MyCluster/
MyService/ecs-svc/1234567890123456789",
        "status": "ACTIVE",
        "taskDefinition": "arn:aws:ecs:us-west-2:123456789012:task-definition/
MyTaskDefinition:2",
        "computedDesiredCount": 0,
        "pendingCount": 0,
        "runningCount": 0,
        "createdAt": 1557128360.711,
        "updatedAt": 1557128360.711,
        "launchType": "EC2",
        "networkConfiguration": {
            "awsvpcConfiguration": {
                "subnets": [
                    "subnet-12344321"
                ],
                "securityGroups": [
                    "sg-12344321"
                ],
                "assignPublicIp": "DISABLED"
            }
        },
        "loadBalancers": [],
        "serviceRegistries": [],
        "scale": {
            "value": 0.0,
            "unit": "PERCENT"
        },
        "stabilityStatus": "STABILIZING",
        "stabilityStatusAt": 1557128360.711
    }
}
```

- Per i dettagli sull'API, consulta CreateTaskSet AWS CLICommand Reference.

`delete-account-setting`

Il seguente esempio di codice mostra come utilizzare`delete-account-setting`.

AWS CLI

Come eliminare le impostazioni dell'account per un utente IAM o un ruolo IAM specifico

L'esempio `delete-account-setting` seguente elimina le impostazioni dell'account per un utente IAM o un ruolo IAM specifico.

```
aws ecs delete-account-setting \
    --name serviceLongArnFormat \
    --principal-arn arn:aws:iam::123456789012:user/MyUser
```

Output:

```
{
    "setting": {
        "name": "serviceLongArnFormat",
        "value": "enabled",
        "principalArn": "arn:aws:iam::123456789012:user/MyUser"
    }
}
```

Per ulteriori informazioni, consulta Amazon Resource Names (ARNs) e IDs la Amazon ECS Developer Guide.

- Per i dettagli sull'API, consulta DeleteAccountSetting AWS CLICommand Reference.

`delete-attributes`

Il seguente esempio di codice mostra come utilizzare`delete-attributes`.

AWS CLI

Come eliminare uno o più attributi personalizzati da una risorsa Amazon ECS

L'esempio `delete-attributes` seguente elimina un attributo denominato `stack` da un'istanza di container.

```
aws ecs delete-attributes \
    --attributes name=stack,targetId=arn:aws:ecs:us-west-2:130757420319:container-instance/1c3be8ed-df30-47b4-8f1e-6e68ebd01f34
```

Output:

```
{
    "attributes": [
        {
            "name": "stack",
            "targetId": "arn:aws:ecs:us-west-2:130757420319:container-instance/1c3be8ed-df30-47b4-8f1e-6e68ebd01f34",
            "value": "production"
        }
    ]
}
```

- Per i dettagli sull'API, consulta DeleteAttributes AWS CLICommand Reference.

`delete-capacity-provider`

Il seguente esempio di codice mostra come utilizzare`delete-capacity-provider`.

AWS CLI

Esempio 1: come eliminare un provider di capacità utilizzando il nome della risorsa Amazon (ARN)

L'esempio `delete-capacity-provider` seguente elimina un provider di capacità specificandone il nome della risorsa Amazon (ARN). Il nome della risorsa Amazon (ARN) e lo stato dell'eliminazione del provider di capacità possono essere recuperati tramite il comando `describe-capacity-providers`.

```
aws ecs delete-capacity-provider \
    --capacity-provider arn:aws:ecs:us-west-2:123456789012:capacity-provider/ExampleCapacityProvider
```

Output:

```
{
    "capacityProvider": {
```

```
        "capacityProviderArn": "arn:aws:ecs:us-west-2:123456789012:capacity-
provider/ExampleCapacityProvider",
        "name": "ExampleCapacityProvider",
        "status": "ACTIVE",
        "autoScalingGroupProvider": {
            "autoScalingGroupArn": "arn:aws:autoscaling:us-
west-2:123456789012:autoScalingGroup:a1b2c3d4-5678-90ab-cdef-
EXAMPLE11111:autoScalingGroupName/MyAutoScalingGroup",
            "managedScaling": {
                "status": "ENABLED",
                "targetCapacity": 100,
                "minimumScalingStepSize": 1,
                "maximumScalingStepSize": 10000
            },
            "managedTerminationProtection": "DISABLED"
        },
        "updateStatus": "DELETE_IN_PROGRESS",
        "tags": []
    }
}
```

Per ulteriori informazioni, consulta Provider di capacità del cluster nella Guida per gli sviluppatori di Amazon ECS.

Esempio 2: come eliminare un provider di capacità utilizzando il nome

L'esempio `delete-capacity-provider` seguente elimina un provider di capacità specificandone il nome breve. Il nome breve e lo stato dell'eliminazione del provider di capacità possono essere recuperati tramite il comando `describe-capacity-providers`.

```
aws ecs delete-capacity-provider \
    --capacity-provider ExampleCapacityProvider
```

Output:

```
{
    "capacityProvider": {
        "capacityProviderArn": "arn:aws:ecs:us-west-2:123456789012:capacity-
provider/ExampleCapacityProvider",
        "name": "ExampleCapacityProvider",
        "status": "ACTIVE",
        "autoScalingGroupProvider": {
```

```
            "autoScalingGroupArn": "arn:aws:autoscaling:us-
west-2:123456789012:autoScalingGroup:a1b2c3d4-5678-90ab-cdef-
EXAMPLE11111:autoScalingGroupName/MyAutoScalingGroup",
            "managedScaling": {
                "status": "ENABLED",
                "targetCapacity": 100,
                "minimumScalingStepSize": 1,
                "maximumScalingStepSize": 10000
            },
            "managedTerminationProtection": "DISABLED"
        },
        "updateStatus": "DELETE_IN_PROGRESS",
        "tags": []
    }
}
```

Per ulteriori informazioni, consulta Provider di capacità del cluster nella Guida per gli sviluppatori di Amazon ECS.

- Per i dettagli sull'API, consulta DeleteCapacityProvider AWS CLICommand Reference.

`delete-cluster`

Il seguente esempio di codice mostra come utilizzare`delete-cluster`.

AWS CLI

Come eliminare un cluster vuoto

L'esempio `delete-cluster` seguente elimina il cluster vuoto specificato.

```
aws ecs delete-cluster --cluster MyCluster
```

Output:

```
{
    "cluster": {
        "clusterArn": "arn:aws:ecs:us-west-2:123456789012:cluster/MyCluster",
        "status": "INACTIVE",
        "clusterName": "MyCluster",
        "registeredContainerInstancesCount": 0,
        "pendingTasksCount": 0,
        "runningTasksCount": 0,
```

```
        "activeServicesCount": 0
        "statistics": [],
        "tags": []
    }
}
```

Per ulteriori informazioni, consulta Eliminazione di un cluster nella Guida per gli sviluppatori di Amazon ECS.

- Per i dettagli sull'API, consulta DeleteCluster AWS CLICommand Reference.

`delete-service`

Il seguente esempio di codice mostra come utilizzare`delete-service`.

AWS CLI

Come eliminare un servizio

L'esempio `ecs delete-service` seguente elimina il servizio specificato da un cluster. Puoi includere il parametro `--force` per eliminare un servizio anche se il numero di attività non è stato ridotto a zero.

```
aws ecs delete-service --cluster MyCluster --service MyService1 --force
```

Per ulteriori informazioni, consulta Eliminazione di un servizio nella Guida per gli sviluppatori di Amazon ECS.

- Per i dettagli sull'API, consulta DeleteService AWS CLICommand Reference.

`delete-task-definitions`

Il seguente esempio di codice mostra come utilizzare`delete-task-definitions`.

AWS CLI

Come eliminare una definizione di attività

L'esempio `delete-task-definitions` seguente elimina una definizione di attività INACTIVE.

```
aws ecs delete-task-definitions \
    --task-definition curltest:1
```

Output:

```
{
"taskDefinitions": [
    {
        "taskDefinitionArn": "arn:aws:ecs:us-east-1:123456789012:task-definition/
curltest:1",
        "containerDefinitions": [
            {
                "name": "ctest",
                "image": "mreferre/eksutils",
                "cpu": 0,
                "portMappings": [],
                "essential": true,
                "entryPoint": [
                    "sh",
                    "-c"
                ],
                "command": [
                    "curl ${ECS_CONTAINER_METADATA_URI_V4}/task"
                ],
                "environment": [],
                "mountPoints": [],
                "volumesFrom": [],
                "logConfiguration": {
                    "logDriver": "awslogs",
                    "options": {
                        "awslogs-create-group": "true",
                        "awslogs-group": "/ecs/curltest",
                        "awslogs-region": "us-east-1",
                        "awslogs-stream-prefix": "ecs"
                    }
                }
            }
        ],
        "family": "curltest",
        "taskRoleArn": "arn:aws:iam::123456789012:role/ecsTaskExecutionRole",
        "executionRoleArn": "arn:aws:iam::123456789012:role/ecsTaskExecutionRole",
        "networkMode": "awsvpc",
        "revision": 1,
        "volumes": [],
        "status": "DELETE_IN_PROGRESS",
        "compatibilities": [
            "EC2",
```

```
            "FARGATE"
        ],
        "requiresCompatibilities": [
            "FARGATE"
        ],
        "cpu": "256",
        "memory": "512",
        "registeredAt": "2021-09-10T12:56:24.704000+00:00",
        "deregisteredAt": "2023-03-14T15:20:59.419000+00:00",
        "registeredBy": "arn:aws:sts::123456789012:assumed-role/Admin/jdoe"
        }
    ],
    "failures": []
}
```

Per ulteriori informazioni, consulta Definizioni di attività Amazon ECS nella Guida per gli sviluppatori di Amazon ECS.

- Per i dettagli sull'API, consulta DeleteTaskDefinitions AWS CLICommand Reference.

`delete-task-set`

Il seguente esempio di codice mostra come utilizzare`delete-task-set`.

AWS CLI

Come eliminare set di attività

L'esempio `delete-task-set` seguente mostra come eliminare un set di attività. È possibile includere il parametro `--force` per eliminare un set di attività anche se il numero di attività non è stato ridotto a zero.

```
aws ecs delete-task-set \
    --cluster MyCluster \
    --service MyService \
    --task-set arn:aws:ecs:us-west-2:123456789012:task-set/MyCluster/MyService/ecs-
svc/1234567890123456789 \
    --force
```

Output:

```
{
    "taskSet": {
```

```
        "id": "ecs-svc/1234567890123456789",
        "taskSetArn": "arn:aws:ecs:us-west-2:123456789012:task-set/MyCluster/
MyService/ecs-svc/1234567890123456789",
        "status": "DRAINING",
        "taskDefinition": "arn:aws:ecs:us-west-2:123456789012:task-definition/
sample-fargate:2",
        "computedDesiredCount": 0,
        "pendingCount": 0,
        "runningCount": 0,
        "createdAt": 1557130260.276,
        "updatedAt": 1557130290.707,
        "launchType": "EC2",
        "networkConfiguration": {
            "awsvpcConfiguration": {
                "subnets": [
                    "subnet-12345678"
                ],
                "securityGroups": [
                    "sg-12345678"
                ],
                "assignPublicIp": "DISABLED"
            }
        },
        "loadBalancers": [],
        "serviceRegistries": [],
        "scale": {
            "value": 0.0,
            "unit": "PERCENT"
        },
        "stabilityStatus": "STABILIZING",
        "stabilityStatusAt": 1557130290.707
    }
}
```

- Per i dettagli sull'API, consulta DeleteTaskSet AWS CLICommand Reference.

`deregister-container-instance`

Il seguente esempio di codice mostra come utilizzare`deregister-container-instance`.

AWS CLI

Come annullare la registrazione di un'istanza di container da un cluster

L'esempio `deregister-container-instance` seguente annulla la registrazione di un'istanza di container dal cluster specificato. Se nell'istanza di container sono ancora presenti attività in esecuzione, è necessario arrestarle prima di annullarne la registrazione oppure utilizzare l'opzione `--force`.

```
aws ecs deregister-container-instance \
    --cluster arn:aws:ecs:us-west-2:123456789012:cluster/MyCluster \
    --container-instance arn:aws:ecs:us-west-2:123456789012:container-instance/
a1b2c3d4-5678-90ab-cdef-11111EXAMPLE \
    --force
```

Output:

```
{
    "containerInstance": {
        "remainingResources": [
            {
                "integerValue": 1024,
                "doubleValue": 0.0,
                "type": "INTEGER",
                "longValue": 0,
                "name": "CPU"
            },
            {
                "integerValue": 985,
                "doubleValue": 0.0,
                "type": "INTEGER",
                "longValue": 0,
                "name": "MEMORY"
            },
            {
                "type": "STRINGSET",
                "integerValue": 0,
                "name": "PORTS",
                "stringSetValue": [
                    "22",
                    "2376",
                    "2375",
                    "51678",
                    "51679"
                ],
                "longValue": 0,
```

```
                "doubleValue": 0.0
            },
            {
                "type": "STRINGSET",
                "integerValue": 0,
                "name": "PORTS_UDP",
                "stringSetValue": [],
                "longValue": 0,
                "doubleValue": 0.0
            }
        ],
        "agentConnected": true,
        "attributes": [
            {
                "name": "ecs.capability.secrets.asm.environment-variables"
            },
            {
                "name": "com.amazonaws.ecs.capability.logging-driver.syslog"
            },
            {
                "value": "ami-01a82c3fce2c3ba58",
                "name": "ecs.ami-id"
            },
            {
                "name": "ecs.capability.secrets.asm.bootstrap.log-driver"
            },
            {
                "name": "com.amazonaws.ecs.capability.logging-driver.none"
            },
            {
                "name": "ecs.capability.ecr-endpoint"
            },
            {
                "name": "com.amazonaws.ecs.capability.logging-driver.json-file"
            },
            {
                "value": "vpc-1234567890123467",
                "name": "ecs.vpc-id"
            },
            {
                "name": "ecs.capability.execution-role-awslogs"
            },
            {
                "name": "com.amazonaws.ecs.capability.docker-remote-api.1.17"
```

```
            },
            {
                "name": "com.amazonaws.ecs.capability.docker-remote-api.1.18"
            },
            {
                "name": "com.amazonaws.ecs.capability.docker-remote-api.1.19"
            },
            {
                "name": "ecs.capability.docker-plugin.local"
            },
            {
                "name": "ecs.capability.task-eni"
            },
            {
                "name": "ecs.capability.task-cpu-mem-limit"
            },
            {
                "name": "ecs.capability.secrets.ssm.bootstrap.log-driver"
            },
            {
                "name": "com.amazonaws.ecs.capability.docker-remote-api.1.30"
            },
            {
                "name": "com.amazonaws.ecs.capability.docker-remote-api.1.31"
            },
            {
                "name": "com.amazonaws.ecs.capability.docker-remote-api.1.32"
            },
            {
                "name": "ecs.capability.execution-role-ecr-pull"
            },
            {
                "name": "ecs.capability.container-health-check"
            },
            {
                "value": "subnet-1234567890123467",
                "name": "ecs.subnet-id"
            },
            {
                "value": "us-west-2a",
                "name": "ecs.availability-zone"
            },
            {
                "value": "t2.micro",
```

```
            "name": "ecs.instance-type"
        },
        {
            "name": "com.amazonaws.ecs.capability.task-iam-role-network-host"
        },
        {
            "name": "ecs.capability.aws-appmesh"
        },
        {
            "name": "com.amazonaws.ecs.capability.logging-driver.awslogs"
        },
        {
            "name": "com.amazonaws.ecs.capability.docker-remote-api.1.24"
        },
        {
            "name": "com.amazonaws.ecs.capability.docker-remote-api.1.25"
        },
        {
            "name": "com.amazonaws.ecs.capability.docker-remote-api.1.26"
        },
        {
            "name": "com.amazonaws.ecs.capability.docker-remote-api.1.27"
        },
        {
            "name": "com.amazonaws.ecs.capability.privileged-container"
        },
        {
            "name": "ecs.capability.container-ordering"
        },
        {
            "name": "com.amazonaws.ecs.capability.docker-remote-api.1.28"
        },
        {
            "name": "com.amazonaws.ecs.capability.docker-remote-api.1.29"
        },
        {
            "value": "x86_64",
            "name": "ecs.cpu-architecture"
        },
        {
            "value": "93f43776-2018.10.0",
            "name": "ecs.capability.cni-plugin-version"
        },
        {
```

```
                "name": "ecs.capability.secrets.ssm.environment-variables"
            },
            {
                "name": "ecs.capability.pid-ipc-namespace-sharing"
            },
            {
                "name": "com.amazonaws.ecs.capability.ecr-auth"
            },
            {
                "value": "linux",
                "name": "ecs.os-type"
            },
            {
                "name": "com.amazonaws.ecs.capability.docker-remote-api.1.20"
            },
            {
                "name": "com.amazonaws.ecs.capability.docker-remote-api.1.21"
            },
            {
                "name": "com.amazonaws.ecs.capability.docker-remote-api.1.22"
            },
            {
                "name": "ecs.capability.task-eia"
            },
            {
                "name": "ecs.capability.private-registry-
authentication.secretsmanager"
            },
            {
                "name": "com.amazonaws.ecs.capability.task-iam-role"
            },
            {
                "name": "com.amazonaws.ecs.capability.docker-remote-api.1.23"
            }
        ],
        "pendingTasksCount": 0,
        "tags": [],
        "containerInstanceArn": "arn:aws:ecs:us-west-2:123456789012:container-
instance/a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
        "registeredResources": [
            {
                "integerValue": 1024,
                "doubleValue": 0.0,
                "type": "INTEGER",
```

```
                "longValue": 0,
                "name": "CPU"
            },
            {
                "integerValue": 985,
                "doubleValue": 0.0,
                "type": "INTEGER",
                "longValue": 0,
                "name": "MEMORY"
            },
            {
                "type": "STRINGSET",
                "integerValue": 0,
                "name": "PORTS",
                "stringSetValue": [
                    "22",
                    "2376",
                    "2375",
                    "51678",
                    "51679"
                ],
                "longValue": 0,
                "doubleValue": 0.0
            },
            {
                "type": "STRINGSET",
                "integerValue": 0,
                "name": "PORTS_UDP",
                "stringSetValue": [],
                "longValue": 0,
                "doubleValue": 0.0
            }
        ],
        "status": "INACTIVE",
        "registeredAt": 1557768075.681,
        "version": 4,
        "versionInfo": {
            "agentVersion": "1.27.0",
            "agentHash": "aabe65ee",
            "dockerVersion": "DockerVersion: 18.06.1-ce"
        },
        "attachments": [],
        "runningTasksCount": 0,
        "ec2InstanceId": "i-12345678901234678"
```

```
    }
}
```

Per ulteriori informazioni, consulta Annullare la registrazione di un'istanza di container nella Guida per gli sviluppatori di Amazon ECS.

- Per i dettagli sull'API, consulta DeregisterContainerInstance AWS CLICommand Reference.

`deregister-task-definition`

Il seguente esempio di codice mostra come utilizzare`deregister-task-definition`.

AWS CLI

Come revocare la registrazione di una definizione di attività

L'esempio `deregister-task-definition` seguente annulla la registrazione della prima revisione della definizione di attività `curler` nella Regione predefinita.

```
aws ecs deregister-task-definition --task-definition curler:1
```

Nota che nell'output risultante lo stato della definizione dell'attività è `INACTIVE`:

```
{
    "taskDefinition": {
        "status": "INACTIVE",
        "family": "curler",
        "volumes": [],
        "taskDefinitionArn": "arn:aws:ecs:us-west-2:123456789012:task-definition/
curler:1",
        "containerDefinitions": [
            {
                "environment": [],
                "name": "curler",
                "mountPoints": [],
                "image": "curl:latest",
                "cpu": 100,
                "portMappings": [],
                "entryPoint": [],
                "memory": 256,
                "command": [
                    "curl -v http://example.com/"
                ],
```

```
                "essential": true,
                "volumesFrom": []
            }
        ],
        "revision": 1
    }
}
```

Per ulteriori informazioni, consulta Definizioni di attività Amazon ECS nella Guida per gli sviluppatori di Amazon ECS.

- Per i dettagli sull'API, consulta DeregisterTaskDefinition AWS CLICommand Reference.

`describe-capacity-providers`

Il seguente esempio di codice mostra come utilizzare`describe-capacity-providers`.

AWS CLI

Esempio 1: come descrivere tutti i provider di capacità

L'esempio `describe-capacity-providers` seguente recupera i dettagli di tutti i provider di capacità.

```
aws ecs describe-capacity-providers
```

Output:

```
{
    "capacityProviders": [
        {
            "capacityProviderArn": "arn:aws:ecs:us-west-2:123456789012:capacity-
provider/MyCapacityProvider",
            "name": "MyCapacityProvider",
            "status": "ACTIVE",
            "autoScalingGroupProvider": {
                "autoScalingGroupArn": "arn:aws:autoscaling:us-
west-2:123456789012:autoScalingGroup:a1b2c3d4-5678-90ab-cdef-
EXAMPLE11111:autoScalingGroupName/MyAutoScalingGroup",
                "managedScaling": {
                    "status": "ENABLED",
                    "targetCapacity": 100,
                    "minimumScalingStepSize": 1,
```

```
                    "maximumScalingStepSize": 1000
                },
                "managedTerminationProtection": "ENABLED"
            },
            "tags": []
        },
        {
            "capacityProviderArn": "arn:aws:ecs:us-west-2:123456789012:capacity-
provider/FARGATE",
            "name": "FARGATE",
            "status": "ACTIVE",
            "tags": []
        },
        {
            "capacityProviderArn": "arn:aws:ecs:us-west-2:123456789012:capacity-
provider/FARGATE_SPOT",
            "name": "FARGATE_SPOT",
            "status": "ACTIVE",
            "tags": []
        }
    ]
}
```

Per ulteriori informazioni, consulta Provider di capacità del cluster nella Guida per gli sviluppatori di Amazon ECS.

Esempio 2: come descrivere un provider di capacità specifico

L'esempio `describe-capacity-providers` seguente recupera i dettagli di un provider di capacità specifico. Se si utilizza il parametro `--include TAGS`, i tag associati al provider di capacità vengono aggiunti all'output.

```
aws ecs describe-capacity-providers \
    --capacity-providers MyCapacityProvider \
    --include TAGS
```

Output:

```
{
    "capacityProviders": [
        {
            "capacityProviderArn": "arn:aws:ecs:us-west-2:123456789012:capacity-
provider/MyCapacityProvider",
```

```
            "name": "MyCapacityProvider",
            "status": "ACTIVE",
            "autoScalingGroupProvider": {
                "autoScalingGroupArn": "arn:aws:autoscaling:us-
west-2:123456789012:autoScalingGroup:a1b2c3d4-5678-90ab-cdef-
EXAMPLE11111:autoScalingGroupName/MyAutoScalingGroup",
                "managedScaling": {
                    "status": "ENABLED",
                    "targetCapacity": 100,
                    "minimumScalingStepSize": 1,
                    "maximumScalingStepSize": 1000
                },
                "managedTerminationProtection": "ENABLED"
            },
            "tags": [
                {
                    "key": "environment",
                    "value": "production"
                }
            ]
        }
    ]
}
```

Per ulteriori informazioni, consulta Provider di capacità del cluster nella Guida per gli sviluppatori di Amazon ECS.

- Per i dettagli sull'API, consulta DescribeCapacityProviders AWS CLICommand Reference.

`describe-clusters`

Il seguente esempio di codice mostra come utilizzare`describe-clusters`.

AWS CLI

Esempio 1: come descrivere un cluster

L'esempio `describe-clusters` seguente recupera i dettagli del cluster specificato.

```
aws ecs describe-clusters \
    --cluster default
```

Output:

```
{
    "clusters": [
        {
            "status": "ACTIVE",
            "clusterName": "default",
            "registeredContainerInstancesCount": 0,
            "pendingTasksCount": 0,
            "runningTasksCount": 0,
            "activeServicesCount": 1,
            "clusterArn": "arn:aws:ecs:us-west-2:123456789012:cluster/default"
        }
    ],
    "failures": []
}
```

Per ulteriori informazioni, consulta Cluster Amazon ECS nella Guida per gli sviluppatori di Amazon ECS.

Esempio 2: come descrivere un cluster con l'opzione ATTACHMENTS

L'esempio `describe-clusters` seguente specifica l'opzione ATTACHMENTS. Recupera i dettagli del cluster specificato e un elenco di risorse collegate al cluster in forma di allegati. Quando si utilizza un provider di capacità con un cluster, le risorse, sia che si tratti di AutoScaling piani o politiche di scalabilità, saranno rappresentate come asp o as_policy ATTACHMENTS.

```
aws ecs describe-clusters \
    --include ATTACHMENTS \
    --clusters sampleCluster
```

Output:

```
{
    "clusters": [
        {
            "clusterArn": "arn:aws:ecs:af-south-1:123456789222:cluster/
sampleCluster",
            "clusterName": "sampleCluster",
            "status": "ACTIVE",
            "registeredContainerInstancesCount": 0,
            "runningTasksCount": 0,
            "pendingTasksCount": 0,
            "activeServicesCount": 0,
```

```
            "statistics": [],
            "tags": [],
            "settings": [],
            "capacityProviders": [
                "sampleCapacityProvider"
            ],
            "defaultCapacityProviderStrategy": [],
            "attachments": [
                {
                    "id": "a1b2c3d4-5678-901b-cdef-EXAMPLE22222",
                    "type": "as_policy",
                    "status": "CREATED",
                    "details": [
                        {
                            "name": "capacityProviderName",
                            "value": "sampleCapacityProvider"
                        },
                        {
                            "name": "scalingPolicyName",
                            "value": "ECSManagedAutoScalingPolicy-3048e262-
fe39-4eaf-826d-6f975d303188"
                        }
                    ]
                }
            ],
            "attachmentsStatus": "UPDATE_COMPLETE"
        }
    ],
    "failures": []
}
```

Per ulteriori informazioni, consulta Cluster Amazon ECS nella Guida per gli sviluppatori di Amazon ECS.

- Per i dettagli sull'API, consulta Command Reference. DescribeClustersAWS CLI

`describe-container-instances`

Il seguente esempio di codice mostra come utilizzare`describe-container-instances`.

AWS CLI

Come descrivere un'istanza di container

L'esempio `describe-container-instances` seguente recupera i dettagli di un'istanza di container nel cluster `update` utilizzando l'UUID dell'istanza di container come identificatore.

```
aws ecs describe-container-instances \
    --cluster update \
    --container-instances a1b2c3d4-5678-90ab-cdef-11111EXAMPLE
```

Output:

```
{
    "failures": [],
    "containerInstances": [
        {
            "status": "ACTIVE",
            "registeredResources": [
                {
                    "integerValue": 2048,
                    "longValue": 0,
                    "type": "INTEGER",
                    "name": "CPU",
                    "doubleValue": 0.0
                },
                {
                    "integerValue": 3955,
                    "longValue": 0,
                    "type": "INTEGER",
                    "name": "MEMORY",
                    "doubleValue": 0.0
                },
                {
                    "name": "PORTS",
                    "longValue": 0,
                    "doubleValue": 0.0,
                    "stringSetValue": [
                        "22",
                        "2376",
                        "2375",
                        "51678"
                    ],
                    "type": "STRINGSET",
                    "integerValue": 0
                }
            ],
```

```
            "ec2InstanceId": "i-A1B2C3D4",
            "agentConnected": true,
            "containerInstanceArn": "arn:aws:ecs:us-west-2:123456789012:container-
instance/a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
            "pendingTasksCount": 0,
            "remainingResources": [
                {
                    "integerValue": 2048,
                    "longValue": 0,
                    "type": "INTEGER",
                    "name": "CPU",
                    "doubleValue": 0.0
                },
                {
                    "integerValue": 3955,
                    "longValue": 0,
                    "type": "INTEGER",
                    "name": "MEMORY",
                    "doubleValue": 0.0
                },
                {
                    "name": "PORTS",
                    "longValue": 0,
                    "doubleValue": 0.0,
                    "stringSetValue": [
                        "22",
                        "2376",
                        "2375",
                        "51678"
                    ],
                    "type": "STRINGSET",
                    "integerValue": 0
                }
            ],
            "runningTasksCount": 0,
            "versionInfo": {
                "agentVersion": "1.0.0",
                "agentHash": "4023248",
                "dockerVersion": "DockerVersion: 1.5.0"
            }
        }
    ]
}
```

Per ulteriori informazioni, consulta Istanze di container Amazon ECS nella Guida per gli sviluppatori di Amazon ECS.

- Per i dettagli sull'API, consulta DescribeContainerInstances AWS CLICommand Reference.

`describe-service-deployments`

Il seguente esempio di codice mostra come utilizzare`describe-service-deployments`.

AWS CLI

Come descrivere i dettagli dell'implementazione del servizio

L'esempio `describe-service-deployments` seguente restituisce i dettagli dell'implementazione del servizio con l'ARN `arn:aws:ecs:us-east-1:123456789012:service-deployment/example-cluster/example-service/ejGvqq2ilnbKT9qj0vLJe`.

```
aws ecs describe-service-deployments \
    --service-deployment-arn arn:aws:ecs:us-east-1:123456789012:service-deployment/example-cluster/example-service/ejGvqq2ilnbKT9qj0vLJe
```

Output:

```
{
    "serviceDeployments": [
        {
            "serviceDeploymentArn": "arn:aws:ecs:us-east-1:123456789012:service-deployment/example-cluster/example-service/ejGvqq2ilnbKT9qj0vLJe",
            "serviceArn": "arn:aws:ecs:us-east-1:123456789012:service/example-cluster/example-service",
            "clusterArn": "arn:aws:ecs:us-east-1:123456789012:cluster/example-cluster",
            "createdAt": "2024-10-31T08:03:30.917000-04:00",
            "startedAt": "2024-10-31T08:03:32.510000-04:00",
            "finishedAt": "2024-10-31T08:05:04.527000-04:00",
            "updatedAt": "2024-10-31T08:05:04.527000-04:00",
            "sourceServiceRevisions": [],
            "targetServiceRevision": {
                "arn": "arn:aws:ecs:us-east-1:123456789012:service-revision/example-cluster/example-service/1485800978477494678",
                "requestedTaskCount": 1,
```

```
                "runningTaskCount": 1,
                "pendingTaskCount": 0
            },
            "status": "SUCCESSFUL",
            "deploymentConfiguration": {
                "deploymentCircuitBreaker": {
                    "enable": true,
                    "rollback": true
                },
                "maximumPercent": 200,
                "minimumHealthyPercent": 100,
                "alarms": {
                    "alarmNames": [],
                    "rollback": false,
                    "enable": false
                }
            },
            "deploymentCircuitBreaker": {
                "status": "MONITORING_COMPLETE",
                "failureCount": 0,
                "threshold": 3
            },
            "alarms": {
                "status": "DISABLED"
            }
        }
    ],
    "failures": []
}
```

Per ulteriori informazioni, consulta Visualizzare la cronologia del servizio utilizzando le implementazioni del servizio Amazon ECS nella Guida per gli sviluppatori di Amazon ECS.

- Per i dettagli sull'API, consulta DescribeServiceDeployments AWS CLICommand Reference.

`describe-service-revisions`

Il seguente esempio di codice mostra come utilizzare`describe-service-revisions`.

AWS CLI

Come descrivere i dettagli della revisione del servizio

L'esempio `describe-service-revisions` seguente restituisce i dettagli della revisione del servizio con l'ARN `arn:aws:ecs:us-east-1:123456789012:service-revision/example-cluster/example-service/1485800978477494678`.

```
aws ecs describe-service-revisions \
    --service-revision-arns arn:aws:ecs:us-east-1:123456789012:service-revision/example-cluster/example-service/1485800978477494678
```

Output:

```
{
    "serviceRevisions": [
        {
            "serviceRevisionArn": "arn:aws:ecs:us-east-1:123456789012:service-revision/example-cluster/example-service/1485800978477494678",
            "serviceArn": "arn:aws:ecs:us-east-1:123456789012:service/example-cluster/example-service",
            "clusterArn": "arn:aws:ecs:us-east-1:123456789012:cluster/example-cluster",
            "taskDefinition": "arn:aws:ecs:us-east-1:123456789012:task-definition/webserver:5",
            "capacityProviderStrategy": [
                {
                    "capacityProvider": "FARGATE",
                    "weight": 1,
                    "base": 0
                }
            ],
            "platformVersion": "1.4.0",
            "platformFamily": "Linux",
            "networkConfiguration": {
                "awsvpcConfiguration": {
                    "subnets": [
                        "subnet-0d0eab1bb38d5ca64",
                        "subnet-0db5010045995c2d5"
                    ],
                    "securityGroups": [
                        "sg-02556bf85a191f59a"
                    ],
                    "assignPublicIp": "ENABLED"
                }
            },
            "containerImages": [
```

```
                {
                    "containerName": "aws-otel-collector",
                    "imageDigest":
 "sha256:7a1b3560655071bcacd66902c20ebe9a69470d5691fe3bd36baace7c2f3c4640",
                    "image": "public.ecr.aws/aws-observability/aws-otel-
collector:v0.32.0"
                },
                {
                    "containerName": "web",
                    "imageDigest":
 "sha256:28402db69fec7c17e179ea87882667f1e054391138f77ffaf0c3eb388efc3ffb",
                    "image": "nginx"
                }
            ],
            "guardDutyEnabled": false,
            "serviceConnectConfiguration": {
                "enabled": false
            },
            "createdAt": "2024-10-31T08:03:29.302000-04:00"
        }
    ],
    "failures": []
}
```

Per ulteriori informazioni, consulta Revisioni del servizio Amazon ECS nella Guida per gli sviluppatori di Amazon ECS.

- Per i dettagli sull'API, consulta DescribeServiceRevisions AWS CLICommand Reference.

`describe-services`

Il seguente esempio di codice mostra come utilizzare`describe-services`.

AWS CLI

Come descrivere un servizio

L'esempio `describe-services` seguente recupera i dettagli del servizio `my-http-service` nel cluster predefinito.

```
aws ecs describe-services --services my-http-service
```

Output:

```
{
    "services": [
        {
            "status": "ACTIVE",
            "taskDefinition": "arn:aws:ecs:us-west-2:123456789012:task-definition/
amazon-ecs-sample:1",
            "pendingCount": 0,
            "loadBalancers": [],
            "desiredCount": 10,
            "createdAt": 1466801808.595,
            "serviceName": "my-http-service",
            "clusterArn": "arn:aws:ecs:us-west-2:123456789012:cluster/default",
            "serviceArn": "arn:aws:ecs:us-west-2:123456789012:service/my-http-
service",
            "deployments": [
                {
                    "status": "PRIMARY",
                    "pendingCount": 0,
                    "createdAt": 1466801808.595,
                    "desiredCount": 10,
                    "taskDefinition": "arn:aws:ecs:us-west-2:123456789012:task-
definition/amazon-ecs-sample:1",
                    "updatedAt": 1428326312.703,
                    "id": "ecs-svc/1234567890123456789",
                    "runningCount": 10
                }
            ],
            "events": [
                {
                    "message": "(service my-http-service) has reached a steady
 state.",
                    "id": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
                    "createdAt": 1466801812.435
                }
            ],
            "runningCount": 10
        }
    ],
    "failures": []
}
```


Per ulteriori informazioni, consulta Servizi nella Guida per gli sviluppatori di Amazon ECS.

- Per i dettagli sull'API, consulta DescribeServices AWS CLICommand Reference.

`describe-task-definition`

Il seguente esempio di codice mostra come utilizzare`describe-task-definition`.

AWS CLI

Come descrivere una definizione di attività

L'esempio `describe-task-definition` seguente recupera i dettagli di una definizione di attività.

```
aws ecs describe-task-definition \
    --task-definition hello_world:8
```

Output:

```
{
    "taskDefinition": {
        "taskDefinitionArn": "arn:aws:ecs:us-east-1:012345678910:task-definition/
hello_world:8",
        "containerDefinitions": [
            {
                "cpu": 10,
                "environment": [],
                "essential": true,
                "image": "wordpress",
                "links": [
                    "mysql"
                ] ,
                "memory": 500,
                "mountPoints": [],
                "name": "wordpress",
                "portMappings": [
                    {
                        "containerPort": 80,
                        "hostPort": 80
                    }
                ],
                "volumesFrom": []
            },
            {
```

```
                "cpu": 10,
                "environment": [
                    {
                        "name": "MYSQL_ROOT_PASSWORD",
                        "value": "password"
                    }
                ],
                "essential": true,
                "image": "mysql",
                "memory": 500,
                "mountPoints": [],
                "name": "mysql",
                "portMappings": [],
                "volumesFrom": []
            }
        ],
    "family": "hello_world",
    "revision": 8,
    "volumes": [],
    "status": "ACTIVE",
    "placementConstraints": [],
    "compatibilities": [
        "EXTERNAL",
        "EC2"
    ],
    "registeredAt": "2024-06-21T11:15:12.669000-05:00",
    "registeredBy": "arn:aws:sts::012345678910:assumed-role/demo-role/jane-doe"
    },
    "tags": []
}
```


Per ulteriori informazioni, consulta Definizioni di attività Amazon ECS nella Guida per gli sviluppatori di Amazon ECS.

- Per i dettagli sull'API, consulta DescribeTaskDefinition AWS CLICommand Reference.

`describe-task-sets`

Il seguente esempio di codice mostra come utilizzare`describe-task-sets`.

AWS CLI

Come descrivere un set di attività

L'esempio `describe-task-sets` seguente descrive un set di attività in un servizio che utilizza un deployer esterno.

```
aws ecs describe-task-sets \
    --cluster MyCluster \
    --service MyService \
    --task-sets arn:aws:ecs:us-west-2:123456789012:task-set/MyCluster/MyService/ecs-svc/1234567890123456789
```

Output:

```
{
    "taskSets": [
        {
            "id": "ecs-svc/1234567890123456789",
            "taskSetArn": "arn:aws:ecs:us-west-2:123456789012:task-set/MyCluster/
MyService/ecs-svc/1234567890123456789",
            "status": "ACTIVE",
            "taskDefinition": "arn:aws:ecs:us-west-2:123456789012:task-definition/
sample-fargate:2",
            "computedDesiredCount": 0,
            "pendingCount": 0,
            "runningCount": 0,
            "createdAt": 1557207715.195,
            "updatedAt": 1557207740.014,
            "launchType": "EC2",
            "networkConfiguration": {
                "awsvpcConfiguration": {
                    "subnets": [
                        "subnet-12344321"
                    ],
                    "securityGroups": [
                        "sg-1234431"
                    ],
                    "assignPublicIp": "DISABLED"
                }
            },
            "loadBalancers": [],
            "serviceRegistries": [],
            "scale": {
                "value": 0.0,
                "unit": "PERCENT"
            },
```

```
            "stabilityStatus": "STEADY_STATE",
            "stabilityStatusAt": 1557207740.014
        }
    ],
    "failures": []
}
```

- Per i dettagli sull'API, consulta DescribeTaskSets AWS CLICommand Reference.

`describe-tasks`

Il seguente esempio di codice mostra come utilizzare`describe-tasks`.

AWS CLI

Esempio 1: come descrivere attività ad attività singola

L'esempio `describe-tasks` seguente recupera i dettagli di un'attività in un cluster. È possibile specificare l'attività utilizzando l'ID o l'ARN completo dell'attività. In questo esempio viene utilizzato l'ARN completo dell'attività.

```
aws ecs describe-tasks \
    --cluster MyCluster \
    --tasks arn:aws:ecs:us-east-1:123456789012:task/
MyCluster/4d590253bb114126b7afa7b58EXAMPLE
```

Output:

```
{
    "tasks": [
        {
            "attachments": [],
            "attributes": [
                {
                    "name": "ecs.cpu-architecture",
                    "value": "x86_64"
                }
            ],
            "availabilityZone": "us-east-1b",
            "clusterArn": "arn:aws:ecs:us-east-1:123456789012:cluster/MyCluster",
            "connectivity": "CONNECTED",
            "connectivityAt": "2021-08-11T12:21:26.681000-04:00",
```

```
            "containerInstanceArn": "arn:aws:ecs:us-east-1:123456789012:container-
instance/test/025c7e2c5e054a6790a29fc1fEXAMPLE",
            "containers": [
                {
                    "containerArn": "arn:aws:ecs:us-east-1:123456789012:container/
MyCluster/4d590253bb114126b7afa7b58eea9221/a992d1cc-ea46-474a-b6e8-24688EXAMPLE",
                    "taskArn": "arn:aws:ecs:us-east-1:123456789012:task/
MyCluster/4d590253bb114126b7afa7b58EXAMPLE",
                    "name": "simple-app",
                    "image": "httpd:2.4",
                    "runtimeId":
 "91251eed27db90006ad67b1a08187290869f216557717dd5c39b37c94EXAMPLE",
                    "lastStatus": "RUNNING",
                    "networkBindings": [
                        {
                            "bindIP": "0.0.0.0",
                            "containerPort": 80,
                            "hostPort": 80,
                            "protocol": "tcp"
                        }
                    ],
                    "networkInterfaces": [],
                    "healthStatus": "UNKNOWN",
                    "cpu": "10",
                    "memory": "300"
                }
            ],
            "cpu": "10",
            "createdAt": "2021-08-11T12:21:26.681000-04:00",
            "desiredStatus": "RUNNING",
            "enableExecuteCommand": false,
            "group": "service:testupdate",
            "healthStatus": "UNKNOWN",
            "lastStatus": "RUNNING",
            "launchType": "EC2",
            "memory": "300",
            "overrides": {
                "containerOverrides": [
                    {
                        "name": "simple-app"
                    }
                ],
                "inferenceAcceleratorOverrides": []
            },
```

```
            "pullStartedAt": "2021-08-11T12:21:28.234000-04:00",
            "pullStoppedAt": "2021-08-11T12:21:33.793000-04:00",
            "startedAt": "2021-08-11T12:21:34.945000-04:00",
            "startedBy": "ecs-svc/968695068243EXAMPLE",
            "tags": [],
            "taskArn": "arn:aws:ecs:us-east-1:123456789012:task/
MyCluster/4d590253bb114126b7afa7b58eea9221",
            "taskDefinitionArn": "arn:aws:ecs:us-east-1:123456789012:task-
definition/console-sample-app-static2:1",
            "version": 2
        }
    ],
    "failures": []
}
```

Per ulteriori informazioni, consulta Definizioni di attività Amazon ECS nella Guida per gli sviluppatori di Amazon ECS.

Esempio 2: come descrivere più attività

L'esempio `describe-tasks` seguente recupera i dettagli di più attività in un cluster. È possibile specificare l'attività utilizzando l'ID o l'ARN completo dell'attività. In questo esempio viene utilizzata la totalità IDs delle attività.

```
aws ecs describe-tasks \
    --cluster MyCluster \
    --tasks "74de0355a10a4f979ac495c14EXAMPLE" "d789e94343414c25b9f6bd59eEXAMPLE"
```

Output:

```
{
    "tasks": [
        {
            "attachments": [
                {
                    "id": "d9e7735a-16aa-4128-bc7a-b2d51EXAMPLE",
                    "type": "ElasticNetworkInterface",
                    "status": "ATTACHED",
                    "details": [
                        {
                            "name": "subnetId",
                            "value": "subnet-0d0eab1bb3EXAMPLE"
                        },
```

```
                        {
                            "name": "networkInterfaceId",
                            "value": "eni-0fa40520aeEXAMPLE"
                        },
                        {
                            "name": "macAddress",
                            "value": "0e:89:76:28:07:b3"
                        },
                        {
                            "name": "privateDnsName",
                            "value": "ip-10-0-1-184.ec2.internal"
                        },
                        {
                            "name": "privateIPv4Address",
                            "value": "10.0.1.184"
                        }
                    ]
                }
            ],
            "attributes": [
                {
                    "name": "ecs.cpu-architecture",
                    "value": "x86_64"
                }
            ],
            "availabilityZone": "us-east-1b",
            "clusterArn": "arn:aws:ecs:us-east-1:123456789012:cluster/MyCluster",
            "connectivity": "CONNECTED",
            "connectivityAt": "2021-12-20T12:13:37.875000-05:00",
            "containers": [
                {
                    "containerArn": "arn:aws:ecs:us-east-1:123456789012:container/
MyCluster/74de0355a10a4f979ac495c14EXAMPLE/aad3ba00-83b3-4dac-84d4-11f8cEXAMPLE",
                    "taskArn": "arn:aws:ecs:us-east-1:123456789012:task/
MyCluster/74de0355a10a4f979ac495c14EXAMPLE",
                    "name": "web",
                    "image": "nginx",
                    "runtimeId": "74de0355a10a4f979ac495c14EXAMPLE-265927825",
                    "lastStatus": "RUNNING",
                    "networkBindings": [],
                    "networkInterfaces": [
                        {
                            "attachmentId": "d9e7735a-16aa-4128-bc7a-b2d51EXAMPLE",
                            "privateIpv4Address": "10.0.1.184"
```

```
                        }
                    ],
                    "healthStatus": "UNKNOWN",
                    "cpu": "99",
                    "memory": "100"
                }
            ],
            "cpu": "256",
            "createdAt": "2021-12-20T12:13:20.226000-05:00",
            "desiredStatus": "RUNNING",
            "enableExecuteCommand": false,
            "group": "service:tdsevicetag",
            "healthStatus": "UNKNOWN",
            "lastStatus": "RUNNING",
            "launchType": "FARGATE",
            "memory": "512",
            "overrides": {
                "containerOverrides": [
                    {
                        "name": "web"
                    }
                ],
                "inferenceAcceleratorOverrides": []
            },
            "platformVersion": "1.4.0",
            "platformFamily": "Linux",
            "pullStartedAt": "2021-12-20T12:13:42.665000-05:00",
            "pullStoppedAt": "2021-12-20T12:13:46.543000-05:00",
            "startedAt": "2021-12-20T12:13:48.086000-05:00",
            "startedBy": "ecs-svc/988401040018EXAMPLE",
            "tags": [],
            "taskArn": "arn:aws:ecs:us-east-1:123456789012:task/
MyCluster/74de0355a10a4f979ac495c14EXAMPLE",
            "taskDefinitionArn": "arn:aws:ecs:us-east-1:123456789012:task-
definition/webserver:2",
            "version": 3,
            "ephemeralStorage": {
            "sizeInGiB": 20
            }
        },
        {
            "attachments": [
                {
                    "id": "214eb5a9-45cd-4bf8-87bc-57fefEXAMPLE",
```

```
                    "type": "ElasticNetworkInterface",
                    "status": "ATTACHED",
                    "details": [
                        {
                            "name": "subnetId",
                            "value": "subnet-0d0eab1bb3EXAMPLE"
                        },
                        {
                            "name": "networkInterfaceId",
                            "value": "eni-064c7766daEXAMPLE"
                        },
                        {
                            "name": "macAddress",
                            "value": "0e:76:83:01:17:a9"
                        },
                        {
                            "name": "privateDnsName",
                            "value": "ip-10-0-1-41.ec2.internal"
                        },
                        {
                            "name": "privateIPv4Address",
                            "value": "10.0.1.41"
                        }
                    ]
                }
            ],
            "attributes": [
                {
                    "name": "ecs.cpu-architecture",
                    "value": "x86_64"
                }
            ],
            "availabilityZone": "us-east-1b",
            "clusterArn": "arn:aws:ecs:us-east-1:123456789012:cluster/MyCluster",
            "connectivity": "CONNECTED",
            "connectivityAt": "2021-12-20T12:13:35.243000-05:00",
            "containers": [
                {
                    "containerArn": "arn:aws:ecs:us-east-1:123456789012:container/
MyCluster/d789e94343414c25b9f6bd59eEXAMPLE/9afef792-609b-43a5-bb6a-3efdbEXAMPLE",
                    "taskArn": "arn:aws:ecs:us-east-1:123456789012:task/MyCluster/
d789e94343414c25b9f6bd59eEXAMPLE",
                    "name": "web",
                    "image": "nginx",
```

```
                    "runtimeId": "d789e94343414c25b9f6bd59eEXAMPLE-265927825",
                    "lastStatus": "RUNNING",
                    "networkBindings": [],
                    "networkInterfaces": [
                        {
                            "attachmentId": "214eb5a9-45cd-4bf8-87bc-57fefEXAMPLE",
                            "privateIpv4Address": "10.0.1.41"
                        }
                    ],
                    "healthStatus": "UNKNOWN",
                    "cpu": "99",
                    "memory": "100"
                }
            ],
            "cpu": "256",
            "createdAt": "2021-12-20T12:13:20.226000-05:00",
            "desiredStatus": "RUNNING",
            "enableExecuteCommand": false,
            "group": "service:tdsevicetag",
            "healthStatus": "UNKNOWN",
            "lastStatus": "RUNNING",
            "launchType": "FARGATE",
            "memory": "512",
            "overrides": {
                "containerOverrides": [
                    {
                        "name": "web"
                    }
                ],
                "inferenceAcceleratorOverrides": []
            },
            "platformVersion": "1.4.0",
            "platformFamily": "Linux",
            "pullStartedAt": "2021-12-20T12:13:44.611000-05:00",
            "pullStoppedAt": "2021-12-20T12:13:48.251000-05:00",
            "startedAt": "2021-12-20T12:13:49.326000-05:00",
            "startedBy": "ecs-svc/988401040018EXAMPLE",
            "tags": [],
            "taskArn": "arn:aws:ecs:us-east-1:123456789012:task/MyCluster/
d789e94343414c25b9f6bd59eEXAMPLE",
            "taskDefinitionArn": "arn:aws:ecs:us-east-1:123456789012:task-
definition/webserver:2",
            "version": 3,
            "ephemeralStorage": {
```

```
                "sizeInGiB": 20
            }
        }
    ],
    "failures": []
}
```

Per ulteriori informazioni, consulta Definizioni di attività Amazon ECS nella Guida per gli sviluppatori di Amazon ECS.

- Per i dettagli sull'API, consulta DescribeTasks AWS CLICommand Reference.

`execute-command`

Il seguente esempio di codice mostra come utilizzare`execute-command`.

AWS CLI

Come eseguire un comando /bin/sh interattivo

L'`execute-command`esempio seguente esegue un comando interattivo /bin/sh su un contenitore denominato MyContainer in base a un'attività con un id di. `arn:aws:ecs:us-east-1:123456789012:task/MyCluster/d789e94343414c25b9f6bd59eEXAMPLE`

```
aws ecs execute-command \
    --cluster MyCluster \
    --task arn:aws:ecs:us-east-1:123456789012:task/MyCluster/
d789e94343414c25b9f6bd59eEXAMPLE \
    --container MyContainer \
    --interactive \
    --command "/bin/sh"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Utilizzo di Amazon ECS Exec per il debug nella Guida per gli sviluppatori di Amazon ECS.

- Per i dettagli sull'API, vedere ExecuteCommandin AWS CLI Command Reference.

`get-task-protection`

Il seguente esempio di codice mostra come utilizzare`get-task-protection`.

AWS CLI

Come recuperare lo stato di protezione di un'attività nel servizio ECS

L'esempio `get-task-protection` seguente recupera lo stato di protezione delle attività ECS appartenenti al servizio Amazon ECS.

```
aws ecs get-task-protection \
    --cluster ECS-project-update-cluster \
    --tasks c43ed3b1331041f289316f958adb6a24
```

Output:

```
{
    "protectedTasks": [
        {
            "taskArn": "arn:aws:ecs:us-west-2:123456789012:task/
c43ed3b1331041f289316f958adb6a24",
            "protectionEnabled": false
        }
    ],
    "failures": []
}
```

Per ulteriori informazioni sulla protezione delle attività, consulta Proteggere le attività di Amazon ECS dall'interruzione causata da eventi di riduzione orizzontale nella Guida per gli sviluppatori Amazon ECS.

- Per i dettagli sull'API, consulta GetTaskProtection AWS CLICommand Reference.

`list-account-settings`

Il seguente esempio di codice mostra come utilizzare`list-account-settings`.

AWS CLI

Esempio 1: come visualizzare le impostazioni di un account

L'esempio `list-account-settings` seguente visualizza le impostazioni effettive di un account.

```
aws ecs list-account-settings --effective-settings
```

Output:

```
{
    "settings": [
        {
            "name": "containerInstanceLongArnFormat",
            "value": "enabled",
            "principalArn": "arn:aws:iam::123456789012:root"
        },
        {
            "name": "serviceLongArnFormat",
            "value": "enabled",
            "principalArn": "arn:aws:iam::123456789012:root"
        },
        {
            "name": "taskLongArnFormat",
            "value": "enabled",
            "principalArn": "arn:aws:iam::123456789012:root"
        }
    ]
}
```

Esempio 2: come visualizzare le impostazioni dell'account per un utente IAM o un ruolo IAM specifico

L'esempio `list-account-settings` seguente visualizza le impostazioni dell'account per un utente IAM o un ruolo IAM specifico.

```
aws ecs list-account-settings --principal-arn arn:aws:iam::123456789012:user/MyUser
```

Output:

```
{
    "settings": [
        {
            "name": "serviceLongArnFormat",
            "value": "enabled",
            "principalArn": "arn:aws:iam::123456789012:user/MyUser"
        }
    ]
}
```

Per ulteriori informazioni, consulta Amazon Resource Names (ARNs) e IDs la Amazon ECS Developer Guide.

- Per i dettagli sull'API, consulta ListAccountSettings AWS CLICommand Reference.

`list-attributes`

Il seguente esempio di codice mostra come utilizzare`list-attributes`.

AWS CLI

Come elencare le istanze di container che contengono un attributo specifico

L'esempio seguente elenca gli attributi per le istanze di container con l'attributo `stack=production` nel cluster predefinito.

```
aws ecs list-attributes \
    --target-type container-instance \
    --attribute-name stack \
    --attribute-value production \
    --cluster default
```

Output:

```
{
    "attributes": [
        {
            "name": "stack",
            "targetId": "arn:aws:ecs:us-west-2:130757420319:container-
instance/1c3be8ed-df30-47b4-8f1e-6e68ebd01f34",
            "value": "production"
        }
    ]
}
```

Per ulteriori informazioni, consulta Configurazione dell'agente del container di Amazon ECS nella Guida per gli sviluppatori di Amazon ECS.

- Per i dettagli sull'API, consulta ListAttributes AWS CLICommand Reference.

`list-clusters`

Il seguente esempio di codice mostra come utilizzare`list-clusters`.

AWS CLI

Come elencare i cluster disponibili

L'esempio `list-clusters` seguente elenca tutti i cluster disponibili.

```
aws ecs list-clusters
```

Output:

```
{
    "clusterArns": [
        "arn:aws:ecs:us-west-2:123456789012:cluster/MyECSCluster1",
        "arn:aws:ecs:us-west-2:123456789012:cluster/AnotherECSCluster"
    ]
}
```

Per ulteriori informazioni, consulta Cluster Amazon ECS nella Guida per gli sviluppatori di Amazon ECS.

- Per i dettagli sull'API, consulta ListClusters AWS CLICommand Reference.

`list-container-instances`

Il seguente esempio di codice mostra come utilizzare`list-container-instances`.

AWS CLI

Come elencare le istanze di container in un cluster

L'esempio `list-container-instances` seguente elenca tutte le istanze di container disponibili in un cluster.

```
aws ecs list-container-instances --cluster MyCluster
```

Output:

```
{
    "containerInstanceArns": [
        "arn:aws:ecs:us-west-2:123456789012:container-instance/MyCluster/
a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
        "arn:aws:ecs:us-west-2:123456789012:container-instance/MyCluster/
a1b2c3d4-5678-90ab-cdef-22222EXAMPLE"
    ]
}
```

Per ulteriori informazioni, consulta Istanze di container Amazon ECS nella Guida per gli sviluppatori di Amazon ECS.

- Per i dettagli sull'API, consulta ListContainerInstances AWS CLICommand Reference.

`list-service-deployments`

Il seguente esempio di codice mostra come utilizzare`list-service-deployments`.

AWS CLI

Come elencare le implementazioni del servizio

L'esempio `list-service-deployments` seguente recupera le implementazioni del servizio denominato `example-service`.

```
aws ecs list-service-deployments \
    --service arn:aws:ecs:us-east-1:123456789012:service/example-cluster/example-
service
```

Output:

```
{
    "serviceDeployments": [
        {
            "serviceDeploymentArn": "arn:aws:ecs:us-east-1:123456789012:service-
deployment/example-cluster/example-service/ejGvqq2ilnbKT9qj0vLJe",
            "serviceArn": "arn:aws:ecs:us-east-1:123456789012:service/example-
cluster/example-service",
            "clusterArn": "arn:aws:ecs:us-east-1:123456789012:cluster/example-
cluster",
            "startedAt": "2024-10-31T08:03:32.510000-04:00",
```

```
            "createdAt": "2024-10-31T08:03:30.917000-04:00",
            "finishedAt": "2024-10-31T08:05:04.527000-04:00",
            "targetServiceRevisionArn": "arn:aws:ecs:us-east-1:123456789012:service-
revision/example-cluster/example-service/1485800978477494678",
            "status": "SUCCESSFUL"
        }
    ]
}
```

Per ulteriori informazioni, consulta Visualizzare la cronologia del servizio utilizzando le implementazioni del servizio Amazon ECS nella Guida per gli sviluppatori di Amazon ECS.

- Per i dettagli sull'API, consulta ListServiceDeployments AWS CLICommand Reference.

`list-services-by-namespace`

Il seguente esempio di codice mostra come utilizzare`list-services-by-namespace`.

AWS CLI

Come elencare i servizi in un namespace

L'esempio `list-services-by-namespace` seguente elenca tutti i servizi configurati per il namespace specificato nella Regione predefinita.

```
aws ecs list-services-by-namespace \
    --namespace service-connect
```

Output:

```
{
    "serviceArns": [
        "arn:aws:ecs:us-west-2:123456789012:service/MyCluster/MyService",
        "arn:aws:ecs:us-west-2:123456789012:service/tutorial/service-connect-nginx-
service"
    ]
}
```

Per ulteriori informazioni, consulta Service Connect nella Guida per gli sviluppatori di Amazon ECS.

- Per i dettagli sull'API, consulta ListServicesByNamespace AWS CLICommand Reference.

`list-services`

Il seguente esempio di codice mostra come utilizzare`list-services`.

AWS CLI

Come elencare i servizi in un cluster

L'esempio `list-services` seguente elenca i servizi in esecuzione in un cluster.

```
aws ecs list-services --cluster MyCluster
```

Output:

```
 {
     "serviceArns": [
         "arn:aws:ecs:us-west-2:123456789012:service/MyCluster/MyService"
     ]
}
```

Per ulteriori informazioni, consulta Servizi nella Guida per gli sviluppatori di Amazon ECS.

- Per i dettagli sull'API, consulta ListServices AWS CLICommand Reference.

`list-tags-for-resource`

Il seguente esempio di codice mostra come utilizzare`list-tags-for-resource`.

AWS CLI

Come elencare i tag per una risorsa

L'esempio `list-tags-for-resource` seguente elenca i tag per un cluster specifico.

```
aws ecs list-tags-for-resource \
    --resource-arn arn:aws:ecs:us-west-2:123456789012:cluster/MyCluster
```

Output:

```
{
    "tags": [
        {
```

```
            "key": "key1",
            "value": "value1"
        },
        {
            "key": "key2",
            "value": "value2"
        },
        {
            "key": "key3",
            "value": "value3"
        }
    ]
}
```

- Per i dettagli sull'API, consulta ListTagsForResource AWS CLICommand Reference.

`list-task-definition-families`

Il seguente esempio di codice mostra come utilizzare`list-task-definition-families`.

AWS CLI

Esempio 1: come elencare le famiglie di definizioni di attività registrate

L'esempio `list-task-definition-families` seguente elenca tutte le famiglie di definizioni di attività registrate.

```
aws ecs list-task-definition-families
```

Output:

```
{
    "families": [
        "node-js-app",
        "web-timer",
        "hpcc",
        "hpcc-c4-8xlarge"
    ]
}
```

Esempio 2: come filtrare le famiglie di definizioni di attività registrate

L'esempio `list-task-definition-families` seguente elenca le revisioni delle definizioni di attività che iniziano con "hpcc".

```
aws ecs list-task-definition-families --family-prefix hpcc
```

Output:

```
{
    "families": [
        "hpcc",
        "hpcc-c4-8xlarge"
    ]
}
```

Per ulteriori informazioni, consulta Parametri di definizione di attività nella Guida per gli sviluppatori di Amazon ECS.

- Per i dettagli sull'API, consulta ListTaskDefinitionFamilies AWS CLICommand Reference.

`list-task-definitions`

Il seguente esempio di codice mostra come utilizzare`list-task-definitions`.

AWS CLI

Esempio 1: come elencare le definizioni di attività registrate

L'esempio `list-task-definitions` seguente elenca tutte le definizioni di attività registrate.

```
aws ecs list-task-definitions
```

Output:

```
{
    "taskDefinitionArns": [
        "arn:aws:ecs:us-west-2:123456789012:task-definition/sleep300:2",
        "arn:aws:ecs:us-west-2:123456789012:task-definition/sleep360:1",
        "arn:aws:ecs:us-west-2:123456789012:task-definition/wordpress:3",
        "arn:aws:ecs:us-west-2:123456789012:task-definition/wordpress:4",
        "arn:aws:ecs:us-west-2:123456789012:task-definition/wordpress:5",
        "arn:aws:ecs:us-west-2:123456789012:task-definition/wordpress:6"
    ]
```

```
}
```

Esempio 2: come elencare le definizioni di attività registrate in una famiglia

L' list-task-definitionsesempio seguente elenca le revisioni delle definizioni delle attività di una famiglia specificata.

```
aws ecs list-task-definitions --family-prefix wordpress
```

Output:

```
{
    "taskDefinitionArns": [
        "arn:aws:ecs:us-west-2:123456789012:task-definition/wordpress:3",
        "arn:aws:ecs:us-west-2:123456789012:task-definition/wordpress:4",
        "arn:aws:ecs:us-west-2:123456789012:task-definition/wordpress:5",
        "arn:aws:ecs:us-west-2:123456789012:task-definition/wordpress:6"
    ]
}
```

Per ulteriori informazioni, consulta Definizioni di attività Amazon ECS nella Guida per gli sviluppatori di Amazon ECS.

- Per i dettagli sull'API, vedere ListTaskDefinitionsin AWS CLI Command Reference.

`list-tasks`

Il seguente esempio di codice mostra come utilizzare`list-tasks`.

AWS CLI

Esempio 1: come elencare le attività in un cluster

L'esempio `list-tasks` seguente elenca tutte le attività in un cluster.

```
aws ecs list-tasks --cluster default
```

Output:

```
{
    "taskArns": [
```

```
        "arn:aws:ecs:us-west-2:123456789012:task/a1b2c3d4-5678-90ab-
cdef-11111EXAMPLE",
        "arn:aws:ecs:us-west-2:123456789012:task/a1b2c3d4-5678-90ab-
cdef-22222EXAMPLE"
    ]
}
```

Esempio 2: come elencare le attività su un'istanza di container specifica

L'esempio `list-tasks` seguente elenca le attività in un'istanza di container utilizzando l'UUID dell'istanza di container come filtro.

```
aws ecs list-tasks --cluster default --container-instance a1b2c3d4-5678-90ab-
cdef-33333EXAMPLE
```

Output:

```
{
    "taskArns": [
        "arn:aws:ecs:us-west-2:123456789012:task/a1b2c3d4-5678-90ab-
cdef-44444EXAMPLE"
    ]
}
```

Per ulteriori informazioni, consulta Definizioni di attività Amazon ECS nella Guida per gli sviluppatori di Amazon ECS.

- Per i dettagli sull'API, consulta ListTasks AWS CLICommand Reference.

`put-account-setting-default`

Il seguente esempio di codice mostra come utilizzare`put-account-setting-default`.

AWS CLI

Come modificare le impostazioni dell'account predefinite

L'esempio `put-account-setting-default` seguente modifica l'impostazione predefinita degli account di tutti gli utenti o ruoli IAM dell'account corrente. Queste modifiche si applicano all'intero AWS account a meno che un utente o un ruolo IAM non sovrascriva esplicitamente queste impostazioni.

```
aws ecs put-account-setting-default --name serviceLongArnFormat --value enabled
```

Output:

```
{
    "setting": {
        "name": "serviceLongArnFormat",
        "value": "enabled",
        "principalArn": "arn:aws:iam::123456789012:root"
    }
}
```

Per ulteriori informazioni, consulta Amazon Resource Names (ARNs) e IDs la Amazon ECS Developer Guide.

- Per i dettagli sull'API, consulta PutAccountSettingDefault AWS CLICommand Reference.

`put-account-setting`

Il seguente esempio di codice mostra come utilizzare`put-account-setting`.

AWS CLI

Come modificare l'impostazione dell'account per l'account utente IAM corrente

L'esempio `put-account-setting` seguente specifica l'impostazione `containerInsights` dell'account su `enhanced` per l'account utente IAM corrente. In tal modo si attivano gli approfondimenti sui container con osservabilità migliorata.

```
aws ecs put-account-setting \
    --name containerInsights \
    --value enhanced
```

Output:

```
{
    "setting": {
        "name": "containerInsights",
        "value": "enhanced",
        "principalArn": "arn:aws:iam::123456789012:user/johndoe",
```

```
        "type": "user"
    }
}
```

Per ulteriori informazioni, consulta Modifica delle impostazioni dell'account nella Guida per gli sviluppatori di Amazon ECS.

- Per i dettagli sull'API, consulta PutAccountSetting AWS CLICommand Reference.

`put-account-settings`

Il seguente esempio di codice mostra come utilizzare`put-account-settings`.

AWS CLI

Come modificare le impostazioni dell'account per un utente IAM o un ruolo IAM

L'esempio `put-account-setting` seguente modifica le impostazioni dell'account per l'utente IAM o il ruolo IAM specifico.

```
aws ecs put-account-setting \
    --name serviceLongArnFormat \
    --value enabled \
    --principal-arn arn:aws:iam::123456789012:user/MyUser
```

Output:

```
{
    "setting": {
        "name": "serviceLongArnFormat",
        "value": "enabled",
        "principalArn": "arn:aws:iam::123456789012:user/MyUser"
    }
}
```

- Per i dettagli sull'API, consulta PutAccountSettings AWS CLICommand Reference.

`put-attributes`

Il seguente esempio di codice mostra come utilizzare`put-attributes`.

AWS CLI

Come creare un attributo e associarlo a una risorsa Amazon ECS

L'esempio `put-attributes` seguente applica un attributo con nome stack e valore production a un'istanza di container.

```
aws ecs put-attributes \
    --attributes name=stack,value=production,targetId=arn:aws:ecs:us-
west-2:130757420319:container-instance/1c3be8ed-df30-47b4-8f1e-6e68ebd01f34
```

Output:

```
{
    "attributes": [
        {
            "name": "stack",
            "targetId": "arn:aws:ecs:us-west-2:130757420319:container-
instance/1c3be8ed-df30-47b4-8f1e-6e68ebd01f34",
            "value": "production"
        }
    ]
}
```

- Per i dettagli sull'API, consulta PutAttributes AWS CLICommand Reference.

`put-cluster-capacity-providers`

Il seguente esempio di codice mostra come utilizzare`put-cluster-capacity-providers`.

AWS CLI

Esempio 1: come aggiungere un provider di capacità esistente a un cluster

L'esempio `put-cluster-capacity-providers` seguente aggiunge un provider di capacità esistente a un cluster. Il comando `create-capacity-provider` viene utilizzato per creare un provider di capacità. Il comando `describe-clusters` viene utilizzato per descrivere i provider di capacità correnti e la relativa strategia predefinita associata a un cluster. Quando si aggiunge un nuovo provider di capacità a un cluster, oltre al nuovo provider di capacità da associare al cluster è necessario specificare tutti i provider di capacità esistenti. È anche necessario specificare

la strategia predefinita del provider di capacità da associare al cluster. In questo esempio, in cui al cluster `MyCluster` è associato il provider di capacità `MyCapacityProvider1`, si desidera aggiungere il provider di capacità `MyCapacityProvider2` per includerlo nella strategia predefinita del provider in modo che le attività siano distribuite uniformemente tra entrambi i provider.

```
aws ecs put-cluster-capacity-providers \
    --cluster MyCluster \
    --capacity-providers MyCapacityProvider1 MyCapacityProvider2 \
    --default-capacity-provider-
strategy capacityProvider=MyCapacityProvider1,weight=1 capacityProvider=MyCapacityProvider2,
```

Output:

```
{
    "cluster": {
        "clusterArn": "arn:aws:ecs:us-west-2:123456789012:cluster/MyCluster",
        "clusterName": "MyCluster",
        "status": "ACTIVE",
        "registeredContainerInstancesCount": 0,
        "runningTasksCount": 0,
        "pendingTasksCount": 0,
        "activeServicesCount": 0,
        "statistics": [],
        "tags": [],
        "settings": [
            {
                "name": "containerInsights",
                "value": "enabled"
            }
        ],
        "capacityProviders": [
            "MyCapacityProvider1",
            "MyCapacityProvider2"
        ],
        "defaultCapacityProviderStrategy": [
            {
                "capacityProvider": "MyCapacityProvider1",
                "weight": 1,
                "base": 0
            },
            {
```

```
                "capacityProvider": "MyCapacityProvider2",
                "weight": 1,
                "base": 0
            }
        ],
        "attachments": [
           {
                "id": "0fb0c8f4-6edd-4de1-9b09-17e470ee1918",
                "type": "as_policy",
                "status": "ACTIVE",
                "details": [
                    {
                        "name": "capacityProviderName",
                        "value": "MyCapacityProvider1"
                    },
                    {
                        "name": "scalingPolicyName",
                        "value": "ECSManagedAutoScalingPolicy-a1b2c3d4-5678-90ab-
cdef-EXAMPLE11111"
                    }
                ]
            },
            {
                "id": "ae592060-2382-4663-9476-b015c685593c",
                "type": "as_policy",
                "status": "ACTIVE",
                "details": [
                    {
                        "name": "capacityProviderName",
                        "value": "MyCapacityProvider2"
                    },
                    {
                        "name": "scalingPolicyName",
                        "value": "ECSManagedAutoScalingPolicy-a1b2c3d4-5678-90ab-
cdef-EXAMPLE22222"
                    }
                ]
            }
        ],
        "attachmentsStatus": "UPDATE_IN_PROGRESS"
    }
}
```

Per ulteriori informazioni, consulta Provider di capacità del cluster nella Guida per gli sviluppatori di Amazon ECS.

Esempio 2: come rimuovere un provider di capacità da un cluster

L'esempio `put-cluster-capacity-providers` seguente rimuove un provider di capacità da un cluster. Il comando `describe-clusters` viene utilizzato per descrivere i provider di capacità correnti e associati a un cluster. Quando si rimuove un provider di capacità da un cluster, è necessario specificare i provider di capacità da lasciare associati al cluster, nonché la strategia predefinita dei provider da associare al cluster. In questo esempio, in cui al cluster sono associati i provider di capacità `MyCapacityProvider1` e `MyCapacityProvider2`, si desidera rimuovere il provider di capacità `MyCapacityProvider2`, che viene quindi specificato da solo nel comando `MyCapacityProvider1` insieme alla strategia predefinita aggiornata del provider.

```
aws ecs put-cluster-capacity-providers \
    --cluster MyCluster \
    --capacity-providers MyCapacityProvider1 \
    --default-capacity-provider-
strategy capacityProvider=MyCapacityProvider1,weight=1,base=0
```

Output:

```
{
    "cluster": {
        "clusterArn": "arn:aws:ecs:us-west-2:123456789012:cluster/MyCluster",
        "clusterName": "MyCluster",
        "status": "ACTIVE",
        "registeredContainerInstancesCount": 0,
        "runningTasksCount": 0,
        "pendingTasksCount": 0,
        "activeServicesCount": 0,
        "statistics": [],
        "tags": [],
        "settings": [
            {
                "name": "containerInsights",
                "value": "enabled"
            }
        ],
        "capacityProviders": [
            "MyCapacityProvider1"
        ],
```

```
        "defaultCapacityProviderStrategy": [
            "capacityProvider": "MyCapacityProvider1",
            "weight": 1,
            "base": 0
        ],
        "attachments": [
           {
                "id": "0fb0c8f4-6edd-4de1-9b09-17e470ee1918",
                "type": "as_policy",
                "status": "ACTIVE",
                "details": [
                    {
                        "name": "capacityProviderName",
                        "value": "MyCapacityProvider1"
                    },
                    {
                        "name": "scalingPolicyName",
                        "value": "ECSManagedAutoScalingPolicy-a1b2c3d4-5678-90ab-
cdef-EXAMPLE11111"
                    }
                ]
            },
            {
                "id": "ae592060-2382-4663-9476-b015c685593c",
                "type": "as_policy",
                "status": "DELETING",
                "details": [
                    {
                        "name": "capacityProviderName",
                        "value": "MyCapacityProvider2"
                    },
                    {
                        "name": "scalingPolicyName",
                        "value": "ECSManagedAutoScalingPolicy-a1b2c3d4-5678-90ab-
cdef-EXAMPLE22222"
                    }
                ]
            }
        ],
        "attachmentsStatus": "UPDATE_IN_PROGRESS"
    }
}
```

Per ulteriori informazioni, consulta Provider di capacità del cluster nella Guida per gli sviluppatori di Amazon ECS.

Esempio 3: come rimuovere tutti i provider di capacità da un cluster

L'esempio `put-cluster-capacity-providers` seguente rimuove tutti i provider di capacità esistenti dal cluster.

```
aws ecs put-cluster-capacity-providers \
    --cluster MyCluster \
    --capacity-providers [] \
    --default-capacity-provider-strategy []
```

Output:

```
{
    "cluster": {
        "clusterArn": "arn:aws:ecs:us-west-2:123456789012:cluster/MyCluster",
        "clusterName": "MyCluster",
        "status": "ACTIVE",
        "registeredContainerInstancesCount": 0,
        "runningTasksCount": 0,
        "pendingTasksCount": 0,
        "activeServicesCount": 0,
        "statistics": [],
        "tags": [],
        "settings": [
            {
                "name": "containerInsights",
                "value": "enabled"
            }
        ],
        "capacityProviders": [],
        "defaultCapacityProviderStrategy": [],
        "attachments": [
           {
                "id": "0fb0c8f4-6edd-4de1-9b09-17e470ee1918",
                "type": "as_policy",
                "status": "DELETING",
                "details": [
                    {
                        "name": "capacityProviderName",
                        "value": "MyCapacityProvider1"
```

```
                    },
                    {
                        "name": "scalingPolicyName",
                        "value": "ECSManagedAutoScalingPolicy-a1b2c3d4-5678-90ab-
cdef-EXAMPLE11111"
                    }
                ]
            },
            {
                "id": "ae592060-2382-4663-9476-b015c685593c",
                "type": "as_policy",
                "status": "DELETING",
                "details": [
                    {
                        "name": "capacityProviderName",
                        "value": "MyCapacityProvider2"
                    },
                    {
                        "name": "scalingPolicyName",
                        "value": "ECSManagedAutoScalingPolicy-a1b2c3d4-5678-90ab-
cdef-EXAMPLE22222"
                    }
                ]
            }
        ],
        "attachmentsStatus": "UPDATE_IN_PROGRESS"
    }
}
```

Per ulteriori informazioni, consulta Provider di capacità del cluster nella Guida per gli sviluppatori di Amazon ECS.

- Per i dettagli sull'API, consulta PutClusterCapacityProviders AWS CLICommand Reference.

`register-task-definition`

Il seguente esempio di codice mostra come utilizzare`register-task-definition`.

AWS CLI

Esempio 1: come registrare una definizione di attività con un file JSON

L'esempio `register-task-definition` seguente registra una definizione di attività nella famiglia specificata. Le definizioni di container vengono salvate in formato JSON nella posizione del file specificata.

```
aws ecs register-task-definition \
    --cli-input-json file://<path_to_json_file>/sleep360.json
```

Contenuto di `sleep360.json`:

```
{
    "containerDefinitions": [
        {
            "name": "sleep",
            "image": "busybox",
            "cpu": 10,
            "command": [
                "sleep",
                "360"
            ],
            "memory": 10,
            "essential": true
        }
    ],
    "family": "sleep360"
}
```

Output:

```
{
    "taskDefinition": {
        "status": "ACTIVE",
        "family": "sleep360",
        "placementConstraints": [],
        "compatibilities": [
                "EXTERNAL",
                "EC2"
        ],
        "volumes": [],
        "taskDefinitionArn": "arn:aws:ecs:us-east-1:123456789012:task-definition/
sleep360:1",
        "containerDefinitions": [
            {
```

```
                "environment": [],
                "name": "sleep",
                "mountPoints": [],
                "image": "busybox",
                "cpu": 10,
                "portMappings": [],
                "command": [
                    "sleep",
                    "360"
                ],
            "memory": 10,
            "essential": true,
            "volumesFrom": []
            }
        ],
            "revision": 1
        }
}
```

Per ulteriori informazioni, consulta Definizioni di attività Amazon ECS nella Guida per gli sviluppatori di Amazon ECS.

Esempio 2: come registrare una definizione di attività con un parametro di stringa JSON

L'esempio `register-task-definition` seguente registra una definizione di attività utilizzando le definizioni di container fornite come parametro di stringa JSON con virgolette doppie con carattere di escape.

```
aws ecs register-task-definition \
    --family sleep360 \
    --container-definitions "[{\"name\":\"sleep\",\"image\":\"busybox\",\"cpu\":10,
\"command\":[\"sleep\",\"360\"],\"memory\":10,\"essential\":true}]"
```

L'output è identico a quello dell'esempio precedente.

Per ulteriori informazioni, consulta Creazione di una definizione di attività nella Guida per gli sviluppatori di Amazon ECS.

- Per i dettagli sull'API, consulta RegisterTaskDefinition AWS CLICommand Reference.

`run-task`

Il seguente esempio di codice mostra come utilizzare`run-task`.

AWS CLI

Esempio 1: come eseguire un'attività nel cluster predefinito

L'esempio `run-task` seguente esegue un'attività nel cluster predefinito e utilizza un token client.

```
aws ecs run-task \
    --cluster default \
    --task-definition sleep360:1 \
    --client-token 550e8400-e29b-41d4-a716-446655440000
```

Output:

```
{
    "tasks": [
        {
            "attachments": [],
            "attributes": [
                {
                    "name": "ecs.cpu-architecture",
                    "value": "x86_64"
                }
            ],
            "availabilityZone": "us-east-1b",
            "capacityProviderName": "example-capacity-provider",
            "clusterArn": "arn:aws:ecs:us-east-1:123456789012:cluster/default",
            "containerInstanceArn": "arn:aws:ecs:us-east-1:123456789012:container-
instance/default/bc4d2ec611d04bb7bb97e83ceEXAMPLE",
            "containers": [
                {
                    "containerArn": "arn:aws:ecs:us-east-1:123456789012:container/
default/d6f51cc5bbc94a47969c92035e9f66f8/75853d2d-711e-458a-8362-0f0aEXAMPLE",
                    "taskArn": "arn:aws:ecs:us-east-1:123456789012:task/default/
d6f51cc5bbc94a47969c9203EXAMPLE",
                    "name": "sleep",
                    "image": "busybox",
                    "lastStatus": "PENDING",
                    "networkInterfaces": [],
                    "cpu": "10",
                    "memory": "10"
                }
            ],
            "cpu": "10",
```

```
            "createdAt": "2023-11-21T16:59:34.403000-05:00",
            "desiredStatus": "RUNNING",
            "enableExecuteCommand": false,
            "group": "family:sleep360",
            "lastStatus": "PENDING",
            "launchType": "EC2",
            "memory": "10",
            "overrides": {
                "containerOverrides": [
                    {
                        "name": "sleep"
                    }
                ],
                "inferenceAcceleratorOverrides": []
            },
            "tags": [],
            "taskArn": "arn:aws:ecs:us-east-1:123456789012:task/default/
d6f51cc5bbc94a47969c9203EXAMPLE",
            "taskDefinitionArn": "arn:aws:ecs:us-east-1:123456789012:task-
definition/sleep360:1",
            "version": 1
        }
    ],
    "failures": []
}
```

Per ulteriori informazioni, consulta Esecuzione di un'applicazione come attività autonoma nella Guida per gli sviluppatori di Amazon ECS.

Esempio 2: come configurare un volume Amazon EBS per un'attività standalone

L'esempio `run-task` seguente configura un volume Amazon EBS crittografato per un'attività Fargate nel cluster predefinito. È necessario disporre di un ruolo dell'infrastruttura Amazon ECS configurato con associata la policy gestita da `AmazonECSInfrastructureRolePolicyForVolumes`. È necessario specificare una definizione di attività con lo stesso nome di volume della richiesta `run-task`. Questo esempio utilizza l'opzione `--cli-input-json` e un file di input JSON denominato `ebs.json`.

```
aws ecs run-task \
    --cli-input-json file://ebs.json
```

Contenuto di `ebs.json`:

```
{
   "cluster": "default",
   "taskDefinition": "mytaskdef",
   "launchType": "FARGATE",
   "networkConfiguration":{
        "awsvpcConfiguration":{
            "assignPublicIp": "ENABLED",
            "securityGroups": ["sg-12344321"],
            "subnets":["subnet-12344321"]
        }
    },
   "volumeConfigurations": [
        {
            "name": "myEBSVolume",
            "managedEBSVolume": {
                "volumeType": "gp3",
                "sizeInGiB": 100,
                "roleArn":"arn:aws:iam::1111222333:role/ecsInfrastructureRole",
                "encrypted": true,
                "kmsKeyId":
 "arn:aws:kms:region:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab"
            }
        }
    ]
}
```

Output:

```
{
    "tasks": [
        {
            "attachments": [
                {
                    "id": "ce868693-15ca-4083-91ac-f782f64000c9",
                    "type": "ElasticNetworkInterface",
                    "status": "PRECREATED",
                    "details": [
                        {
                        "name": "subnetId",
                        "value": "subnet-070982705451dad82"
                        }
                    ]
                },
```

```
                {
                    "id": "a17ed863-786c-4372-b5b3-b23e53f37877",
                    "type": "AmazonElasticBlockStorage",
                    "status": "CREATED",
                    "details": [
                        {
                            "name": "roleArn",
                            "value": "arn:aws:iam::123456789012:role/
ecsInfrastructureRole"
                        },
                        {
                            "name": "volumeName",
                            "value": "myEBSVolume"
                        },
                        {
                            "name": "deleteOnTermination",
                            "value": "true"
                        }
                    ]
                }
            ],
            "attributes": [
                {
                    "name": "ecs.cpu-architecture",
                    "value": "x86_64"
                }
            ],
            "availabilityZone": "us-west-2b",
            "clusterArn": "arn:aws:ecs:us-west-2:123456789012:cluster/default",
            "containers": [
                {
                    "containerArn": "arn:aws:ecs:us-west-2:123456789012:container/
default/7f1fbd3629434cc4b82d72d2f09b67c9/e21962a2-f328-4699-98a3-5161ac2c186a",
                    "taskArn": "arn:aws:ecs:us-west-2:123456789012:task/
default/7f1fbd3629434cc4b82d72d2f09b67c9",
                    "name": "container-using-ebs",
                    "image": "amazonlinux:2",
                    "lastStatus": "PENDING",
                    "networkInterfaces": [],
                    "cpu": "0"
                }
            ],
            "cpu": "1024",
            "createdAt": "2025-01-23T10:29:46.650000-06:00",
```

```
            "desiredStatus": "RUNNING",
            "enableExecuteCommand": false,
            "group": "family:mytaskdef",
            "lastStatus": "PROVISIONING",
            "launchType": "FARGATE",
            "memory": "3072",
            "overrides": {
                "containerOverrides": [
                    {
                        "name": "container-using-ebs"
                    }
                ],
                "inferenceAcceleratorOverrides": []
            },
            "platformVersion": "1.4.0",
            "platformFamily": "Linux",
            "tags": [],
            "taskArn": "arn:aws:ecs:us-west-2:123456789012:task/
default/7f1fbd3629434cc4b82d72d2f09b67c9",
            "taskDefinitionArn": "arn:aws:ecs:us-west-2:123456789012:task-
definition/mytaskdef:4",
            "version": 1,
            "ephemeralStorage": {
                "sizeInGiB": 20
            },
            "fargateEphemeralStorage": {
                "sizeInGiB": 20
            }
        }
    ],
    "failures": []
}
```

Per ulteriori informazioni, consulta Utilizzare i volumi Amazon EBS con Amazon ECS nella Guida per gli sviluppatori di Amazon ECS.

- Per i dettagli sull'API, consulta RunTask AWS CLICommand Reference.

`start-task`

Il seguente esempio di codice mostra come utilizzare`start-task`.

AWS CLI

Esempio 1: come avviare una nuova attività

L'esempio `start-task` seguente avvia un'attività utilizzando la revisione più recente della definizione di attività `sleep360` sull'istanza di container specificata nel cluster predefinito.

```
aws ecs start-task \
    --task-definition sleep360 \
    --container-instances 765936fadbdd46b5991a4bd70c2a43d4
```

Output:

```
{
    "tasks": [
        {
            "taskArn": "arn:aws:ecs:us-west-2:123456789012:task/
default/666fdccc2e2d4b6894dd422f4eeee8f8",
            "clusterArn": "arn:aws:ecs:us-west-2:123456789012:cluster/default",
            "taskDefinitionArn": "arn:aws:ecs:us-west-2:123456789012:task-
definition/sleep360:3",
            "containerInstanceArn": "arn:aws:ecs:us-west-2:123456789012:container-
instance/default/765936fadbdd46b5991a4bd70c2a43d4",
            "overrides": {
                "containerOverrides": [
                    {
                        "name": "sleep"
                    }
                ]
            },
            "lastStatus": "PENDING",
            "desiredStatus": "RUNNING",
            "cpu": "128",
            "memory": "128",
            "containers": [
                {
                    "containerArn": "arn:aws:ecs:us-
west-2:123456789012:container/75f11ed4-8a3d-4f26-a33b-ad1db9e02d41",
                    "taskArn": "arn:aws:ecs:us-west-2:123456789012:task/
default/666fdccc2e2d4b6894dd422f4eeee8f8",
                    "name": "sleep",
                    "lastStatus": "PENDING",
                    "networkInterfaces": [],
```

```
                    "cpu": "10",
                    "memory": "10"
                }
            ],
            "version": 1,
            "createdAt": 1563421494.186,
            "group": "family:sleep360",
            "launchType": "EC2",
            "attachments": [],
            "tags": []
        }
    ],
    "failures": []
}
```

Per ulteriori informazioni, consulta Pianificare i container su Amazon ECS nella Guida per gli sviluppatori di Amazon ECS.

Esempio 2: come configurare un volume Amazon EBS per un'attività all'avvio dell'attività

L'esempio `start-task` seguente configura un volume Amazon EBS crittografato per un'attività nell'istanza di container specificata. È necessario disporre di un ruolo dell'infrastruttura Amazon ECS configurato con associata la policy gestita da `AmazonECSInfrastructureRolePolicyForVolumes`. È necessario specificare una definizione di attività con lo stesso nome di volume della richiesta `start-task`. Questo esempio utilizza l'opzione `--cli-input-json` e un file di input JSON denominato `ebs.json` con il seguente contenuto.

```
aws ecs start-task \
    --cli-input-json file://ebs.json \
    --container-instances 765936fadbdd46b5991a4bd70c2a43d4
```

Contenuto di `ebs.json`:

```
{
   "cluster": "default",
   "taskDefinition": "mytaskdef",
   "networkConfiguration":{
        "awsvpcConfiguration":{
            "assignPublicIp": "ENABLED",
            "securityGroups": ["sg-12344321"],
            "subnets":["subnet-12344321"]
```

```
        }
    },
   "volumeConfigurations": [
        {
            "name": "myEBSVolume",
            "managedEBSVolume": {
                "volumeType": "gp3",
                "sizeInGiB": 100,
                "roleArn":"arn:aws:iam::123456789012:role/ecsInfrastructureRole",
                "encrypted": true,
                "kmsKeyId":
 "arn:aws:kms:region:123456789012:key/1234abcd-12ab-34cd-56ef-1234567890ab"
            }
        }
    ]
}
```

Output:

```
{
    "tasks": [
        {
            "attachments": [
                {
                    "id": "aea29489-9dcd-49f1-8164-4d91566e1113",
                    "type": "ElasticNetworkInterface",
                    "status": "PRECREATED",
                    "details": [
                        {
                            "name": "subnetId",
                            "value": "subnet-12344321"
                        }
                    ]
                },
                {
                    "id": "f29e1222-9a1e-410f-b499-a12a7cd6d42e",
                    "type": "AmazonElasticBlockStorage",
                    "status": "CREATED",
                    "details": [
                        {
                            "name": "roleArn",
                            "value": "arn:aws:iam::123456789012:role/
ecsInfrastructureRole"
```

```
                        },
                        {
                            "name": "volumeName",
                            "value": "myEBSVolume"
                        },
                        {
                            "name": "deleteOnTermination",
                            "value": "true"
                        }
                    ]
                }
            ],
            "attributes": [
                {
                    "name": "ecs.cpu-architecture",
                    "value": "arm64"
                }
            ],
            "availabilityZone": "us-west-2c",
            "clusterArn": "arn:aws:ecs:us-west-2:123456789012:cluster/default",
            "containerInstanceArn": "arn:aws:ecs:us-west-2:123456789012:container-
instance/default/765936fadbdd46b5991a4bd70c2a43d4",
            "containers": [
                {
                    "containerArn": "arn:aws:ecs:us-west-2:123456789012:container/
default/bb122ace3ed84add92c00a351a03c69e/a4a9ed10-51c7-4567-9653-50e71b94f867",
                    "taskArn": "arn:aws:ecs:us-west-2:123456789012:task/default/
bb122ace3ed84add92c00a351a03c69e",
                    "name": "container-using-ebs",
                    "image": "amazonlinux:2",
                    "lastStatus": "PENDING",
                    "networkInterfaces": [],
                    "cpu": "0"
                }
            ],
            "cpu": "1024",
            "createdAt": "2025-01-23T14:51:05.191000-06:00",
            "desiredStatus": "RUNNING",
            "enableExecuteCommand": false,
            "group": "family:mytaskdef",
            "lastStatus": "PROVISIONING",
            "launchType": "EC2",
            "memory": "3072",
            "overrides": {
```

```
                "containerOverrides": [
                    {
                        "name": "container-using-ebs"
                    }
                ],
                "inferenceAcceleratorOverrides": []
            },
             "tags": [],
            "taskArn": "arn:aws:ecs:us-west-2:123456789012:task/default/
bb122ace3ed84add92c00a351a03c69e",
            "taskDefinitionArn": "arn:aws:ecs:us-west-2:123456789012:task-
definition/mytaskdef:4",
            "version": 1
        }
    ],
    "failures": []
}
```

Per ulteriori informazioni, consulta Utilizzare i volumi Amazon EBS con Amazon ECS nella Guida per gli sviluppatori di Amazon ECS.

- Per i dettagli sull'API, consulta StartTask AWS CLICommand Reference.

`stop-task`

Il seguente esempio di codice mostra come utilizzare`stop-task`.

AWS CLI

Come arrestare un'attività

Il comando `stop-task` seguente arresta l'esecuzione dell'attività specificata nel cluster predefinito.

```
aws ecs stop-task \
    --task 666fdccc2e2d4b6894dd422f4eeee8f8
```

Output:

```
{
    "task": {
        "taskArn": "arn:aws:ecs:us-west-2:130757420319:task/
default/666fdccc2e2d4b6894dd422f4eeee8f8",
```

```
        "clusterArn": "arn:aws:ecs:us-west-2:130757420319:cluster/default",
        "taskDefinitionArn": "arn:aws:ecs:us-west-2:130757420319:task-definition/
sleep360:3",
        "containerInstanceArn": "arn:aws:ecs:us-west-2:130757420319:container-
instance/default/765936fadbdd46b5991a4bd70c2a43d4",
        "overrides": {
            "containerOverrides": []
        },
        "lastStatus": "STOPPED",
        "desiredStatus": "STOPPED",
        "cpu": "128",
        "memory": "128",
        "containers": [],
        "version": 2,
        "stoppedReason": "Taskfailedtostart",
        "stopCode": "TaskFailedToStart",
        "connectivity": "CONNECTED",
        "connectivityAt": 1563421494.186,
        "pullStartedAt": 1563421494.252,
        "pullStoppedAt": 1563421496.252,
        "executionStoppedAt": 1563421497,
        "createdAt": 1563421494.186,
        "stoppingAt": 1563421497.252,
        "stoppedAt": 1563421497.252,
        "group": "family:sleep360",
        "launchType": "EC2",
        "attachments": [],
        "tags": []
    }
}
```

- Per i dettagli sull'API, consulta StopTask AWS CLICommand Reference.

`tag-resource`

Il seguente esempio di codice mostra come utilizzare`tag-resource`.

AWS CLI

Come taggare a una risorsa

L'esempio `tag-resource` seguente aggiunge un singolo tag alla risorsa specificata.

```
aws ecs tag-resource \
    --resource-arn arn:aws:ecs:us-west-2:123456789012:cluster/MyCluster
    --tags key=key1,value=value1
```

Questo comando non produce alcun output.

Come aggiungere più tag a una risorsa

L'esempio `tag-resource` seguente aggiunge più tag alla risorsa specificata.

```
aws ecs tag-resource \
--resource-arn arn:aws:ecs:us-west-2:123456789012:cluster/MyCluster \
--tags key=key1,value=value1 key=key2,value=value2 key=key3,value=value3
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta TagResource AWS CLICommand Reference.

`untag-resource`

Il seguente esempio di codice mostra come utilizzare`untag-resource`.

AWS CLI

Come rimuovere un tag da una risorsa

L'esempio `untag-resource` seguente rimuove i tag elencati dalla risorsa specificata.

```
aws ecs untag-resource \
    --resource-arn arn:aws:ecs:us-west-2:123456789012:cluster/MyCluster \
    --tag-keys key1,key2
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta UntagResource AWS CLICommand Reference.

`update-cluster-settings`

Il seguente esempio di codice mostra come utilizzare`update-cluster-settings`.

AWS CLI

Come modificare le impostazioni del cluster

L'`update-cluster-settings`esempio seguente abilita CloudWatch Container Insights con una migliore osservabilità per il `MyCluster` cluster.

```
aws ecs update-cluster-settings \
    --cluster MyCluster \
    --settings name=containerInsights,value=enhanced
```

Output:

```
{
    "cluster": {
        "clusterArn": "arn:aws:ecs:us-esat-1:123456789012:cluster/MyCluster",
        "clusterName": "default",
        "status": "ACTIVE",
        "registeredContainerInstancesCount": 0,
        "runningTasksCount": 0,
        "pendingTasksCount": 0,
        "activeServicesCount": 0,
        "statistics": [],
        "tags": [],
        "settings": [
            {
                "name": "containerInsights",
                "value": "enhanced"
            }
        ]
    }
}
```

Per ulteriori informazioni, consulta Modifica delle impostazioni dell'account nella Guida per gli sviluppatori di Amazon ECS.

- Per i dettagli sull'API, consulta UpdateClusterSettings AWS CLICommand Reference.

`update-cluster`

Il seguente esempio di codice mostra come utilizzare`update-cluster`.

AWS CLI

Esempio 1: aggiorna il cluster ECS abilitando containerInsights

Il comando `update-cluster` seguente aggiorna il valore containerInsights impostandolo su `enabled` in un cluster già creato. Per impostazione predefinita, è disabilitato.

```
aws ecs update-cluster \
    --cluster ECS-project-update-cluster \
    --settings name=containerInsights,value=enabled
```

Output:

```
"cluster": {
    "clusterArn": "arn:aws:ecs:us-west-2:123456789012:cluster/ECS-project-update-
cluster",
    "clusterName": "ECS-project-update-cluster",
    "status": "ACTIVE",
    "registeredContainerInstancesCount": 0,
    "runningTasksCount": 0,
    "pendingTasksCount": 0,
    "activeServicesCount": 0,
    "statistics": [],
    "tags": [],
    "settings": [
        {
            "name": "containerInsights",
            "value": "enabled"
        }
    ],
    "capacityProviders": [
        "Infra-ECS-Cluster-ECS-project-update-cluster-d6bb6d5b-
EC2CapacityProvider-3fIpdkLywwFt"
    ],
    "defaultCapacityProviderStrategy": [
        {
            "capacityProvider": "Infra-ECS-Cluster-ECS-project-update-cluster-
d6bb6d5b-EC2CapacityProvider-3fIpdkLywwFt",
            "weight": 1,
            "base": 0
        }
    ],
    "attachments": [
        {
            "id": "069d002b-7634-42e4-b1d4-544f4c8f6380",
            "type": "as_policy",
            "status": "CREATED",
```

```
            "details": [
                {
                    "name": "capacityProviderName",
                    "value": "Infra-ECS-Cluster-ECS-project-update-cluster-d6bb6d5b-
EC2CapacityProvider-3fIpdkLywwFt"
                },
                {
                    "name": "scalingPolicyName",
                    "value": "ECSManagedAutoScalingPolicy-152363a6-8c65-484c-
b721-42c3e070ae93"
                }
            ]
        },
        {
            "id": "08b5b6ca-45e9-4209-a65d-e962a27c490a",
            "type": "managed_draining",
            "status": "CREATED",
            "details": [
                {
                    "name": "capacityProviderName",
                    "value": "Infra-ECS-Cluster-ECS-project-update-cluster-d6bb6d5b-
EC2CapacityProvider-3fIpdkLywwFt"
                },
                {
                    "name": "autoScalingLifecycleHookName",
                    "value": "ecs-managed-draining-termination-hook"
                }
            ]
        },
        {
            "id": "45d0b36f-8cff-46b6-9380-1288744802ab",
            "type": "sc",
            "status": "ATTACHED",
            "details": []
        }
    ],
    "attachmentsStatus": "UPDATE_COMPLETE",
    "serviceConnectDefaults": {
        "namespace": "arn:aws:servicediscovery:us-west-2:123456789012:namespace/ns-
igwrsylmy3kwvcdx"
    }
}
```

Esempio 2: aggiorna il cluster ECS per impostare un namespace Service Connect predefinito

Il comando `update-cluster` aggiorna il cluster ECS impostando un namespace Service Connect predefinito.

```
aws ecs update-cluster \
    --cluster ECS-project-update-cluster \
    --service-connect-defaults namespace=test
```

Output:

```
 {
     "cluster": {
         "clusterArn": "arn:aws:ecs:us-west-2:123456789012:cluster/ECS-project-
update-cluster",
         "clusterName": "ECS-project-update-cluster",
         "status": "ACTIVE",
         "registeredContainerInstancesCount": 0,
         "runningTasksCount": 0,
         "pendingTasksCount": 0,
         "activeServicesCount": 0,
         "statistics": [],
         "tags": [],
         "settings": [
             {
                 "name": "containerInsights",
                 "value": "enabled"
             }
         ],
         "capacityProviders": [
             "Infra-ECS-Cluster-ECS-project-update-cluster-d6bb6d5b-
EC2CapacityProvider-3fIpdkLywwFt"
         ],
         "defaultCapacityProviderStrategy": [
             {
                 "capacityProvider": "Infra-ECS-Cluster-ECS-project-update-cluster-
d6bb6d5b-EC2CapacityProvider-3fIpdkLywwFt",
                 "weight": 1,
                 "base": 0
             }
         ],
         "attachments": [
             {
```

```
                "id": "069d002b-7634-42e4-b1d4-544f4c8f6380",
                "type": "as_policy",
                "status": "CREATED",
                "details": [
                    {
                        "name": "capacityProviderName",
                        "value": "Infra-ECS-Cluster-ECS-project-update-cluster-
d6bb6d5b-EC2CapacityProvider-3fIpdkLywwFt"
                    },
                    {
                        "name": "scalingPolicyName",
                        "value": "ECSManagedAutoScalingPolicy-152363a6-8c65-484c-
b721-42c3e070ae93"
                    }
                ]
            },
            {
                "id": "08b5b6ca-45e9-4209-a65d-e962a27c490a",
                "type": "managed_draining",
                "status": "CREATED",
                "details": [
                    {
                        "name": "capacityProviderName",
                        "value": "Infra-ECS-Cluster-ECS-project-update-cluster-
d6bb6d5b-EC2CapacityProvider-3fIpdkLywwFt"
                    },
                    {
                        "name": "autoScalingLifecycleHookName",
                        "value": "ecs-managed-draining-termination-hook"
                    }
                ]
            },
            {
                "id": "45d0b36f-8cff-46b6-9380-1288744802ab",
                "type": "sc",
                "status": "DELETED",
                "details": []
            },
            {
                "id": "3e6890c3-609c-4832-91de-d6ca891b3ef1",
                "type": "sc",
                "status": "ATTACHED",
                "details": []
            },
```

```
            {
                "id": "961b8ec1-c2f1-4070-8495-e669b7668e90",
                "type": "sc",
                "status": "DELETED",
                "details": []
            }
        ],
        "attachmentsStatus": "UPDATE_COMPLETE",
        "serviceConnectDefaults": {
            "namespace": "arn:aws:servicediscovery:us-
west-2:123456789012:namespace/ns-dtjmxqpfi46ht7dr"
        }
    }
}
```

Per ulteriori informazioni su Service Connect, consulta Utilizzare Service Connect per connettere i servizi Amazon ECS con nomi brevi nella Guida per gli sviluppatori di Amazon ECS.

- Per i dettagli sull'API, consulta UpdateCluster AWS CLICommand Reference.

`update-container-agent`

Il seguente esempio di codice mostra come utilizzare`update-container-agent`.

AWS CLI

Come aggiornare l'agente del container in un'istanza di container Amazon ECS

L'esempio `update-container-agent` seguente aggiorna l'agente del container nell'istanza di container specificata nel cluster predefinito.

```
aws ecs update-container-agent --cluster default --container-
instance a1b2c3d4-5678-90ab-cdef-11111EXAMPLE
```

Output:

```
{
    "containerInstance": {
        "status": "ACTIVE",
...
        "agentUpdateStatus": "PENDING",
        "versionInfo": {
            "agentVersion": "1.0.0",
```

```
            "agentHash": "4023248",
            "dockerVersion": "DockerVersion: 1.5.0"
        }
    }
}
```

Per ulteriori informazioni, consulta Aggiornamento dell'agente del container di Amazon ECS nella Guida per gli sviluppatori di Amazon ECS.

- Per i dettagli sull'API, consulta UpdateContainerAgent AWS CLICommand Reference.

`update-container-instances-state`

Il seguente esempio di codice mostra come utilizzare`update-container-instances-state`.

AWS CLI

Come aggiornare lo stato di un'istanza di container

Il comando `update-container-instances-state` seguente aggiorna lo stato dell'istanza di container specificata su `DRAINING` per rimuoverla dal cluster in cui è registrata.

```
aws ecs update-container-instances-state \
    --container-instances 765936fadbdd46b5991a4bd70c2a43d4 \
    --status DRAINING
```

Output:

```
{
    "containerInstances": [
        {
            "containerInstanceArn": "arn:aws:ecs:us-west-2:130757420319:container-
instance/default/765936fadbdd46b5991a4bd70c2a43d4",
            "ec2InstanceId": "i-013d87ffbb4d513bf",
            "version": 4390,
            "versionInfo": {
                "agentVersion": "1.29.0",
                "agentHash": "a190a73f",
                "dockerVersion": "DockerVersion:18.06.1-ce"
            },
            "remainingResources": [
                {
                    "name": "CPU",
```

```
                    "type": "INTEGER",
                    "doubleValue": 0,
                    "longValue": 0,
                    "integerValue": 1536
                },
                {
                    "name": "MEMORY",
                    "type": "INTEGER",
                    "doubleValue": 0,
                    "longValue": 0,
                    "integerValue": 2681
                },
                {
                    "name": "PORTS",
                    "type": "STRINGSET",
                    "doubleValue": 0,
                    "longValue": 0,
                    "integerValue": 0,
                    "stringSetValue": [
                        "22",
                        "2376",
                        "2375",
                        "51678",
                        "51679"
                    ]
                },
                {
                    "name": "PORTS_UDP",
                    "type": "STRINGSET",
                    "doubleValue": 0,
                    "longValue": 0,
                    "integerValue": 0,
                    "stringSetValue": []
                }
            ],
            "registeredResources": [
                {
                    "name": "CPU",
                    "type": "INTEGER",
                    "doubleValue": 0,
                    "longValue": 0,
                    "integerValue": 2048
                },
                {
```

```
                    "name": "MEMORY",
                    "type": "INTEGER",
                    "doubleValue": 0,
                    "longValue": 0,
                    "integerValue": 3705
                },
                {
                    "name": "PORTS",
                    "type": "STRINGSET",
                    "doubleValue": 0,
                    "longValue": 0,
                    "integerValue": 0,
                    "stringSetValue": [
                        "22",
                        "2376",
                        "2375",
                        "51678",
                        "51679"
                    ]
                },
                {
                    "name": "PORTS_UDP",
                    "type": "STRINGSET",
                    "doubleValue": 0,
                    "longValue": 0,
                    "integerValue": 0,
                    "stringSetValue": []
                }
            ],
            "status": "DRAINING",
            "agentConnected": true,
            "runningTasksCount": 2,
            "pendingTasksCount": 0,
            "attributes": [
                {
                    "name": "ecs.capability.secrets.asm.environment-variables"
                },
                {
                    "name": "ecs.capability.branch-cni-plugin-version",
                    "value": "e0703516-"
                },
                {
                    "name": "ecs.ami-id",
                    "value": "ami-00e0090ac21971297"
```

```
                },
                {
                    "name": "ecs.capability.secrets.asm.bootstrap.log-driver"
                },
                {
                    "name": "com.amazonaws.ecs.capability.logging-driver.none"
                },
                {
                    "name": "ecs.capability.ecr-endpoint"
                },
                {
                    "name": "ecs.capability.docker-plugin.local"
                },
                {
                    "name": "ecs.capability.task-cpu-mem-limit"
                },
                {
                    "name": "ecs.capability.secrets.ssm.bootstrap.log-driver"
                },
                {
                    "name": "com.amazonaws.ecs.capability.docker-remote-api.1.30"
                },
                {
                    "name": "com.amazonaws.ecs.capability.docker-remote-api.1.31"
                },
                {
                    "name": "com.amazonaws.ecs.capability.docker-remote-api.1.32"
                },
                {
                    "name": "ecs.availability-zone",
                    "value": "us-west-2c"
                },
                {
                    "name": "ecs.capability.aws-appmesh"
                },
                {
                    "name": "com.amazonaws.ecs.capability.logging-driver.awslogs"
                },
                {
                    "name": "com.amazonaws.ecs.capability.docker-remote-api.1.24"
                },
                {
                    "name": "ecs.capability.task-eni-trunking"
                },
```

```
                {
                    "name": "com.amazonaws.ecs.capability.docker-remote-api.1.25"
                },
                {
                    "name": "com.amazonaws.ecs.capability.docker-remote-api.1.26"
                },
                {
                    "name": "com.amazonaws.ecs.capability.docker-remote-api.1.27"
                },
                {
                    "name": "com.amazonaws.ecs.capability.docker-remote-api.1.28"
                },
                {
                    "name": "com.amazonaws.ecs.capability.privileged-container"
                },
                {
                    "name": "com.amazonaws.ecs.capability.docker-remote-api.1.29"
                },
                {
                    "name": "ecs.cpu-architecture",
                    "value": "x86_64"
                },
                {
                    "name": "com.amazonaws.ecs.capability.ecr-auth"
                },
                {
                    "name": "com.amazonaws.ecs.capability.docker-remote-api.1.20"
                },
                {
                    "name": "ecs.os-type",
                    "value": "linux"
                },
                {
                    "name": "com.amazonaws.ecs.capability.docker-remote-api.1.21"
                },
                {
                    "name": "com.amazonaws.ecs.capability.docker-remote-api.1.22"
                },
                {
                    "name": "ecs.capability.task-eia"
                },
                {
                    "name": "com.amazonaws.ecs.capability.docker-remote-api.1.23"
                },
```

```
                {
                    "name": "ecs.capability.private-registry-
authentication.secretsmanager"
                },
                {
                    "name": "com.amazonaws.ecs.capability.logging-driver.syslog"
                },
                {
                    "name": "com.amazonaws.ecs.capability.logging-driver.json-file"
                },
                {
                    "name": "ecs.capability.execution-role-awslogs"
                },
                {
                    "name": "ecs.vpc-id",
                    "value": "vpc-1234"
                },
                {
                    "name": "com.amazonaws.ecs.capability.docker-remote-api.1.17"
                },
                {
                    "name": "com.amazonaws.ecs.capability.docker-remote-api.1.18"
                },
                {
                    "name": "com.amazonaws.ecs.capability.docker-remote-api.1.19"
                },
                {
                    "name": "ecs.capability.task-eni"
                },
                {
                    "name": "ecs.capability.execution-role-ecr-pull"
                },
                {
                    "name": "ecs.capability.container-health-check"
                },
                {
                    "name": "ecs.subnet-id",
                    "value": "subnet-1234"
                },
                {
                    "name": "ecs.instance-type",
                    "value": "c5.large"
                },
                {
```

```
                    "name": "com.amazonaws.ecs.capability.task-iam-role-network-
host"
                },
                {
                    "name": "ecs.capability.container-ordering"
                },
                {
                    "name": "ecs.capability.cni-plugin-version",
                    "value": "91ccefc8-2019.06.0"
                },
                {
                    "name": "ecs.capability.pid-ipc-namespace-sharing"
                },
                {
                    "name": "ecs.capability.secrets.ssm.environment-variables"
                },
                {
                    "name": "com.amazonaws.ecs.capability.task-iam-role"
                }
            ],
            "registeredAt": 1560788724.507,
            "attachments": [],
            "tags": []
        }
    ],
    "failures": []
}
```

- Per i dettagli sull'API, consulta UpdateContainerInstancesState AWS CLICommand Reference.

`update-service-primary-task-set`

Il seguente esempio di codice mostra come utilizzare`update-service-primary-task-set`.

AWS CLI

Come aggiornare il set di attività principale per un servizio

L'esempio `update-service-primary-task-set` seguente aggiorna il set di attività principale per il servizio specificato.

```
aws ecs update-service-primary-task-set \
    --cluster MyCluster \
```

```
    --service MyService \
    --primary-task-set arn:aws:ecs:us-west-2:123456789012:task-set/MyCluster/
MyService/ecs-svc/1234567890123456789
```

Output:

```
{
    "taskSet": {
        "id": "ecs-svc/1234567890123456789",
        "taskSetArn": "arn:aws:ecs:us-west-2:123456789012:task-set/MyCluster/
MyService/ecs-svc/1234567890123456789",
        "status": "PRIMARY",
        "taskDefinition": "arn:aws:ecs:us-west-2:123456789012:task-definition/
sample-fargate:2",
        "computedDesiredCount": 1,
        "pendingCount": 0,
        "runningCount": 0,
        "createdAt": 1557128360.711,
        "updatedAt": 1557129412.653,
        "launchType": "EC2",
        "networkConfiguration": {
            "awsvpcConfiguration": {
                "subnets": [
                    "subnet-12344321"
                ],
                "securityGroups": [
                    "sg-12344312"
                ],
                "assignPublicIp": "DISABLED"
            }
        },
        "loadBalancers": [],
        "serviceRegistries": [],
        "scale": {
            "value": 50.0,
            "unit": "PERCENT"
        },
        "stabilityStatus": "STABILIZING",
        "stabilityStatusAt": 1557129279.914
    }
}
```

- Per i dettagli sull'API, consulta UpdateServicePrimaryTaskSet AWS CLICommand Reference.

update-service

Il seguente esempio di codice mostra come utilizzareupdate-service.

AWS CLI

Esempio 1: come modificare la definizione dell'attività utilizzata in un servizio

L'esempio update-service seguente aggiorna il servizio my-http-service in modo da utilizzare la definizione dell'attività amazon-ecs-sample.

```
aws ecs update-service \
    --cluster test \
    --service my-http-service \
    --task-definition amazon-ecs-sample
```

Output:

```
{
    "service": {
        "serviceArn": "arn:aws:ecs:us-west-2:123456789012:service/test/my-http-
service",
        "serviceName": "my-http-service",
        "clusterArn": "arn:aws:ecs:us-west-2:123456789012:cluster/test",
        "loadBalancers": [],
        "serviceRegistries": [],
        "status": "ACTIVE",
        "desiredCount": 2,
        "runningCount": 2,
        "pendingCount": 0,
        "launchType": "FARGATE",
        "platformVersion": "1.4.0",
        "platformFamily": "Linux",
        "taskDefinition": "arn:aws:ecs:us-west-2:123456789012:task-definition/
amazon-ecs-sample:2",
        "deploymentConfiguration": {
            "deploymentCircuitBreaker": {
                "enable": true,
                "rollback": true
            },
            "maximumPercent": 200,
            "minimumHealthyPercent": 100,
            "alarms": {
```

```
                "alarmNames": [],
                "rollback": false,
                "enable": false
            }
        },
        "deployments": [
            {
                "id": "ecs-svc/7419115625193919142",
                "status": "PRIMARY",
                "taskDefinition": "arn:aws:ecs:us-west-2:123456789012:task-
definition/amazon-ecs-sample:2",
                "desiredCount": 0,
                "pendingCount": 0,
                "runningCount": 0,
                "failedTasks": 0,
                "createdAt": "2025-02-21T13:26:02.734000-06:00",
                "updatedAt": "2025-02-21T13:26:02.734000-06:00",
                "launchType": "FARGATE",
                "platformVersion": "1.4.0",
                "platformFamily": "Linux",
                "networkConfiguration": {
                    "awsvpcConfiguration": {
                        "subnets": [
                            "subnet-12344321"
                        ],
                        "securityGroups": [
                            "sg-12344321"
                        ],
                        "assignPublicIp": "ENABLED"
                    }
                },
                "rolloutState": "IN_PROGRESS",
                "rolloutStateReason": "ECS deployment ecs-svc/7419115625193919142 in
 progress."
            },
            {
                "id": "ecs-svc/1709597507655421668",
                "status": "ACTIVE",
                "taskDefinition": "arn:aws:ecs:us-west-2:123456789012:task-
definition/old-amazon-ecs-sample:4",
                "desiredCount": 2,
                "pendingCount": 0,
                "runningCount": 2,
                "failedTasks": 0,
```

```
                "createdAt": "2025-01-24T11:13:07.621000-06:00",
                "updatedAt": "2025-02-02T16:11:30.838000-06:00",
                "launchType": "FARGATE",
                "platformVersion": "1.4.0",
                "platformFamily": "Linux",
                "networkConfiguration": {
                    "awsvpcConfiguration": {
                        "subnets": [
                            "subnet-12344321"
                        ],
                        "securityGroups": [
                             "sg-12344321"
                        ],
                        "assignPublicIp": "ENABLED"
                    }
                },
                "rolloutState": "COMPLETED",
                "rolloutStateReason": "ECS deployment ecs-svc/1709597507655421668
 completed."
            }
        ],
        "roleArn": "arn:aws:iam::123456789012:role/aws-service-role/
ecs.amazonaws.com/AWSServiceRoleForECS",
        "events": [
            {
                "id": "e40b4d1c-80d9-4834-aaf3-6a268e530e17",
                "createdAt": "2025-02-21T10:31:26.037000-06:00",
                "message": "(my-http-service) has reached a steady state."
            },
            {
                "id": "6ac069ad-fc8b-4e49-a35d-b5574a964c8e",
                "createdAt": "2025-02-21T04:31:22.703000-06:00",
                "message": "(my-http-service) has reached a steady state."
            },
            {
                "id": "265f7d37-dfd1-4880-a846-ec486f341919",
                "createdAt": "2025-02-20T22:31:22.514000-06:00",
                "message": "(my-http-service) has reached a steady state."
            }
        ],
        "createdAt": "2024-10-30T17:12:43.218000-05:00",
        "placementConstraints": [],
        "placementStrategy": [],
        "networkConfiguration": {
```

```
            "awsvpcConfiguration": {
                "subnets": [
                    "subnet-12344321",
                ],
                "securityGroups": [
                    "sg-12344321"
                ],
                "assignPublicIp": "ENABLED"
            }
        },
        "healthCheckGracePeriodSeconds": 0,
        "schedulingStrategy": "REPLICA",
        "deploymentController": {
            "type": "ECS"
        },
        "createdBy": "arn:aws:iam::123456789012:role/AIDACKCEVSQ6C2EXAMPLE",
        "enableECSManagedTags": true,
        "propagateTags": "NONE",
        "enableExecuteCommand": false,
        "availabilityZoneRebalancing": "DISABLED"
    }
}
```

Per ulteriori informazioni, consulta Aggiornare un servizio Amazon ECS utilizzando la console nella Guida per gli sviluppatori di Amazon ECS.

Esempio 2: come modificare il numero di attività in un servizio

L'esempio `update-service` seguente aggiorna il numero di attività desiderato del servizio `my-http-service` impostandolo su 2.

```
aws ecs update-service \
    --cluster MyCluster \
    --service my-http-service \
    --desired-count 2
```

Output:

```
{
    "service": {
        "serviceArn": "arn:aws:ecs:us-east-1:123456789012:service/MyCluster/my-http-
service",
        "serviceName": "my-http-service",
```

```
        "clusterArn": "arn:aws:ecs:us-east-1:123456789012:cluster/MyCluster",
        "loadBalancers": [],
        "serviceRegistries": [],
        "status": "ACTIVE",
        "desiredCount": 2,
        "runningCount": 1,
        "pendingCount": 0,
        "capacityProviderStrategy": [
            {
                "capacityProvider": "FARGATE",
                "weight": 1,
                "base": 0
            }
        ],
        "platformVersion": "LATEST",
        "platformFamily": "Linux",
        "taskDefinition": "arn:aws:ecs:us-east-1:123456789012:task-definition/
MyTaskDefinition",
        "deploymentConfiguration": {
            "deploymentCircuitBreaker": {
                "enable": true,
                "rollback": true
            },
            "maximumPercent": 200,
            "minimumHealthyPercent": 100,
            "alarms": {
                "alarmNames": [],
                "rollback": false,
                "enable": false
            }
        },
        "deployments": [
            {
                "id": "ecs-svc/1976744184940610707",
                "status": "PRIMARY",
                "taskkDefinition": "arn:aws:ecs:us-east-1:123456789012:task-
definition/MyTaskDefinition",
                "desiredCount": 1,
                "pendingCount": 0,
                "runningCount": 1,
                "failedTasks": 0,
                "createdAt": "2024-12-03T16:24:25.225000-05:00",
                "updatedAt": "2024-12-03T16:25:15.837000-05:00",
                "capacityProviderStrategy": [
```

```
                    {
                        "capacityProvider": "FARGATE",
                        "weight": 1,
                        "base": 0
                    }
                ],
                "platformVersion": "1.4.0",
                "platformFamily": "Linux",
                "networkConfiguration": {
                    "awsvpcConfiguration": {
                        "subnets": [
                            "subnet-0d0eab1bb38d5ca64",
                            "subnet-0db5010045995c2d5"
                        ],
                        "securityGroups": [
                            "sg-02556bf85a191f59a"
                        ],
                        "assignPublicIp": "ENABLED"
                    }
                },
                "rolloutState": "COMPLETED",
                "rolloutStateReason": "ECS deployment ecs-svc/1976744184940610707
 completed."
            }
        ],
        "roleArn": "arn:aws:iam::123456789012:role/aws-service-role/
ecs.amazonaws.com/AWSServiceRoleForECS",
        "events": [
            {
                "id": "f27350b9-4b2a-4e2e-b72e-a4b68380de45",
                "createdAt": "2024-12-30T13:24:07.345000-05:00",
                "message": "(service my-http-service) has reached a steady state."
            },
            {
                "id": "e764ec63-f53f-45e3-9af2-d99f922d2957",
                "createdAt": "2024-12-30T12:32:21.600000-05:00",
                "message": "(service my-http-service) has reached a steady state."
            },
            {
                "id": "28444756-c2fa-47f8-bd60-93a8e05f3991",
                "createdAt": "2024-12-08T19:26:10.367000-05:00",
                "message": "(service my-http-service) has reached a steady state."
            }
        ],
```

```
        "createdAt": "2024-12-03T16:24:25.225000-05:00",
        "placementConstraints": [],
        "placementStrategy": [],
        "networkConfiguration": {
            "awsvpcConfiguration": {
                "subnets": [
                    "subnet-0d0eab1bb38d5ca64",
                    "subnet-0db5010045995c2d5"
                ],
                "securityGroups": [
                    "sg-02556bf85a191f59a"
                ],
                "assignPublicIp": "ENABLED"
            }
        },
        "healthCheckGracePeriodSeconds": 0,
        "schedulingStrategy": "REPLICA",
        "deploymentController": {
            "type": "ECS"
        },
        "createdBy": "arn:aws:iam::123456789012:role/Admin",
        "enableECSManagedTags": true,
        "propagateTags": "NONE",
        "enableExecuteCommand": false,
        "availabilityZoneRebalancing": "ENABLED"
    }
}
```

Per ulteriori informazioni, consulta Aggiornamento di un servizio Amazon ECS utilizzando la console nella Guida per gli sviluppatori di Amazon ECS.

Esempio 3: come configurare i volumi Amazon EBS per gli allegati durante l'aggiornamento del servizio

L'esempio `update-service` seguente aggiorna il servizio `my-http-service` in modo da utilizzare i volumi Amazon EBS. È necessario disporre di un ruolo dell'infrastruttura Amazon ECS configurato con associata la policy gestita da `AmazonECSInfrastructureRolePolicyForVolumes`. È inoltre necessario specificare una definizione di attività con lo stesso nome di volume della richiesta `update-service` e con `configuredAtLaunch` impostato su `true`. Questo esempio utilizza l'opzione `--cli-input-json` e un file di input JSON denominato `ebs.json`.

```
aws ecs update-service \
    --cli-input-json file://ebs.json
```

Contenuto di `ebs.json`:

```
{
   "cluster": "mycluster",
   "taskDefinition": "mytaskdef",
   "service": "my-http-service",
   "desiredCount": 2,
   "volumeConfigurations": [
        {
            "name": "myEbsVolume",
            "managedEBSVolume": {
                "roleArn":"arn:aws:iam::123456789012:role/ecsInfrastructureRole",
                "volumeType": "gp3",
                "sizeInGiB": 100,
                "iops": 3000,
                "throughput": 125,
                "filesystemType": "ext4"
            }
        }
   ]
}
```

Output:

```
{
    "service": {
        "serviceArn": "arn:aws:ecs:us-west-2:123456789012:service/mycluster/my-http-
service",
        "serviceName": "my-http-service",
        "clusterArn": "arn:aws:ecs:us-west-2:123456789012:cluster/mycluster",
        "loadBalancers": [],
        "serviceRegistries": [],
        "status": "ACTIVE",
        "desiredCount": 2,
        "runningCount": 2,
        "pendingCount": 0,
        "launchType": "FARGATE",
        "platformVersion": "LATEST",
        "platformFamily": "Linux",
```

```
        "taskDefinition": "arn:aws:ecs:us-west-2:123456789012:task-definition/
mytaskdef:1",
        "deploymentConfiguration": {
            "deploymentCircuitBreaker": {
                "enable": true,
                "rollback": true
            },
            "maximumPercent": 200,
            "minimumHealthyPercent": 100,
            "alarms": {
                "alarmNames": [],
                "rollback": false,
                "enable": false
            }
        },
        "deployments": [
            {
                "id": "ecs-svc/2420458347226626275",
                "status": "PRIMARY",
                "taskDefinition": "arn:aws:ecs:us-west-2:123456789012:task-
definition/mytaskdef:1",
                "desiredCount": 0,
                "pendingCount": 0,
                "runningCount": 0,
                "failedTasks": 0,
                "createdAt": "2025-02-21T15:07:20.519000-06:00",
                "updatedAt": "2025-02-21T15:07:20.519000-06:00",
                "launchType": "FARGATE",
                "platformVersion": "1.4.0",
                "platformFamily": "Linux",
                "networkConfiguration": {
                    "awsvpcConfiguration": {
                        "subnets": [
                            "subnet-12344321",
                        ],
                        "securityGroups": [
                            "sg-12344321"
                        ],
                        "assignPublicIp": "ENABLED"
                    }
                },
                "rolloutState": "IN_PROGRESS",
                "rolloutStateReason": "ECS deployment ecs-svc/2420458347226626275 in
 progress.",
```

```
                "volumeConfigurations": [
                    {
                        "name": "ebs-volume",
                        "managedEBSVolume": {
                            "volumeType": "gp3",
                            "sizeInGiB": 100,
                            "iops": 3000,
                            "throughput": 125,
                            "roleArn": "arn:aws:iam::123456789012:role/
ecsInfrastructureRole",
                            "filesystemType": "ext4"
                        }
                    }
                ]
            },
            {
                "id": "ecs-svc/5191625155316533644",
                "status": "ACTIVE",
                "taskDefinition": "arn:aws:ecs:us-west-2:123456789012:task-
definition/mytaskdef:2",
                "desiredCount": 2,
                "pendingCount": 0,
                "runningCount": 2,
                "failedTasks": 0,
                "createdAt": "2025-02-21T14:54:48.862000-06:00",
                "updatedAt": "2025-02-21T14:57:22.502000-06:00",
                "launchType": "FARGATE",
                "platformVersion": "1.4.0",
                "platformFamily": "Linux",
                "networkConfiguration": {
                    "awsvpcConfiguration": {
                        "subnets": [
                            "subnet-12344321"
                        ],
                        "securityGroups": [
                            "sg-12344321"
                        ],
                        "assignPublicIp": "ENABLED"
                    }
                },
                "rolloutState": "COMPLETED",
                "rolloutStateReason": "ECS deployment ecs-svc/5191625155316533644
 completed."
            }
```

```
        ],
        "roleArn": "arn:aws:iam::123456789012:role/aws-service-role/
ecs.amazonaws.com/AWSServiceRoleForECS",
        "events": [
            {
                "id": "b5823113-c2c5-458e-9649-8c2ed38f23a5",
                "createdAt": "2025-02-21T14:57:22.508000-06:00",
                "message": "(service my-http-service) has reached a steady state."
            },
            {
                "id": "b05a48e8-da35-4074-80aa-37ceb3167357",
                "createdAt": "2025-02-21T14:57:22.507000-06:00",
                "message": "(service my-http-service) (deployment ecs-
svc/5191625155316533644) deployment completed."
            },
            {
                "id": "a10cd55d-4ba6-4cea-a655-5a5d32ada8a0",
                "createdAt": "2025-02-21T14:55:32.833000-06:00",
                "message": "(service my-http-service) has started 1 tasks: (task
 fb9c8df512684aec92f3c57dc3f22361)."
            },
        ],
        "createdAt": "2025-02-21T14:54:48.862000-06:00",
        "placementConstraints": [],
        "placementStrategy": [],
        "networkConfiguration": {
            "awsvpcConfiguration": {
                "subnets": [
                    "subnet-12344321"
                ],
                "securityGroups": [
                    "sg-12344321"
                ],
                "assignPublicIp": "ENABLED"
            }
        },
        "healthCheckGracePeriodSeconds": 0,
        "schedulingStrategy": "REPLICA",
        "deploymentController": {
            "type": "ECS"
        },
        "createdBy": "arn:aws:iam::123456789012:role/AIDACKCEVSQ6C2EXAMPLE",
        "enableECSManagedTags": true,
        "propagateTags": "NONE",
```

```
        "enableExecuteCommand": false,
        "availabilityZoneRebalancing": "ENABLED"
    }
}
```

Per ulteriori informazioni, consulta Utilizzare i volumi Amazon EBS con Amazon ECS nella Guida per gli sviluppatori di Amazon ECS.

Esempio 4: come aggiornare un servizio in modo che non utilizzi più i volumi Amazon EBS

L'esempio `update-service` seguente aggiorna il servizio `my-http-service` in modo da non utilizzare più i volumi Amazon EBS. È necessario specificare una revisione della definizione di attività con `configuredAtLaunch` impostato su `false`.

```
aws ecs update-service \
    --cluster mycluster \
    --task-definition mytaskdef \
    --service my-http-service \
    --desired-count 2 \
    --volume-configurations "[]"
```

Output:

```
{
    "service": {
        "serviceArn": "arn:aws:ecs:us-west-2:123456789012:service/mycluster/my-http-
service",
        "serviceName": "my-http-service",
        "clusterArn": "arn:aws:ecs:us-west-2:123456789012:cluster/mycluster",
        "loadBalancers": [],
        "serviceRegistries": [],
        "status": "ACTIVE",
        "desiredCount": 2,
        "runningCount": 2,
        "pendingCount": 0,
        "launchType": "FARGATE",
        "platformVersion": "LATEST",
        "platformFamily": "Linux",
        "taskDefinition": "arn:aws:ecs:us-west-2:123456789012:task-definition/
mytaskdef:3",
        "deploymentConfiguration": {
            "deploymentCircuitBreaker": {
                "enable": true,
```

```
                "rollback": true
            },
            "maximumPercent": 200,
            "minimumHealthyPercent": 100,
            "alarms": {
                "alarmNames": [],
                "rollback": false,
                "enable": false
            }
        },
        "deployments": [
            {
                "id": "ecs-svc/7522791612543716777",
                "status": "PRIMARY",
                "taskDefinition": "arn:aws:ecs:us-west-2:123456789012:task-
definition/mytaskdef:3",
                "desiredCount": 0,
                "pendingCount": 0,
                "runningCount": 0,
                "failedTasks": 0,
                "createdAt": "2025-02-21T15:25:38.598000-06:00",
                "updatedAt": "2025-02-21T15:25:38.598000-06:00",
                    "launchType": "FARGATE",
                "platformVersion": "1.4.0",
                "platformFamily": "Linux",
                "networkConfiguration": {
                    "awsvpcConfiguration": {
                        "subnets": [
                            "subnet-12344321"
                        ],
                        "securityGroups": [
                            "sg-12344321"
                        ],
                        "assignPublicIp": "ENABLED"
                    }
                },
                "rolloutState": "IN_PROGRESS",
                "rolloutStateReason": "ECS deployment ecs-svc/7522791612543716777 in
 progress."
            },
            {
                "id": "ecs-svc/2420458347226626275",
                "status": "ACTIVE",
```

```
                "taskDefinition": "arn:aws:ecs:us-west-2:123456789012:task-
definition/myoldtaskdef:1",
                "desiredCount": 2,
                "pendingCount": 0,
                "runningCount": 2,
                "failedTasks": 0,
                "createdAt": "2025-02-21T15:07:20.519000-06:00",
                "updatedAt": "2025-02-21T15:10:59.955000-06:00",
                "launchType": "FARGATE",
                "platformVersion": "1.4.0",
                "platformFamily": "Linux",
                "networkConfiguration": {
                    "awsvpcConfiguration": {
                        "subnets": [
                            "subnet-12344321"
                        ],
                        "securityGroups": [
                            "sg-12344321"
                        ],
                        "assignPublicIp": "ENABLED"
                    }
                },
                "rolloutState": "COMPLETED",
                "rolloutStateReason": "ECS deployment ecs-svc/2420458347226626275
 completed.",
                "volumeConfigurations": [
                    {
                        "name": "ebs-volume",
                        "managedEBSVolume": {
                            "volumeType": "gp3",
                            "sizeInGiB": 100,
                            "iops": 3000,
                            "throughput": 125,
                            "roleArn": "arn:aws:iam::123456789012:role/
ecsInfrastructureRole",
                            "filesystemType": "ext4"
                        }
                    }
                ]
            }
        ],
        "roleArn": "arn:aws:iam::123456789012:role/aws-service-role/
ecs.amazonaws.com/AWSServiceRoleForECS",
        "events": [
```

```
            {
                "id": "4f2c3ca1-7800-4048-ba57-bba210ada2ad",
                "createdAt": "2025-02-21T15:10:59.959000-06:00",
                "message": "(service my-http-service) has reached a steady state."
            },
            {
                "id": "4b36a593-2d40-4ed6-8be8-b9b699eb6198",
                "createdAt": "2025-02-21T15:10:59.958000-06:00",
                "message": "(service my-http-service) (deployment ecs-
svc/2420458347226626275) deployment completed."
            },
            {
                "id": "88380089-14e2-4ef0-8dbb-a33991683371",
                "createdAt": "2025-02-21T15:09:39.055000-06:00",
                "message": "(service my-http-service) has stopped 1 running tasks:
 (task fb9c8df512684aec92f3c57dc3f22361)."
            },
            {
                "id": "97d84243-d52f-4255-89bb-9311391c61f6",
                "createdAt": "2025-02-21T15:08:57.653000-06:00",
                "message": "(service my-http-service) has stopped 1 running tasks:
 (task 33eff090ad2c40539daa837e6503a9bc)."
            },
            {
                "id": "672ece6c-e2d0-4021-b5da-eefb14001687",
                "createdAt": "2025-02-21T15:08:15.631000-06:00",
                "message": "(service my-http-service) has started 1 tasks: (task
 996c02a66ff24f3190a4a8e0c841740f)."
            },
            {
                "id": "a3cf9bea-9be6-4175-ac28-4c68360986eb",
                "createdAt": "2025-02-21T15:07:36.931000-06:00",
                "message": "(service my-http-service) has started 1 tasks: (task
 d5d23c39f89e46cf9a647b9cc6572feb)."
            },
            {
                "id": "b5823113-c2c5-458e-9649-8c2ed38f23a5",
                "createdAt": "2025-02-21T14:57:22.508000-06:00",
                "message": "(service my-http-service) has reached a steady state."
            },
            {
                "id": "b05a48e8-da35-4074-80aa-37ceb3167357",
                "createdAt": "2025-02-21T14:57:22.507000-06:00",
```

```
                "message": "(service my-http-service) (deployment ecs-
svc/5191625155316533644) deployment completed."
            },
            {
                "id": "a10cd55d-4ba6-4cea-a655-5a5d32ada8a0",
                "createdAt": "2025-02-21T14:55:32.833000-06:00",
                "message": "(service my-http-service) has started 1 tasks: (task
 fb9c8df512684aec92f3c57dc3f22361)."
            },
            {
                "id": "42da91fa-e26d-42ef-88c3-bb5965c56b2f",
                "createdAt": "2025-02-21T14:55:02.703000-06:00",
                "message": "(service my-http-service) has started 1 tasks: (task
 33eff090ad2c40539daa837e6503a9bc)."
            }
        ],
        "createdAt": "2025-02-21T14:54:48.862000-06:00",
        "placementConstraints": [],
        "placementStrategy": [],
        "networkConfiguration": {
            "awsvpcConfiguration": {
                "subnets": [
                    "subnet-12344321"
                ],
                "securityGroups": [
                    "sg-12344321"
                ],
                "assignPublicIp": "ENABLED"
            }
        },
        "healthCheckGracePeriodSeconds": 0,
        "schedulingStrategy": "REPLICA",
        "deploymentController": {
            "type": "ECS"
        },
        "createdBy": "arn:aws:iam::123456789012:role/AIDACKCEVSQ6C2EXAMPLE",
        "enableECSManagedTags": true,
        "propagateTags": "NONE",
        "enableExecuteCommand": false,
        "availabilityZoneRebalancing": "ENABLED"
    }
}
```

Per ulteriori informazioni, consulta Utilizzare i volumi Amazon EBS con Amazon ECS nella Guida per gli sviluppatori di Amazon ECS.

Esempio 5: come attivare il ribilanciamento delle zone di disponibilità per un servizio

L'esempio `update-service` seguente attiva il ribilanciamento delle zone di disponibilità per il servizio `my-http-service`.

```
aws ecs update-service \
    --cluster MyCluster \
    --service my-http-service \
    --availability-zone-rebalancing ENABLED
```

Output:

```
{
    "service": {
        "serviceArn": "arn:aws:ecs:us-east-1:123456789012:service/MyCluster/my-http-
service",
        "serviceName": "my-http-service",
        "clusterArn": "arn:aws:ecs:us-east-1:123456789012:cluster/MyCluster",
        "loadBalancers": [],
        "serviceRegistries": [],
        "status": "ACTIVE",
        "desiredCount": 2,
        "runningCount": 1,
        "pendingCount": 0,
        "capacityProviderStrategy": [
            {
                "capacityProvider": "FARGATE",
                "weight": 1,
                "base": 0
            }
        ],
        "platformVersion": "LATEST",
        "platformFamily": "Linux",
        "taskDefinition": "arn:aws:ecs:us-east-1:123456789012:task-definition/
MyTaskDefinition",
        "deploymentConfiguration": {
            "deploymentCircuitBreaker": {
                "enable": true,
                "rollback": true
            },
```

```
            "maximumPercent": 200,
            "minimumHealthyPercent": 100,
            "alarms": {
                "alarmNames": [],
                "rollback": false,
                "enable": false
            }
        },
        "deployments": [
            {
                "id": "ecs-svc/1976744184940610707",
                "status": "PRIMARY",
                "taskkDefinition": "arn:aws:ecs:us-east-1:123456789012:task-
definition/MyTaskDefinition",
                "desiredCount": 1,
                "pendingCount": 0,
                "runningCount": 1,
                "failedTasks": 0,
                "createdAt": "2024-12-03T16:24:25.225000-05:00",
                "updatedAt": "2024-12-03T16:25:15.837000-05:00",
                "capacityProviderStrategy": [
                    {
                        "capacityProvider": "FARGATE",
                        "weight": 1,
                        "base": 0
                    }
                ],
                "platformVersion": "1.4.0",
                "platformFamily": "Linux",
                "networkConfiguration": {
                    "awsvpcConfiguration": {
                        "subnets": [
                            "subnet-0d0eab1bb38d5ca64",
                            "subnet-0db5010045995c2d5"
                        ],
                        "securityGroups": [
                            "sg-02556bf85a191f59a"
                        ],
                        "assignPublicIp": "ENABLED"
                    }
                },
                "rolloutState": "COMPLETED",
                "rolloutStateReason": "ECS deployment ecs-svc/1976744184940610707
 completed."
```

```
            }
        ],
        "roleArn": "arn:aws:iam::123456789012:role/aws-service-role/
ecs.amazonaws.com/AWSServiceRoleForECS",
        "events": [],
        "createdAt": "2024-12-03T16:24:25.225000-05:00",
        "placementConstraints": [],
        "placementStrategy": [],
        "networkConfiguration": {
            "awsvpcConfiguration": {
                "subnets": [
                    "subnet-0d0eab1bb38d5ca64",
                    "subnet-0db5010045995c2d5"
                ],
                "securityGroups": [
                    "sg-02556bf85a191f59a"
                ],
                "assignPublicIp": "ENABLED"
            }
        },
        "healthCheckGracePeriodSeconds": 0,
        "schedulingStrategy": "REPLICA",
        "deploymentController": {
            "type": "ECS"
        },
        "createdBy": "arn:aws:iam::123456789012:role/Admin",
        "enableECSManagedTags": true,
        "propagateTags": "NONE",
        "enableExecuteCommand": false,
        "availabilityZoneRebalancing": "ENABLED"
    }
}
```

Per ulteriori informazioni, consulta Aggiornamento di un servizio Amazon ECS utilizzando la console nella Guida per gli sviluppatori di Amazon ECS.

- Per i dettagli sull'API, consulta UpdateService AWS CLICommand Reference.

`update-task-protection`

Il seguente esempio di codice mostra come utilizzareup`date-task-protection`.

AWS CLI

Esempio 1: abilita la protezione delle attività per le attività ECS

Quanto segue `update-task-protection` protegge l'attività ECS dall'interruzione durante la scalabilità da Deployments o Service. AutoScaling È possibile specificare un periodo di scadenza personalizzato per la protezione delle attività con intervallo da 1 a 2.880 minuti (48 ore). Se non si specifica il periodo di scadenza, il tempo predefinito di attivazione della protezione delle attività è di 2 ore.

```
aws ecs update-task-protection \
    --cluster ECS-project-update-cluster \
    --tasks c43ed3b1331041f289316f958adb6a24 \
    --protection-enabled \
    --expires-in-minutes 300
```

Output:

```
{
"protectedTasks": [
    {
        "taskArn": "arn:aws:ecs:us-west-2:123456789012:task/
c43ed3b1331041f289316f958adb6a24",
        "protectionEnabled": true,
        "expirationDate": "2024-09-14T19:53:36.687000-05:00"
    }
],
"failures": []
}
```

Esempio 2: disabilita la protezione delle attività per le attività ECS

Quanto segue `update-task-protection` disabilita le attività protette dalla scalabilità di Deployments o Service. AutoScaling

```
aws ecs update-task-protection \
    --cluster ECS-project-update-cluster \
    --tasks c43ed3b1331041f289316f958adb6a24 \
    --no-protection-enabled
```

Output:

```
{
    "protectedTasks": [
        {
            "taskArn": "arn:aws:ecs:us-west-2:123456789012:task/
c43ed3b1331041f289316f958adb6a24",
            "protectionEnabled": false
        }
    ],
    "failures": []
}
```

Per ulteriori informazioni sulla protezione delle attività, consulta Proteggere le attività di Amazon ECS dall'interruzione causata da eventi di riduzione orizzontale nella Guida per gli sviluppatori Amazon ECS.

- Per i dettagli sull'API, consulta Command UpdateTaskProtectionReference AWS CLI .

`update-task-set`

Il seguente esempio di codice mostra come utilizzare`update-task-set`.

AWS CLI

Come aggiornare un set di attività

L'esempio `update-task-set` seguente aggiorna un set di attività per regolare il dimensionamento.

```
aws ecs update-task-set \
    --cluster MyCluster \
    --service MyService \
    --task-set arn:aws:ecs:us-west-2:123456789012:task-set/MyCluster/MyService/ecs-
svc/1234567890123456789 \
    --scale value=50,unit=PERCENT
```

Output:

```
{
    "taskSet": {
        "id": "ecs-svc/1234567890123456789",
        "taskSetArn": "arn:aws:ecs:us-west-2:123456789012:task-set/MyCluster/
MyService/ecs-svc/1234567890123456789",
```

```
        "status": "ACTIVE",
        "taskDefinition": "arn:aws:ecs:us-west-2:123456789012:task-definition/
sample-fargate:2",
        "computedDesiredCount": 0,
        "pendingCount": 0,
        "runningCount": 0,
        "createdAt": 1557128360.711,
        "updatedAt": 1557129279.914,
        "launchType": "EC2",
        "networkConfiguration": {
            "awsvpcConfiguration": {
                "subnets": [
                    "subnet-12344321"
                ],
                "securityGroups": [
                    "sg-12344321"
                ],
                "assignPublicIp": "DISABLED"
            }
        },
        "loadBalancers": [],
        "serviceRegistries": [],
        "scale": {
            "value": 50.0,
            "unit": "PERCENT"
        },
        "stabilityStatus": "STABILIZING",
        "stabilityStatusAt": 1557129279.914
    }
}
```

- Per i dettagli sull'API, consulta UpdateTaskSet AWS CLICommand Reference.

Esempi di utilizzo di Amazon EFS AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando Amazon EFS. AWS Command Line Interface

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`create-file-system`

Il seguente esempio di codice mostra come usare`create-file-system`.

AWS CLI

Creare un file system crittografato

L'esempio `create-file-system` seguente crea un file system crittografato utilizzando la chiave gestita dal cliente (CMK) predefinita. Aggiunge anche il tag `Name=my-file-system`.

```
aws efs create-file-system \
    --performance-mode generalPurpose \
    --throughput-mode bursting \
    --encrypted \
    --tags Key=Name,Value=my-file-system
```

Output:

```
{
    "OwnerId": "123456789012",
    "CreationToken": "console-d7f56c5f-e433-41ca-8307-9d9c0example",
    "FileSystemId": "fs-c7a0456e",
    "FileSystemArn": "arn:aws:elasticfilesystem:us-west-2:123456789012:file-system/
fs-48499b4d",
    "CreationTime": 1595286880.0,
    "LifeCycleState": "creating",
    "Name": "my-file-system",
    "NumberOfMountTargets": 0,
    "SizeInBytes": {
        "Value": 0,
        "ValueInIA": 0,
        "ValueInStandard": 0
```

```
    },
    "PerformanceMode": "generalPurpose",
    "Encrypted": true,
    "KmsKeyId": "arn:aws:kms:us-west-2:123456789012:key/a59b3472-e62c-42e4-
adcf-30d92example",
    "ThroughputMode": "bursting",
    "Tags": [
        {
            "Key": "Name",
            "Value": "my-file-system"
        }
    ]
}
```

Per ulteriori informazioni, consulta Creazione di file system Amazon EFS nella Guida per l'utente di Amazon Elastic File System.

- Per i dettagli sull'API, consulta CreateFileSystem AWS CLICommand Reference.

`create-mount-target`

Il seguente esempio di codice mostra come utilizzare`create-mount-target`.

AWS CLI

Come creare una destinazione di montaggio

L'esempio `create-mount-target` seguente crea una destinazione di montaggio per il file system specificato.

```
aws efs create-mount-target \
    --file-system-id fs-c7a0456e \
    --subnet-id subnet-02bf4c428bexample \
    --security-groups sg-068f739363example
```

Output:

```
{
    "OwnerId": "123456789012",
    "MountTargetId": "fsmt-f9a14450",
    "FileSystemId": "fs-c7a0456e",
    "SubnetId": "subnet-02bf4c428bexample",
```

```
    "LifeCycleState": "creating",
    "IpAddress": "10.0.1.24",
    "NetworkInterfaceId": "eni-02d542216aexample",
    "AvailabilityZoneId": "use2-az2",
    "AvailabilityZoneName": "us-east-2b",
    "VpcId": "vpc-0123456789abcdef0"
}
```

Per ulteriori informazioni, consulta Creazione di destinazioni di montaggio nella Guida per l'utente di Amazon Elastic File System.

- Per i dettagli sull'API, consulta CreateMountTarget AWS CLICommand Reference.

`delete-file-system`

Il seguente esempio di codice mostra come utilizzare`delete-file-system`.

AWS CLI

Come eliminare un file system

L'esempio `delete-file-system` seguente elimina il file system specificato.

```
aws efs delete-file-system \
    --file-system-id fs-c7a0456e
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Cancellazione di un file system Amazon EFS nella Guida per l'utente di Amazon Elastic File System.

- Per i dettagli sull'API, consulta DeleteFileSystem AWS CLICommand Reference.

`delete-mount-target`

Il seguente esempio di codice mostra come utilizzare`delete-mount-target`.

AWS CLI

Come eliminare una destinazione di montaggio

L'esempio `delete-mount-target` seguente elimina la destinazione di montaggio specificata.

```
aws efs delete-mount-target \
    --mount-target-id fsmt-f9a14450
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Creazione di destinazioni di montaggio nella Guida per l'utente di Amazon Elastic File System.

- Per i dettagli sull'API, consulta DeleteMountTarget AWS CLICommand Reference.

`describe-file-systems`

Il seguente esempio di codice mostra come utilizzare`describe-file-systems`.

AWS CLI

Come descrivere un file system

L'esempio `describe-file-systems` seguente descrive il file system specificato.

```
aws efs describe-file-systems \
    --file-system-id fs-c7a0456e
```

Output:

```
{
    "FileSystems": [
        {
            "OwnerId": "123456789012",
            "CreationToken": "console-d7f56c5f-e433-41ca-8307-9d9c0example",
            "FileSystemId": "fs-c7a0456e",
            "FileSystemArn": "arn:aws:elasticfilesystem:us-west-2:123456789012:file-
system/fs-48499b4d",
            "CreationTime": 1595286880.0,
            "LifeCycleState": "available",
            "Name": "my-file-system",
            "NumberOfMountTargets": 3,
            "SizeInBytes": {
                "Value": 6144,
                "Timestamp": 1600991437.0,
                "ValueInIA": 0,
```

```
                "ValueInStandard": 6144
            },
            "PerformanceMode": "generalPurpose",
            "Encrypted": true,
            "KmsKeyId": "arn:aws:kms:us-west-2:123456789012:key/a59b3472-e62c-42e4-
adcf-30d92example",
            "ThroughputMode": "bursting",
            "Tags": [
                {
                    "Key": "Name",
                    "Value": "my-file-system"
                }
            ]
        }
    ]
}
```

Per ulteriori informazioni, consulta Gestione dei file system Amazon EFS nella Guida per l'utente di Amazon Elastic File System.

- Per i dettagli sull'API, consulta DescribeFileSystems AWS CLICommand Reference.

`describe-mount-targets`

Il seguente esempio di codice mostra come utilizzare`describe-mount-targets`.

AWS CLI

Come descrivere una destinazione di montaggio

L'esempio `describe-mount-targets` seguente descrive la destinazione di montaggio specificata.

```
aws efs describe-mount-targets \
    --mount-target-id fsmt-f9a14450
```

Output:

```
{
    "MountTargets": [
        {
            "OwnerId": "123456789012",
```

```
            "MountTargetId": "fsmt-f9a14450",
            "FileSystemId": "fs-c7a0456e",
            "SubnetId": "subnet-02bf4c428bexample",
            "LifeCycleState": "creating",
            "IpAddress": "10.0.1.24",
            "NetworkInterfaceId": "eni-02d542216aexample",
            "AvailabilityZoneId": "use2-az2",
            "AvailabilityZoneName": "us-east-2b",
            "VpcId": "vpc-0123456789abcdef0"
        }
    ]
}
```

Per ulteriori informazioni, consulta Creazione di destinazioni di montaggio nella Guida per l'utente di Amazon Elastic File System.

- Per i dettagli sull'API, consulta DescribeMountTargets AWS CLICommand Reference.

`describe-tags`

Il seguente esempio di codice mostra come utilizzare`describe-tags`.

AWS CLI

Come descrivere i tag di un file system

L'esempio `describe-tags` seguente descrive i tag per il file system specificato.

```
aws efs describe-tags \
    --file-system-id fs-c7a0456e
```

Output:

```
{
    "Tags": [
        {
            "Key": "Name",
            "Value": "my-file-system"
        },
        {
            "Key": "Department",
            "Value": "Business Intelligence"
```

```
        }
    ]
}
```

Per ulteriori informazioni, consulta Gestione dei tag del file system nella Guida per l'utente di Amazon Elastic File System.

- Per i dettagli sull'API, consulta DescribeTags AWS CLICommand Reference.

`list-tags-for-resource`

Il seguente esempio di codice mostra come utilizzare`list-tags-for-resource`.

AWS CLI

Come elencare i tag per una risorsa

L'esempio `list-tags-for-resource` seguente recupera i tag associati al file system specificato.

```
aws efs list-tags-for-resource \
    --resource-id fs-c7a0456e
```

Output:

```
{
    "Tags": [
        {
            "Key": "Name",
            "Value": "my-file-system"
        },
        {
            "Key": "Department",
            "Value": "Business Intelligence"
        }
    ]
}
```

Per ulteriori informazioni, consulta Gestione dei tag del file system nella Guida per l'utente di Amazon Elastic File System.

- Per i dettagli sull'API, consulta ListTagsForResource AWS CLICommand Reference.

`tag-resource`

Il seguente esempio di codice mostra come utilizzare`tag-resource`.

AWS CLI

Come taggare a una risorsa

L'esempio `tag-resource` seguente aggiunge il tag `Department=Business Intelligence` al file system specificato.

```
aws efs tag-resource \
    --resource-id fs-c7a0456e \
    --tags Key=Department,Value="Business Intelligence"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Gestione dei tag del file system nella Guida per l'utente di Amazon Elastic File System.

- Per i dettagli sull'API, consulta TagResource AWS CLICommand Reference.

`untag-resource`

Il seguente esempio di codice mostra come utilizzare`untag-resource`.

AWS CLI

Come rimuovere un tag da una risorsa

L'esempio `untag-resource` seguente rimuove un tag con la chiave `Department` dal file system specificato.

```
aws efs untag-resource \
    --resource-id fs-c7a0456e \
    --tag-keys Department
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Gestione dei tag del file system nella Guida per l'utente di Amazon Elastic File System.

- Per i dettagli sull'API, consulta UntagResource AWS CLICommand Reference.

Esempi di utilizzo di Amazon EKS AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface con Amazon EKS.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`associate-access-policy`

Il seguente esempio di codice mostra come usare`associate-access-policy`.

AWS CLI

Come associare una policy di accesso e il relativo ambito alla voce di accesso del cluster

Il comando `associate-access-policy` seguente associa una policy di accesso e il relativo ambito alla voce di accesso del cluster specificato.

```
aws eks associate-access-policy \
    --cluster-name eks-customer \
    --principal-arn arn:aws:iam::111122223333:role/Admin \
    --policy-arn arn:aws:eks::aws:cluster-access-policy/AmazonEKSEditPolicy \
    --access-scope type=namespace,namespaces=default
```

Output:

```
{
    "clusterName": "eks-customer",
    "principalArn": "arn:aws:iam::111122223333:role/Admin",
    "associatedAccessPolicy": {
        "policyArn": "arn:aws:eks::aws:cluster-access-policy/AmazonEKSEditPolicy",
```

```
        "accessScope": {
            "type": "namespace",
            "namespaces": [
                "default"
            ]
        },
        "associatedAt": "2025-05-24T15:59:51.981000-05:00",
        "modifiedAt": "2025-05-24T15:59:51.981000-05:00"
    }
}
```

Per ulteriori informazioni, consulta Associare policy di accesso a voci di accesso nella Guida per l'utente di Amazon EKS.

- Per i dettagli sull'API, consulta AssociateAccessPolicy AWS CLICommand Reference.

`associate-encryption-config`

Il seguente esempio di codice mostra come utilizzare`associate-encryption-config`.

AWS CLI

Come associare una configurazione di crittografia a un cluster esistente

L'esempio `associate-encryption-config` seguente abilita la crittografia su un cluster EKS esistente in cui la crittografia non è già abilitata.

```
aws eks associate-encryption-config \
    --cluster-name my-eks-cluster \
    --encryption-config '[{"resources":["secrets"],"provider":
{"keyArn":"arn:aws:kms:region-code:account:key/key"}}]'
```

Output:

```
{
    "update": {
        "id": "3141b835-8103-423a-8e68-12c2521ffa4d",
        "status": "InProgress",
        "type": "AssociateEncryptionConfig",
        "params": [
            {
                "type": "EncryptionConfig",
```

```
                "value": "[{\"resources\":[\"secrets\"],\"provider\":{\"keyArn\":
\"arn:aws:kms:region-code:account:key/key\"}}]"
            }
        ],
        "createdAt": "2024-03-14T11:01:26.297000-04:00",
        "errors": []
    }
}
```

Per ulteriori informazioni, consulta Abilitazione della crittografia segreta dei dati in transito su un cluster esistente nella Guida per l'utente di Amazon EKS.

- Per i dettagli sull'API, consulta AssociateEncryptionConfig AWS CLICommand Reference.

`associate-identity-provider-config`

Il seguente esempio di codice mostra come utilizzare`associate-identity-provider-config`.

AWS CLI

Come associare un gestore dell'identità al cluster Amazon EKS

L'esempio `associate-identity-provider-config` seguente associa un gestore dell'identità al cluster Amazon EKS in uso.

```
aws eks associate-identity-provider-config \
    --cluster-name my-eks-cluster \
    --oidc 'identityProviderConfigName=my-identity-provider,issuerUrl=https://
oidc.eks.us-east-2.amazonaws.com/
id/38D6A4619A0A69E342B113ED7F1A7652,clientId=kubernetes,usernameClaim=email,usernamePrefix=m
username-prefix,groupsClaim=my-claim,groupsPrefix=my-groups-
prefix,requiredClaims={Claim1=value1,Claim2=value2}' \
    --tags env=dev
```

Output:

```
{
    "update": {
        "id": "8c6c1bef-61fe-42ac-a242-89412387b8e7",
        "status": "InProgress",
        "type": "AssociateIdentityProviderConfig",
        "params": [
            {
```

```
                "type": "IdentityProviderConfig",
                "value": "[{\"type\":\"oidc\",\"name\":\"my-identity-provider\"}]"
            }
        ],
        "createdAt": "2024-04-11T13:46:49.648000-04:00",
        "errors": []
    },
    "tags": {
        "env": "dev"
    }
}
```

Per ulteriori informazioni, consulta Autenticazione degli utenti per il cluster da un gestore dell'identità digitale OpenID Connect - Associazione di un provider di identità OIDC nella Guida per l'utente di Amazon EKS.

- Per i dettagli sull'API, consulta AssociateIdentityProviderConfig AWS CLICommand Reference.

`create-access-entry`

Il seguente esempio di codice mostra come utilizzare`create-access-entry`.

AWS CLI

Esempio 1: come creare la voce di accesso per il cluster EKS

L'esempio `create-access-entry` seguente crea una voce di accesso che consente a un principale IAM di accedere al cluster EKS.

```
aws eks create-access-entry \
    --cluster-name eks-customer \
    --principal-arn arn:aws:iam::111122223333:user/eks-user
```

Output:

```
{
    "accessEntry": {
        "clusterName": "eks-customer",
        "principalArn": "arn:aws:iam::111122223333:user/eks-user",
        "kubernetesGroups": [],
        "accessEntryArn": "arn:aws:eks:us-west-2:111122223333:access-entry/eks-
customer/user/111122223333/eks-user/a1b2c3d4-5678-90ab-cdef-a6506e3d36p0",
```

```
        "createdAt": "2025-04-14T22:45:48.097000-05:00",
        "modifiedAt": "2025-04-14T22:45:48.097000-05:00",
        "tags": {},
        "username": "arn:aws:iam::111122223333:user/eks-user",
        "type": "STANDARD"
    }
}
```

Per ulteriori informazioni, consulta Creare voci di accesso nella Guida per l'utente di Amazon EKS.

Esempio 2: come creare la voce di accesso per il cluster EKS specificandone il tipo

L'esempio `create-access-entry` seguente crea una voce di accesso di tipo `EC2_LINUX` nel cluster EKS. Per impostazione predefinita, viene creata una voce di accesso di tipo `STANDARD`. Oltre all'impostazione predefinita, se si specificano altri tipi di accesso, è necessario passare un ARN del ruolo IAM nella CLI.

```
aws eks create-access-entry \
    --cluster-name eks-customer \
    --principal-arn arn:aws:iam::111122223333:role/admin-test-ip \
    --type EC2_LINUX
```

Output:

```
{
    "accessEntry": {
        "clusterName": "eks-customer",
        "principalArn": "arn:aws:iam::111122223333:role/admin-test-ip",
        "kubernetesGroups": [
            "system:nodes"
        ],
        "accessEntryArn": "arn:aws:eks:us-west-2:111122223333:access-entry/eks-
customer/role/111122223333/admin-test-ip/accb5418-f493-f390-3e6e-c3f19f725fcp",
        "createdAt": "2025-05-06T19:42:45.453000-05:00",
        "modifiedAt": "2025-05-06T19:42:45.453000-05:00",
        "tags": {},
        "username": "system:node:{{EC2PrivateDNSName}}",
        "type": "EC2_LINUX"
    }
}
```

Per ulteriori informazioni, consulta Creare voci di accesso nella Guida per l'utente di Amazon EKS.

- Per i dettagli sull'API, consulta CreateAccessEntry AWS CLICommand Reference.

`create-addon`

Il seguente esempio di codice mostra come utilizzare`create-addon`.

AWS CLI

Esempio 1: come creare un componente aggiuntivo Amazon EKS con una versione compatibile predefinita per la rispettiva versione del cluster EKS

L'esempio `create-addon` seguente crea un componente aggiuntivo Amazon EKS con una versione compatibile predefinita per la rispettiva versione del cluster EKS.

```
aws eks create-addon \
    --cluster-name my-eks-cluster \
    --addon-name my-eks-addon \
    --service-account-role-arn arn:aws:iam::111122223333:role/role-name
```

Output:

```
{
    "addon": {
        "addonName": "my-eks-addon",
        "clusterName": "my-eks-cluster",
        "status": "CREATING",
        "addonVersion": "v1.15.1-eksbuild.1",
        "health": {
            "issues": []
        },
        "addonArn": "arn:aws:eks:us-east-2:111122223333:addon/my-eks-cluster/my-eks-
addon/1ec71ee1-b9c2-8915-4e17-e8be0a55a149",
        "createdAt": "2024-03-14T12:20:03.264000-04:00",
        "modifiedAt": "2024-03-14T12:20:03.283000-04:00",
        "serviceAccountRoleArn": "arn:aws:iam::111122223333:role/role-name",
        "tags": {}
    }
}
```

Per ulteriori informazioni, consulta Gestione dei componenti aggiuntivi di Amazon EKS - Creazione di un componente aggiuntivo nella Guida per l'utente di Amazon EKS.

Esempio 2: come creare un componente aggiuntivo Amazon EKS con una versione specifica

L'esempio `create-addon` seguente crea un componente aggiuntivo Amazon EKS con una versione specifica.

```
aws eks create-addon \
    --cluster-name my-eks-cluster \
    --addon-name my-eks-addon \
    --service-account-role-arn arn:aws:iam::111122223333:role/role-name \
    --addon-version v1.16.4-eksbuild.2
```

Output:

```
{
    "addon": {
        "addonName": "my-eks-addon",
        "clusterName": "my-eks-cluster",
        "status": "CREATING",
        "addonVersion": "v1.16.4-eksbuild.2",
        "health": {
            "issues": []
        },
        "addonArn": "arn:aws:eks:us-east-2:111122223333:addon/my-eks-cluster/my-eks-
addon/34c71ee6-7738-6c8b-c6bd-3921a176b5ff",
        "createdAt": "2024-03-14T12:30:24.507000-04:00",
        "modifiedAt": "2024-03-14T12:30:24.521000-04:00",
        "serviceAccountRoleArn": "arn:aws:iam::111122223333:role/role-name",
        "tags": {}
    }
}
```

Per ulteriori informazioni, consulta Gestione dei componenti aggiuntivi di Amazon EKS - Creazione di un componente aggiuntivo nella Guida per l'utente di Amazon EKS.

Esempio 3: come creare un componente aggiuntivo Amazon EKS con valori di configurazione personalizzati e risolvere i dettagli dei conflitti

L'esempio `create-addon` seguente crea un componente aggiuntivo Amazon EKS con valori di configurazione personalizzati e risolve i dettagli dei conflitti.

```
aws eks create-addon \
    --cluster-name my-eks-cluster \
    --addon-name my-eks-addon \
    --service-account-role-arn arn:aws:iam::111122223333:role/role-name \
    --addon-version v1.16.4-eksbuild.2 \
    --configuration-values '{"resources":{"limits":{"cpu":"100m"}}}' \
    --resolve-conflicts OVERWRITE
```

Output:

```
{
    "addon": {
        "addonName": "my-eks-addon",
        "clusterName": "my-eks-cluster",
        "status": "CREATING",
        "addonVersion": "v1.16.4-eksbuild.2",
        "health": {
            "issues": []
        },
        "addonArn": "arn:aws:eks:us-east-2:111122223333:addon/my-eks-cluster/my-eks-
addon/a6c71ee9-0304-9237-1be8-25af1b0f1ffb",
        "createdAt": "2024-03-14T12:35:58.313000-04:00",
        "modifiedAt": "2024-03-14T12:35:58.327000-04:00",
        "serviceAccountRoleArn": "arn:aws:iam::111122223333:role/role-name",
        "tags": {},
        "configurationValues": "{\"resources\":{\"limits\":{\"cpu\":\"100m\"}}}"
    }
}
```

Per ulteriori informazioni, consulta Gestione dei componenti aggiuntivi di Amazon EKS - Creazione di un componente aggiuntivo nella Guida per l'utente di Amazon EKS.

Esempio 4: come creare un componente aggiuntivo Amazon EKS con un file di valori di configurazione JSON personalizzati

L'esempio `create-addon` seguente crea un componente aggiuntivo Amazon EKS con valori di configurazione personalizzati e risolve i dettagli dei conflitti.

```
aws eks create-addon \
    --cluster-name my-eks-cluster \
    --addon-name my-eks-addon \
```

```
    --service-account-role-arn arn:aws:iam::111122223333:role/role-name \
    --addon-version v1.16.4-eksbuild.2 \
    --configuration-values 'file://configuration-values.json' \
    --resolve-conflicts OVERWRITE \
    --tags '{"eks-addon-key-1": "value-1" , "eks-addon-key-2": "value-2"}'
```

Contenuto di `configuration-values.json`:

```
{
    "resources": {
        "limits": {
            "cpu": "150m"
        }
    },
    "env": {
        "AWS_VPC_K8S_CNI_LOGLEVEL": "ERROR"
    }
}
```

Output:

```
{
    "addon": {
        "addonName": "my-eks-addon",
        "clusterName": "my-eks-cluster",
        "status": "CREATING",
        "addonVersion": "v1.16.4-eksbuild.2",
        "health": {
            "issues": []
        },
        "addonArn": "arn:aws:eks:us-east-2:111122223333:addon/my-eks-cluster/my-eks-
addon/d8c71ef8-fbd8-07d0-fb32-6a7be19ececd",
        "createdAt": "2024-03-14T13:10:51.763000-04:00",
        "modifiedAt": "2024-03-14T13:10:51.777000-04:00",
        "serviceAccountRoleArn": "arn:aws:iam::111122223333:role/role-name",
        "tags": {
            "eks-addon-key-1": "value-1",
            "eks-addon-key-2": "value-2"
        },
        "configurationValues": "{\n    \"resources\": {\n        \"limits\":
 {\n            \"cpu\": \"150m\"\n        }\n    },\n    \"env\": {\n
 \"AWS_VPC_K8S_CNI_LOGLEVEL\": \"ERROR\"\n    }\n}"
    }
```

```
}
```

Per ulteriori informazioni, consulta Gestione dei componenti aggiuntivi di Amazon EKS - Creazione di un componente aggiuntivo nella Guida per l'utente di Amazon EKS.

Esempio 5: come creare un componente aggiuntivo Amazon EKS con un file di valori di configurazione YAML personalizzati

L'esempio `create-addon` seguente crea un componente aggiuntivo Amazon EKS con valori di configurazione personalizzati e risolve i dettagli dei conflitti.

```
aws eks create-addon \
    --cluster-name my-eks-cluster \
    --addon-name my-eks-addon \
    --service-account-role-arn arn:aws:iam::111122223333:role/role-name \
    --addon-version v1.16.4-eksbuild.2 \
    --configuration-values 'file://configuration-values.yaml' \
    --resolve-conflicts OVERWRITE \
    --tags '{"eks-addon-key-1": "value-1" , "eks-addon-key-2": "value-2"}'
```

Contenuto di `configuration-values.yaml`:

```
resources:
    limits:
        cpu: '100m'
env:
    AWS_VPC_K8S_CNI_LOGLEVEL: 'DEBUG'
```

Output:

```
{
    "addon": {
        "addonName": "my-eks-addon",
        "clusterName": "my-eks-cluster",
        "status": "CREATING",
        "addonVersion": "v1.16.4-eksbuild.2",
        "health": {
            "issues": []
        },
        "addonArn": "arn:aws:eks:us-east-2:111122223333:addon/my-eks-cluster/my-eks-
addon/d4c71efb-3909-6f36-a548-402cd4b5d59e",
        "createdAt": "2024-03-14T13:15:45.220000-04:00",
```

```
        "modifiedAt": "2024-03-14T13:15:45.237000-04:00",
        "serviceAccountRoleArn": "arn:aws:iam::111122223333:role/role-name",
        "tags": {
            "eks-addon-key-3": "value-3",
            "eks-addon-key-4": "value-4"
        },
        "configurationValues": "resources:\n    limits:\n        cpu: '100m'\nenv:\n
    AWS_VPC_K8S_CNI_LOGLEVEL: 'INFO'"
    }
}
```

Per ulteriori informazioni, consulta Gestione dei componenti aggiuntivi di Amazon EKS - Creazione di un componente aggiuntivo nella Guida per l'utente di Amazon EKS.

- Per i dettagli sull'API, consulta CreateAddon AWS CLICommand Reference.

`create-cluster`

Il seguente esempio di codice mostra come utilizzare`create-cluster`.

AWS CLI

Come creare un nuovo cluster

Questo comando di esempio crea un cluster denominato `prod` nella Regione predefinita.

Comando:

```
aws eks create-cluster --name prod \
--role-arn arn:aws:iam::012345678910:role/eks-service-role-
AWSServiceRoleForAmazonEKS-J70NKE3BQ4PI \
--resources-vpc-config subnetIds=subnet-6782e71e,subnet-
e7e761ac,securityGroupIds=sg-6979fe18
```

Output:

```
{
    "cluster": {
        "name": "prod",
        "arn": "arn:aws:eks:us-west-2:012345678910:cluster/prod",
        "createdAt": 1527808069.147,
        "version": "1.10",
```

```
        "roleArn": "arn:aws:iam::012345678910:role/eks-service-role-
AWSServiceRoleForAmazonEKS-J7ONKE3BQ4PI",
        "resourcesVpcConfig": {
            "subnetIds": [
                "subnet-6782e71e",
                "subnet-e7e761ac"
            ],
            "securityGroupIds": [
                "sg-6979fe18"
            ],
            "vpcId": "vpc-950809ec"
        },
        "status": "CREATING",
        "certificateAuthority": {}
    }
}
```

Come creare un nuovo cluster con accesso privato agli endpoint e registrazione dei log abilitati

Questo comando di esempio crea un cluster denominato `example` nell'area predefinita con l'accesso pubblico agli endpoint disabilitato, l'accesso privato agli endpoint abilitato e tutti i tipi di registrazione di log abilitati.

Comando:

```
aws eks create-cluster --name example --kubernetes-version 1.12 \
--role-arn arn:aws:iam::012345678910:role/example-cluster-ServiceRole-1XWBQWYSFRE2Q
 \
--resources-vpc-
config subnetIds=subnet-0a188dccd2f9a632f,subnet-09290d93da4278664,subnet-0f21dd86e0e91134a,
 \
--logging '{"clusterLogging":[{"types":
["api","audit","authenticator","controllerManager","scheduler"],"enabled":true}]}'
```

Output:

```
{
    "cluster": {
        "name": "example",
        "arn": "arn:aws:eks:us-west-2:012345678910:cluster/example",
        "createdAt": 1565804921.901,
        "version": "1.12",
```

```
        "roleArn": "arn:aws:iam::012345678910:role/example-cluster-
ServiceRole-1XWBQWYSFRE2Q",
        "resourcesVpcConfig": {
            "subnetIds": [
                "subnet-0a188dccd2f9a632f",
                "subnet-09290d93da4278664",
                "subnet-0f21dd86e0e91134a",
                "subnet-0173dead68481a583",
                "subnet-051f70a57ed6fcab6",
                "subnet-01322339c5c7de9b4"
            ],
            "securityGroupIds": [
                "sg-0c5b580845a031c10"
            ],
            "vpcId": "vpc-0f622c01f68d4afec",
            "endpointPublicAccess": false,
            "endpointPrivateAccess": true
        },
        "logging": {
            "clusterLogging": [
                {
                    "types": [
                        "api",
                        "audit",
                        "authenticator",
                        "controllerManager",
                        "scheduler"
                    ],
                    "enabled": true
                }
            ]
        },
        "status": "CREATING",
        "certificateAuthority": {},
        "platformVersion": "eks.3"
    }
}
```

- Per i dettagli sull'API, consulta CreateCluster AWS CLICommand Reference.

`create-fargate-profile`

Il seguente esempio di codice mostra come utilizzare`create-fargate-profile`.

AWS CLI

Esempio 1: crea il profilo EKS Fargate per un selettore con un namespace

L'esempio `create-fargate-profile` seguente crea un profilo EKS Fargate per un selettore con un namespace.

```
aws eks create-fargate-profile \
    --cluster-name my-eks-cluster \
    --pod-execution-role-arn arn:aws:iam::111122223333:role/role-name \
    --fargate-profile-name my-fargate-profile \
    --selectors '[{"namespace": "default"}]'
```

Output:

```
{
    "fargateProfile": {
        "fargateProfileName": "my-fargate-profile",
        "fargateProfileArn": "arn:aws:eks:us-east-2:111122223333:fargateprofile/my-
eks-cluster/my-fargate-profile/a2c72bca-318e-abe8-8ed1-27c6d4892e9e",
        "clusterName": "my-eks-cluster",
        "createdAt": "2024-03-19T12:38:47.368000-04:00",
        "podExecutionRoleArn": "arn:aws:iam::111122223333:role/role-name",
        "subnets": [
            "subnet-09d912bb63ef21b9a",
            "subnet-04ad87f71c6e5ab4d",
            "subnet-0e2907431c9988b72"
        ],
        "selectors": [
            {
                "namespace": "default"
            }
        ],
        "status": "CREATING",
        "tags": {}
    }
}
```

Per ulteriori informazioni, consulta Profilo AWS Fargate - Creazione di un profilo Fargate nella Guida per l'utente di Amazon EKS.

Esempio 2: crea il profilo EKS Fargate per un selettore con namespace ed etichette

L'esempio `create-fargate-profile` seguente crea un profilo EKS Fargate per un selettore con namespace ed etichette.

```
aws eks create-fargate-profile \
    --cluster-name my-eks-cluster \
    --pod-execution-role-arn arn:aws:iam::111122223333:role/role-name \
    --fargate-profile-name my-fargate-profile \
    --selectors '[{"namespace": "default", "labels": {"labelname1":
 "labelvalue1"}}]'
```

Output:

```
{
    "fargateProfile": {
        "fargateProfileName": "my-fargate-profile",
        "fargateProfileArn": "arn:aws:eks:us-east-2:111122223333:fargateprofile/my-
eks-cluster/my-fargate-profile/88c72bc7-e8a4-fa34-44e4-2f1397224bb3",
        "clusterName": "my-eks-cluster",
        "createdAt": "2024-03-19T12:33:48.125000-04:00",
        "podExecutionRoleArn": "arn:aws:iam::111122223333:role/role-name",
        "subnets": [
            "subnet-09d912bb63ef21b9a",
            "subnet-04ad87f71c6e5ab4d",
            "subnet-0e2907431c9988b72"
        ],
        "selectors": [
            {
                "namespace": "default",
                "labels": {
                    "labelname1": "labelvalue1"
                }
            }
        ],
        "status": "CREATING",
        "tags": {}
    }
}
```

Per ulteriori informazioni, consulta Profilo AWS Fargate - Creazione di un profilo Fargate nella Guida per l'utente di Amazon EKS.

Esempio 3: creazione del profilo EKS Fargate per un selettore con namespace ed etichette, insieme IDs alle sottoreti in cui lanciare un Pod.

L'`create-fargate-profile`esempio seguente crea il profilo EKS Fargate per un selettore con uno spazio dei nomi e delle etichette, insieme IDs alle sottoreti in cui lanciare un Pod.

```
aws eks create-fargate-profile \
    --cluster-name my-eks-cluster \
    --pod-execution-role-arn arn:aws:iam::111122223333:role/role-name \
    --fargate-profile-name my-fargate-profile \
    --selectors '[{"namespace": "default", "labels": {"labelname1":
 "labelvalue1"}}]' \
    --subnets '["subnet-09d912bb63ef21b9a", "subnet-04ad87f71c6e5ab4d",
 "subnet-0e2907431c9988b72"]'
```

Output:

```
{
    "fargateProfile": {
        "fargateProfileName": "my-fargate-profile",
        "fargateProfileArn": "arn:aws:eks:us-east-2:111122223333:fargateprofile/my-
eks-cluster/my-fargate-profile/e8c72bc8-e87b-5eb6-57cb-ed4fe57577e3",
        "clusterName": "my-eks-cluster",
        "createdAt": "2024-03-19T12:35:58.640000-04:00",
        "podExecutionRoleArn": "arn:aws:iam::111122223333:role/role-name",
        "subnets": [
            "subnet-09d912bb63ef21b9a",
            "subnet-04ad87f71c6e5ab4d",
            "subnet-0e2907431c9988b72"
        ],
        "selectors": [
            {
                "namespace": "default",
                "labels": {
                    "labelname1": "labelvalue1"
                }
            }
        ],
        "status": "CREATING",
        "tags": {}
    }
}
```

Per ulteriori informazioni, consulta Profilo AWS Fargate - Creazione di un profilo Fargate nella Guida per l'utente di Amazon EKS.

Esempio 4: creazione del profilo EKS Fargate per un selettore con più namespace ed etichette, insieme IDs a sottoreti in cui lanciare un Pod

L'`create-fargate-profile`esempio seguente crea un profilo EKS Fargate per un selettore con più namespace ed etichette, insieme IDs a sottoreti in cui lanciare un Pod.

```
aws eks create-fargate-profile \
    --cluster-name my-eks-cluster \
    --pod-execution-role-arn arn:aws:iam::111122223333:role/role-name \
    --fargate-profile-name my-fargate-profile \
    --selectors '[{"namespace": "default1", "labels": {"labelname1": "labelvalue1",
 "labelname2": "labelvalue2"}}, {"namespace": "default2", "labels": {"labelname1":
 "labelvalue1", "labelname2": "labelvalue2"}}]' \
    --subnets '["subnet-09d912bb63ef21b9a", "subnet-04ad87f71c6e5ab4d",
 "subnet-0e2907431c9988b72"]' \
    --tags '{"eks-fargate-profile-key-1": "value-1" , "eks-fargate-profile-key-2":
 "value-2"}'
```

Output:

```
{
    "fargateProfile": {
        "fargateProfileName": "my-fargate-profile",
        "fargateProfileArn": "arn:aws:eks:us-east-2:111122223333:fargateprofile/my-
eks-cluster/my-fargate-profile/4cc72bbf-b766-8ee6-8d29-e62748feb3cd",
        "clusterName": "my-eks-cluster",
        "createdAt": "2024-03-19T12:15:55.271000-04:00",
        "podExecutionRoleArn": "arn:aws:iam::111122223333:role/role-name",
        "subnets": [
            "subnet-09d912bb63ef21b9a",
            "subnet-04ad87f71c6e5ab4d",
            "subnet-0e2907431c9988b72"
        ],
        "selectors": [
            {
                "namespace": "default1",
                "labels": {
                    "labelname2": "labelvalue2",
                    "labelname1": "labelvalue1"
                }
```

```
            },
            {
                "namespace": "default2",
                "labels": {
                    "labelname2": "labelvalue2",
                    "labelname1": "labelvalue1"
                }
            }
        ],
        "status": "CREATING",
        "tags": {
            "eks-fargate-profile-key-2": "value-2",
            "eks-fargate-profile-key-1": "value-1"
        }
    }
}
```

Per ulteriori informazioni, consulta Profilo AWS Fargate - Creazione di un profilo Fargate nella Guida per l'utente di Amazon EKS.

Esempio 5: creazione del profilo EKS Fargate con un selettore di caratteri jolly per namespace ed etichette, insieme IDs alle sottoreti in cui lanciare un Pod

L'`create-fargate-profile`esempio seguente crea un profilo EKS Fargate per un selettore con più namespace ed etichette, insieme IDs a sottoreti in cui lanciare un Pod.

```
aws eks create-fargate-profile \
    --cluster-name my-eks-cluster \
    --pod-execution-role-arn arn:aws:iam::111122223333:role/role-name \
    --fargate-profile-name my-fargate-profile \
    --selectors '[{"namespace": "prod*", "labels": {"labelname*?": "*value1"}},
 {"namespace": "*dev*", "labels": {"labelname*?": "*value*"}}]' \
    --subnets '["subnet-09d912bb63ef21b9a", "subnet-04ad87f71c6e5ab4d",
 "subnet-0e2907431c9988b72"]' \
    --tags '{"eks-fargate-profile-key-1": "value-1" , "eks-fargate-profile-key-2":
 "value-2"}'
```

Output:

```
{
    "fargateProfile": {
        "fargateProfileName": "my-fargate-profile",
```

```
        "fargateProfileArn": "arn:aws:eks:us-east-2:111122223333:fargateprofile/my-
eks-cluster/my-fargate-profile/e8c72bd6-5966-0bfe-b77b-1802893e5a6f",
        "clusterName": "my-eks-cluster",
        "createdAt": "2024-03-19T13:05:20.550000-04:00",
        "podExecutionRoleArn": "arn:aws:iam::111122223333:role/role-name",
        "subnets": [
            "subnet-09d912bb63ef21b9a",
            "subnet-04ad87f71c6e5ab4d",
            "subnet-0e2907431c9988b72"
        ],
        "selectors": [
            {
                "namespace": "prod*",
                "labels": {
                    "labelname*?": "*value1"
                }
            },
            {
                "namespace": "*dev*",
                "labels": {
                    "labelname*?": "*value*"
                }
            }
        ],
        "status": "CREATING",
        "tags": {
            "eks-fargate-profile-key-2": "value-2",
            "eks-fargate-profile-key-1": "value-1"
        }
    }
}
```

Per ulteriori informazioni, consulta Profilo AWS Fargate - Creazione di un profilo Fargate nella Guida per l'utente di Amazon EKS.

- Per i dettagli sull'API, consulta Command Reference. CreateFargateProfileAWS CLI

`create-nodegroup`

Il seguente esempio di codice mostra come utilizzare`create-nodegroup`.

AWS CLI

Esempio 1: crea un gruppo di nodi gestiti per un cluster Amazon EKS

L'esempio `create-nodegroup` seguente crea un gruppo di nodi gestiti per un cluster Amazon EKS.

```
aws eks create-nodegroup \
    --cluster-name my-eks-cluster \
    --nodegroup-name my-eks-nodegroup \
    --node-role arn:aws:iam::111122223333:role/role-name \
    --
subnets "subnet-0e2907431c9988b72" "subnet-04ad87f71c6e5ab4d" "subnet-09d912bb63ef21b9a"
 \
    --scaling-config minSize=1,maxSize=3,desiredSize=1 \
    --region us-east-2
```

Output:

```
{
    "nodegroup": {
        "nodegroupName": "my-eks-nodegroup",
        "nodegroupArn": "arn:aws:eks:us-east-2:111122223333:nodegroup/my-eks-
cluster/my-eks-nodegroup/bac7550f-b8b8-5fbb-4f3e-7502a931119e",
        "clusterName": "my-eks-cluster",
        "version": "1.26",
        "releaseVersion": "1.26.12-20240329",
        "createdAt": "2024-04-04T13:19:32.260000-04:00",
        "modifiedAt": "2024-04-04T13:19:32.260000-04:00",
        "status": "CREATING",
        "capacityType": "ON_DEMAND",
        "scalingConfig": {
            "minSize": 1,
            "maxSize": 3,
            "desiredSize": 1
        },
        "instanceTypes": [
            "t3.medium"
        ],
        "subnets": [
            "subnet-0e2907431c9988b72, subnet-04ad87f71c6e5ab4d,
 subnet-09d912bb63ef21b9a"
        ],
```

```
        "amiType": "AL2_x86_64",
        "nodeRole": "arn:aws:iam::111122223333:role/role-name",
        "diskSize": 20,
        "health": {
            "issues": []
        },
        "updateConfig": {
            "maxUnavailable": 1
        },
        "tags": {}
    }
}
```

Per ulteriori informazioni, consulta Creazione di un gruppo di nodi gestiti nella Guida per l'utente di Amazon EKS.

Esempio 2: crea un gruppo di nodi gestiti per un cluster Amazon EKS con tipi di istanza e dimensioni del disco personalizzati

L'esempio `create-nodegroup` seguente crea un gruppo di nodi gestiti per un cluster Amazon EKS con valori personalizzati per instance-types e disk-size.

```
aws eks create-nodegroup \
    --cluster-name my-eks-cluster \
    --nodegroup-name my-eks-nodegroup \
    --node-role arn:aws:iam::111122223333:role/role-name \
    --
subnets "subnet-0e2907431c9988b72" "subnet-04ad87f71c6e5ab4d" "subnet-09d912bb63ef21b9a"
 \
    --scaling-config minSize=1,maxSize=3,desiredSize=1 \
    --capacity-type ON_DEMAND \
    --instance-types 'm5.large' \
    --disk-size 50 \
    --region us-east-2
```

Output:

```
{
    "nodegroup": {
        "nodegroupName": "my-eks-nodegroup",
        "nodegroupArn": "arn:aws:eks:us-east-2:111122223333:nodegroup/my-eks-
cluster/my-eks-nodegroup/c0c7551b-e4f9-73d9-992c-a450fdb82322",
        "clusterName": "my-eks-cluster",
```

```
        "version": "1.26",
        "releaseVersion": "1.26.12-20240329",
        "createdAt": "2024-04-04T13:46:07.595000-04:00",
        "modifiedAt": "2024-04-04T13:46:07.595000-04:00",
        "status": "CREATING",
        "capacityType": "ON_DEMAND",
        "scalingConfig": {
            "minSize": 1,
            "maxSize": 3,
            "desiredSize": 1
        },
        "instanceTypes": [
            "m5.large"
        ],
        "subnets": [
            "subnet-0e2907431c9988b72",
            "subnet-04ad87f71c6e5ab4d",
            "subnet-09d912bb63ef21b9a"
        ],
        "amiType": "AL2_x86_64",
        "nodeRole": "arn:aws:iam::111122223333:role/role-name",
        "diskSize": 50,
        "health": {
            "issues": []
        },
        "updateConfig": {
            "maxUnavailable": 1
        },
        "tags": {}
    }
}
```

Per ulteriori informazioni, consulta Creazione di un gruppo di nodi gestiti nella Guida per l'utente di Amazon EKS.

Esempio 3: crea un gruppo di nodi gestiti per un cluster Amazon EKS con valori personalizzati per instance-types, disk-size, ami-type, capacity-type, update-config, labels, taints e tags.

L'esempio `create-nodegroup` seguente crea un gruppo di nodi gestiti per un cluster Amazon EKS con valori personalizzati per instance-types, disk-size, ami-type, capacity-type, update-config, labels, taints e tags.

```
aws eks create-nodegroup  \
```

```
    --cluster-name my-eks-cluster \
    --nodegroup-name my-eks-nodegroup \
    --node-role arn:aws:iam::111122223333:role/role-name \
    --
subnets "subnet-0e2907431c9988b72" "subnet-04ad87f71c6e5ab4d" "subnet-09d912bb63ef21b9a"
 \
    --scaling-config minSize=1,maxSize=5,desiredSize=4 \
    --instance-types 't3.large' \
    --disk-size 50 \
    --ami-type AL2_x86_64 \
    --capacity-type SPOT \
    --update-config maxUnavailable=2 \
    --labels '{"my-eks-nodegroup-label-1": "value-1" , "my-eks-nodegroup-label-2":
 "value-2"}' \
    --taints '{"key": "taint-key-1" , "value": "taint-value-1", "effect":
 "NO_EXECUTE"}' \
    --tags '{"my-eks-nodegroup-key-1": "value-1" , "my-eks-nodegroup-key-2":
 "value-2"}'
```

Output:

```
{
    "nodegroup": {
        "nodegroupName": "my-eks-nodegroup",
        "nodegroupArn": "arn:aws:eks:us-east-2:111122223333:nodegroup/my-eks-
cluster/my-eks-nodegroup/88c75524-97af-0cb9-a9c5-7c0423ab5314",
        "clusterName": "my-eks-cluster",
        "version": "1.26",
        "releaseVersion": "1.26.12-20240329",
        "createdAt": "2024-04-04T14:05:07.940000-04:00",
        "modifiedAt": "2024-04-04T14:05:07.940000-04:00",
        "status": "CREATING",
        "capacityType": "SPOT",
        "scalingConfig": {
            "minSize": 1,
            "maxSize": 5,
            "desiredSize": 4
        },
        "instanceTypes": [
            "t3.large"
        ],
        "subnets": [
            "subnet-0e2907431c9988b72",
```

```
            "subnet-04ad87f71c6e5ab4d",
            "subnet-09d912bb63ef21b9a"
        ],
        "amiType": "AL2_x86_64",
        "nodeRole": "arn:aws:iam::111122223333:role/role-name",
        "labels": {
            "my-eks-nodegroup-label-2": "value-2",
            "my-eks-nodegroup-label-1": "value-1"
        },
        "taints": [
            {
                "key": "taint-key-1",
                "value": "taint-value-1",
                "effect": "NO_EXECUTE"
            }
        ],
        "diskSize": 50,
        "health": {
            "issues": []
        },
        "updateConfig": {
            "maxUnavailable": 2
        },
        "tags": {
            "my-eks-nodegroup-key-1": "value-1",
            "my-eks-nodegroup-key-2": "value-2"
        }
    }
}
```

Per ulteriori informazioni, consulta Creazione di un gruppo di nodi gestiti nella Guida per l'utente di Amazon EKS.

- Per i dettagli sull'API, consulta CreateNodegroup AWS CLICommand Reference.

`create-pod-identity-association`

Il seguente esempio di codice mostra come utilizzare`create-pod-identity-association`.

AWS CLI

Esempio 1: come creare un'associazione EKS Pod Identity nel cluster EKS

L'esempio `create-pod-identity-association` seguente crea un'associazione EKS Pod Identity tra un account del servizio nel cluster EKS e un ruolo IAM.

```
aws eks create-pod-identity-association \
    --cluster-name eks-customer \
    --namespace default \
    --service-account default \
    --role-arn arn:aws:iam::111122223333:role/my-role
```

Output:

```
{
    "association": {
        "clusterName": "eks-customer",
        "namespace": "default",
        "serviceAccount": "default",
        "roleArn": "arn:aws:iam::111122223333:role/my-role",
        "associationArn": "arn:aws:eks:us-
west-2:111122223333:podidentityassociation/eks-customer/a-8mvwvh57cu74mgcst",
        "associationId": "a-8mvwvh57cu74mgcst",
        "tags": {},
        "createdAt": "2025-05-24T19:40:13.961000-05:00",
        "modifiedAt": "2025-05-24T19:40:13.961000-05:00"
    }
}
```

Per ulteriori informazioni, consulta Scopri come EKS Pod Identity concede ai pod l'accesso ai AWS servizi nella Amazon EKS User Guide.

Esempio 2: come creare un'associazione EKS Pod Identity nel cluster EKS con tag

Il comando `create-pod-identity-association` seguente crea un'associazione EKS Pod Identity tra un account del servizio e un ruolo IAM nel cluster EKS con tag.

```
aws eks create-pod-identity-association \
    --cluster-name eks-customer \
    --namespace default \
    --service-account default \
    --role-arn arn:aws:iam::111122223333:role/my-role \
    --tags Key1=value1,Key2=value2
```

Output:

```
{
    "association": {
        "clusterName": "eks-customer",
        "namespace": "default",
        "serviceAccount": "default",
        "roleArn": "arn:aws:iam::111122223333:role/my-role",
        "associationArn": "arn:aws:eks:us-
west-2:111122223333:podidentityassociation/eks-customer/a-9njjin9gfghecgoda",
        "associationId": "a-9njjin9gfghecgoda",
        "tags": {
            "Key2": "value2",
            "Key1": "value1"
        },
        "createdAt": "2025-05-24T19:52:14.135000-05:00",
        "modifiedAt": "2025-05-24T19:52:14.135000-05:00"
    }
}
```

Per ulteriori informazioni, consulta Scopri come EKS Pod Identity concede ai pod l'accesso ai AWS servizi nella Amazon EKS User Guide.

- Per i dettagli sull'API, consulta AWS CLI Command CreatePodIdentityAssociationReference.

`delete-access-entry`

Il seguente esempio di codice mostra come utilizzare`delete-access-entry`.

AWS CLI

Come eliminare una voce di accesso associata al cluster

Il comando `delete-access-entry` seguente elimina una voce di accesso associata al cluster EKS denominato `eks-customer`.

```
aws eks delete-access-entry \
    --cluster-name eks-customer \
    --principal-arn arn:aws:iam::111122223333:role/Admin
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Eliminare le voci di accesso nella Guida per l'utente di Amazon EKS.

- Per i dettagli sull'API, consulta DeleteAccessEntry AWS CLICommand Reference.

`delete-addon`

Il seguente esempio di codice mostra come utilizzare`delete-addon`.

AWS CLI

Esempio 1: Come eliminare un componente aggiuntivo Amazon EKS mantenendo il software relativo nel cluster EKS

L'esempio `delete-addon` seguente elimina un componente aggiuntivo Amazon EKS mantenendo il software relativo nel cluster EKS.

```
aws eks delete-addon \
    --cluster-name my-eks-cluster \
    --addon-name my-eks-addon \
    --preserve
```

Output:

```
{
    "addon": {
        "addonName": "my-eks-addon",
        "clusterName": "my-eks-cluster",
        "status": "DELETING",
        "addonVersion": "v1.9.3-eksbuild.7",
        "health": {
            "issues": []
        },
        "addonArn": "arn:aws:eks:us-east-2:111122223333:addon/my-eks-cluster/my-eks-
addon/a8c71ed3-944e-898b-9167-c763856af4b8",
        "createdAt": "2024-03-14T11:49:09.009000-04:00",
        "modifiedAt": "2024-03-14T12:03:49.776000-04:00",
        "tags": {}
    }
}
```

Per ulteriori informazioni, consulta Gestione dei componenti aggiuntivi di Amazon EKS - Aggiornamento di un componente aggiuntivo nella Guida per l'utente di Amazon EKS.

Esempio 2: Come eliminare un componente aggiuntivo Amazon EKS e il software relativo dal cluster EKS

L'esempio `delete-addon` seguente elimina un componente aggiuntivo Amazon EKS e il software relativo dal cluster EKS.

```
aws eks delete-addon \
    --cluster-name my-eks-cluster \
    --addon-name my-eks-addon
```

Output:

```
{
    "addon": {
        "addonName": "my-eks-addon",
        "clusterName": "my-eks-cluster",
        "status": "DELETING",
        "addonVersion": "v1.15.1-eksbuild.1",
        "health": {
            "issues": []
        },
        "addonArn": "arn:aws:eks:us-east-2:111122223333:addon/my-eks-cluster/my-eks-
addon/bac71ed1-ec43-3bb6-88ea-f243cdb58954",
        "createdAt": "2024-03-14T11:45:31.983000-04:00",
        "modifiedAt": "2024-03-14T11:58:40.136000-04:00",
        "serviceAccountRoleArn": "arn:aws:iam::111122223333:role/role-name",
        "tags": {}
    }
}
```

Per ulteriori informazioni, consulta Gestione dei componenti aggiuntivi di Amazon EKS - Aggiornamento di un componente aggiuntivo nella Guida per l'utente di Amazon EKS.

- Per i dettagli sull'API, consulta DeleteAddon AWS CLICommand Reference.

`delete-cluster`

Il seguente esempio di codice mostra come utilizzare`delete-cluster`.

AWS CLI

Come eliminare un piano di controllo (control-plane) del cluster Amazon EKS

L'esempio `delete-cluster` seguente elimina un piano di controllo (control-plane) del cluster Amazon EKS.

```
aws eks delete-cluster \
    --name my-eks-cluster
```

Output:

```
{
    "cluster": {
        "name": "my-eks-cluster",
        "arn": "arn:aws:eks:us-east-2:111122223333:cluster/my-eks-cluster",
        "createdAt": "2024-03-14T11:31:44.348000-04:00",
        "version": "1.27",
        "endpoint": "https://DALSJ343KE23J3RN45653DSKJTT647TYD.yl4.us-
east-2.eks.amazonaws.com",
        "roleArn": "arn:aws:iam::111122223333:role/eksctl-my-eks-cluster-cluster-
ServiceRole-zMF6CBakwwbW",
        "resourcesVpcConfig": {
            "subnetIds": [
                "subnet-0fb75d2d8401716e7",
                "subnet-02184492f67a3d0f9",
                "subnet-04098063527aab776",
                "subnet-0e2907431c9988b72",
                "subnet-04ad87f71c6e5ab4d",
                "subnet-09d912bb63ef21b9a"
            ],
            "securityGroupIds": [
                "sg-0c1327f6270afbb36"
            ],
            "clusterSecurityGroupId": "sg-01c84d09d70f39a7f",
            "vpcId": "vpc-0012b8e1cc0abb17d",
            "endpointPublicAccess": true,
            "endpointPrivateAccess": true,
            "publicAccessCidrs": [
                "0.0.0.0/0"
            ]
        },
        "kubernetesNetworkConfig": {
            "serviceIpv4Cidr": "10.100.0.0/16",
            "ipFamily": "ipv4"
        },
        "logging": {
```

```
            "clusterLogging": [
                {
                    "types": [
                        "api",
                        "audit",
                        "authenticator",
                        "controllerManager",
                        "scheduler"
                    ],
                    "enabled": true
                }
            ]
        },
        "identity": {
            "oidc": {
                "issuer": "https://oidc.eks.us-east-2.amazonaws.com/id/
DALSJ343KE23J3RN45653DSKJTT647TYD"
            }
        },
        "status": "DELETING",
        "certificateAuthority": {
            "data": "XXX_CA_DATA_XXX"
        },
        "platformVersion": "eks.16",
        "tags": {
            "aws:cloudformation:stack-name": "eksctl-my-eks-cluster-cluster",
            "alpha.eksctl.io/cluster-name": "my-eks-cluster",
            "karpenter.sh/discovery": "my-eks-cluster",
            "aws:cloudformation:stack-id": "arn:aws:cloudformation:us-
east-2:111122223333:stack/eksctl-my-eks-cluster-cluster/e752ea00-e217-11ee-
beae-0a9599c8c7ed",
            "auto-delete": "no",
            "eksctl.cluster.k8s.io/v1alpha1/cluster-name": "my-eks-cluster",
            "EKS-Cluster-Name": "my-eks-cluster",
            "alpha.eksctl.io/cluster-oidc-enabled": "true",
            "aws:cloudformation:logical-id": "ControlPlane",
            "alpha.eksctl.io/eksctl-version": "0.173.0-dev
+a7ee89342.2024-03-01T03:40:57Z",
            "Name": "eksctl-my-eks-cluster-cluster/ControlPlane"
        },
        "accessConfig": {
            "authenticationMode": "API_AND_CONFIG_MAP"
        }
    }
```

```
}
```

Per ulteriori informazioni, consulta Eliminazione di un cluster Amazon EKS nella Guida per l'utente di Amazon EKS.

- Per i dettagli sull'API, consulta DeleteCluster AWS CLICommand Reference.

`delete-fargate-profile`

Il seguente esempio di codice mostra come utilizzare`delete-fargate-profile`.

AWS CLI

Esempio 1: crea il profilo EKS Fargate per un selettore con un namespace

L'esempio `delete-fargate-profile` seguente crea un profilo EKS Fargate per un selettore con un namespace.

```
aws eks delete-fargate-profile \
    --cluster-name my-eks-cluster \
    --fargate-profile-name my-fargate-profile
```

Output:

```
{
    "fargateProfile": {
        "fargateProfileName": "my-fargate-profile",
        "fargateProfileArn": "arn:aws:eks:us-east-2:111122223333:fargateprofile/my-
eks-cluster/my-fargate-profile/1ac72bb3-3fc6-2631-f1e1-98bff53bed62",
        "clusterName": "my-eks-cluster",
        "createdAt": "2024-03-19T11:48:39.975000-04:00",
        "podExecutionRoleArn": "arn:aws:iam::111122223333:role/role-name",
        "subnets": [
            "subnet-09d912bb63ef21b9a",
            "subnet-04ad87f71c6e5ab4d",
            "subnet-0e2907431c9988b72"
        ],
        "selectors": [
            {
                "namespace": "default",
                "labels": {
                    "foo": "bar"
                }
```

```
            }
        ],
        "status": "DELETING",
        "tags": {}
    }
}
```

Per ulteriori informazioni, consulta Profilo AWS Fargate - Eliminazione di un profilo Fargate nella Guida per l'utente di Amazon EKS.

- Per i dettagli sull'API, consulta DeleteFargateProfile AWS CLICommand Reference.

`delete-nodegroup`

Il seguente esempio di codice mostra come utilizzare`delete-nodegroup`.

AWS CLI

Esempio 1: elimina un gruppo di nodi gestiti per un cluster Amazon EKS

L'esempio `delete-nodegroup` seguente elimina un gruppo di nodi gestiti per un cluster Amazon EKS.

```
aws eks delete-nodegroup \
    --cluster-name my-eks-cluster \
    --nodegroup-name my-eks-nodegroup
```

Output:

```
{
    "nodegroup": {
        "nodegroupName": "my-eks-nodegroup",
        "nodegroupArn": "arn:aws:eks:us-east-2:111122223333:nodegroup/my-eks-
cluster/my-eks-nodegroup/1ec75f5f-0e21-dcc0-b46e-f9c442685cd8",
        "clusterName": "my-eks-cluster",
        "version": "1.26",
        "releaseVersion": "1.26.12-20240329",
        "createdAt": "2024-04-08T13:25:15.033000-04:00",
        "modifiedAt": "2024-04-08T13:25:31.252000-04:00",
        "status": "DELETING",
        "capacityType": "SPOT",
        "scalingConfig": {
            "minSize": 1,
```

```
            "maxSize": 5,
            "desiredSize": 4
        },
        "instanceTypes": [
            "t3.large"
        ],
        "subnets": [
            "subnet-0e2907431c9988b72",
            "subnet-04ad87f71c6e5ab4d",
            "subnet-09d912bb63ef21b9a"
        ],
        "amiType": "AL2_x86_64",
        "nodeRole": "arn:aws:iam::111122223333:role/role-name",
        "labels": {
            "my-eks-nodegroup-label-2": "value-2",
            "my-eks-nodegroup-label-1": "value-1"
        },
        "taints": [
            {
                "key": "taint-key-1",
                "value": "taint-value-1",
                "effect": "NO_EXECUTE"
            }
        ],
        "diskSize": 50,
        "health": {
            "issues": []
        },
        "updateConfig": {
            "maxUnavailable": 2
        },
        "tags": {
            "my-eks-nodegroup-key-1": "value-1",
            "my-eks-nodegroup-key-2": "value-2"
        }
    }
}
```

- Per i dettagli sull'API, consulta DeleteNodegroup AWS CLICommand Reference.

`delete-pod-identity-association`

Il seguente esempio di codice mostra come utilizzare`delete-pod-identity-association`.

AWS CLI

Come eliminare l'associazione EKS Pod Identity

L'esempio `delete-pod-identity-association` seguente elimina l'associazione EKS Pod Identity con ID associazione `a-9njjin9gfghecgocd` dal cluster EKS denominato `eks-customer`.

```
aws eks delete-pod-identity-association \
    --cluster-name eks-customer \
    --association-id a-9njjin9gfghecgocd
```

Output:

```
{
    "association": {
        "clusterName": "eks-customer",
        "namespace": "default",
        "serviceAccount": "default",
        "roleArn": "arn:aws:iam::111122223333:role/s3-role",
        "associationArn": "arn:aws:eks:us-
west-2:111122223333:podidentityassociation/eks-customer/a-9njjin9gfghecgocd",
        "associationId": "a-9njjin9gfghecgocd",
        "tags": {
            "Key2": "value2",
            "Key1": "value1"
        },
        "createdAt": "2025-05-24T19:52:14.135000-05:00",
        "modifiedAt": "2025-05-25T21:10:56.923000-05:00"
    }
}
```

Per ulteriori informazioni, consulta Scopri come EKS Pod Identity concede ai pod l'accesso ai AWS servizi nella Amazon EKS User Guide.

- Per i dettagli sull'API, consulta AWS CLI Command DeletePodIdentityAssociationReference.

`deregister-cluster`

Il seguente esempio di codice mostra come utilizzare`deregister-cluster`.

AWS CLI

Come annullare la registrazione di un cluster connesso per rimuoverlo dal piano di controllo (control-plane) di Amazon EKS

L'esempio `deregister-cluster` seguente annulla la registrazione di un cluster connesso per rimuoverlo dal piano di controllo (control-plane) di Amazon EKS.

```
aws eks deregister-cluster \
    --name my-eks-anywhere-cluster
```

Output:

```
{
    "cluster": {
        "name": "my-eks-anywhere-cluster",
        "arn": "arn:aws:eks:us-east-2:111122223333:cluster/my-eks-anywhere-cluster",
        "createdAt": "2024-04-12T12:38:37.561000-04:00",
        "status": "DELETING",
        "tags": {},
        "connectorConfig": {
            "activationId": "dfb5ad28-13c3-4e26-8a19-5b2457638c74",
            "activationExpiry": "2024-04-15T12:38:37.082000-04:00",
            "provider": "EKS_ANYWHERE",
            "roleArn": "arn:aws:iam::111122223333:role/AmazonEKSConnectorAgentRole"
        }
    }
}
```

Per ulteriori informazioni, consulta Annullamento della registrazione di un cluster nella Guida per l'utente di Amazon EKS.

- Per i dettagli sull'API, consulta DeregisterCluster AWS CLICommand Reference.

`describe-access-entry`

Il seguente esempio di codice mostra come utilizzare`describe-access-entry`.

AWS CLI

Come descrivere la voce di accesso per il cluster EKS

L'esempio `describe-access-entry` seguente descrive una voce di accesso per il cluster EKS.

```
aws eks describe-access-entry \
    --cluster-name eks-customer \
    --principal-arn arn:aws:iam::111122223333:user/eks-admin-user
```

Output:

```
{
    "accessEntry": {
        "clusterName": "eks-customer",
        "principalArn": "arn:aws:iam::111122223333:user/eks-admin-user",
        "kubernetesGroups": [],
        "accessEntryArn": "arn:aws:eks:us-west-2:111122223333:access-entry/eks-
customer/user/111122223333/eks-admin-user/0acb1bc6-cb0a-ede6-11ae-a6506e3d36p0",
        "createdAt": "2025-04-14T22:45:48.097000-05:00",
        "modifiedAt": "2025-04-14T22:45:48.097000-05:00",
        "tags": {},
        "username": "arn:aws:iam::111122223333:user/eks-admin-user",
        "type": "STANDARD"
    }
}
```

Per ulteriori informazioni, consulta Concedere a utenti IAM l'accesso a Kubernetes con le voci di accesso EKS nella Guida per l'utente di Amazon EKS.

- Per i dettagli sull'API, consulta DescribeAccessEntry AWS CLICommand Reference.

`describe-addon-configuration`

Il seguente esempio di codice mostra come utilizzare`describe-addon-configuration`.

AWS CLI

Esempio 1: opzioni di configurazione disponibili durante la creazione o l'aggiornamento di Amazon vpc-cni AddOns

L'esempio `describe-addon-configuration` seguente restituisce tutti gli schemi di configurazione disponibili utilizzati quando si crea o si aggiorna un componente aggiuntivo per vpc-cni con la rispettiva versione.

```
aws eks describe-addon-configuration \
    --addon-name vpc-cni \
```

```
    --addon-version v1.15.1-eksbuild.1
```

Output:


```
{
    "addonName": "vpc-cni",
    "addonVersion": "v1.15.1-eksbuild.1",
    "configurationSchema": "{\"$ref\":\"#/definitions/VpcCni\",\"$schema\":\"http://
json-schema.org/draft-06/schema#\",\"definitions\":{\"Affinity\":{\"type\":
[\"object\",\"null\"]},\"EniConfig\":{\"additionalProperties\":false,\"properties
\":{\"create\":{\"type\":\"boolean\"},\"region\":{\"type\":\"string\"},\"subnets\":
{\"additionalProperties\":{\"additionalProperties\":false,\"properties\":{\"id\":
{\"type\":\"string\"},\"securityGroups\":{\"items\":{\"type\":\"string\"},\"type\":
\"array\"}},\"required\":[\"id\"],\"type\":\"object\"},\"minProperties\":1,\"type
\":\"object\"}},\"required\":[\"create\",\"region\",\"subnets\"],\"type\":\"object
\"},\"Env\":{\"additionalProperties\":false,\"properties\":{\"ADDITIONAL_ENI_TAGS
\":{\"type\":\"string\"},\"ANNOTATE_POD_IP\":{\"format\":\"boolean\",\"type\":
\"string\"},\"AWS_EC2_ENDPOINT\":{\"type\":\"string\"},\"AWS_EXTERNAL_SERVICE_CIDRS
\":{\"type\":\"string\"},\"AWS_MANAGE_ENIS_NON_SCHEDULABLE\":{\"format\":\"boolean
\",\"type\":\"string\"},\"AWS_VPC_CNI_NODE_PORT_SUPPORT\":{\"format\":\"boolean
\",\"type\":\"string\"},\"AWS_VPC_ENI_MTU\":{\"format\":\"integer\",\"type\":
\"string\"},\"AWS_VPC_K8S_CNI_CUSTOM_NETWORK_CFG\":{\"format\":\"boolean\",\"type
\":\"string\"},\"AWS_VPC_K8S_CNI_EXCLUDE_SNAT_CIDRS\":{\"type\":\"string\"},
\"AWS_VPC_K8S_CNI_EXTERNALSNAT\":{\"format\":\"boolean\",\"type\":\"string\"},
\"AWS_VPC_K8S_CNI_LOGLEVEL\":{\"type\":\"string\"},\"AWS_VPC_K8S_CNI_LOG_FILE\":
{\"type\":\"string\"},\"AWS_VPC_K8S_CNI_RANDOMIZESNAT\":{\"type\":\"string\"},
\"AWS_VPC_K8S_CNI_VETHPREFIX\":{\"type\":\"string\"},\"AWS_VPC_K8S_PLUGIN_LOG_FILE
\":{\"type\":\"string\"},\"AWS_VPC_K8S_PLUGIN_LOG_LEVEL\":{\"type\":\"string
\"},\"CLUSTER_ENDPOINT\":{\"type\":\"string\"},\"DISABLE_INTROSPECTION\":
{\"format\":\"boolean\",\"type\":\"string\"},\"DISABLE_LEAKED_ENI_CLEANUP\":
{\"format\":\"boolean\",\"type\":\"string\"},\"DISABLE_METRICS\":{\"format
\":\"boolean\",\"type\":\"string\"},\"DISABLE_NETWORK_RESOURCE_PROVISIONING
\":{\"format\":\"boolean\",\"type\":\"string\"},\"DISABLE_POD_V6\":{\"format
\":\"boolean\",\"type\":\"string\"},\"ENABLE_BANDWIDTH_PLUGIN\":{\"format\":
\"boolean\",\"type\":\"string\"},\"ENABLE_POD_ENI\":{\"format\":\"boolean\",
\"type\":\"string\"},\"ENABLE_PREFIX_DELEGATION\":{\"format\":\"boolean\",
\"type\":\"string\"},\"ENABLE_V4_EGRESS\":{\"format\":\"boolean\",\"type\":
\"string\"},\"ENABLE_V6_EGRESS\":{\"format\":\"boolean\",\"type\":\"string\"},
\"ENI_CONFIG_ANNOTATION_DEF\":{\"type\":\"string\"},\"ENI_CONFIG_LABEL_DEF\":
{\"type\":\"string\"},\"INTROSPECTION_BIND_ADDRESS\":{\"type\":\"string\"},
\"IP_COOLDOWN_PERIOD\":{\"format\":\"integer\",\"type\":\"string\"},\"MAX_ENI
\":{\"format\":\"integer\",\"type\":\"string\"},\"MINIMUM_IP_TARGET\":{\"format
\":\"integer\",\"type\":\"string\"},\"POD_SECURITY_GROUP_ENFORCING_MODE\":
```

```
{\"type\":\"string\"},\"WARM_ENI_TARGET\":{\"format\":\"integer\",\"type\":
\"string\"},\"WARM_IP_TARGET\":{\"format\":\"integer\",\"type\":\"string\"},
\"WARM_PREFIX_TARGET\":{\"format\":\"integer\",\"type\":\"string\"}},\"title
\":\"Env\",\"type\":\"object\"},\"Init\":{\"additionalProperties\":false,
\"properties\":{\"env\":{\"$ref\":\"#/definitions/InitEnv\"}},\"title\":\"Init
\",\"type\":\"object\"},\"InitEnv\":{\"additionalProperties\":false,\"properties
\":{\"DISABLE_TCP_EARLY_DEMUX\":{\"format\":\"boolean\",\"type\":\"string\"},
\"ENABLE_V6_EGRESS\":{\"format\":\"boolean\",\"type\":\"string\"}},\"title\":
\"InitEnv\",\"type\":\"object\"},\"Limits\":{\"additionalProperties\":false,
\"properties\":{\"cpu\":{\"type\":\"string\"},\"memory\":{\"type\":\"string\"}},
\"title\":\"Limits\",\"type\":\"object\"},\"NodeAgent\":{\"additionalProperties
\":false,\"properties\":{\"enableCloudWatchLogs\":{\"format\":\"boolean\",
\"type\":\"string\"},\"enablePolicyEventLogs\":{\"format\":\"boolean\",\"type\":
\"string\"},\"healthProbeBindAddr\":{\"format\":\"integer\",\"type\":\"string
\"},\"metricsBindAddr\":{\"format\":\"integer\",\"type\":\"string\"}},\"title\":
\"NodeAgent\",\"type\":\"object\"},\"Resources\":{\"additionalProperties\":false,
\"properties\":{\"limits\":{\"$ref\":\"#/definitions/Limits\"},\"requests\":
{\"$ref\":\"#/definitions/Limits\"}},\"title\":\"Resources\",\"type\":\"object
\"},\"Tolerations\":{\"additionalProperties\":false,\"items\":{\"type\":\"object
\"},\"type\":\"array\"},\"VpcCni\":{\"additionalProperties\":false,\"properties
\":{\"affinity\":{\"$ref\":\"#/definitions/Affinity\"},\"enableNetworkPolicy\":
{\"format\":\"boolean\",\"type\":\"string\"},\"enableWindowsIpam\":{\"format\":
\"boolean\",\"type\":\"string\"},\"eniConfig\":{\"$ref\":\"#/definitions/EniConfig
\"},\"env\":{\"$ref\":\"#/definitions/Env\"},\"init\":{\"$ref\":\"#/definitions/Init
\"},\"livenessProbeTimeoutSeconds\":{\"type\":\"integer\"},\"nodeAgent\":{\"$ref\":
\"#/definitions/NodeAgent\"},\"readinessProbeTimeoutSeconds\":{\"type\":\"integer
\"},\"resources\":{\"$ref\":\"#/definitions/Resources\"},\"tolerations\":{\"$ref
\":\"#/definitions/Tolerations\"}},\"title\":\"VpcCni\",\"type\":\"object\"}},
\"description\":\"vpc-cni\"}"
}
```


Esempio 2: opzioni di configurazione disponibili durante la creazione o l'aggiornamento di Amazon cordns AddOns

L'esempio `describe-addon-configuration` seguente restituisce tutti gli schemi di configurazione disponibili utilizzati quando si crea o si aggiorna un componente aggiuntivo per coredns con la rispettiva versione.

```
aws eks describe-addon-configuration \
    --addon-name coredns \
    --addon-version v1.8.7-eksbuild.4
```

Output:

```
{
    "addonName": "coredns",
    "addonVersion": "v1.8.7-eksbuild.4",
    "configurationSchema": "{\"$ref\":\"#/definitions/Coredns\",\"$schema
\":\"http://json-schema.org/draft-06/schema#\",\"definitions\":{\"Coredns\":
{\"additionalProperties\":false,\"properties\":{\"computeType\":{\"type\":
\"string\"},\"corefile\":{\"description\":\"Entire corefile contents to use with
 installation\",\"type\":\"string\"},\"nodeSelector\":{\"additionalProperties\":
{\"type\":\"string\"},\"type\":\"object\"},\"replicaCount\":{\"type\":\"integer
\"},\"resources\":{\"$ref\":\"#/definitions/Resources\"}},\"title\":\"Coredns\",
\"type\":\"object\"},\"Limits\":{\"additionalProperties\":false,\"properties\":
{\"cpu\":{\"type\":\"string\"},\"memory\":{\"type\":\"string\"}},\"title\":\"Limits
\",\"type\":\"object\"},\"Resources\":{\"additionalProperties\":false,\"properties
\":{\"limits\":{\"$ref\":\"#/definitions/Limits\"},\"requests\":{\"$ref\":\"#/
definitions/Limits\"}},\"title\":\"Resources\",\"type\":\"object\"}}}"
}
```


Per ulteriori informazioni, consulta Creazione o aggiornamento di un file kubeconfig per un cluster Amazon EKS nella Guida per l'utente di Amazon EKS.

- Per i dettagli sull'API, consulta AWS CLI Command DescribeAddonConfigurationReference.

`describe-addon-versions`

Il seguente esempio di codice mostra come utilizzare`describe-addon-versions`.

AWS CLI

Esempio 1: elenca tutti i componenti aggiuntivi disponibili per un cluster EKS

L'`describe-addon-versions`esempio seguente elenca tutti i AWS componenti aggiuntivi disponibili.

```
aws eks describe-addon-versions  \
    --query 'sort_by(addons  &owner)[].{publisher: publisher, owner: owner,
 addonName: addonName, type: type}' \
    --output table
```

Output:

```
-----------------------------------------------------------------------------------------
```

```
|                                                DescribeAddonVersions
                              |
+---------------------------------------------+------------------
+----------------------+-------------------------+
|                  addonName                  |      owner       |       publisher
     |          type           |
+---------------------------------------------+------------------
+----------------------+-------------------------+
|  vpc-cni                                    |  aws             |  eks
     |  networking             |
|  snapshot-controller                        |  aws             |  eks
     |  storage                |
|  kube-proxy                                 |  aws             |  eks
     |  networking             |
|  eks-pod-identity-agent                     |  aws             |  eks
     |  security               |
|  coredns                                    |  aws             |  eks
     |  networking             |
|  aws-mountpoint-s3-csi-driver               |  aws             |  s3
     |  storage                |
|  aws-guardduty-agent                        |  aws             |  eks
     |  security               |
|  aws-efs-csi-driver                         |  aws             |  eks
     |  storage                |
|  aws-ebs-csi-driver                         |  aws             |  eks
     |  storage                |
|  amazon-cloudwatch-observability            |  aws             |  eks
     |  observability          |
|  adot                                       |  aws             |  eks
     |  observability          |
|  upwind-security_upwind-operator            |  aws-marketplace |  Upwind Security
     |  security               |
|  upbound_universal-crossplane               |  aws-marketplace |  upbound
     |  infra-management       |
|  tetrate-io_istio-distro                    |  aws-marketplace |  tetrate-io
     |  policy-management      |
|  teleport_teleport                          |  aws-marketplace |  teleport
     |  policy-management      |
|  stormforge_optimize-live                   |  aws-marketplace |  StormForge
     |  cost-management        |
|  splunk_splunk-otel-collector-chart         |  aws-marketplace |  Splunk
     |  monitoring             |
|  solo-io_istio-distro                       |  aws-marketplace |  Solo.io
     |  service-mesh           |
```

```
|  rafay-systems_rafay-operator                |  aws-marketplace |  rafay-systems
     |  kubernetes-management  |
|  new-relic_kubernetes-operator               |  aws-marketplace |  New Relic
     |  observability          |
|  netapp_trident-operator                     |  aws-marketplace |  NetApp Inc.
     |  storage                |
|  leaksignal_leakagent                        |  aws-marketplace |  leaksignal
     |  monitoring             |
|  kubecost_kubecost                           |  aws-marketplace |  kubecost
     |  cost-management        |
|  kong_konnect-ri                             |  aws-marketplace |  kong
     |  ingress-service-type   |
|  kasten_k10                                  |  aws-marketplace |  Kasten by Veeam
     |  data-protection        |
|  haproxy-technologies_kubernetes-ingress-ee |  aws-marketplace |  HAProxy
 Technologies |  ingress-controller     |
|  groundcover_agent                           |  aws-marketplace |  groundcover
     |  monitoring             |
|  grafana-labs_kubernetes-monitoring          |  aws-marketplace |  Grafana Labs
     |  monitoring             |
|  factorhouse_kpow                            |  aws-marketplace |  factorhouse
     |  monitoring             |
|  dynatrace_dynatrace-operator                |  aws-marketplace |  dynatrace
     |  monitoring             |
|  datree_engine-pro                           |  aws-marketplace |  datree
     |  policy-management      |
|  datadog_operator                            |  aws-marketplace |  Datadog
     |  monitoring             |
|  cribl_cribledge                             |  aws-marketplace |  Cribl
     |  observability          |
|  calyptia_fluent-bit                         |  aws-marketplace |  Calyptia Inc
     |  observability          |
|  accuknox_kubearmor                          |  aws-marketplace |  AccuKnox
     |  security               |
+----------------------------------------------+------------------
+----------------------+-------------------------+
```

Per ulteriori informazioni, consulta Gestione dei componenti aggiuntivi di Amazon EKS - Creazione di un componente aggiuntivo nella Guida per l'utente di Amazon EKS.

Esempio 2: elenca tutti i componenti aggiuntivi disponibili per la versione di Kubernetes specificata supportata da EKS

L'esempio `describe-addon-versions` seguente elenca tutti i componenti aggiuntivi disponibili per la versione di Kubernetes specificata supportata da EKS.

```
aws eks describe-addon-versions  \
    --kubernetes-version=1.26 \
    --query 'sort_by(addons  &owner)[].{publisher: publisher, owner: owner,
 addonName: addonName, type: type}' \
    --output table
```

Output:

```
-------------------------------------------------------------------------------------------------
|                                                DescribeAddonVersions
                                |
+-----------------------------------------------+-----------------
+-----------------------+-------------------------+
|                   addonName                   |      owner       |       publisher
     |          type           |
+-----------------------------------------------+-----------------
+-----------------------+-------------------------+
|  vpc-cni                                      |  aws             |  eks
     |  networking             |
|  snapshot-controller                          |  aws             |  eks
     |  storage                |
|  kube-proxy                                   |  aws             |  eks
     |  networking             |
|  eks-pod-identity-agent                       |  aws             |  eks
     |  security               |
|  coredns                                      |  aws             |  eks
     |  networking             |
|  aws-mountpoint-s3-csi-driver                 |  aws             |  s3
     |  storage                |
|  aws-guardduty-agent                          |  aws             |  eks
     |  security               |
|  aws-efs-csi-driver                           |  aws             |  eks
     |  storage                |
|  aws-ebs-csi-driver                           |  aws             |  eks
     |  storage                |
|  amazon-cloudwatch-observability              |  aws             |  eks
     |  observability          |
|  adot                                         |  aws             |  eks
     |  observability          |
```

```
|  upwind-security_upwind-operator            |  aws-marketplace |  Upwind Security
     |  security               |
|  tetrate-io_istio-distro                    |  aws-marketplace |  tetrate-io
     |  policy-management      |
|  stormforge_optimize-live                   |  aws-marketplace |  StormForge
     |  cost-management        |
|  splunk_splunk-otel-collector-chart         |  aws-marketplace |  Splunk
     |  monitoring             |
|  solo-io_istio-distro                       |  aws-marketplace |  Solo.io
     |  service-mesh           |
|  rafay-systems_rafay-operator               |  aws-marketplace |  rafay-systems
     |  kubernetes-management  |
|  new-relic_kubernetes-operator              |  aws-marketplace |  New Relic
     |  observability          |
|  netapp_trident-operator                    |  aws-marketplace |  NetApp Inc.
     |  storage                |
|  leaksignal_leakagent                       |  aws-marketplace |  leaksignal
     |  monitoring             |
|  kubecost_kubecost                          |  aws-marketplace |  kubecost
     |  cost-management        |
|  kong_konnect-ri                            |  aws-marketplace |  kong
     |  ingress-service-type   |
|  haproxy-technologies_kubernetes-ingress-ee |  aws-marketplace |  HAProxy
 Technologies |  ingress-controller     |
|  groundcover_agent                          |  aws-marketplace |  groundcover
     |  monitoring             |
|  grafana-labs_kubernetes-monitoring         |  aws-marketplace |  Grafana Labs
     |  monitoring             |
|  dynatrace_dynatrace-operator               |  aws-marketplace |  dynatrace
     |  monitoring             |
|  datadog_operator                           |  aws-marketplace |  Datadog
     |  monitoring             |
|  cribl_cribledge                            |  aws-marketplace |  Cribl
     |  observability          |
|  calyptia_fluent-bit                        |  aws-marketplace |  Calyptia Inc
     |  observability          |
|  accuknox_kubearmor                         |  aws-marketplace |  AccuKnox
     |  security               |
+---------------------------------------------+------------------
+----------------------+-------------------------+
```

Per ulteriori informazioni, consulta Gestione dei componenti aggiuntivi di Amazon EKS - Creazione di un componente aggiuntivo nella Guida per l'utente di Amazon EKS.

Esempio 3: elenca tutte le versioni dei componenti aggiuntivi vpc-cni disponibili per la versione di Kubernetes specificata supportata da EKS

L'esempio `describe-addon-versions` seguente elenca tutte le versioni dei componenti aggiuntivi vpc-cni disponibili per la versione di Kubernetes specificata supportata da EKS.

```
aws eks describe-addon-versions \
    --kubernetes-version=1.26 \
    --addon-name=vpc-cni \
    --query='addons[].addonVersions[].addonVersion'
```

Output:

```
[
    "v1.18.0-eksbuild.1",
    "v1.17.1-eksbuild.1",
    "v1.16.4-eksbuild.2",
    "v1.16.3-eksbuild.2",
    "v1.16.2-eksbuild.1",
    "v1.16.0-eksbuild.1",
    "v1.15.5-eksbuild.1",
    "v1.15.4-eksbuild.1",
    "v1.15.3-eksbuild.1",
    "v1.15.1-eksbuild.1",
    "v1.15.0-eksbuild.2",
    "v1.14.1-eksbuild.1",
    "v1.14.0-eksbuild.3",
    "v1.13.4-eksbuild.1",
    "v1.13.3-eksbuild.1",
    "v1.13.2-eksbuild.1",
    "v1.13.0-eksbuild.1",
    "v1.12.6-eksbuild.2",
    "v1.12.6-eksbuild.1",
    "v1.12.5-eksbuild.2",
    "v1.12.0-eksbuild.2"
]
```

Per ulteriori informazioni, consulta Gestione dei componenti aggiuntivi di Amazon EKS - Creazione di un componente aggiuntivo nella Guida per l'utente di Amazon EKS.

- Per i dettagli sull'API, consultate AWS CLI Command DescribeAddonVersionsReference.

`describe-addon`

Il seguente esempio di codice mostra come utilizzare`describe-addon`.

AWS CLI

Come descrivere il componente aggiuntivo EKS attualmente in esecuzione nel cluster Amazon EKS

L'esempio `describe-addon` seguente descrive il componente aggiuntivo EKS attualmente in esecuzione nel cluster Amazon EKS.

```
aws eks describe-addon \
    --cluster-name my-eks-cluster \
    --addon-name vpc-cni
```

Output:

```
{
    "addon": {
        "addonName": "vpc-cni",
        "clusterName": "my-eks-cluster",
        "status": "ACTIVE",
        "addonVersion": "v1.16.4-eksbuild.2",
        "health": {
            "issues": []
        },
        "addonArn": "arn:aws:eks:us-east-2:111122223333:addon/my-eks-cluster/vpc-
cni/0ec71efc-98dd-3203-60b0-4b939b2a5e5f",
        "createdAt": "2024-03-14T13:18:45.417000-04:00",
        "modifiedAt": "2024-03-14T13:18:49.557000-04:00",
        "serviceAccountRoleArn": "arn:aws:iam::111122223333:role/eksctl-my-eks-
cluster-addon-vpc-cni-Role1-YfakrqOC1UTm",
        "tags": {
            "eks-addon-key-3": "value-3",
            "eks-addon-key-4": "value-4"
        },
        "configurationValues": "resources:\n    limits:\n        cpu: '100m'\nenv:\n
    AWS_VPC_K8S_CNI_LOGLEVEL: 'DEBUG'"
    }
}
```

- Per i dettagli sull'API, consulta DescribeAddon AWS CLICommand Reference.

`describe-cluster`

Il seguente esempio di codice mostra come utilizzare`describe-cluster`.

AWS CLI

Come descrivere il componente aggiuntivo EKS attualmente in esecuzione nel cluster Amazon EKS

L'esempio `describe-cluster` seguente descrive il componente aggiuntivo EKS attualmente in esecuzione nel cluster Amazon EKS.

```
aws eks describe-cluster \
    --name my-eks-cluster
```

Output:

```
{
    "cluster": {
        "name": "my-eks-cluster",
        "arn": "arn:aws:eks:us-east-2:111122223333:cluster/my-eks-cluster",
        "createdAt": "2024-03-14T11:31:44.348000-04:00",
        "version": "1.26",
        "endpoint": "https://JSA79429HJDASKJDJ8223829MNDNASW.yl4.us-
east-2.eks.amazonaws.com",
        "roleArn": "arn:aws:iam::111122223333:role/eksctl-my-eks-cluster-cluster-
ServiceRole-zMF6CBakwwbW",
        "resourcesVpcConfig": {
            "subnetIds": [
                "subnet-0fb75d2d8401716e7",
                "subnet-02184492f67a3d0f9",
                "subnet-04098063527aab776",
                "subnet-0e2907431c9988b72",
                "subnet-04ad87f71c6e5ab4d",
                "subnet-09d912bb63ef21b9a"
            ],
            "securityGroupIds": [
                "sg-0c1327f6270afbb36"
            ],
            "clusterSecurityGroupId": "sg-01c84d09d70f39a7f",
            "vpcId": "vpc-0012b8e1cc0abb17d",
            "endpointPublicAccess": true,
            "endpointPrivateAccess": true,
```

```
            "publicAccessCidrs": [
                "22.19.18.2/32"
            ]
        },
        "kubernetesNetworkConfig": {
            "serviceIpv4Cidr": "10.100.0.0/16",
            "ipFamily": "ipv4"
        },
        "logging": {
            "clusterLogging": [
                {
                    "types": [
                        "api",
                        "audit",
                        "authenticator",
                        "controllerManager",
                        "scheduler"
                    ],
                    "enabled": true
                }
            ]
        },
        "identity": {
            "oidc": {
                "issuer": "https://oidc.eks.us-east-2.amazonaws.com/id/
JSA79429HJDASKJDJ8223829MNDNASW"
            }
        },
        "status": "ACTIVE",
        "certificateAuthority": {
            "data": "CA_DATA_STRING..."
        },
        "platformVersion": "eks.14",
        "tags": {
            "aws:cloudformation:stack-name": "eksctl-my-eks-cluster-cluster",
            "alpha.eksctl.io/cluster-name": "my-eks-cluster",
            "karpenter.sh/discovery": "my-eks-cluster",
            "aws:cloudformation:stack-id": "arn:aws:cloudformation:us-
east-2:111122223333:stack/eksctl-my-eks-cluster-cluster/e752ea00-e217-11ee-
beae-0a9599c8c7ed",
            "auto-delete": "no",
            "eksctl.cluster.k8s.io/v1alpha1/cluster-name": "my-eks-cluster",
            "EKS-Cluster-Name": "my-eks-cluster",
            "alpha.eksctl.io/cluster-oidc-enabled": "true",
```

```
            "aws:cloudformation:logical-id": "ControlPlane",
            "alpha.eksctl.io/eksctl-version": "0.173.0-dev
+a7ee89342.2024-03-01T03:40:57Z",
            "Name": "eksctl-my-eks-cluster-cluster/ControlPlane"
        },
        "health": {
            "issues": []
        },
        "accessConfig": {
            "authenticationMode": "API_AND_CONFIG_MAP"
        }
    }
}
```

- Per i dettagli sull'API, consulta DescribeCluster AWS CLICommand Reference.

`describe-fargate-profile`

Il seguente esempio di codice mostra come utilizzare`describe-fargate-profile`.

AWS CLI

Come descrivere un profilo Fargate

L'esempio `describe-fargate-profile` seguente descrive un profilo Fargate.

```
aws eks describe-fargate-profile \
    --cluster-name my-eks-cluster \
    --fargate-profile-name my-fargate-profile
```

Output:

```
{
    "fargateProfile": {
        "fargateProfileName": "my-fargate-profile",
        "fargateProfileArn": "arn:aws:eks:us-east-2:111122223333:fargateprofile/my-
eks-cluster/my-fargate-profile/96c766ce-43d2-f9c9-954c-647334391198",
        "clusterName": "my-eks-cluster",
        "createdAt": "2024-04-11T10:42:52.486000-04:00",
        "podExecutionRoleArn": "arn:aws:iam::111122223333:role/eksctl-my-eks-
cluster-farga-FargatePodExecutionRole-1htfAaJdJUEO",
        "subnets": [
            "subnet-09d912bb63ef21b9a",
```

```
            "subnet-04ad87f71c6e5ab4d",
            "subnet-0e2907431c9988b72"
        ],
        "selectors": [
            {
                "namespace": "prod*",
                "labels": {
                    "labelname*?": "*value1"
                }
            },
            {
                "namespace": "*dev*",
                "labels": {
                    "labelname*?": "*value*"
                }
            }
        ],
        "status": "ACTIVE",
        "tags": {
            "eks-fargate-profile-key-2": "value-2",
            "eks-fargate-profile-key-1": "value-1"
        }
    }
}
```

- Per i dettagli sull'API, consulta DescribeFargateProfile AWS CLICommand Reference.

`describe-identity-provider-config`

Il seguente esempio di codice mostra come utilizzare`describe-identity-provider-config`.

AWS CLI

Come descrivere una configurazione del gestore dell'identità associata al cluster Amazon EKS

L'esempio `describe-identity-provider-config` seguente descrive una configurazione del gestore dell'identità associata al cluster Amazon EKS in uso.

```
aws eks describe-identity-provider-config \
    --cluster-name my-eks-cluster \
    --identity-provider-config type=oidc,name=my-identity-provider
```

Output:

```
{
    "identityProviderConfig": {
        "oidc": {
            "identityProviderConfigName": "my-identity-provider",
            "identityProviderConfigArn": "arn:aws:eks:us-
east-2:111122223333:identityproviderconfig/my-eks-cluster/oidc/my-identity-
provider/8ac76722-78e4-cec1-ed76-d49eea058622",
            "clusterName": "my-eks-cluster",
            "issuerUrl": "https://oidc.eks.us-east-2.amazonaws.com/
id/38D6A4619A0A69E342B113ED7F1A7652",
            "clientId": "kubernetes",
            "usernameClaim": "email",
            "usernamePrefix": "my-username-prefix",
            "groupsClaim": "my-claim",
            "groupsPrefix": "my-groups-prefix",
            "requiredClaims": {
                "Claim1": "value1",
                "Claim2": "value2"
            },
            "tags": {
                "env": "dev"
            },
            "status": "ACTIVE"
        }
    }
}
```

Per ulteriori informazioni, consulta Autenticazione degli utenti per il cluster da un gestore dell'identità digitale OpenID Connect nella Guida per l'utente di Amazon EKS.

- Per i dettagli sull'API, consulta DescribeIdentityProviderConfig AWS CLICommand Reference.

`describe-insight`

Il seguente esempio di codice mostra come utilizzare`describe-insight`.

AWS CLI

Come ottenere i dettagli di un approfondimento per un cluster EKS utilizzando il relativo ID

L'esempio `describe-insight` seguente restituisce i dettagli di un approfondimento specificato utilizzando il nome del cluster e l'ID approfondimento.

```
aws eks describe-insight \
    --cluster-name eks-customer \
    --id 38ea7a64-a14f-4e0e-95c7-8dbcab3c3623
```

Output:

```
{
    "insight": {
        "id": "38ea7a64-a14f-4e0e-95c7-8dbcab3c3623",
        "name": "Kubelet version skew",
        "category": "UPGRADE_READINESS",
        "kubernetesVersion": "1.33",
        "lastRefreshTime": "2025-05-24T11:22:50-05:00",
        "lastTransitionTime": "2025-05-24T11:22:50-05:00",
        "description": "Checks for kubelet versions of worker nodes in the cluster
 to see if upgrade would cause noncompliance with supported Kubernetes kubelet
 version skew policy.",
        "insightStatus": {
            "status": "PASSING",
            "reason": "Node kubelet versions match the cluster control plane
 version."
        },
        "recommendation": "Upgrade your worker nodes to match the Kubernetes version
 of your cluster control plane.",
        "additionalInfo": {
            "Kubelet version skew policy": "https://kubernetes.io/releases/version-
skew-policy/#kubelet",
            "Updating a managed node group": "https://docs.aws.amazon.com/eks/
latest/userguide/update-managed-node-group.html"
        },
        "resources": [],
        "categorySpecificSummary": {
            "deprecationDetails": []
        }
    }
}
```

Per ulteriori informazioni, consulta Visualizzare gli approfondimenti sui cluster nella Guida per l'utente di Amazon EKS.

- Per i dettagli sull'API, consulta DescribeInsight AWS CLICommand Reference.

`describe-nodegroup`

Il seguente esempio di codice mostra come utilizzare`describe-nodegroup`.

AWS CLI

Come descrivere un gruppo di nodi gestiti per un cluster Amazon EKS

L'esempio `describe-nodegroup` seguente descrive un gruppo di nodi gestiti per un cluster Amazon EKS.

```
aws eks describe-nodegroup \
    --cluster-name my-eks-cluster \
    --nodegroup-name my-eks-nodegroup
```

Output:

```
{
    "nodegroup": {
        "nodegroupName": "my-eks-nodegroup",
        "nodegroupArn": "arn:aws:eks:us-east-2:111122223333:nodegroup/my-eks-
cluster/my-eks-nodegroup/a8c75f2f-df78-a72f-4063-4b69af3de5b1",
        "clusterName": "my-eks-cluster",
        "version": "1.26",
        "releaseVersion": "1.26.12-20240329",
        "createdAt": "2024-04-08T11:42:10.555000-04:00",
        "modifiedAt": "2024-04-08T11:44:12.402000-04:00",
        "status": "ACTIVE",
        "capacityType": "ON_DEMAND",
        "scalingConfig": {
            "minSize": 1,
            "maxSize": 3,
            "desiredSize": 1
        },
        "instanceTypes": [
            "t3.medium"
        ],
        "subnets": [
            "subnet-0e2907431c9988b72",
            "subnet-04ad87f71c6e5ab4d",
            "subnet-09d912bb63ef21b9a"
        ],
        "amiType": "AL2_x86_64",
```

```
        "nodeRole": "arn:aws:iam::111122223333:role/role-name",
        "labels": {},
        "resources": {
            "autoScalingGroups": [
                {
                    "name": "eks-my-eks-nodegroup-a8c75f2f-df78-
a72f-4063-4b69af3de5b1"
                }
            ]
        },
        "diskSize": 20,
        "health": {
            "issues": []
        },
        "updateConfig": {
            "maxUnavailable": 1
        },
        "tags": {}
    }
}
```

- Per i dettagli sull'API, consulta DescribeNodegroup AWS CLICommand Reference.

`describe-pod-identity-association`

Il seguente esempio di codice mostra come utilizzare`describe-pod-identity-association`.

AWS CLI

Come fornire i dettagli sull'associazione Pod Identity

L'esempio `describe-pod-identity-association` seguente descrive un'associazione Pod Identity nel cluster EKS.

```
aws eks describe-pod-identity-association \
    --cluster-name eks-customer \
    --association-id a-9njjin9gfghecgocd
```

Output:

```
{
```

```
    "association": {
        "clusterName": "eks-customer",
        "namespace": "default",
        "serviceAccount": "default",
        "roleArn": "arn:aws:iam::111122223333:role/my-role",
        "associationArn": "arn:aws:eks:us-
west-2:111122223333:podidentityassociation/eks-customer/a-9njjin9gfghecgocd",
        "associationId": "a-9njjin9gfghecgocd",
        "tags": {
            "Key2": "value2",
            "Key1": "value1"
        },
        "createdAt": "2025-05-24T19:52:14.135000-05:00",
        "modifiedAt": "2025-05-24T19:52:14.135000-05:00"
    }
}
```

Per ulteriori informazioni, consulta Scopri come EKS Pod Identity concede ai pod l'accesso ai AWS servizi nella Amazon EKS User Guide.

- Per i dettagli sull'API, consulta AWS CLI Command DescribePodIdentityAssociationReference.

`describe-update`

Il seguente esempio di codice mostra come utilizzare`describe-update`.

AWS CLI

Esempio 1: come descrivere un aggiornamento per un cluster

L'esempio `describe-update` seguente descrive un aggiornamento per un cluster denominato.

```
aws eks describe-update \
    --name my-eks-cluster \
    --update-id 10bddb13-a71b-425a-b0a6-71cd03e59161
```

Output:

```
{
    "update": {
        "id": "10bddb13-a71b-425a-b0a6-71cd03e59161",
        "status": "Successful",
```

```
        "type": "EndpointAccessUpdate",
        "params": [
            {
                "type": "EndpointPublicAccess",
                "value": "false"
            },
            {
                "type": "EndpointPrivateAccess",
                "value": "true"
            }
        ],
        "createdAt": "2024-03-14T10:01:26.297000-04:00",
        "errors": []
    }
}
```

Per ulteriori informazioni, consulta Aggiornamento della versione Kubernetes del cluster Amazon EKS nella Guida per l'utente di Amazon EKS.

Esempio 2: come descrivere un aggiornamento per un cluster

L'esempio `describe-update` seguente descrive un aggiornamento per un cluster denominato.

```
aws eks describe-update \
    --name my-eks-cluster \
    --update-id e4994991-4c0f-475a-a040-427e6da52966
```

Output:

```
{
    "update": {
        "id": "e4994991-4c0f-475a-a040-427e6da52966",
        "status": "Successful",
        "type": "AssociateEncryptionConfig",
        "params": [
            {
                "type": "EncryptionConfig",
                "value": "[{\"resources\":[\"secrets\"],\"provider\":{\"keyArn\":
\"arn:aws:kms:region-code:account:key/key\"}}]"
            }
        ],
        "createdAt": "2024-03-14T11:01:26.297000-04:00",
```

```
        "errors": []
    }
}
```

Per ulteriori informazioni, consulta Aggiornamento della versione Kubernetes del cluster Amazon EKS nella Guida per l'utente di Amazon EKS.

Esempio 3: come descrivere un aggiornamento per un cluster

L'esempio `describe-update` seguente descrive un aggiornamento per un cluster denominato.

```
aws eks describe-update \
    --name my-eks-cluster \
    --update-id b5f0ba18-9a87-4450-b5a0-825e6e84496f
```

Output:

```
{
    "update": {
        "id": "b5f0ba18-9a87-4450-b5a0-825e6e84496f",
        "status": "Successful",
        "type": "VersionUpdate",
        "params": [
            {
                "type": "Version",
                "value": "1.29"
            },
            {
                "type": "PlatformVersion",
                "value": "eks.1"
            }
        ],
        "createdAt": "2024-03-14T12:05:26.297000-04:00",
        "errors": []
    }
}
```

Per ulteriori informazioni, consulta Aggiornamento della versione Kubernetes del cluster Amazon EKS nella Guida per l'utente di Amazon EKS.

- Per i dettagli sull'API, consulta DescribeUpdate AWS CLICommand Reference.

`disassociate-access-policy`

Il seguente esempio di codice mostra come utilizzare`disassociate-access-policy`.

AWS CLI

Come annullare l'associazione tra una policy di accesso e una voce di accesso

Il comando `disassociate-access-policy` seguente rimuove la policy di accesso associata alla voce di accesso.

```
aws eks disassociate-access-policy \
    --cluster-name eks-customer \
    --principal-arn arn:aws:iam::111122223333:role/Admin \
    --policy-arn arn:aws:eks::aws:cluster-access-policy/AmazonEKSEditPolicy
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Associare policy di accesso a voci di accesso nella Guida per l'utente di Amazon EKS.

- Per i dettagli sull'API, consulta DisassociateAccessPolicy AWS CLICommand Reference.

`disassociate-identity-provider-config`

Il seguente esempio di codice mostra come utilizzare`disassociate-identity-provider-config`.

AWS CLI

Come annullare l'associazione tra un gestore dell'identità e il cluster Amazon EKS in uso

L'esempio `disassociate-identity-provider-config` seguente annulla l'associazione tra un gestore dell'identità e il cluster Amazon EKS in uso.

```
aws eks disassociate-identity-provider-config \
    --cluster-name my-eks-cluster \
    --identity-provider-config 'type=oidc,name=my-identity-provider'
```

Output:

```
{
    "update": {
```

```
        "id": "5f78d14e-c57b-4857-a3e4-cf664ae20949",
        "status": "InProgress",
        "type": "DisassociateIdentityProviderConfig",
        "params": [
            {
                "type": "IdentityProviderConfig",
                "value": "[]"
            }
        ],
        "createdAt": "2024-04-11T13:53:43.314000-04:00",
        "errors": []
    }
}
```

Per ulteriori informazioni, consulta Autenticazione degli utenti per il cluster da un gestore dell'identità digitale OpenID Connect - Dissociazione di un gestore dell'identità OIDC dal cluster nella Guida per l'utente di Amazon EKS.

- Per i dettagli sull'API, consulta DisassociateIdentityProviderConfig AWS CLICommand Reference.

`get-token`

Il seguente esempio di codice mostra come utilizzare`get-token`.

AWS CLI

Esempio 1: ottieni un token di autenticazione per un cluster Amazon EKS denominato `my-eks-cluster`

L'`get-token`esempio seguente ottiene un token di autenticazione per un cluster Amazon EKS denominato my-eks-cluster.

```
aws eks get-token \
    --cluster-name my-eks-cluster
```

Output:

```
{
    "kind": "ExecCredential",
    "apiVersion": "client.authentication.k8s.io/v1beta1",
    "spec": {},
```

```
    "status": {
        "expirationTimestamp": "2024-04-11T20:59:56Z",
        "token": "k8s-aws-v1.EXAMPLE_TOKEN_DATA_STRING..."
    }
}
```

Esempio 2: ottiene un token di autenticazione per un cluster Amazon EKS denominato `my-eks-cluster` assumendo questo ROLearn per le credenziali al momento della firma del token

L'`get-token`esempio seguente ottiene un token di autenticazione per un cluster Amazon EKS denominato my-eks-cluster assumendo questo ROLearn per le credenziali al momento della firma del token.

```
aws eks get-token \
    --cluster-name my-eks-cluster \
    --role-arn arn:aws:iam::111122223333:role/eksctl-EKS-Linux-Cluster-v1-24-cluster-ServiceRole-j1k7AfTIQtnM
```

Output:

```
{
    "kind": "ExecCredential",
    "apiVersion": "client.authentication.k8s.io/v1beta1",
    "spec": {},
    "status": {
        "expirationTimestamp": "2024-04-11T21:05:26Z",
        "token": "k8s-aws-v1.EXAMPLE_TOKEN_DATA_STRING..."
    }
}
```

- Per i dettagli sull'API, consulta Command GetTokenReference AWS CLI .

`list-access-entries`

Il seguente esempio di codice mostra come utilizzare`list-access-entries`.

AWS CLI

Come elencare le voci di accesso per un cluster EKS

Il comando `list-access-entries` seguente restituisce l'elenco delle voci di accesso associate al cluster EKS `eks-customer`.

```
aws eks list-access-entries \
    --cluster-name eks-customer
```

Output:

```
{
    "accessEntries": [
        "arn:aws:iam::111122223333:role/Admin",
        "arn:aws:iam::111122223333:role/admin-test-ip",
        "arn:aws:iam::111122223333:role/assume-worker-node-role",
        "arn:aws:iam::111122223333:user/eks-admin-user"
    ]
}
```

Per ulteriori informazioni, consulta Concedere a utenti IAM l'accesso a Kubernetes con le voci di accesso EKS nella Guida per l'utente di Amazon EKS.

- Per i dettagli sull'API, consulta ListAccessEntries AWS CLICommand Reference.

`list-access-policies`

Il seguente esempio di codice mostra come utilizzare`list-access-policies`.

AWS CLI

Come elencare tutte le policy di accesso disponibili

L'esempio `list-access-policies` seguente restituisce l'elenco di tutte le policy di accesso disponibili.

```
aws eks list-access-policies
```

Output:

```
{
    "accessPolicies": [
        {
            "name": "AmazonEKSAdminPolicy",
            "arn": "arn:aws:eks::aws:cluster-access-policy/AmazonEKSAdminPolicy"
        },
        {
            "name": "AmazonEKSAdminViewPolicy",
```

```
            "arn": "arn:aws:eks::aws:cluster-access-policy/AmazonEKSAdminViewPolicy"
        },
        {
            "name": "AmazonEKSAutoNodePolicy",
            "arn": "arn:aws:eks::aws:cluster-access-policy/AmazonEKSAutoNodePolicy"
        },
        {
            "name": "AmazonEKSBlockStorageClusterPolicy",
            "arn": "arn:aws:eks::aws:cluster-access-policy/
AmazonEKSBlockStorageClusterPolicy"
        },
        {
            "name": "AmazonEKSBlockStoragePolicy",
            "arn": "arn:aws:eks::aws:cluster-access-policy/
AmazonEKSBlockStoragePolicy"
        },
        {
            "name": "AmazonEKSClusterAdminPolicy",
            "arn": "arn:aws:eks::aws:cluster-access-policy/
AmazonEKSClusterAdminPolicy"
        },
        {
            "name": "AmazonEKSComputeClusterPolicy",
            "arn": "arn:aws:eks::aws:cluster-access-policy/
AmazonEKSComputeClusterPolicy"
        },
        {
            "name": "AmazonEKSComputePolicy",
            "arn": "arn:aws:eks::aws:cluster-access-policy/AmazonEKSComputePolicy"
        },
        {
            "name": "AmazonEKSEditPolicy",
            "arn": "arn:aws:eks::aws:cluster-access-policy/AmazonEKSEditPolicy"
        },
        {
            "name": "AmazonEKSHybridPolicy",
            "arn": "arn:aws:eks::aws:cluster-access-policy/AmazonEKSHybridPolicy"
        },
        {
            "name": "AmazonEKSLoadBalancingClusterPolicy",
            "arn": "arn:aws:eks::aws:cluster-access-policy/
AmazonEKSLoadBalancingClusterPolicy"
        },
        {
```

```
            "name": "AmazonEKSLoadBalancingPolicy",
            "arn": "arn:aws:eks::aws:cluster-access-policy/
AmazonEKSLoadBalancingPolicy"
        },
        {
            "name": "AmazonEKSNetworkingClusterPolicy",
            "arn": "arn:aws:eks::aws:cluster-access-policy/
AmazonEKSNetworkingClusterPolicy"
        },
        {
            "name": "AmazonEKSNetworkingPolicy",
            "arn": "arn:aws:eks::aws:cluster-access-policy/
AmazonEKSNetworkingPolicy"
        },
        {
            "name": "AmazonEKSViewPolicy",
            "arn": "arn:aws:eks::aws:cluster-access-policy/AmazonEKSViewPolicy"
        },
        {
            "name": "AmazonEMRJobPolicy",
            "arn": "arn:aws:eks::aws:cluster-access-policy/AmazonEMRJobPolicy"
        },
        {
            "name": "AmazonSagemakerHyperpodClusterPolicy",
            "arn": "arn:aws:eks::aws:cluster-access-policy/
AmazonSagemakerHyperpodClusterPolicy"
        },
        {
            "name": "AmazonSagemakerHyperpodControllerPolicy",
            "arn": "arn:aws:eks::aws:cluster-access-policy/
AmazonSagemakerHyperpodControllerPolicy"
        },
        {
            "name": "AmazonSagemakerHyperpodSystemNamespacePolicy",
            "arn": "arn:aws:eks::aws:cluster-access-policy/
AmazonSagemakerHyperpodSystemNamespacePolicy"
        }
    ]
}
```

Per ulteriori informazioni, consulta Associare policy di accesso a voci di accesso nella Guida per l'utente di Amazon EKS.

- Per i dettagli sull'API, consulta ListAccessPolicies AWS CLICommand Reference.

`list-addons`

Il seguente esempio di codice mostra come utilizzare`list-addons`.

AWS CLI

Elenca tutti i componenti aggiuntivi installati nel tuo cluster Amazon EKS denominati `` my-eks-cluster

L'`list-addons`esempio seguente elenca tutti i componenti aggiuntivi installati nel cluster Amazon EKS denominato my-eks-cluster.

```
aws eks list-addons \
    --cluster-name my-eks-cluster
```

Output:

```
{
    "addons": [
        "kube-proxy",
        "vpc-cni"
    ]
}
```

- Per i dettagli sull'API, consulta ListAddons AWS CLICommand Reference.

`list-associated-access-policies`

Il seguente esempio di codice mostra come utilizzare`list-associated-access-policies`.

AWS CLI

Come elencare le policy di accesso associate a una voce di accesso

L'esempio `list-associated-access-policies` seguente restituisce l'elenco delle policy di accesso associate a una voce di accesso nel cluster EKS.

```
aws eks list-associated-access-policies \
    --cluster-name eks-customer \
    --principal-arn arn:aws:iam::111122223333:role/Admin
```

Output:

```
{
    "associatedAccessPolicies": [
        {
            "policyArn": "arn:aws:eks::aws:cluster-access-policy/
AmazonEKSAdminPolicy",
            "accessScope": {
                "type": "cluster",
                "namespaces": []
            },
            "associatedAt": "2025-05-24T17:26:22.935000-05:00",
            "modifiedAt": "2025-05-24T17:26:22.935000-05:00"
        }
    ],
    "clusterName": "eks-customer",
    "principalArn": "arn:aws:iam::111122223333:role/Admin"
}
```

Per ulteriori informazioni, consulta Concedere a utenti IAM l'accesso a Kubernetes con le voci di accesso EKS nella Guida per l'utente di Amazon EKS.

- Per i dettagli sull'API, consulta ListAssociatedAccessPolicies AWS CLICommand Reference.

`list-clusters`

Il seguente esempio di codice mostra come utilizzare`list-clusters`.

AWS CLI

Per elencare tutti i componenti aggiuntivi installati nel cluster Amazon EKS denominati `` my-eks-cluster

L'`list-clusters`esempio seguente elenca tutti i componenti aggiuntivi installati nel cluster Amazon EKS denominato my-eks-cluster.

```
aws eks list-clusters
```

Output:

```
{
    "clusters": [
        "prod",
        "qa",
```

```
        "stage",
        "my-eks-cluster"
    ]
}
```

- Per i dettagli sull'API, consulta ListClusters AWS CLICommand Reference.

`list-fargate-profiles`

Il seguente esempio di codice mostra come utilizzare`list-fargate-profiles`.

AWS CLI

Per elencare tutti i profili fargate nel tuo cluster Amazon EKS denominati `` my-eks-cluster

L'`list-fargate-profiles`esempio seguente elenca tutti i profili fargate my-eks-cluster denominati nel cluster Amazon EKS.

```
aws eks list-fargate-profiles \
    --cluster-name my-eks-cluster
```

Output:

```
{
    "fargateProfileNames": [
        "my-fargate-profile"
    ]
}
```

- Per i dettagli sull'API, consulta ListFargateProfiles AWS CLICommand Reference.

`list-identity-provider-configs`

Il seguente esempio di codice mostra come utilizzare`list-identity-provider-configs`.

AWS CLI

Come elencare i gestori dell'identità associati al cluster Amazon EKS

L'esempio `list-identity-provider-configs` seguente elenca i gestori dell'identità associati al cluster Amazon EKS in uso.

```
aws eks list-identity-provider-configs \
    --cluster-name my-eks-cluster
```

Output:

```
{
    "identityProviderConfigs": [
        {
            "type": "oidc",
            "name": "my-identity-provider"
        }
    ]
}
```

Per ulteriori informazioni, consulta Autenticazione degli utenti per il cluster da un gestore dell'identità digitale OpenID Connect nella Guida per l'utente di Amazon EKS.

- Per i dettagli sull'API, consulta ListIdentityProviderConfigs AWS CLICommand Reference.

`list-insights`

Il seguente esempio di codice mostra come utilizzare`list-insights`.

AWS CLI

Come elencare tutti gli approfondimenti per il cluster specificato

L'esempio `list-insights` seguente restituisce l'elenco di tutti gli approfondimenti verificati rispetto al cluster specificato.

```
aws eks list-insights \
--cluster-name eks-customer
```

Output:

```
{
    "insights": [
        {
            "id": "38ea7a64-a14f-4e0e-95c7-8dbcab3c3616",
            "name": "Kubelet version skew",
            "category": "UPGRADE_READINESS",
            "kubernetesVersion": "1.33",
```

```
            "lastRefreshTime": "2025-05-24T11:22:50-05:00",
            "lastTransitionTime": "2025-05-24T11:22:50-05:00",
            "description": "Checks for kubelet versions of worker nodes in the
cluster to see if upgrade would cause noncompliance with supported Kubernetes
kubelet version skew policy.",
            "insightStatus": {
                "status": "PASSING",
                "reason": "Node kubelet versions match the cluster control plane
version."
            }
        },
        {
            "id": "9cd91472-f99c-45a9-b7d7-54d4900dee23",
            "name": "EKS add-on version compatibility",
            "category": "UPGRADE_READINESS",
            "kubernetesVersion": "1.33",
            "lastRefreshTime": "2025-05-24T11:22:59-05:00",
            "lastTransitionTime": "2025-05-24T11:22:50-05:00",
            "description": "Checks version of installed EKS add-ons to ensure they
are compatible with the next version of Kubernetes. ",
            "insightStatus": {
                "status": "PASSING",
                "reason": "All installed EKS add-on versions are compatible with
next Kubernetes version."
            }
        },
        {
            "id": "0deb269d-b1e1-458c-a2b4-7a57f940c875",
            "name": "Cluster health issues",
            "category": "UPGRADE_READINESS",
            "kubernetesVersion": "1.33",
            "lastRefreshTime": "2025-05-24T11:22:59-05:00",
            "lastTransitionTime": "2025-05-24T11:22:50-05:00",
            "description": "Checks for any cluster health issues that prevent
successful upgrade to the next Kubernetes version on EKS.",
            "insightStatus": {
                "status": "PASSING",
                "reason": "No cluster health issues detected."
            }
        },
        {
            "id": "214fa274-344f-420b-812a-5049ce72c9ww",
            "name": "kube-proxy version skew",
            "category": "UPGRADE_READINESS",
```

```
            "kubernetesVersion": "1.33",
            "lastRefreshTime": "2025-05-24T11:22:50-05:00",
            "lastTransitionTime": "2025-05-24T11:22:50-05:00",
            "description": "Checks version of kube-proxy in cluster to see if
 upgrade would cause noncompliance with supported Kubernetes kube-proxy version skew
 policy.",
            "insightStatus": {
                "status": "PASSING",
                "reason": "kube-proxy versions match the cluster control plane
 version."
            }
        }
    ]
}
```

Per ulteriori informazioni, consulta Visualizzare gli approfondimenti sui cluster nella Guida per l'utente di Amazon EKS.

- Per i dettagli sull'API, consulta ListInsights AWS CLICommand Reference.

`list-nodegroups`

Il seguente esempio di codice mostra come utilizzare`list-nodegroups`.

AWS CLI

Come elencare tutti i gruppi di nodi in un cluster Amazon EKS

L'esempio `list-nodegroups` seguente elenca tutti i gruppi di nodi in un cluster Amazon EKS.

```
aws eks list-nodegroups \
    --cluster-name my-eks-cluster
```

Output:

```
{
    "nodegroups": [
        "my-eks-managed-node-group",
        "my-eks-nodegroup"
    ]
}
```

- Per i dettagli sull'API, consulta ListNodegroups AWS CLICommand Reference.

`list-pod-identity-associations`

Il seguente esempio di codice mostra come utilizzare`list-pod-identity-associations`.

AWS CLI

Esempio 1: come elencare le associazioni Pod Identity in un cluster EKS

Il comando `list-pod-identity-associations` seguente restituisce l'elenco delle associazioni Pod Identity nel cluster EKS denominate `eks-customer` in tutti i namespace e gli account del servizio.

```
aws eks list-pod-identity-associations \
    --cluster-name eks-customer
```

Output:

```
{
    "associations": [
        {
            "clusterName": "eks-customer",
            "namespace": "default",
            "serviceAccount": "default",
            "associationArn": "arn:aws:eks:us-
west-2:111122223333:podidentityassociation/eks-customer/a-9njjin9gfghecgocd",
            "associationId": "a-9njjin9gfghecgocd"
        },
        {
            "clusterName": "eks-customer",
            "namespace": "kube-system",
            "serviceAccount": "eks-customer",
            "associationArn": "arn:aws:eks:us-
west-2:111122223333:podidentityassociation/eks-customer/a-dvtacahdvjn01ffbc",
            "associationId": "a-dvtacahdvjn01ffbc"
        },
        {
            "clusterName": "eks-customer",
            "namespace": "kube-system",
            "serviceAccount": "coredns",
            "associationArn": "arn:aws:eks:us-
west-2:111122223333:podidentityassociation/eks-customer/a-yrpsdroc4ei7k6xps",
            "associationId": "a-yrpsdroc4ei7k6xps"
```

```
        }
    ]
}
```

Per ulteriori informazioni, consulta Scopri come EKS Pod Identity concede ai pod l'accesso ai AWS servizi nella Amazon EKS User Guide.

Esempio 2: come elencare le associazioni Pod Identity in un cluster EKS in base al namespace e all'account del servizio

Il comando `list-pod-identity-associations` seguente restituisce l'elenco delle associazioni Pod Identity nel cluster EKS in base al namespace e all'account del servizio.

```
aws eks list-pod-identity-associations \
    --cluster-name eks-customer \
    --namespace kube-system \
    --service-account eks-customer
```

Output:

```
{
    "associations": [
        {
            "clusterName": "eks-customer",
            "namespace": "kube-system",
            "serviceAccount": "eks-customer",
            "associationArn": "arn:aws:eks:us-
west-2:111122223333:podidentityassociation/eks-customer/a-dvtacahdvjn01ffbc",
            "associationId": "a-dvtacahdvjn01ffbc"
        }
    ]
}
```

Per ulteriori informazioni, consulta Scopri come EKS Pod Identity concede ai pod l'accesso ai AWS servizi nella Amazon EKS User Guide.

- Per i dettagli sull'API, consulta AWS CLI Command ListPodIdentityAssociationsReference.

`list-tags-for-resource`

Il seguente esempio di codice mostra come utilizzare`list-tags-for-resource`.

AWS CLI

Esempio 1: come elencare tutti i tag per l'ARN di un cluster Amazon EKS

L'esempio `list-tags-for-resource` seguente elenca tutti i tag per l'ARN di un cluster Amazon EKS.

```
aws eks list-tags-for-resource \
    --resource-arn arn:aws:eks:us-east-2:111122223333:cluster/my-eks-cluster
```

Output:

```
{
    "tags": {
        "aws:cloudformation:stack-name": "eksctl-my-eks-cluster-cluster",
        "alpha.eksctl.io/cluster-name": "my-eks-cluster",
        "karpenter.sh/discovery": "my-eks-cluster",
        "aws:cloudformation:stack-id": "arn:aws:cloudformation:us-
east-2:111122223333:stack/eksctl-my-eks-cluster-cluster/e752ea00-e217-11ee-
beae-0a9599c8c7ed",
        "auto-delete": "no",
        "eksctl.cluster.k8s.io/v1alpha1/cluster-name": "my-eks-cluster",
        "EKS-Cluster-Name": "my-eks-cluster",
        "alpha.eksctl.io/cluster-oidc-enabled": "true",
        "aws:cloudformation:logical-id": "ControlPlane",
        "alpha.eksctl.io/eksctl-version": "0.173.0-dev
+a7ee89342.2024-03-01T03:40:57Z",
        "Name": "eksctl-my-eks-cluster-cluster/ControlPlane"
    }
}
```

Esempio 2: come elencare tutti i tag per l'ARN di un gruppo di nodi Amazon EKS

L'esempio `list-tags-for-resource` seguente elenca tutti i tag per l'ARN di un gruppo di nodi Amazon EKS.

```
aws eks list-tags-for-resource \
    --resource-arn arn:aws:eks:us-east-2:111122223333:nodegroup/my-eks-cluster/my-
eks-managed-node-group/60c71ed2-2cfb-020f-a5f4-ad32477f198c
```

Output:

```
{
    "tags": {
        "aws:cloudformation:stack-name": "eksctl-my-eks-cluster-nodegroup-my-eks-
managed-node-group",
        "aws:cloudformation:stack-id": "arn:aws:cloudformation:us-
east-2:111122223333:stack/eksctl-my-eks-cluster-nodegroup-my-eks-managed-node-group/
eaa20310-e219-11ee-b851-0ab9ad8228ff",
        "eksctl.cluster.k8s.io/v1alpha1/cluster-name": "my-eks-cluster",
        "EKS-Cluster-Name": "my-eks-cluster",
        "alpha.eksctl.io/nodegroup-type": "managed",
        "NodeGroup Name 1": "my-eks-managed-node-group",
        "k8s.io/cluster-autoscaler/enabled": "true",
        "nodegroup-role": "worker",
        "alpha.eksctl.io/cluster-name": "my-eks-cluster",
        "alpha.eksctl.io/nodegroup-name": "my-eks-managed-node-group",
        "karpenter.sh/discovery": "my-eks-cluster",
        "NodeGroup Name 2": "AmazonLinux-Linux-Managed-NG-v1-26-v1",
        "auto-delete": "no",
        "k8s.io/cluster-autoscaler/my-eks-cluster": "owned",
        "aws:cloudformation:logical-id": "ManagedNodeGroup",
        "alpha.eksctl.io/eksctl-version": "0.173.0-dev
+a7ee89342.2024-03-01T03:40:57Z"
    }
}
```

Esempio 3: elencare tutti i tag su un profilo Amazon EKS Fargate ARNe

L'esempio `list-tags-for-resource` seguente elenca tutti i tag per l'ARN di un profilo Fargate in Amazon EKS.

```
aws eks list-tags-for-resource  \
    --resource-arn arn:aws:eks:us-east-2:111122223333:fargateprofile/my-eks-cluster/
my-fargate-profile/d6c76780-e541-0725-c816-36754cab734b
```

Output:

```
{
    "tags": {
        "eks-fargate-profile-key-2": "value-2",
        "eks-fargate-profile-key-1": "value-1"
    }
}
```

Esempio 4: come elencare tutti i tag per l'ARN di un componente aggiuntivo Amazon EKS

L'esempio `list-tags-for-resource` seguente elenca tutti i tag per l'ARN di un componente aggiuntivo di Amazon EKS.

```
aws eks list-tags-for-resource \
    --resource-arn arn:aws:eks:us-east-2:111122223333:addon/my-eks-cluster/vpc-cni/0ec71efc-98dd-3203-60b0-4b939b2a5e5f
```

Output:

```
{
    "tags": {
        "eks-addon-key-2": "value-2",
        "eks-addon-key-1": "value-1"
    }
}
```

Esempio 5: come elencare tutti i tag per l'ARN di un gestore dell'identità OIDC di Amazon EKS

L'esempio `list-tags-for-resource` seguente elenca tutti i tag per l'ARN di un gestore dell'identità OIDC in Amazon EKS.

```
aws eks list-tags-for-resource \
    --resource-arn arn:aws:eks:us-east-2:111122223333:identityproviderconfig/my-eks-cluster/oidc/my-identity-provider/8ac76722-78e4-cec1-ed76-d49eea058622
```

Output:

```
{
    "tags": {
        "my-identity-provider": "test"
    }
}
```

- Per i dettagli sull'API, consulta ListTagsForResource AWS CLICommand Reference.

`list-update`

Il seguente esempio di codice mostra come utilizzare`list-update`.

AWS CLI

Esempio 1: come elencare gli aggiornamenti associati a un nome di cluster Amazon EKS

L'`list-updates`esempio seguente elenca tutti gli aggiornamenti IDs per un nome di cluster Amazon EKS.

```
aws eks list-updates \
    --name my-eks-cluster
```

Output:

```
{
    "updateIds": [
        "5f78d14e-c57b-4857-a3e4-cf664ae20949",
        "760e5a3f-adad-48c7-88d3-7ac283c09c26",
        "cd4ec863-bc55-47d5-a377-3971502f529b",
        "f12657ce-e869-4f17-b158-a82ab8b7d937"
    ]
}
```

Esempio 2: elencare tutti gli aggiornamenti IDs per un gruppo di nodi Amazon EKS

L'`list-updates`esempio seguente elenca tutti gli aggiornamenti IDs per un gruppo di nodi Amazon EKS.

```
aws eks list-updates \
    --name my-eks-cluster \
    --nodegroup-name my-eks-managed-node-group
```

Output:

```
{
    "updateIds": [
        "8c6c1bef-61fe-42ac-a242-89412387b8e7"
    ]
}
```

Esempio 3: elencare tutti gli aggiornamenti IDs su un componente aggiuntivo Amazon EKS

L'`list-updates`esempio seguente elenca tutti gli aggiornamenti IDs per un componente aggiuntivo Amazon EKS.

```
aws eks list-updates \
    --name my-eks-cluster \
    --addon-name vpc-cni
```

Output:

```
{
    "updateIds": [
        "9cdba8d4-79fb-3c83-afe8-00b508d33268"
    ]
}
```

- Per i dettagli sull'API, consulta ListUpdate AWS CLICommand Reference.

`list-updates`

Il seguente esempio di codice mostra come utilizzare`list-updates`.

AWS CLI

Come elencare gli aggiornamenti per un cluster

Questo comando di esempio elenca gli aggiornamenti correnti per un cluster denominato `example` nella Regione predefinita.

Comando:

```
aws eks list-updates --name example
```

Output:

```
{
    "updateIds": [
        "10bddb13-a71b-425a-b0a6-71cd03e59161"
    ]
}
```

- Per i dettagli sull'API, consulta ListUpdates AWS CLICommand Reference.

`register-cluster`

Il seguente esempio di codice mostra come utilizzare`register-cluster`.

AWS CLI

Esempio 1: registra un cluster Kubernetes EKS_ANYWHERE esterno in Amazon EKS

L'esempio `register-cluster` seguente registra un cluster Kubernetes esterno in Amazon EKS.

```
aws eks register-cluster \
    --name my-eks-anywhere-cluster \
    --connector-config 'roleArn=arn:aws:iam::111122223333:role/
AmazonEKSConnectorAgentRole,provider=EKS_ANYWHERE'
```

Output:

```
{
    "cluster": {
        "name": "my-eks-anywhere-cluster",
        "arn": "arn:aws:eks:us-east-2:111122223333:cluster/my-eks-anywhere-cluster",
        "createdAt": "2024-04-12T12:38:37.561000-04:00",
        "status": "PENDING",
        "tags": {},
        "connectorConfig": {
            "activationId": "xxxxxxxxACTIVATION_IDxxxxxxxx",
            "activationCode": "xxxxxxxxACTIVATION_CODExxxxxxxx",
            "activationExpiry": "2024-04-15T12:38:37.082000-04:00",
            "provider": "EKS_ANYWHERE",
            "roleArn": "arn:aws:iam::111122223333:role/AmazonEKSConnectorAgentRole"
        }
    }
}
```

Per ulteriori informazioni, consulta Connessione di un cluster esterno nella Guida per l'utente di Amazon EKS.

Esempio 2: registra un cluster Kubernetes esterno in Amazon EKS

L'esempio `register-cluster` seguente registra un cluster Kubernetes esterno in Amazon EKS.

```
aws eks register-cluster \
    --name my-eks-anywhere-cluster \
    --connector-config 'roleArn=arn:aws:iam::111122223333:role/
AmazonEKSConnectorAgentRole,provider=OTHER'
```

Output:

```
{
    "cluster": {
        "name": "my-onprem-k8s-cluster",
        "arn": "arn:aws:eks:us-east-2:111122223333:cluster/my-onprem-k8s-cluster",
        "createdAt": "2024-04-12T12:42:10.861000-04:00",
        "status": "PENDING",
        "tags": {},
        "connectorConfig": {
            "activationId": "xxxxxxxxACTIVATION_IDxxxxxxxx",
            "activationCode": "xxxxxxxxACTIVATION_CODExxxxxxxx",
            "activationExpiry": "2024-04-15T12:42:10.339000-04:00",
            "provider": "OTHER",
            "roleArn": "arn:aws:iam::111122223333:role/AmazonEKSConnectorAgentRole"
        }
    }
}
```

Per ulteriori informazioni, consulta Connessione di un cluster esterno nella Guida per l'utente di Amazon EKS.

- Per i dettagli sull'API, consulta RegisterCluster AWS CLICommand Reference.

`tag-resource`

Il seguente esempio di codice mostra come utilizzare`tag-resource`.

AWS CLI

Esempio 1: come aggiungere i tag specificati a un cluster Amazon EKS

L'esempio `tag-resource` seguente aggiunge i tag specificati a un cluster Amazon EKS.

```
aws eks tag-resource \
    --resource-arn arn:aws:eks:us-east-2:111122223333:cluster/my-eks-cluster \
```

```
    --tag 'my-eks-cluster-test-1=test-value-1,my-eks-cluster-dev-1=dev-value-2'
```

Questo comando non produce alcun output.

Esempio 2: come aggiungere i tag specificati a un gruppo di nodi Amazon EKS

L'esempio `tag-resource` seguente aggiunge i tag specificati a un gruppo di nodi Amazon EKS.

```
aws eks tag-resource \
    --resource-arn arn:aws:eks:us-east-2:111122223333:nodegroup/my-eks-cluster/my-eks-managed-node-group/60c71ed2-2cfb-020f-a5f4-ad32477f198c \
    --tag 'my-eks-nodegroup-test-1=test-value-1,my-eks-nodegroup-dev-1=dev-value-2'
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta TagResource AWS CLICommand Reference.

`untag-resource`

Il seguente esempio di codice mostra come utilizzare`untag-resource`.

AWS CLI

Esempio 1: come eliminare i tag specificati da un cluster Amazon EKS

L'esempio `untag-resource` seguente elimina i tag specificati da un cluster Amazon EKS.

```
aws eks untag-resource \
    --resource-arn arn:aws:eks:us-east-2:111122223333:cluster/my-eks-cluster \
    --tag-keys "my-eks-cluster-test-1" "my-eks-cluster-dev-1"
```

Questo comando non produce alcun output.

Esempio 2: come eliminare i tag specificati da un gruppo di nodi Amazon EKS

L'esempio `untag-resource` seguente elimina i tag specificati da un gruppo di nodi Amazon EKS.

```
aws eks untag-resource \
    --resource-arn arn:aws:eks:us-east-2:111122223333:nodegroup/my-eks-cluster/my-eks-managed-node-group/60c71ed2-2cfb-020f-a5f4-ad32477f198c \
    --tag-keys "my-eks-nodegroup-test-1" "my-eks-nodegroup-dev-1"
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta UntagResource AWS CLICommand Reference.

`update-access-entry`

Il seguente esempio di codice mostra come utilizzare`update-access-entry`.

AWS CLI

Come aggiornare una voce di accesso per un cluster EKS

Il comando `update-access-entry` seguente aggiorna una voce di accesso per il cluster EKS aggiungendo il gruppo Kubernetes `tester`.

```
aws eks update-access-entry \
    --cluster-name eks-customer \
    --principal-arn arn:aws:iam::111122223333:role/Admin \
    --kubernetes-groups tester
```

Output:

```
{
    "accessEntry": {
        "clusterName": "eks-customer",
        "principalArn": "arn:aws:iam::111122223333:role/Admin",
        "kubernetesGroups": [
            "tester"
        ],
        "accessEntryArn": "arn:aws:eks:us-west-2:111122223333:access-entry/eks-
customer/role/111122223333/Admin/d2cb8183-d6ec-b82a-d967-eca21902a4b4",
        "createdAt": "2025-05-24T11:02:04.432000-05:00",
        "modifiedAt": "2025-05-24T17:08:01.608000-05:00",
        "tags": {},
        "username": "arn:aws:sts::111122223333:assumed-role/Admin/{{SessionName}}",
        "type": "STANDARD"
    }
}
```

Per ulteriori informazioni, consulta Aggiornare le voci di accesso nella Guida per l'utente di Amazon EKS.

- Per i dettagli sull'API, consulta UpdateAccessEntry AWS CLICommand Reference.

`update-addon`

Il seguente esempio di codice mostra come utilizzare`update-addon`.

AWS CLI

Esempio 1: Come aggiornare un componente aggiuntivo Amazon EKS con l'ARN del ruolo dell'account del servizio

L'esempio `update-addon` seguente aggiorna un componente aggiuntivo Amazon EKS con l'ARN del ruolo dell'account del servizio.

```
aws eks update-addon \
    --cluster-name my-eks-cluster \
    --addon-name vpc-cni \
    --service-account-role-arn arn:aws:iam::111122223333:role/eksctl-my-eks-cluster-addon-vpc-cni-Role1-YfakrqOC1UTm
```

Output:

```
{
    "update": {
        "id": "c00d2de2-c2e4-3d30-929e-46b8edec2ce4",
        "status": "InProgress",
        "type": "AddonUpdate",
        "params": [
            {
                "type": "ServiceAccountRoleArn",
                "value": "arn:aws:iam::111122223333:role/eksctl-my-eks-cluster-addon-vpc-cni-Role1-YfakrqOC1UTm"
            }
        ],
        "updatedAt": "2024-04-12T16:04:55.614000-04:00",
        "errors": []
    }
}
```

Per ulteriori informazioni, consulta Gestione dei componenti aggiuntivi di Amazon EKS - Aggiornamento di un componente aggiuntivo nella Guida per l'utente di Amazon EKS.

Esempio 2: Come aggiornare un componente aggiuntivo Amazon EKS con una versione specifica

L'esempio `update-addon` seguente aggiorna un componente aggiuntivo Amazon EKS con una versione specifica.

```
aws eks update-addon \
    --cluster-name my-eks-cluster \
    --addon-name vpc-cni \
    --service-account-role-arn arn:aws:iam::111122223333:role/eksctl-my-eks-cluster-addon-vpc-cni-Role1-YfakrqOC1UTm \
    --addon-version v1.16.4-eksbuild.2
```

Output:

```
{
    "update": {
        "id": "f58dc0b0-2b18-34bd-bc6a-e4abc0011f36",
        "status": "InProgress",
        "type": "AddonUpdate",
        "params": [
            {
                "type": "AddonVersion",
                "value": "v1.16.4-eksbuild.2"
            },
            {
                "type": "ServiceAccountRoleArn",
                "value": "arn:aws:iam::111122223333:role/eksctl-my-eks-cluster-addon-vpc-cni-Role1-YfakrqOC1UTm"
            }
        ],
        "createdAt": "2024-04-12T16:07:16.550000-04:00",
        "errors": []
    }
}
```

Per ulteriori informazioni, consulta Gestione dei componenti aggiuntivi di Amazon EKS - Aggiornamento di un componente aggiuntivo nella Guida per l'utente di Amazon EKS.

Esempio 3: Come aggiornare un componente aggiuntivo Amazon EKS con valori di configurazione personalizzati e risolvere i dettagli dei conflitti

L'esempio `update-addon` seguente aggiorna un componente aggiuntivo Amazon EKS con valori di configurazione personalizzati e risolve i dettagli dei conflitti.

```
aws eks update-addon \
    --cluster-name my-eks-cluster \
    --addon-name vpc-cni \
    --service-account-role-arn arn:aws:iam::111122223333:role/eksctl-my-eks-cluster-addon-vpc-cni-Role1-YfakrqOC1UTm \
    --addon-version v1.16.4-eksbuild.2 \
    --configuration-values '{"resources": {"limits":{"cpu":"100m"}, "requests": {"cpu":"50m"}}}' \
    --resolve-conflicts PRESERVE
```

Output:

```
{
    "update": {
        "id": "cd9f2173-a8d8-3004-a90f-032f14326520",
        "status": "InProgress",
        "type": "AddonUpdate",
        "params": [
            {
                "type": "AddonVersion",
                "value": "v1.16.4-eksbuild.2"
            },
            {
                "type": "ServiceAccountRoleArn",
                "value": "arn:aws:iam::111122223333:role/eksctl-my-eks-cluster-addon-vpc-cni-Role1-YfakrqOC1UTm"
            },
            {
                "type": "ResolveConflicts",
                "value": "PRESERVE"
            },
            {
                "type": "ConfigurationValues",
                "value": "{\"resources\": {\"limits\":{\"cpu\":\"100m\"}, \"requests\":{\"cpu\":\"50m\"}}}"
            }
        ],
        "createdAt": "2024-04-12T16:16:27.363000-04:00",
        "errors": []
    }
}
```

Per ulteriori informazioni, consulta Gestione dei componenti aggiuntivi di Amazon EKS - Aggiornamento di un componente aggiuntivo nella Guida per l'utente di Amazon EKS.

Esempio 4: Come aggiornare un componente aggiuntivo Amazon EKS con un file di valori di configurazione JSON personalizzati

L'esempio `update-addon` seguente aggiorna un componente aggiuntivo Amazon EKS con valori di configurazione JSON personalizzati e risolve i dettagli dei conflitti.

```
aws eks update-addon \
    --cluster-name my-eks-cluster \
    --addon-name vpc-cni \
    --service-account-role-arn arn:aws:iam::111122223333:role/eksctl-my-eks-cluster-addon-vpc-cni-Role1-YfakrqOC1UTm \
    --addon-version v1.17.1-eksbuild.1 \
    --configuration-values 'file://configuration-values.json' \
    --resolve-conflicts PRESERVE
```

Contenuto di `configuration-values.json`:

```
{
    "resources": {
        "limits": {
            "cpu": "100m"
        },
        "requests": {
            "cpu": "50m"
        }
    },
    "env": {
        "AWS_VPC_K8S_CNI_LOGLEVEL": "ERROR"
    }
}
```

Output:

```
{
    "update": {
        "id": "6881a437-174f-346b-9a63-6e91763507cc",
        "status": "InProgress",
        "type": "AddonUpdate",
```

```
        "params": [
            {
                "type": "AddonVersion",
                "value": "v1.17.1-eksbuild.1"
            },
            {
                "type": "ServiceAccountRoleArn",
                "value": "arn:aws:iam::111122223333:role/eksctl-my-eks-cluster-
addon-vpc-cni-Role1-YfakrqOC1UTm"
            },
            {
                "type": "ResolveConflicts",
                "value": "PRESERVE"
            },
            {
                "type": "ConfigurationValues",
                "value": "{\n    \"resources\": {\n        \"limits\": {\n
  \"cpu\": \"100m\"\n        },\n        \"requests\": {\n            \"cpu\": \"50m
\"\n        }\n    },\n    \"env\": {\n        \"AWS_VPC_K8S_CNI_LOGLEVEL\": \"ERROR
\"\n    }\n}"
            }
        ],
        "createdAt": "2024-04-12T16:22:55.519000-04:00",
        "errors": []
    }
}
```

Per ulteriori informazioni, consulta Gestione dei componenti aggiuntivi di Amazon EKS - Aggiornamento di un componente aggiuntivo nella Guida per l'utente di Amazon EKS.

Esempio 5: Come aggiornare un componente aggiuntivo Amazon EKS con un file di valori di configurazione YAML personalizzati

L'esempio `update-addon` seguente aggiorna un componente aggiuntivo Amazon EKS con valori di configurazione YAML personalizzati e risolve i dettagli dei conflitti.

```
aws eks update-addon \
    --cluster-name my-eks-cluster \
    --addon-name vpc-cni \
    --service-account-role-arn arn:aws:iam::111122223333:role/eksctl-my-eks-cluster-
addon-vpc-cni-Role1-YfakrqOC1UTm \
    --addon-version v1.18.0-eksbuild.1 \
    --configuration-values 'file://configuration-values.yaml' \
```

```
    --resolve-conflicts PRESERVE
```

Contenuto di `configuration-values.yaml`:

```
resources:
    limits:
        cpu: '100m'
    requests:
        cpu: '50m'
env:
    AWS_VPC_K8S_CNI_LOGLEVEL: 'DEBUG'
```

Output:

```
{
    "update": {
        "id": "a067a4c9-69d0-3769-ace9-d235c5b16701",
        "status": "InProgress",
        "type": "AddonUpdate",
        "params": [
            {
                "type": "AddonVersion",
                "value": "v1.18.0-eksbuild.1"
            },
            {
                "type": "ServiceAccountRoleArn",
                "value": "arn:aws:iam::111122223333:role/eksctl-my-eks-cluster-
addon-vpc-cni-Role1-YfakrqOC1UTm"
            },
            {
                "type": "ResolveConflicts",
                "value": "PRESERVE"
            },
            {
                "type": "ConfigurationValues",
                "value": "resources:\n    limits:\n        cpu: '100m'\n
 requests:\n        cpu: '50m'\nenv:\n    AWS_VPC_K8S_CNI_LOGLEVEL: 'DEBUG'"
            }
        ],
        "createdAt": "2024-04-12T16:25:07.212000-04:00",
        "errors": []
    }
```

```
}
```

Per ulteriori informazioni, consulta Gestione dei componenti aggiuntivi di Amazon EKS - Aggiornamento di un componente aggiuntivo nella Guida per l'utente di Amazon EKS.

- Per i dettagli sull'API, consulta UpdateAddon AWS CLICommand Reference.

`update-cluster-config`

Il seguente esempio di codice mostra come utilizzare`update-cluster-config`.

AWS CLI

Come aggiornare l'accesso agli endpoint del cluster

Questo comando di esempio aggiorna un cluster in modo da disabilitare l'accesso pubblico agli endpoint e abilitare quello privato.

Comando:

```
aws eks update-cluster-config --name example \
--resources-vpc-config endpointPublicAccess=false,endpointPrivateAccess=true
```

Output:

```
{
    "update": {
        "id": "ec883c93-2e9e-407c-a22f-8f6fa6e67d4f",
        "status": "InProgress",
        "type": "EndpointAccessUpdate",
        "params": [
            {
                "type": "EndpointPublicAccess",
                "value": "false"
            },
            {
                "type": "EndpointPrivateAccess",
                "value": "true"
            }
        ],
        "createdAt": 1565806986.506,
        "errors": []
    }
```

```
}
```

Come abilitare la registrazione dei log per un cluster

Questo comando di esempio abilita tutti i tipi di registrazione di log del piano di controllo (control-plane) per il cluster denominato `example`.

Comando:

```
aws eks update-cluster-config --name example \
--logging '{"clusterLogging":[{"types":
["api","audit","authenticator","controllerManager","scheduler"],"enabled":true}]}'
```

Output:

```
{
    "update": {
        "id": "7551c64b-1d27-4b1e-9f8e-c45f056eb6fd",
        "status": "InProgress",
        "type": "LoggingUpdate",
        "params": [
            {
                "type": "ClusterLogging",
                "value": "{\"clusterLogging\":[{\"types\":[\"api\",\"audit\",
\"authenticator\",\"controllerManager\",\"scheduler\"],\"enabled\":true}]}"
            }
        ],
        "createdAt": 1565807210.37,
        "errors": []
    }
}
```

- Per i dettagli sull'API, consulta UpdateClusterConfig AWS CLICommand Reference.

`update-cluster-version`

Il seguente esempio di codice mostra come utilizzare`update-cluster-version`.

AWS CLI

Per aggiornare un cluster Amazon EKS denominato `my-eks-cluster` alla versione Kubernetes specificata

L'esempio `update-cluster-version` seguente aggiorna un cluster Amazon EKS alla versione Kubernetes specificata.

```
aws eks update-cluster-version \
    --name my-eks-cluster \
    --kubernetes-version 1.27
```

Output:

```
{
    "update": {
        "id": "e4091a28-ea14-48fd-a8c7-975aeb469e8a",
        "status": "InProgress",
        "type": "VersionUpdate",
        "params": [
            {
                "type": "Version",
                "value": "1.27"
            },
            {
                "type": "PlatformVersion",
                "value": "eks.16"
            }
        ],
        "createdAt": "2024-04-12T16:56:01.082000-04:00",
        "errors": []
    }
}
```

Per ulteriori informazioni, consulta Aggiornamento della versione Kubernetes del cluster Amazon EKS nella Guida per l'utente di Amazon EKS.

- Per i dettagli sull'API, consulta Command UpdateClusterVersionReference AWS CLI .

`update-kubeconfig`

Il seguente esempio di codice mostra come utilizzare`update-kubeconfig`.

AWS CLI

Esempio 1: configura il tuo kubectl creando o aggiornando kubeconfig in modo da poterti connettere a un cluster Amazon EKS denominato `` my-eks-cluster

L'`update-kubeconfig`esempio seguente configura il tuo kubectl creando o aggiornando kubeconfig in modo che tu possa connetterti a un cluster Amazon EKS denominato. my-eks-cluster

```
aws eks update-kubeconfig \
    --name my-eks-cluster
```

Output:

```
Updated context arn:aws:eks:us-east-2:111122223333:cluster/my-eks-cluster in /Users/
xxx/.kube/config
```

Per ulteriori informazioni, consulta Creazione o aggiornamento di un file kubeconfig per un cluster Amazon EKS nella Guida per l'utente di Amazon EKS.

Esempio 2: configura il tuo kubectl creando o aggiornando kubeconfig (con l'opzione role-arn per assumere un ruolo per l'autenticazione del cluster) in modo da poterti connettere a un cluster Amazon EKS denominato `` my-eks-cluster

L'`update-kubeconfig`esempio seguente configura il tuo kubectl creando o aggiornando kubeconfig (con l'opzione role-arn per assumere un ruolo per l'autenticazione del cluster) in modo da poterti connettere a un cluster Amazon EKS denominato. my-eks-cluster

```
aws eks update-kubeconfig \
    --name my-eks-cluster \
    --role-arn arn:aws:iam::111122223333:role/eksctl-EKS-Linux-Cluster-v1-24-
cluster-ServiceRole-j1k7AfTIQtnM
```

Output:

```
Updated context arn:aws:eks:us-east-2:111122223333:cluster/my-eks-cluster in /Users/
xxx/.kube/config
```

Per ulteriori informazioni, consulta Creazione o aggiornamento di un file kubeconfig per un cluster Amazon EKS nella Guida per l'utente di Amazon EKS.

Esempio 3: configura il tuo kubectl creando o aggiornando kubeconfig (con l'opzione role-arn per assumere un ruolo per l'autenticazione del cluster insieme a alias cluster e alias utente personalizzati) in modo da poterti connettere a un cluster Amazon EKS denominato `` my-eks-cluster

L'`update-kubeconfig`esempio seguente configura il tuo kubectl creando o aggiornando kubeconfig (con l'opzione role-arn per assumere un ruolo per l'autenticazione del cluster insieme a alias cluster e alias utente personalizzati) in modo da poterti connettere a un cluster Amazon EKS denominato. my-eks-cluster

```
aws eks update-kubeconfig \
    --name my-eks-cluster \
    --role-arn arn:aws:iam::111122223333:role/eksctl-EKS-Linux-Cluster-v1-24-cluster-ServiceRole-j1k7AfTIQtnM \
    --alias stage-eks-cluster \
    --user-alias john
```

Output:

```
Updated context stage-eks-cluster in /Users/dubaria/.kube/config
```

Per ulteriori informazioni, consulta Creazione o aggiornamento di un file kubeconfig per un cluster Amazon EKS nella Guida per l'utente di Amazon EKS.

Esempio 4: stampa le voci del file kubeconfig per la revisione e configura il tuo kubectl in modo da poterti connettere a un cluster Amazon EKS denominato `` my-eks-cluster

L'`update-kubeconfig`esempio seguente configura il tuo kubectl creando o aggiornando kubeconfig (con l'opzione role-arn per assumere un ruolo per l'autenticazione del cluster insieme a alias cluster e alias utente personalizzati) in modo da poterti connettere a un cluster Amazon EKS denominato. my-eks-cluster

```
aws eks update-kubeconfig \
    --name my-eks-cluster \
    --role-arn arn:aws:iam::111122223333:role/eksctl-EKS-Linux-Cluster-v1-24-cluster-ServiceRole-j1k7AfTIQtnM \
    --alias stage-eks-cluster \
    --user-alias john \
    --verbose
```

Output:

```
Updated context stage-eks-cluster in /Users/dubaria/.kube/config
Entries:

context:
```

```
cluster: arn:aws:eks:us-east-2:111122223333:cluster/my-eks-cluster
user: john
name: stage-eks-cluster

name: john
user:
exec:
    apiVersion: client.authentication.k8s.io/v1beta1
    args:
    - --region
    - us-east-2
    - eks
    - get-token
    - --cluster-name
    - my-eks-cluster
    - --output
    - json
    - --role
    - arn:aws:iam::111122223333:role/eksctl-EKS-Linux-Cluster-v1-24-cluster-
ServiceRole-j1k7AfTIQtnM
    command: aws

cluster:
certificate-authority-data: xxx_CA_DATA_xxx
server: https://DALSJ343KE23J3RN45653DSKJTT647TYD.yl4.us-east-2.eks.amazonaws.com
name: arn:aws:eks:us-east-2:111122223333:cluster/my-eks-cluster
```

Per ulteriori informazioni, consulta Creazione o aggiornamento di un file kubeconfig per un cluster Amazon EKS nella Guida per l'utente di Amazon EKS.

- Per UpdateKubeconfigi dettagli AWS CLI sull'API, consulta Command Reference.

`update-nodegroup-config`

Il seguente esempio di codice mostra come utilizzare`update-nodegroup-config`.

AWS CLI

Esempio 1: aggiorna un gruppo di nodi gestiti per aggiungere nuovi taint ed etichette al nodo worker EKS per un cluster Amazon EKS

L'esempio `update-nodegroup-config` seguente aggiorna un gruppo di nodi gestiti per aggiungere nuovi taint ed etichette al nodo worker EKS per un cluster Amazon EKS.

```
aws eks update-nodegroup-config \
    --cluster-name my-eks-cluster \
    --nodegroup-name my-eks-nodegroup \
    --labels 'addOrUpdateLabels={my-eks-nodegroup-label-1=value-1,my-eks-nodegroup-label-2=value-2}' \
    --taints 'addOrUpdateTaints=[{key=taint-key-1,value=taint-value-1,effect=NO_EXECUTE}]'
```

Output:

```
{
    "update": {
        "id": "e66d21d3-bd8b-3ad1-a5aa-b196dc08c7c1",
        "status": "InProgress",
        "type": "ConfigUpdate",
        "params": [
            {
                "type": "LabelsToAdd",
                "value": "{\"my-eks-nodegroup-label-2\":\"value-2\",\"my-eks-nodegroup-label-1\":\"value-1\"}"
            },
            {
                "type": "TaintsToAdd",
                "value": "[{\"effect\":\"NO_EXECUTE\",\"value\":\"taint-value-1\",\"key\":\"taint-key-1\"}]"
            }
        ],
        "createdAt": "2024-04-08T12:05:19.161000-04:00",
        "errors": []
    }
}
```

Per ulteriori informazioni, consulta Aggiornamento di un gruppo di nodi gestiti nella Guida per l'utente di Amazon EKS.

Esempio 2: aggiorna un gruppo di nodi gestiti per rimuovere etichette e taint dal nodo worker EKS per un cluster Amazon EKS

L'esempio `update-nodegroup-config` seguente aggiorna un gruppo di nodi gestiti per rimuovere etichette e taint dal nodo worker EKS per un cluster Amazon EKS.

```
aws eks update-nodegroup-config \
```

```
    --cluster-name my-eks-cluster \
    --nodegroup-name my-eks-nodegroup \
    --labels 'removeLabels=my-eks-nodegroup-label-1, my-eks-nodegroup-label-2' \
    --taints 'removeTaints=[{key=taint-key-1,value=taint-
value-1,effect=NO_EXECUTE}]'
```

Output:

```
{
    "update": {
        "id": "67a08692-9e59-3ace-a916-13929f44cec3",
        "status": "InProgress",
        "type": "ConfigUpdate",
        "params": [
            {
                "type": "LabelsToRemove",
                "value": "[\"my-eks-nodegroup-label-1\",\"my-eks-nodegroup-
label-2\"]"
            },
            {
                "type": "TaintsToRemove",
                "value": "[{\"effect\":\"NO_EXECUTE\",\"value\":\"taint-value-1\",
\"key\":\"taint-key-1\"}]"
            }
        ],
        "createdAt": "2024-04-08T12:17:31.817000-04:00",
        "errors": []
    }
}
```

Per ulteriori informazioni, consulta Aggiornamento di un gruppo di nodi gestiti nella Guida per l'utente di Amazon EKS.

Esempio 3: aggiorna un gruppo di nodi gestiti per rimuovere e aggiungere etichette e taint dal e al nodo worker per un cluster Amazon EKS

L'esempio `update-nodegroup-config` seguente aggiorna un gruppo di nodi gestiti per rimuovere e aggiungere etichette e taint dal e al nodo worker EKS per un cluster Amazon EKS.

```
aws eks update-nodegroup-config \
    --cluster-name my-eks-cluster \
    --nodegroup-name my-eks-nodegroup \
```

```
    --labels 'addOrUpdateLabels={my-eks-nodegroup-new-label-1=new-value-1,my-eks-
nodegroup-new-label-2=new-value-2},removeLabels=my-eks-nodegroup-label-1, my-eks-
nodegroup-label-2' \
    --taints 'addOrUpdateTaints=[{key=taint-new-key-1,value=taint-new-
value-1,effect=PREFER_NO_SCHEDULE}],removeTaints=[{key=taint-key-1,value=taint-
value-1,effect=NO_EXECUTE}]'
```

Output:

```
{
    "update": {
        "id": "4a9c8c45-6ac7-3115-be71-d6412a2339b7",
        "status": "InProgress",
        "type": "ConfigUpdate",
        "params": [
            {
                "type": "LabelsToAdd",
                "value": "{\"my-eks-nodegroup-new-label-1\":\"new-value-1\",\"my-
eks-nodegroup-new-label-2\":\"new-value-2\"}"
            },
            {
                "type": "LabelsToRemove",
                "value": "[\"my-eks-nodegroup-label-1\",\"my-eks-nodegroup-
label-2\"]"
            },
            {
                "type": "TaintsToAdd",
                "value": "[{\"effect\":\"PREFER_NO_SCHEDULE\",\"value\":\"taint-new-
value-1\",\"key\":\"taint-new-key-1\"}]"
            },
            {
                "type": "TaintsToRemove",
                "value": "[{\"effect\":\"NO_EXECUTE\",\"value\":\"taint-value-1\",
\"key\":\"taint-key-1\"}]"
            }
        ],
        "createdAt": "2024-04-08T12:30:55.486000-04:00",
        "errors": []
    }
}
```

Per ulteriori informazioni, consulta Aggiornamento di un gruppo di nodi gestiti nella Guida per l'utente di Amazon EKS.

Esempio 4: aggiorna un gruppo di nodi gestiti per aggiornare i file scaling-config e update-config per il nodo worker EKS per un cluster Amazon EKS

L'esempio `update-nodegroup-config` seguente aggiorna un gruppo di nodi gestiti per aggiornare i file scaling-config e update-config per il nodo worker EKS per un cluster Amazon EKS.

```
aws eks update-nodegroup-config \
    --cluster-name my-eks-cluster \
    --nodegroup-name my-eks-nodegroup \
    --scaling-config minSize=1,maxSize=5,desiredSize=2 \
    --update-config maxUnavailable=2
```

Output:

```
{
    "update": {
        "id": "a977160f-59bf-3023-805d-c9826e460aea",
        "status": "InProgress",
        "type": "ConfigUpdate",
        "params": [
            {
                "type": "MinSize",
                "value": "1"
            },
            {
                "type": "MaxSize",
                "value": "5"
            },
            {
                "type": "DesiredSize",
                "value": "2"
            },
            {
                "type": "MaxUnavailable",
                "value": "2"
            }
        ],
        "createdAt": "2024-04-08T12:35:17.036000-04:00",
        "errors": []
    }
}
```

Per ulteriori informazioni, consulta Aggiornamento di un gruppo di nodi gestiti nella Guida per l'utente di Amazon EKS.

- Per i dettagli sull'API, consulta UpdateNodegroupConfig AWS CLICommand Reference.

`update-nodegroup-version`

Il seguente esempio di codice mostra come utilizzare`update-nodegroup-version`.

AWS CLI

Esempio 1: aggiorna la versione Kubernetes o la versione AMI di un gruppo di nodi gestiti Amazon EKS

L'esempio `update-nodegroup-version` seguente aggiorna la versione Kubernetes o la versione AMI di un gruppo di nodi gestiti Amazon EKS all'ultima versione disponibile per il cluster Kubernetes in uso.

```
aws eks update-nodegroup-version \
    --cluster-name my-eks-cluster \
    --nodegroup-name my-eks-nodegroup \
    --no-force
```

Output:

```
{
    "update": {
        "id": "a94ebfc3-6bf8-307a-89e6-7dbaa36421f7",
        "status": "InProgress",
        "type": "VersionUpdate",
        "params": [
            {
                "type": "Version",
                "value": "1.26"
            },
            {
                "type": "ReleaseVersion",
                "value": "1.26.12-20240329"
            }
        ],
        "createdAt": "2024-04-08T13:16:00.724000-04:00",
        "errors": []
```

```
    }
}
```

Per ulteriori informazioni, consulta Aggiornamento di un gruppo di nodi gestiti nella Guida per l'utente di Amazon EKS.

Esempio 2: aggiorna la versione Kubernetes o la versione AMI di un gruppo di nodi gestiti Amazon EKS

L'esempio `update-nodegroup-version` seguente aggiorna la versione Kubernetes o la versione AMI di un gruppo di nodi gestiti Amazon EKS alla versione AMI specificata.

```
aws eks update-nodegroup-version \
    --cluster-name my-eks-cluster \
    --nodegroup-name my-eks-nodegroup \
    --kubernetes-version '1.26' \
    --release-version '1.26.12-20240307' \
    --no-force
```

Output:

```
{
    "update": {
        "id": "4db06fe1-088d-336b-bdcd-3fdb94995fb7",
        "status": "InProgress",
        "type": "VersionUpdate",
        "params": [
            {
                "type": "Version",
                "value": "1.26"
            },
            {
                "type": "ReleaseVersion",
                "value": "1.26.12-20240307"
            }
        ],
        "createdAt": "2024-04-08T13:13:58.595000-04:00",
        "errors": []
    }
}
```

Per ulteriori informazioni, consulta la sezione Aggiornamento di un gruppo di nodi gestiti - < https://docs.aws.amazon.com/eks/ latest/userguide/update - managed-node-group .html>` nella Guida per l'utente di Amazon EKS.

- Per i dettagli sull'API, consulta UpdateNodegroupVersionCommand Reference.AWS CLI

`update-pod-identity-association`

Il seguente esempio di codice mostra come utilizzare`update-pod-identity-association`.

AWS CLI

Come aggiornare l'associazione EKS Pod Identity

L'esempio `update-pod-identity-association` seguente aggiorna un'associazione EKS Pod Identity modificando il ruolo IAM associato da `arn:aws:iam::111122223333:role/my-role` a `arn:aws:iam::111122223333:role/s3-role` per l'ID associazione `a-9njjin9gfghecgocd`. Questa API consente solo l'aggiornamento del ruolo IAM associato.

```
aws eks update-pod-identity-association \
    --cluster-name eks-customer \
    --association-id a-9njjin9gfghecgocd \
    --role-arn arn:aws:iam::111122223333:role/s3-role
```

Output:

```
{
    "association": {
        "clusterName": "eks-customer",
        "namespace": "default",
        "serviceAccount": "default",
        "roleArn": "arn:aws:iam::111122223333:role/s3-role",
        "associationArn": "arn:aws:eks:us-
west-2:111122223333:podidentityassociation/eks-customer/a-9njjin9gfghecgocd",
        "associationId": "a-9njjin9gfghecgocd",
        "tags": {
            "Key2": "value2",
            "Key1": "value1"
        },
        "createdAt": "2025-05-24T19:52:14.135000-05:00",
        "modifiedAt": "2025-05-25T21:01:53.120000-05:00"
```

```
    }
}
```

Per ulteriori informazioni, consulta Scopri come EKS Pod Identity concede ai pod l'accesso ai AWS servizi nella Amazon EKS User Guide.

- Per i dettagli sull'API, consulta AWS CLI Command UpdatePodIdentityAssociationReference.

Esempi di utilizzo di Elastic Beanstalk AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface con Elastic Beanstalk.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`abort-environment-update`

Il seguente esempio di codice mostra come utilizzare. `abort-environment-update`

AWS CLI

Come interrompere un'implementazione

Il comando seguente interrompe un'implementazione in corso di una versione dell'applicazione per un ambiente denominato `my-env`:

```
aws elasticbeanstalk abort-environment-update --environment-name my-env
```

- Per i dettagli sull'API, consulta AbortEnvironmentUpdate AWS CLICommand Reference.

`check-dns-availability`

Il seguente esempio di codice mostra come utilizzare`check-dns-availability`.

AWS CLI

Come verificare la disponibilità di un CNAME

Il comando seguente verifica la disponibilità del sottodominio `my-cname.elasticbeanstalk.com`:

```
aws elasticbeanstalk check-dns-availability --cname-prefix my-cname
```

Output:

```
{
    "Available": true,
    "FullyQualifiedCNAME": "my-cname.elasticbeanstalk.com"
}
```

- Per i dettagli sull'API, consulta CheckDnsAvailability AWS CLICommand Reference.

`create-application-version`

Il seguente esempio di codice mostra come utilizzare`create-application-version`.

AWS CLI

Come creare una nuova versione dell'applicazione

Il comando seguente crea una nuova versione, «v1", di un'applicazione denominata" «: MyApp

```
aws elasticbeanstalk create-application-version --application-name MyApp --
version-label v1 --description MyAppv1 --source-bundle S3Bucket="amzn-s3-demo-
bucket",S3Key="sample.war" --auto-create-application
```

L'applicazione verrà creata automaticamente se non esiste già, grazie all' auto-create-applicationopzione. Il bundle di origine è un file .war archiviato in un bucket S3 denominato "amzn-s3-demo-bucket" che contiene l'applicazione di esempio Apache Tomcat.

Output:

```
{
  "ApplicationVersion": {
      "ApplicationName": "MyApp",
      "VersionLabel": "v1",
      "Description": "MyAppv1",
      "DateCreated": "2015-02-03T23:01:25.412Z",
      "DateUpdated": "2015-02-03T23:01:25.412Z",
      "SourceBundle": {
          "S3Bucket": "amzn-s3-demo-bucket",
          "S3Key": "sample.war"
      }
  }
}
```

- Per i dettagli sull'API, consulta CreateApplicationVersion AWS CLICommand Reference.

`create-application`

Il seguente esempio di codice mostra come utilizzare`create-application`.

AWS CLI

Come creare una nuova applicazione

Il comando seguente crea una nuova applicazione denominata "MyApp«:

```
aws elasticbeanstalk create-application --application-name MyApp --description "my
 application"
```

Il comando `create-application` configura solo il nome e la descrizione dell'applicazione. Per caricare il codice sorgente dell'applicazione, crea una versione iniziale dell'applicazione utilizzando `create-application-version`. `create-application-version` ha anche un'opzione `auto-create-application` che consente di creare l'applicazione e la versione dell'applicazione in un unico passaggio.

Output:

```
{
```

```
  "Application": {
      "ApplicationName": "MyApp",
      "ConfigurationTemplates": [],
      "DateUpdated": "2015-02-12T18:32:21.181Z",
      "Description": "my application",
      "DateCreated": "2015-02-12T18:32:21.181Z"
  }
}
```

- Per i dettagli sull'API, vedere CreateApplicationin AWS CLI Command Reference.

`create-configuration-template`

Il seguente esempio di codice mostra come utilizzare`create-configuration-template`.

AWS CLI

Come creare un modello di configurazione

Il comando seguente crea un modello di configurazione denominato `my-app-v1` dalle impostazioni applicate a un ambiente con ID `e-rpqsewtp2j`:

```
aws elasticbeanstalk create-configuration-template --application-name my-app --
template-name my-app-v1 --environment-id e-rpqsewtp2j
```

Output:

```
{
    "ApplicationName": "my-app",
    "TemplateName": "my-app-v1",
    "DateCreated": "2015-08-12T18:40:39Z",
    "DateUpdated": "2015-08-12T18:40:39Z",
    "SolutionStackName": "64bit Amazon Linux 2015.03 v2.0.0 running Tomcat 8 Java 8"
}
```

- Per i dettagli sull'API, consulta CreateConfigurationTemplate AWS CLICommand Reference.

`create-environment`

Il seguente esempio di codice mostra come utilizzare`create-environment`.

AWS CLI

Come creare un nuovo ambiente per un'applicazione

Il comando seguente crea un nuovo ambiente per la versione "v1" di un'applicazione Java denominata "my-app":

```
aws elasticbeanstalk create-environment --application-name my-app --environment-
name my-env --cname-prefix my-app --version-label v1 --solution-stack-name "64bit
 Amazon Linux 2015.03 v2.0.0 running Tomcat 8 Java 8"
```

Output:

```
{
  "ApplicationName": "my-app",
  "EnvironmentName": "my-env",
  "VersionLabel": "v1",
  "Status": "Launching",
  "EnvironmentId": "e-izqpassy4h",
  "SolutionStackName": "64bit Amazon Linux 2015.03 v2.0.0 running Tomcat 8 Java 8",
  "CNAME": "my-app.elasticbeanstalk.com",
  "Health": "Grey",
  "Tier": {
      "Type": "Standard",
      "Name": "WebServer",
      "Version": " "
  },
  "DateUpdated": "2015-02-03T23:04:54.479Z",
  "DateCreated": "2015-02-03T23:04:54.479Z"
}
```

`v1`è l'etichetta di una versione dell'applicazione precedentemente caricata con create-application-version.

Come specificare un file JSON per definire le opzioni di configurazione dell'ambiente

L'esempio `create-environment` seguente specifica che un file JSON denominato `myoptions.json` deve essere utilizzato per sovrascrivere i valori ottenuti dallo stack di soluzioni o dal modello di configurazione:

```
aws elasticbeanstalk create-environment --environment-name sample-env --application-
name sampleapp --option-settings file://myoptions.json
```

`myoptions.json` è un oggetto JSON che definisce diverse impostazioni:

```
[
  {
    "Namespace": "aws:elb:healthcheck",
    "OptionName": "Interval",
    "Value": "15"
  },
  {
    "Namespace": "aws:elb:healthcheck",
    "OptionName": "Timeout",
    "Value": "8"
  },
  {
    "Namespace": "aws:elb:healthcheck",
    "OptionName": "HealthyThreshold",
    "Value": "2"
  },
  {
    "Namespace": "aws:elb:healthcheck",
    "OptionName": "UnhealthyThreshold",
    "Value": "3"
  }
]
```

Per ulteriori informazioni, consulta Valori opzione nella Guida per gli sviluppatori di AWS Elastic Beanstalk.

- Per i dettagli sull'API, consulta CreateEnvironment AWS CLICommand Reference.

`create-storage-location`

Il seguente esempio di codice mostra come utilizzare`create-storage-location`.

AWS CLI

Come creare una posizione di archiviazione

Il comando seguente crea un percorso di archiviazione in Amazon S3:

```
aws elasticbeanstalk create-storage-location
```

Output:

```
{
    "S3Bucket": "elasticbeanstalk-us-west-2-0123456789012"
}
```

- Per i dettagli sull'API, consulta CreateStorageLocation AWS CLICommand Reference.

`delete-application-version`

Il seguente esempio di codice mostra come utilizzare`delete-application-version`.

AWS CLI

Come eliminare una versione di un'applicazione

Il comando seguente elimina una versione dell'applicazione denominata `22a0-stage-150819_182129` per un'applicazione denominata `my-app`:

```
aws elasticbeanstalk delete-application-version --version-label 22a0-stage-150819_182129 --application-name my-app
```

- Per i dettagli sull'API, consulta DeleteApplicationVersion AWS CLICommand Reference.

`delete-application`

Il seguente esempio di codice mostra come utilizzare`delete-application`.

AWS CLI

Come eliminare un'applicazione

Il comando seguente elimina un'applicazione denominata `my-app`:

```
aws elasticbeanstalk delete-application --application-name my-app
```

- Per i dettagli sull'API, consulta DeleteApplication AWS CLICommand Reference.

`delete-configuration-template`

Il seguente esempio di codice mostra come utilizzare`delete-configuration-template`.

AWS CLI

Come eliminare un modello di configurazione

Il comando seguente elimina un modello di configurazione denominato `my-template` per un'applicazione denominata `my-app`:

```
aws elasticbeanstalk delete-configuration-template --template-name my-template --
application-name my-app
```

- Per i dettagli sull'API, consulta DeleteConfigurationTemplate AWS CLICommand Reference.

`delete-environment-configuration`

Il seguente esempio di codice mostra come utilizzare`delete-environment-configuration`.

AWS CLI

Come eliminare una bozza di configurazione

Il comando seguente elimina una bozza di configurazione per un ambiente denominato `my-env`:

```
aws elasticbeanstalk delete-environment-configuration --environment-name my-env --
application-name my-app
```

- Per i dettagli sull'API, consulta DeleteEnvironmentConfiguration AWS CLICommand Reference.

`describe-application-versions`

Il seguente esempio di codice mostra come utilizzare`describe-application-versions`.

AWS CLI

Come visualizzare informazioni su una versione dell'applicazione

Il seguente comando recupera informazioni su una versione dell'applicazione con etichetta v2:

```
aws elasticbeanstalk describe-application-versions --application-name my-app --
version-label "v2"
```

Output:

```
{
    "ApplicationVersions": [
        {
            "ApplicationName": "my-app",
            "VersionLabel": "v2",
            "Description": "update cover page",
            "DateCreated": "2015-07-23T01:32:26.079Z",
            "DateUpdated": "2015-07-23T01:32:26.079Z",
            "SourceBundle": {
                "S3Bucket": "elasticbeanstalk-us-west-2-015321684451",
                "S3Key": "my-app/5026-stage-150723_224258.war"
            }
        },
      {
          "ApplicationName": "my-app",
          "VersionLabel": "v1",
          "Description": "initial version",
          "DateCreated": "2015-07-23T22:26:10.816Z",
          "DateUpdated": "2015-07-23T22:26:10.816Z",
          "SourceBundle": {
              "S3Bucket": "elasticbeanstalk-us-west-2-015321684451",
              "S3Key": "my-app/5026-stage-150723_222618.war"
          }
      }
    ]
}
```

- Per i dettagli sull'API, consulta DescribeApplicationVersions AWS CLICommand Reference.

`describe-applications`

Il seguente esempio di codice mostra come utilizzare`describe-applications`.

AWS CLI

Come visualizzare un elenco di applicazioni

Il comando seguente recupera le informazioni sulle applicazioni nella Regione corrente.

```
aws elasticbeanstalk describe-applications
```

Output:

```
{
    "Applications": [
        {
            "ApplicationName": "ruby",
            "ConfigurationTemplates": [],
            "DateUpdated": "2015-08-13T21:05:44.376Z",
            "Versions": [
                "Sample Application"
            ],
            "DateCreated": "2015-08-13T21:05:44.376Z"
        },
        {
            "ApplicationName": "pythonsample",
            "Description": "Application created from the EB CLI using \"eb init\"",
            "Versions": [
                "Sample Application"
            ],
            "DateCreated": "2015-08-13T19:05:43.637Z",
            "ConfigurationTemplates": [],
            "DateUpdated": "2015-08-13T19:05:43.637Z"
        },
        {
            "ApplicationName": "nodejs-example",
            "ConfigurationTemplates": [],
            "DateUpdated": "2015-08-06T17:50:02.486Z",
            "Versions": [
                "add elasticache",
                "First Release"
            ],
            "DateCreated": "2015-08-06T17:50:02.486Z"
        }
    ]
}
```

- Per i dettagli sull'API, consulta DescribeApplications AWS CLICommand Reference.

`describe-configuration-options`

Il seguente esempio di codice mostra come utilizzare`describe-configuration-options`.

AWS CLI

Come visualizzare le opzioni di configurazione di un ambiente

Il comando seguente recupera la descrizione di tutte le opzioni di configurazione disponibili per un ambiente denominato `my-env`:

```
aws elasticbeanstalk describe-configuration-options --environment-name my-env --
application-name my-app
```

Output (abbreviato):

```
{
    "Options": [
        {
            "Name": "JVMOptions",
            "UserDefined": false,
            "DefaultValue": "Xms=256m,Xmx=256m,XX:MaxPermSize=64m,JVM Options=",
            "ChangeSeverity": "RestartApplicationServer",
            "Namespace": "aws:cloudformation:template:parameter",
            "ValueType": "KeyValueList"
        },
        {
            "Name": "Interval",
            "UserDefined": false,
            "DefaultValue": "30",
            "ChangeSeverity": "NoInterruption",
            "Namespace": "aws:elb:healthcheck",
            "MaxValue": 300,
            "MinValue": 5,
            "ValueType": "Scalar"
        },
        ...
        {
            "Name": "LowerThreshold",
            "UserDefined": false,
            "DefaultValue": "2000000",
            "ChangeSeverity": "NoInterruption",
            "Namespace": "aws:autoscaling:trigger",
            "MinValue": 0,
            "ValueType": "Scalar"
        },
        {
            "Name": "ListenerEnabled",
            "UserDefined": false,
            "DefaultValue": "true",
            "ChangeSeverity": "Unknown",
```

```
            "Namespace": "aws:elb:listener",
            "ValueType": "Boolean"
        }
    ]
}
```

Le opzioni di configurazione disponibili variano in base alla piattaforma e alla versione della configurazione. Per ulteriori informazioni sui namespace e sulle opzioni supportate, consulta Valori opzione nella Guida per gli sviluppatori di AWS Elastic Beanstalk.

- Per i dettagli sull'API, consulta DescribeConfigurationOptions AWS CLICommand Reference.

`describe-configuration-settings`

Il seguente esempio di codice mostra come utilizzare`describe-configuration-settings`.

AWS CLI

Come visualizzare le impostazioni di configurazione di un ambiente

Il comando seguente recupera le impostazioni di configurazione per un ambiente denominato `my-env`:

```
aws elasticbeanstalk describe-configuration-settings --environment-name my-env --
application-name my-app
```

Output (abbreviato):

```
{
    "ConfigurationSettings": [
        {
            "ApplicationName": "my-app",
            "EnvironmentName": "my-env",
            "Description": "Environment created from the EB CLI using \"eb create
\"",
            "DeploymentStatus": "deployed",
            "DateCreated": "2015-08-13T19:16:25Z",
            "OptionSettings": [
                {
                    "OptionName": "Availability Zones",
                    "ResourceName": "AWSEBAutoScalingGroup",
                    "Namespace": "aws:autoscaling:asg",
                    "Value": "Any"
```

```
                },
                {
                    "OptionName": "Cooldown",
                    "ResourceName": "AWSEBAutoScalingGroup",
                    "Namespace": "aws:autoscaling:asg",
                    "Value": "360"
                },
                ...
                {
                    "OptionName": "ConnectionDrainingTimeout",
                    "ResourceName": "AWSEBLoadBalancer",
                    "Namespace": "aws:elb:policies",
                    "Value": "20"
                },
                {
                    "OptionName": "ConnectionSettingIdleTimeout",
                    "ResourceName": "AWSEBLoadBalancer",
                    "Namespace": "aws:elb:policies",
                    "Value": "60"
                }
            ],
            "DateUpdated": "2015-08-13T23:30:07Z",
            "SolutionStackName": "64bit Amazon Linux 2015.03 v2.0.0 running Tomcat 8
 Java 8"
        }
    ]
}
```

Per ulteriori informazioni sui namespace e sulle opzioni supportate, consulta Valori opzione nella Guida per gli sviluppatori di AWS Elastic Beanstalk.

- Per i dettagli sull'API, consulta DescribeConfigurationSettings AWS CLICommand Reference.

`describe-environment-health`

Il seguente esempio di codice mostra come utilizzare`describe-environment-health`.

AWS CLI

Come visualizzare l'integrità dell'ambiente

Il comando seguente recupera informazioni generali sull'integrità di un ambiente denominato `my-env`:

```
aws elasticbeanstalk describe-environment-health --environment-name my-env --
attribute-names All
```

Output:


```
{
    "Status": "Ready",
    "EnvironmentName": "my-env",
    "Color": "Green",
    "ApplicationMetrics": {
        "Duration": 10,
        "Latency": {
            "P99": 0.004,
            "P75": 0.002,
            "P90": 0.003,
            "P95": 0.004,
            "P85": 0.003,
            "P10": 0.001,
            "P999": 0.004,
            "P50": 0.001
        },
        "RequestCount": 45,
        "StatusCodes": {
            "Status3xx": 0,
            "Status2xx": 45,
            "Status5xx": 0,
            "Status4xx": 0
        }
    },
    "RefreshedAt": "2015-08-20T21:09:18Z",
    "HealthStatus": "Ok",
    "InstancesHealth": {
        "Info": 0,
        "Ok": 1,
        "Unknown": 0,
        "Severe": 0,
        "Warning": 0,
        "Degraded": 0,
        "NoData": 0,
        "Pending": 0
    },
    "Causes": []
```

```
}
```

Le informazioni sull'integrità sono disponibili solo per gli ambienti in cui il reporting dello stato avanzato è abilitato. Per ulteriori informazioni, consulta Reporting e monitoraggio dello stato avanzati nella Guida per gli sviluppatori di AWS Elastic Beanstalk.

- Per i dettagli sull'API, consulta DescribeEnvironmentHealth AWS CLICommand Reference.

`describe-environment-resources`

Il seguente esempio di codice mostra come utilizzare`describe-environment-resources`.

AWS CLI

Per visualizzare informazioni sulle AWS risorse presenti nell'ambiente

Il comando seguente recupera informazioni sulle risorse in un ambiente denominato `my-env`:

```
aws elasticbeanstalk describe-environment-resources --environment-name my-env
```

Output:

```
{
    "EnvironmentResources": {
        "EnvironmentName": "my-env",
        "AutoScalingGroups": [
            {
                "Name": "awseb-e-qu3fyyjyjs-stack-AWSEBAutoScalingGroup-
QSB2ZO88SXZT"
            }
        ],
        "Triggers": [],
        "LoadBalancers": [
            {
                "Name": "awseb-e-q-AWSEBLoa-1EEPZ0K98BIF0"
            }
        ],
        "Queues": [],
        "Instances": [
            {
                "Id": "i-0c91c786"
            }
        ],
```

```
        "LaunchConfigurations": [
            {
                "Name": "awseb-e-qu3fyyjyjs-stack-
AWSEBAutoScalingLaunchConfiguration-1UUVQIBC96TQ2"
            }
        ]
    }
}
```

- Per i dettagli sull'API, consulta DescribeEnvironmentResources AWS CLICommand Reference.

`describe-environments`

Il seguente esempio di codice mostra come utilizzare`describe-environments`.

AWS CLI

Come visualizzare informazioni su un ambiente

Il comando seguente recupera informazioni su un ambiente denominato `my-env`:

```
aws elasticbeanstalk describe-environments --environment-names my-env
```

Output:

```
{
    "Environments": [
        {
            "ApplicationName": "my-app",
            "EnvironmentName": "my-env",
            "VersionLabel": "7f58-stage-150812_025409",
            "Status": "Ready",
            "EnvironmentId": "e-rpqsewtp2j",
            "EndpointURL": "awseb-e-w-AWSEBLoa-1483140XB0Q4L-109QXY8121.us-
west-2.elb.amazonaws.com",
            "SolutionStackName": "64bit Amazon Linux 2015.03 v2.0.0 running Tomcat 8
 Java 8",
            "CNAME": "my-env.elasticbeanstalk.com",
            "Health": "Green",
            "AbortableOperationInProgress": false,
            "Tier": {
                "Version": " ",
                "Type": "Standard",
```

```
                "Name": "WebServer"
            },
            "DateUpdated": "2015-08-12T18:16:55.019Z",
            "DateCreated": "2015-08-07T20:48:49.599Z"
        }
    ]
}
```

- Per i dettagli sull'API, consulta DescribeEnvironments AWS CLICommand Reference.

`describe-events`

Il seguente esempio di codice mostra come utilizzare`describe-events`.

AWS CLI

Come visualizzare gli eventi per un ambiente

Il comando seguente recupera gli eventi per un ambiente denominato `my-env`:

```
aws elasticbeanstalk describe-events --environment-name my-env
```

Output (abbreviato):

```
{
    "Events": [
        {
            "ApplicationName": "my-app",
            "EnvironmentName": "my-env",
            "Message": "Environment health has transitioned from Info to Ok.",
            "EventDate": "2015-08-20T07:06:53.535Z",
            "Severity": "INFO"
        },
        {
            "ApplicationName": "my-app",
            "EnvironmentName": "my-env",
            "Severity": "INFO",
            "RequestId": "b7f3960b-4709-11e5-ba1e-07e16200da41",
            "Message": "Environment update completed successfully.",
            "EventDate": "2015-08-20T07:06:02.049Z"
        },
        ...
        {
```

```
            "ApplicationName": "my-app",
            "EnvironmentName": "my-env",
            "Severity": "INFO",
            "RequestId": "ca8dfbf6-41ef-11e5-988b-651aa638f46b",
            "Message": "Using elasticbeanstalk-us-west-2-012445113685 as Amazon S3
 storage bucket for environment data.",
            "EventDate": "2015-08-13T19:16:27.561Z"
        },
        {
            "ApplicationName": "my-app",
            "EnvironmentName": "my-env",
            "Severity": "INFO",
            "RequestId": "cdfba8f6-41ef-11e5-988b-65638f41aa6b",
            "Message": "createEnvironment is starting.",
            "EventDate": "2015-08-13T19:16:26.581Z"
        }
    ]
}
```

- Per i dettagli sull'API, consulta DescribeEvents AWS CLICommand Reference.

`describe-instances-health`

Il seguente esempio di codice mostra come utilizzare`describe-instances-health`.

AWS CLI

Come visualizzare l'integrità dell'ambiente

Il comando seguente recupera informazioni sull'integrità delle istanze in un ambiente denominato `my-env`:

```
aws elasticbeanstalk describe-instances-health --environment-name my-env --
attribute-names All
```

Output:

```
{
    "InstanceHealthList": [
        {
            "InstanceId": "i-08691cc7",
            "ApplicationMetrics": {
```

```
                "Duration": 10,
                "Latency": {
                    "P99": 0.006,
                    "P75": 0.002,
                    "P90": 0.004,
                    "P95": 0.005,
                    "P85": 0.003,
                    "P10": 0.0,
                    "P999": 0.006,
                    "P50": 0.001
                },
                "RequestCount": 48,
                "StatusCodes": {
                    "Status3xx": 0,
                    "Status2xx": 47,
                    "Status5xx": 0,
                    "Status4xx": 1
                }
            },
            "System": {
                "LoadAverage": [
                    0.0,
                    0.02,
                    0.05
                ],
                "CPUUtilization": {
                    "SoftIRQ": 0.1,
                    "IOWait": 0.2,
                    "System": 0.3,
                    "Idle": 97.8,
                    "User": 1.5,
                    "IRQ": 0.0,
                    "Nice": 0.1
                }
            },
            "Color": "Green",
            "HealthStatus": "Ok",
            "LaunchedAt": "2015-08-13T19:17:09Z",
            "Causes": []
        }
    ],
    "RefreshedAt": "2015-08-20T21:09:08Z"
}
```

Le informazioni sull'integrità sono disponibili solo per gli ambienti in cui il reporting dello stato avanzato è abilitato. Per ulteriori informazioni, consulta Reporting e monitoraggio dello stato avanzati nella Guida per gli sviluppatori di AWS Elastic Beanstalk.

- Per i dettagli sull'API, consulta DescribeInstancesHealth AWS CLICommand Reference.

`list-available-solution-stacks`

Il seguente esempio di codice mostra come utilizzare`list-available-solution-stacks`.

AWS CLI

Come visualizzare gli stack di soluzioni

Il comando seguente elenca gli stack di soluzioni per tutte le configurazioni di piattaforma attualmente disponibili e per quelle utilizzate in passato.

```
aws elasticbeanstalk list-available-solution-stacks
```

Output (abbreviato):

```
{
    "SolutionStacks": [
        "64bit Amazon Linux 2015.03 v2.0.0 running Node.js",
        "64bit Amazon Linux 2015.03 v2.0.0 running PHP 5.6",
        "64bit Amazon Linux 2015.03 v2.0.0 running PHP 5.5",
        "64bit Amazon Linux 2015.03 v2.0.0 running PHP 5.4",
        "64bit Amazon Linux 2015.03 v2.0.0 running Python 3.4",
        "64bit Amazon Linux 2015.03 v2.0.0 running Python 2.7",
        "64bit Amazon Linux 2015.03 v2.0.0 running Python",
        "64bit Amazon Linux 2015.03 v2.0.0 running Ruby 2.2 (Puma)",
        "64bit Amazon Linux 2015.03 v2.0.0 running Ruby 2.2 (Passenger Standalone)",
        "64bit Amazon Linux 2015.03 v2.0.0 running Ruby 2.1 (Puma)",
        "64bit Amazon Linux 2015.03 v2.0.0 running Ruby 2.1 (Passenger Standalone)",
        "64bit Amazon Linux 2015.03 v2.0.0 running Ruby 2.0 (Puma)",
        "64bit Amazon Linux 2015.03 v2.0.0 running Ruby 2.0 (Passenger Standalone)",
        "64bit Amazon Linux 2015.03 v2.0.0 running Ruby 1.9.3",
        "64bit Amazon Linux 2015.03 v2.0.0 running Tomcat 8 Java 8",
        "64bit Amazon Linux 2015.03 v2.0.0 running Tomcat 7 Java 7",
        "64bit Amazon Linux 2015.03 v2.0.0 running Tomcat 7 Java 6",
        "64bit Windows Server Core 2012 R2 running IIS 8.5",
        "64bit Windows Server 2012 R2 running IIS 8.5",
        "64bit Windows Server 2012 running IIS 8",
```

```
        "64bit Windows Server 2008 R2 running IIS 7.5",
        "64bit Amazon Linux 2015.03 v2.0.0 running Docker 1.6.2",
        "64bit Amazon Linux 2015.03 v2.0.0 running Multi-container Docker 1.6.2
 (Generic)",
        "64bit Debian jessie v2.0.0 running GlassFish 4.1 Java 8 (Preconfigured -
 Docker)",
        "64bit Debian jessie v2.0.0 running GlassFish 4.0 Java 7 (Preconfigured -
 Docker)",
        "64bit Debian jessie v2.0.0 running Go 1.4 (Preconfigured - Docker)",
        "64bit Debian jessie v2.0.0 running Go 1.3 (Preconfigured - Docker)",
        "64bit Debian jessie v2.0.0 running Python 3.4 (Preconfigured - Docker)",
    ],
    "SolutionStackDetails": [
        {
            "PermittedFileTypes": [
                "zip"
            ],
            "SolutionStackName": "64bit Amazon Linux 2015.03 v2.0.0 running Node.js"
        },
        ...
    ]
}
```

- Per i dettagli sull'API, consulta ListAvailableSolutionStacks AWS CLICommand Reference.

`rebuild-environment`

Il seguente esempio di codice mostra come utilizzare`rebuild-environment`.

AWS CLI

Come ricompilare un ambiente

Il comando seguente termina e ricrea le risorse in un ambiente denominato `my-env`:

```
aws elasticbeanstalk rebuild-environment --environment-name my-env
```

- Per i dettagli sull'API, consulta RebuildEnvironment AWS CLICommand Reference.

`request-environment-info`

Il seguente esempio di codice mostra come utilizzare`request-environment-info`.

AWS CLI

Come richiedere log in tempo reale

Il comando seguente richiede i log da un ambiente denominato `my-env`:

```
aws elasticbeanstalk request-environment-info --environment-name my-env --info-type tail
```

Dopo aver richiesto i log, recupera la loro posizione con. retrieve-environment-info

- Per i dettagli sull'API, consulta Command RequestEnvironmentInfoReference AWS CLI .

`restart-app-server`

Il seguente esempio di codice mostra come utilizzare`restart-app-server`.

AWS CLI

Come riavviare i server di applicazioni

Il comando seguente riavvia i server di applicazioni su tutte le istanze in un ambiente denominato `my-env`:

```
aws elasticbeanstalk restart-app-server --environment-name my-env
```

- Per i dettagli sull'API, consulta RestartAppServer AWS CLICommand Reference.

`retrieve-environment-info`

Il seguente esempio di codice mostra come utilizzare`retrieve-environment-info`.

AWS CLI

Come recuperare log in tempo reale

Il comando seguente recupera un link ai log da un ambiente denominato `my-env`:

```
aws elasticbeanstalk retrieve-environment-info --environment-name my-env --info-type tail
```

Output:

```
{
    "EnvironmentInfo": [
        {
            "SampleTimestamp": "2015-08-20T22:23:17.703Z",
            "Message": "https://elasticbeanstalk-us-
west-2-0123456789012.s3.amazonaws.com/resources/environments/
logs/tail/e-fyqyju3yjs/i-09c1c867/TailLogs-1440109397703.out?
AWSAccessKeyId=AKGPT4J56IAJ2EUBL5CQ&Expires=1440195891&Signature=n
%2BEalOV6A2HIOx4Rcfb7LT16bBM%3D",
            "InfoType": "tail",
            "Ec2InstanceId": "i-09c1c867"
        }
    ]
}
```

Visualizza il link in un browser. Prima del recupero, i log devono essere richiesti con. request-environment-info

- Per i dettagli sull'API, vedere RetrieveEnvironmentInfoin AWS CLI Command Reference.

`swap-environment-cnames`

Il seguente esempio di codice mostra come utilizzare`swap-environment-cnames`.

AWS CLI

Come scambiare i CNAME di ambiente

Il comando seguente scambia i sottodomini assegnati a due ambienti:

```
aws elasticbeanstalk swap-environment-cnames --source-environment-name my-env-blue
 --destination-environment-name my-env-green
```

- Per i dettagli sull'API, consulta SwapEnvironmentCnames AWS CLICommand Reference.

`terminate-environment`

Il seguente esempio di codice mostra come utilizzare`terminate-environment`.

AWS CLI

Come terminare un ambiente

Il comando seguente termina un ambiente Elastic Beanstalk denominato `my-env`:

```
aws elasticbeanstalk terminate-environment --environment-name my-env
```

Output:

```
{
    "ApplicationName": "my-app",
    "EnvironmentName": "my-env",
    "Status": "Terminating",
    "EnvironmentId": "e-fh2eravpns",
    "EndpointURL": "awseb-e-f-AWSEBLoa-1I9XUMP4-8492WNUP202574.us-
west-2.elb.amazonaws.com",
    "SolutionStackName": "64bit Amazon Linux 2015.03 v2.0.0 running Tomcat 8 Java
 8",
    "CNAME": "my-env.elasticbeanstalk.com",
    "Health": "Grey",
    "AbortableOperationInProgress": false,
    "Tier": {
        "Version": " ",
        "Type": "Standard",
        "Name": "WebServer"
    },
    "DateUpdated": "2015-08-12T19:05:54.744Z",
    "DateCreated": "2015-08-12T18:52:53.622Z"
}
```

- Per i dettagli sull'API, consulta TerminateEnvironment AWS CLICommand Reference.

`update-application-version`

Il seguente esempio di codice mostra come utilizzare`update-application-version`.

AWS CLI

Come modificare la descrizione di una versione dell'applicazione

Il comando seguente aggiorna la descrizione della versione di un'applicazione denominata `22a0-stage-150819_185942`:

```
aws elasticbeanstalk update-application-version --version-label 22a0-
stage-150819_185942 --application-name my-app --description "new description"
```

Output:

```
{
    "ApplicationVersion": {
        "ApplicationName": "my-app",
        "VersionLabel": "22a0-stage-150819_185942",
        "Description": "new description",
        "DateCreated": "2015-08-19T18:59:17.646Z",
        "DateUpdated": "2015-08-20T22:53:28.871Z",
        "SourceBundle": {
            "S3Bucket": "elasticbeanstalk-us-west-2-0123456789012",
            "S3Key": "my-app/22a0-stage-150819_185942.war"
        }
    }
}
```

- Per i dettagli sull'API, consulta UpdateApplicationVersion AWS CLICommand Reference.

`update-application`

Il seguente esempio di codice mostra come utilizzare`update-application`.

AWS CLI

Come modificare la descrizione di un'applicazione

Il comando seguente aggiorna la descrizione di un'applicazione denominata `my-app`:

```
aws elasticbeanstalk update-application --application-name my-app --description "my
 Elastic Beanstalk application"
```

Output:

```
{
    "Application": {
        "ApplicationName": "my-app",
        "Description": "my Elastic Beanstalk application",
        "Versions": [
            "2fba-stage-150819_234450",
            "bf07-stage-150820_214945",
            "93f8",
```

```
            "fd7c-stage-150820_000431",
            "22a0-stage-150819_185942"
        ],
        "DateCreated": "2015-08-13T19:15:50.449Z",
        "ConfigurationTemplates": [],
        "DateUpdated": "2015-08-20T22:34:56.195Z"
    }
}
```

- Per i dettagli sull'API, consulta UpdateApplication AWS CLICommand Reference.

`update-configuration-template`

Il seguente esempio di codice mostra come utilizzare`update-configuration-template`.

AWS CLI

Come aggiornare un modello di configurazione

Il comando seguente rimuove la configurazione delle metriche sanitarie CloudWatch personalizzate configurate `ConfigDocument` da un modello di configurazione salvato denominato`my-template`:

```
aws elasticbeanstalk update-configuration-template --template-
name my-template --application-name my-app --options-to-
remove Namespace=aws:elasticbeanstalk:healthreporting:system,OptionName=ConfigDocument
```

Output:

```
{
    "ApplicationName": "my-app",
    "TemplateName": "my-template",
    "DateCreated": "2015-08-20T22:39:31Z",
    "DateUpdated": "2015-08-20T22:43:11Z",
    "SolutionStackName": "64bit Amazon Linux 2015.03 v2.0.0 running Tomcat 8 Java 8"
}
```

Per ulteriori informazioni sui namespace e sulle opzioni supportate, consulta Valori opzione nella Guida per gli sviluppatori di AWS Elastic Beanstalk.

- Per i dettagli sull'API, vedere UpdateConfigurationTemplatein AWS CLI Command Reference.

`update-environment`

Il seguente esempio di codice mostra come utilizzare`update-environment`.

AWS CLI

Come aggiornare un ambiente a una nuova versione

Il comando seguente aggiorna un ambiente denominato "my-env" alla versione "v2" dell'applicazione a cui appartiene:

```
aws elasticbeanstalk update-environment --environment-name my-env --version-label v2
```

Questo comando richiede che l'ambiente "my-env" esista già e appartenga a un'applicazione che disponga di una versione valida con etichetta "v2".

Output:

```
{
  "ApplicationName": "my-app",
  "EnvironmentName": "my-env",
  "VersionLabel": "v2",
  "Status": "Updating",
  "EnvironmentId": "e-szqipays4h",
  "EndpointURL": "awseb-e-i-AWSEBLoa-1RDLX6TC9VUAO-0123456789.us-
west-2.elb.amazonaws.com",
  "SolutionStackName": "64bit Amazon Linux running Tomcat 7",
  "CNAME": "my-env.elasticbeanstalk.com",
  "Health": "Grey",
  "Tier": {
      "Version": " ",
      "Type": "Standard",
      "Name": "WebServer"
  },
  "DateUpdated": "2015-02-03T23:12:29.119Z",
  "DateCreated": "2015-02-03T23:04:54.453Z"
}
```

Come impostare una variabile di ambiente

Il comando seguente imposta il valore della variabile "PARAM1" nell'ambiente «my-env» su "«: ParamValue

```
aws elasticbeanstalk update-environment --environment-name my-env --option-
settings Namespace=aws:elasticbeanstalk:application:environment,OptionName=PARAM1,Value=Para
```

Il parametro `option-settings` accetta un namespace, oltre al nome e al valore della variabile. Elastic Beanstalk supporta diversi namespace per le opzioni, oltre alle variabili di ambiente.

Come configurare le impostazioni delle opzioni da un file

Il comando seguente configura diverse opzioni nel namespace `aws:elb:loadbalancer` utilizzando un file:

```
aws elasticbeanstalk update-environment --environment-name my-env --option-
settings file://options.json
```

`options.json` è un oggetto JSON che definisce diverse impostazioni:

```
[
  {
    "Namespace": "aws:elb:healthcheck",
    "OptionName": "Interval",
    "Value": "15"
  },
  {
    "Namespace": "aws:elb:healthcheck",
    "OptionName": "Timeout",
    "Value": "8"
  },
  {
    "Namespace": "aws:elb:healthcheck",
    "OptionName": "HealthyThreshold",
    "Value": "2"
  },
  {
    "Namespace": "aws:elb:healthcheck",
    "OptionName": "UnhealthyThreshold",
    "Value": "3"
  }
]
```

Output:

```
{
```

```
    "ApplicationName": "my-app",
    "EnvironmentName": "my-env",
    "VersionLabel": "7f58-stage-150812_025409",
    "Status": "Updating",
    "EnvironmentId": "e-wtp2rpqsej",
    "EndpointURL": "awseb-e-w-AWSEBLoa-14XB83101Q4L-104QXY80921.sa-
east-1.elb.amazonaws.com",
    "SolutionStackName": "64bit Amazon Linux 2015.03 v2.0.0 running Tomcat 8 Java
 8",
    "CNAME": "my-env.elasticbeanstalk.com",
    "Health": "Grey",
    "AbortableOperationInProgress": true,
    "Tier": {
        "Version": " ",
        "Type": "Standard",
        "Name": "WebServer"
    },
    "DateUpdated": "2015-08-12T18:15:23.804Z",
    "DateCreated": "2015-08-07T20:48:49.599Z"
}
```

Per ulteriori informazioni sui namespace e sulle opzioni supportate, consulta Valori opzione nella Guida per gli sviluppatori di AWS Elastic Beanstalk.

- Per i dettagli sull'API, vedere UpdateEnvironmentin AWS CLI Command Reference.

`validate-configuration-settings`

Il seguente esempio di codice mostra come utilizzare`validate-configuration-settings`.

AWS CLI

Come convalidare le impostazioni di configurazione

Il comando seguente convalida un documento di configurazione delle metriche CloudWatch personalizzato:

```
aws elasticbeanstalk validate-configuration-settings --application-name my-app --
environment-name my-env --option-settings file://options.json
```

`options.json` è un documento JSON che include una o più impostazioni di configurazione da convalidare:

```
[
    {
        "Namespace": "aws:elasticbeanstalk:healthreporting:system",
        "OptionName": "ConfigDocument",
        "Value": "{\"CloudWatchMetrics\": {\"Environment\":
 {\"ApplicationLatencyP99.9\": null,\"InstancesSevere\": 60,
\"ApplicationLatencyP90\": 60,\"ApplicationLatencyP99\": null,
\"ApplicationLatencyP95\": 60,\"InstancesUnknown\": 60,\"ApplicationLatencyP85\":
 60,\"InstancesInfo\": null,\"ApplicationRequests2xx\": null,\"InstancesDegraded
\": null,\"InstancesWarning\": 60,\"ApplicationLatencyP50\": 60,
\"ApplicationRequestsTotal\": null,\"InstancesNoData\": null,\"InstancesPending
\": 60,\"ApplicationLatencyP10\": null,\"ApplicationRequests5xx\": null,
\"ApplicationLatencyP75\": null,\"InstancesOk\": 60,\"ApplicationRequests3xx\":
 null,\"ApplicationRequests4xx\": null},\"Instance\": {\"ApplicationLatencyP99.9\":
 null,\"ApplicationLatencyP90\": 60,\"ApplicationLatencyP99\": null,
\"ApplicationLatencyP95\": null,\"ApplicationLatencyP85\": null,\"CPUUser\": 60,
\"ApplicationRequests2xx\": null,\"CPUIdle\": null,\"ApplicationLatencyP50\":
 null,\"ApplicationRequestsTotal\": 60,\"RootFilesystemUtil\": null,
\"LoadAverage1min\": null,\"CPUIrq\": null,\"CPUNice\": 60,\"CPUIowait\": 60,
\"ApplicationLatencyP10\": null,\"LoadAverage5min\": null,\"ApplicationRequests5xx
\": null,\"ApplicationLatencyP75\": 60,\"CPUSystem\": 60,\"ApplicationRequests3xx\":
 60,\"ApplicationRequests4xx\": null,\"InstanceHealth\": null,\"CPUSoftirq\": 60}},
\"Version\": 1}"
    }
]
```

Se le opzioni specificate sono valide per l'ambiente indicato, Elastic Beanstalk restituisce un array Messages vuoto:

```
{
    "Messages": []
}
```

Se la convalida non riesce, la risposta includerà informazioni sull'errore:

```
{
    "Messages": [
        {
            "OptionName": "ConfigDocumet",
            "Message": "Invalid option specification (Namespace:
 'aws:elasticbeanstalk:healthreporting:system', OptionName: 'ConfigDocumet'):
 Unknown configuration setting.",
```

```
            "Namespace": "aws:elasticbeanstalk:healthreporting:system",
            "Severity": "error"
        }
    ]
}
```

Per ulteriori informazioni sui namespace e sulle opzioni supportate, consulta Valori opzione nella Guida per gli sviluppatori di AWS Elastic Beanstalk.

- Per i dettagli sull'API, consultate Command ValidateConfigurationSettingsReference AWS CLI .

Elastic Load Balancing - Esempi di utilizzo della versione 1 AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with Elastic Load Balancing - Versione 1.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`add-tags`

Il seguente esempio di codice mostra come utilizzare`add-tags`.

AWS CLI

Come aggiungere un tag a un bilanciatore del carico

Questo esempio aggiunge tag al bilanciatore del carico specificato.

Comando:

```
aws elb add-tags --load-balancer-name my-load-balancer --
tags "Key=project,Value=lima" "Key=department,Value=digital-media"
```

- Per i dettagli sull'API, consulta AddTags AWS CLICommand Reference.

`apply-security-groups-to-load-balancer`

Il seguente esempio di codice mostra come utilizzare`apply-security-groups-to-load-balancer`.

AWS CLI

Come associare un gruppo di sicurezza a un bilanciatore del carico in un VPC

Questo esempio associa un gruppo di sicurezza al bilanciatore del carico specificato in un VPC.

Comando:

```
aws elb apply-security-groups-to-load-balancer --load-balancer-name my-load-balancer
 --security-groups sg-fc448899
```

Output:

```
{
  "SecurityGroups": [
      "sg-fc448899"
  ]
}
```

- Per i dettagli sull'API, consulta ApplySecurityGroupsToLoadBalancer AWS CLICommand Reference.

`attach-load-balancer-to-subnets`

Il seguente esempio di codice mostra come utilizzare`attach-load-balancer-to-subnets`.

AWS CLI

Come associare sottoreti a un bilanciatore del carico

Questo esempio aggiunge la sottorete specificata al set di sottoreti configurate per il bilanciatore del carico specificato.

Comando:

```
aws elb attach-load-balancer-to-subnets --load-balancer-name my-load-balancer --
subnets subnet-0ecac448
```

Output:

```
{
    "Subnets": [
        "subnet-15aaab61",
        "subnet-0ecac448"
    ]
}
```

- Per i dettagli sull'API, consulta AttachLoadBalancerToSubnets AWS CLICommand Reference.

`configure-health-check`

Il seguente esempio di codice mostra come utilizzare`configure-health-check`.

AWS CLI

Per specificare le impostazioni del controllo dello stato delle istanze di backend EC2

Questo esempio specifica le impostazioni del controllo dello stato utilizzate per valutare lo stato delle istanze di backend. EC2

Comando:

```
aws elb configure-health-check --load-balancer-name my-load-balancer --health-
check Target=HTTP:80/
png,Interval=30,UnhealthyThreshold=2,HealthyThreshold=2,Timeout=3
```

Output:

```
{
    "HealthCheck": {
        "HealthyThreshold": 2,
        "Interval": 30,
```

```
        "Target": "HTTP:80/png",
        "Timeout": 3,
        "UnhealthyThreshold": 2
    }
}
```

- Per i dettagli sull'API, consulta Command ConfigureHealthCheckReference AWS CLI .

`create-app-cookie-stickiness-policy`

Il seguente esempio di codice mostra come utilizzare`create-app-cookie-stickiness-policy`.

AWS CLI

Come generare una policy di persistenza per il bilanciatore del carico HTTPS

Questo esempio genera una policy di persistenza conforme alla durata delle sessioni persistenti del cookie generato dall'applicazione.

Comando:

```
aws elb create-app-cookie-stickiness-policy --load-balancer-name my-load-balancer --policy-name my-app-cookie-policy --cookie-name my-app-cookie
```

- Per i dettagli sull'API, consulta CreateAppCookieStickinessPolicy AWS CLICommand Reference.

`create-lb-cookie-stickiness-policy`

Il seguente esempio di codice mostra come utilizzare`create-lb-cookie-stickiness-policy`.

AWS CLI

Come generare una policy di persistenza basata sulla durata per il bilanciatore del carico HTTPS

Questo esempio genera una policy di persistenza con durate delle sessioni persistenti controllate dal periodo di scadenza specificato.

Comando:

```
aws elb create-lb-cookie-stickiness-policy --load-balancer-name my-load-balancer --policy-name my-duration-cookie-policy --cookie-expiration-period 60
```

- Per i dettagli sull'API, consulta CreateLbCookieStickinessPolicy AWS CLICommand Reference.

`create-load-balancer-listeners`

Il seguente esempio di codice mostra come utilizzare`create-load-balancer-listeners`.

AWS CLI

Come creare listener HTTP per un bilanciatore del carico

Questo esempio crea un listener per il bilanciatore del carico sulla porta 80 utilizzando il protocollo HTTP.

Comando:

```
aws elb create-load-balancer-listeners --load-balancer-name my-load-balancer --
listeners "Protocol=HTTP,LoadBalancerPort=80,InstanceProtocol=HTTP,InstancePort=80"
```

Come creare listener HTTPS per un bilanciatore del carico

Questo esempio crea un listener per il bilanciatore del carico sulla porta 443 utilizzando il protocollo HTTPS.

Comando:

```
aws elb create-load-balancer-listeners --load-balancer-name my-load-balancer --
listeners "Protocol=HTTPS,LoadBalancerPort=443,InstanceProtocol=HTTP,InstancePort=80"
```

- Per i dettagli sull'API, consulta CreateLoadBalancerListeners AWS CLICommand Reference.

`create-load-balancer-policy`

Il seguente esempio di codice mostra come utilizzare`create-load-balancer-policy`.

AWS CLI

Come creare una policy che abiliti il protocollo proxy su un bilanciatore del carico

Questo esempio crea una policy che abilita il protocollo proxy per il bilanciatore del carico specificato.

Comando:

```
aws elb create-load-balancer-policy --load-balancer-name my-load-balancer --policy-
name my-ProxyProtocol-policy --policy-type-name ProxyProtocolPolicyType --policy-
attributes AttributeName=ProxyProtocol,AttributeValue=true
```

Come creare una policy di negoziazione SSL utilizzando la policy di sicurezza consigliata

Questo esempio crea una policy di negoziazione SSL per il bilanciatore del carico HTTPS specificato utilizzando la policy di sicurezza consigliata.

Comando:

```
aws elb create-load-balancer-policy --load-balancer-name my-load-
balancer --policy-name my-SSLNegotiation-policy --policy-type-
name SSLNegotiationPolicyType --policy-attributes AttributeName=Reference-Security-
Policy,AttributeValue=ELBSecurityPolicy-2015-03
```

Come creare una policy di negoziazione SSL utilizzando la policy di sicurezza personalizzata

Questo esempio crea una policy di negoziazione SSL per il bilanciatore del carico HTTPS utilizzando una policy di sicurezza personalizzata abilitando i protocolli e la crittografia.

Comando:

```
aws elb create-load-balancer-policy --load-balancer-name my-load-balancer --policy-
name my-SSLNegotiation-policy --policy-type-name SSLNegotiationPolicyType --policy-
attributes AttributeName=Protocol-SSLv3,AttributeValue=true AttributeName=Protocol-
TLSv1.1,AttributeValue=true AttributeName=DHE-RSA-AES256-
SHA256,AttributeValue=true AttributeName=Server-Defined-Cipher-
Order,AttributeValue=true
```

Come creare una policy delle chiavi pubbliche

Questo esempio crea una policy delle chiavi pubbliche.

Comando:

```
aws elb create-load-balancer-policy --load-balancer-name my-load-balancer --
policy-name my-PublicKey-policy --policy-type-name PublicKeyPolicyType --policy-
attributes AttributeName=PublicKey,AttributeValue=MIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQ
+dS74kj//c6x7ROtusUaeQCTgIUkayttRDWchuqo1pHC1u
+n5xxXnBBe2ejbb2WRsKIQ5rXEeixsjFpFsojpSQKkzhVGI6mJVZBJDVKSHmswnwLBdofLhzvllpovBPTHe
+o4haAWvDBALJU0pkSI1FecPHcs2hwxf14zHoXy1e2k36A64nXW43wtfx5qcVSIxtCEOjnYRg7RPvybaGfQ
```

```
+v6Iaxb/+7J5kEvZhTFQId+bSiJImF1FSUT1W1xwzBZPUbcUkkXDj45vC2s3Z8E
+Lk7a3uZhvsQHLZnrfuWjBWGWvZ/MhZYgEXAMPLE
```

Come creare una policy di autenticazione del server di backend

Questo esempio crea una policy di autenticazione del server di backend che abilita l'autenticazione sull'istanza di backend utilizzando una policy delle chiavi pubbliche.

Comando:

```
aws elb create-load-balancer-policy --load-balancer-name my-
load-balancer --policy-name my-authentication-policy --policy-
type-name BackendServerAuthenticationPolicyType --policy-
attributes AttributeName=PublicKeyPolicyName,AttributeValue=my-PublicKey-policy
```

- Per i dettagli sull'API, consulta CreateLoadBalancerPolicy AWS CLICommand Reference.

`create-load-balancer`

Il seguente esempio di codice mostra come utilizzare`create-load-balancer`.

AWS CLI

Come creare un bilanciatore del carico HTTP

Questo esempio crea un bilanciatore del carico con un listener HTTP in un VPC.

Comando:

```
aws elb create-load-balancer --load-balancer-name my-load-balancer --
listeners "Protocol=HTTP,LoadBalancerPort=80,InstanceProtocol=HTTP,InstancePort=80"
 --subnets subnet-15aaab61 --security-groups sg-a61988c3
```

Output:

```
{
    "DNSName": "my-load-balancer-1234567890.us-west-2.elb.amazonaws.com"
}
```

Questo esempio crea un sistema di bilanciamento del carico con un listener HTTP in EC2 - Classic.

Comando:

```
aws elb create-load-balancer --load-balancer-name my-load-balancer --
listeners "Protocol=HTTP,LoadBalancerPort=80,InstanceProtocol=HTTP,InstancePort=80"
 --availability-zones us-west-2a us-west-2b
```

Output:

```
{
    "DNSName": "my-load-balancer-123456789.us-west-2.elb.amazonaws.com"
}
```

Come creare un bilanciatore del carico HTTPS

Questo esempio crea un bilanciatore del carico con un listener HTTPS in un VPC.

Comando:

```
aws elb create-load-balancer --load-balancer-name my-load-balancer --
listeners "Protocol=HTTP,LoadBalancerPort=80,InstanceProtocol=HTTP,InstancePort=80" "Protoco
certificate/my-server-cert" --subnets subnet-15aaab61 --security-groups sg-a61988c3
```

Output:

```
{
    "DNSName": "my-load-balancer-1234567890.us-west-2.elb.amazonaws.com"
}
```

Questo esempio crea un sistema di bilanciamento del carico con un listener HTTPS in -Classic. EC2

Comando:

```
aws elb create-load-balancer --load-balancer-name my-load-balancer --
listeners "Protocol=HTTP,LoadBalancerPort=80,InstanceProtocol=HTTP,InstancePort=80" "Protoco
certificate/my-server-cert" --availability-zones us-west-2a us-west-2b
```

Output:

```
{
```

```
    "DNSName": "my-load-balancer-123456789.us-west-2.elb.amazonaws.com"
}
```

Come creare un bilanciatore del carico interno

Questo esempio crea un bilanciatore del carico interno con un listener HTTP in un VPC.

Comando:

```
aws elb create-load-balancer --load-balancer-name my-load-balancer --
listeners "Protocol=HTTP,LoadBalancerPort=80,InstanceProtocol=HTTP,InstancePort=80"
 --scheme internal --subnets subnet-a85db0df --security-groups sg-a61988c3
```

Output:

```
{
    "DNSName": "internal-my-load-balancer-123456789.us-west-2.elb.amazonaws.com"
}
```

- Per i dettagli sull'API, consulta Command CreateLoadBalancerReference AWS CLI .

`delete-load-balancer-listeners`

Il seguente esempio di codice mostra come utilizzare`delete-load-balancer-listeners`.

AWS CLI

Come eliminare un listener dal bilanciatore del carico

Questo esempio elimina il listener per la porta specificata dal bilanciatore del carico specificato.

Comando:

```
aws elb delete-load-balancer-listeners --load-balancer-name my-load-balancer --load-
balancer-ports 80
```

- Per i dettagli sull'API, consulta DeleteLoadBalancerListeners AWS CLICommand Reference.

`delete-load-balancer-policy`

Il seguente esempio di codice mostra come utilizzare`delete-load-balancer-policy`.

AWS CLI

Come eliminare una policy dal bilanciatore del carico

Questo esempio elimina la policy specificata dal bilanciatore del carico specificato. La policy non deve essere abilitata in nessun listener.

Comando:

```
aws elb delete-load-balancer-policy --load-balancer-name my-load-balancer --policy-name my-duration-cookie-policy
```

- Per i dettagli sull'API, consulta DeleteLoadBalancerPolicy AWS CLICommand Reference.

`delete-load-balancer`

Il seguente esempio di codice mostra come utilizzare`delete-load-balancer`.

AWS CLI

Come eliminare un bilanciatore del carico

Questo esempio elimina il bilanciatore del carico specificato.

Comando:

```
aws elb delete-load-balancer --load-balancer-name my-load-balancer
```

- Per i dettagli sull'API, consulta DeleteLoadBalancer AWS CLICommand Reference.

`deregister-instances-from-load-balancer`

Il seguente esempio di codice mostra come utilizzare`deregister-instances-from-load-balancer`.

AWS CLI

Come annullare la registrazione delle istanze al bilanciatore del carico

Questo esempio annulla la registrazione dell'istanza specificata al bilanciatore del carico specificato.

Comando:

```
aws elb deregister-instances-from-load-balancer --load-balancer-name my-load-balancer --instances i-d6f6fae3
```

Output:

```
{
    "Instances": [
        {
            "InstanceId": "i-207d9717"
        },
        {
            "InstanceId": "i-afefb49b"
        }
    ]
}
```

- Per i dettagli sull'API, consulta DeregisterInstancesFromLoadBalancer AWS CLICommand Reference.

`describe-account-limits`

Il seguente esempio di codice mostra come utilizzare`describe-account-limits`.

AWS CLI

Come descrivere i limiti del Classic Load Balancer

L'`describe-account-limits`esempio seguente mostra i dettagli sui limiti del Classic Load Balancer per il tuo AWS account.

```
aws elb describe-account-limits
```

Output:

```
{
    "Limits": [
        {
            "Name": "classic-load-balancers",
            "Max": "20"
```

```
        },
        {
            "Name": "classic-listeners",
            "Max": "100"
        },
        {
            "Name": "classic-registered-instances",
            "Max": "1000"
        }
    ]
}
```

- Per i dettagli sull'API, consulta DescribeAccountLimits AWS CLICommand Reference.

`describe-instance-health`

Il seguente esempio di codice mostra come utilizzare`describe-instance-health`.

AWS CLI

Come descrivere l'integrità delle istanze per un bilanciatore del carico

Questo esempio descrive l'integrità delle istanze per il bilanciatore del carico specificato.

Comando:

```
aws elb describe-instance-health --load-balancer-name my-load-balancer
```

Output:

```
{
  "InstanceStates": [
      {
          "InstanceId": "i-207d9717",
          "ReasonCode": "N/A",
          "State": "InService",
          "Description": "N/A"
      },
      {
          "InstanceId": "i-afefb49b",
          "ReasonCode": "N/A",
          "State": "InService",
          "Description": "N/A"
```

```
        }
    ]
}
```

Come descrivere l'integrità di un'istanza per un bilanciatore del carico

Questo esempio descrive l'integrità dell'istanza specificata per il bilanciatore del carico specificato.

Comando:

```
aws elb describe-instance-health --load-balancer-name my-load-balancer --
instances i-7299c809
```

Di seguito è riportato un esempio di risposta per un'istanza in corso di registrazione.

Output:

```
{
  "InstanceStates": [
      {
          "InstanceId": "i-7299c809",
          "ReasonCode": "ELB",
          "State": "OutOfService",
          "Description": "Instance registration is still in progress."
    }
  ]
}
```

Di seguito è riportato un esempio di risposta per un'istanza non integra.

Output:

```
{
  "InstanceStates": [
      {
          "InstanceId": "i-7299c809",
          "ReasonCode": "Instance",
          "State": "OutOfService",
          "Description": "Instance has failed at least the UnhealthyThreshold number
 of health checks consecutively."
      }
  ]
}
```

- Per i dettagli sull'API, consulta DescribeInstanceHealth AWS CLICommand Reference.

`describe-load-balancer-attributes`

Il seguente esempio di codice mostra come utilizzare`describe-load-balancer-attributes`.

AWS CLI

Come descrivere gli attributi di un bilanciatore del carico

Questo esempio descrive gli attributi del bilanciatore del carico specificato.

Comando:

```
aws elb describe-load-balancer-attributes --load-balancer-name my-load-balancer
```

Output:

```
{
  "LoadBalancerAttributes": {
      "ConnectionDraining": {
          "Enabled": false,
          "Timeout": 300
      },
      "CrossZoneLoadBalancing": {
          "Enabled": true
      },
      "ConnectionSettings": {
          "IdleTimeout": 30
      },
      "AccessLog": {
          "Enabled": false
    }
  }
}
```

- Per i dettagli sull'API, consulta DescribeLoadBalancerAttributes AWS CLICommand Reference.

`describe-load-balancer-policies`

Il seguente esempio di codice mostra come utilizzare`describe-load-balancer-policies`.

AWS CLI

Come descrivere tutte le policy associate a un bilanciatore del carico

Questo esempio descrive tutte le policy associate al bilanciatore del carico specificato.

Comando:

```
aws elb describe-load-balancer-policies --load-balancer-name my-load-balancer
```

Output:

```
{
  "PolicyDescriptions": [
    {
      "PolicyAttributeDescriptions": [
        {
          "AttributeName": "ProxyProtocol",
          "AttributeValue": "true"
        }
      ],
      "PolicyName": "my-ProxyProtocol-policy",
      "PolicyTypeName": "ProxyProtocolPolicyType"
    },
    {
        "PolicyAttributeDescriptions": [
            {
                "AttributeName": "CookieName",
                "AttributeValue": "my-app-cookie"
            }
        ],
        "PolicyName": "my-app-cookie-policy",
        "PolicyTypeName": "AppCookieStickinessPolicyType"
    },
    {
      "PolicyAttributeDescriptions": [
        {
          "AttributeName": "CookieExpirationPeriod",
          "AttributeValue": "60"
        }
      ],
      "PolicyName": "my-duration-cookie-policy",
      "PolicyTypeName": "LBCookieStickinessPolicyType"
```

```
    },
    .
    .
    .
  ]
}
```

Come descrivere una policy specifica associata a un bilanciatore del carico

Questo esempio descrive la policy specificata associate al bilanciatore del carico specificato.

Comando:

```
aws elb describe-load-balancer-policies --load-balancer-name my-load-balancer --
policy-name my-authentication-policy
```

Output:

```
{
  "PolicyDescriptions": [
      {
          "PolicyAttributeDescriptions": [
              {
                  "AttributeName": "PublicKeyPolicyName",
                  "AttributeValue": "my-PublicKey-policy"
              }
          ],
          "PolicyName": "my-authentication-policy",
          "PolicyTypeName": "BackendServerAuthenticationPolicyType"
      }
  ]
}
```

- Per i dettagli sull'API, consulta DescribeLoadBalancerPolicies AWS CLICommand Reference.

`describe-load-balancer-policy-types`

Il seguente esempio di codice mostra come utilizzare`describe-load-balancer-policy-types`.

AWS CLI

Come descrivere i tipi di policy del bilanciatore del carico definiti da Elastic Load Balancing

Questo esempio descrive i tipi di policy del bilanciatore del carico che è possibile utilizzare per creare configurazioni di policy per il bilanciatore del carico.

Comando:

```
aws elb describe-load-balancer-policy-types
```

Output:

```
{
  "PolicyTypeDescriptions": [
      {
          "PolicyAttributeTypeDescriptions": [
              {
                  "Cardinality": "ONE",
                  "AttributeName": "ProxyProtocol",
                  "AttributeType": "Boolean"
              }
          ],
          "PolicyTypeName": "ProxyProtocolPolicyType",
          "Description": "Policy that controls whether to include the IP address and
 port of the originating request for TCP messages. This policy operates on TCP/SSL
 listeners only"
      },
      {
          "PolicyAttributeTypeDescriptions": [
              {
                  "Cardinality": "ONE",
                  "AttributeName": "PublicKey",
                  "AttributeType": "String"
              }
          ],
          "PolicyTypeName": "PublicKeyPolicyType",
          "Description": "Policy containing a list of public keys to
 accept when authenticating the back-end server(s). This policy cannot be
 applied directly to back-end servers or listeners but must be part of a
 BackendServerAuthenticationPolicyType."
      },
      {
          "PolicyAttributeTypeDescriptions": [
              {
                  "Cardinality": "ONE",
                  "AttributeName": "CookieName",
```

```
                "AttributeType": "String"
            }
        ],
        "PolicyTypeName": "AppCookieStickinessPolicyType",
        "Description": "Stickiness policy with session lifetimes controlled by the
lifetime of the application-generated cookie. This policy can be associated only
with HTTP/HTTPS listeners."
    },
    {
        "PolicyAttributeTypeDescriptions": [
            {
                "Cardinality": "ZERO_OR_ONE",
                "AttributeName": "CookieExpirationPeriod",
                "AttributeType": "Long"
            }
        ],
        "PolicyTypeName": "LBCookieStickinessPolicyType",
        "Description": "Stickiness policy with session lifetimes controlled by
the browser (user-agent) or a specified expiration period. This policy can be
associated only with HTTP/HTTPS listeners."
    },
    {
        "PolicyAttributeTypeDescriptions": [
            .
            .
            .
        ],
        "PolicyTypeName": "SSLNegotiationPolicyType",
        "Description": "Listener policy that defines the ciphers and protocols
that will be accepted by the load balancer. This policy can be associated only with
HTTPS/SSL listeners."
    },
    {
        "PolicyAttributeTypeDescriptions": [
            {
                "Cardinality": "ONE_OR_MORE",
                "AttributeName": "PublicKeyPolicyName",
                "AttributeType": "PolicyName"
            }
        ],
        "PolicyTypeName": "BackendServerAuthenticationPolicyType",
        "Description": "Policy that controls authentication to back-end server(s)
and contains one or more policies, such as an instance of a PublicKeyPolicyType.
This policy can be associated only with back-end servers that are using HTTPS/SSL."
```

```
      }
  ]
}
```

- Per i dettagli sull'API, consulta DescribeLoadBalancerPolicyTypes AWS CLICommand Reference.

`describe-load-balancers`

Il seguente esempio di codice mostra come utilizzare`describe-load-balancers`.

AWS CLI

Come descrivere i bilanciatori del carico

Questo esempio descrive tutti i bilanciatori del carico.

Comando:

```
aws elb describe-load-balancers
```

Come descrivere uno dei bilanciatori del carico in uso

Questo esempio descrive il bilanciatore del carico specificato.

Comando:

```
aws elb describe-load-balancers --load-balancer-name my-load-balancer
```

Di seguito è riportata una risposta di esempio per un bilanciatore del carico HTTPS in un VPC.

Output:

```
{
  "LoadBalancerDescriptions": [
    {
      "Subnets": [
          "subnet-15aaab61"
      ],
      "CanonicalHostedZoneNameID": "Z3DZXE0EXAMPLE",
      "CanonicalHostedZoneName": "my-load-balancer-1234567890.us-
west-2.elb.amazonaws.com",
      "ListenerDescriptions": [
```

```
            {
                "Listener": {
                    "InstancePort": 80,
                    "LoadBalancerPort": 80,
                    "Protocol": "HTTP",
                    "InstanceProtocol": "HTTP"
                },
                "PolicyNames": []
            },
            {
                "Listener": {
                    "InstancePort": 443,
                    "SSLCertificateId": "arn:aws:iam::123456789012:server-certificate/
my-server-cert",
                    "LoadBalancerPort": 443,
                    "Protocol": "HTTPS",
                    "InstanceProtocol": "HTTPS"
                },
                "PolicyNames": [
                    "ELBSecurityPolicy-2015-03"
                ]
            }
        ],
        "HealthCheck": {
            "HealthyThreshold": 2,
            "Interval": 30,
            "Target": "HTTP:80/png",
            "Timeout": 3,
            "UnhealthyThreshold": 2
        },
        "VPCId": "vpc-a01106c2",
        "BackendServerDescriptions": [
            {
                "InstancePort": 80,
                "PolicyNames": [
                    "my-ProxyProtocol-policy"
                ]
            }
        ],
        "Instances": [
            {
                "InstanceId": "i-207d9717"
            },
            {
```

```
                "InstanceId": "i-afefb49b"
            }
        ],
        "DNSName": "my-load-balancer-1234567890.us-west-2.elb.amazonaws.com",
        "SecurityGroups": [
            "sg-a61988c3"
        ],
        "Policies": {
            "LBCookieStickinessPolicies": [
                {
                    "PolicyName": "my-duration-cookie-policy",
                    "CookieExpirationPeriod": 60
                }
            ],
            "AppCookieStickinessPolicies": [],
            "OtherPolicies": [
                "my-PublicKey-policy",
                "my-authentication-policy",
                "my-SSLNegotiation-policy",
                "my-ProxyProtocol-policy",
                "ELBSecurityPolicy-2015-03"
            ]
        },
        "LoadBalancerName": "my-load-balancer",
        "CreatedTime": "2015-03-19T03:24:02.650Z",
        "AvailabilityZones": [
            "us-west-2a"
        ],
        "Scheme": "internet-facing",
        "SourceSecurityGroup": {
            "OwnerAlias": "123456789012",
            "GroupName": "my-elb-sg"
        }
    }
  ]
}
```

- Per i dettagli sull'API, consulta DescribeLoadBalancers AWS CLICommand Reference.

`describe-tags`

Il seguente esempio di codice mostra come utilizzare`describe-tags`.

AWS CLI

Come descrivere i tag assegnati a un bilanciatore del carico

Questo esempio descrive i tag assegnati al bilanciatore del carico specificato.

Comando:

```
aws elb describe-tags --load-balancer-name my-load-balancer
```

Output:

```
{
  "TagDescriptions": [
      {
          "Tags": [
              {
                  "Value": "lima",
                  "Key": "project"
              },
              {
                  "Value": "digital-media",
                  "Key": "department"
              }
          ],
          "LoadBalancerName": "my-load-balancer"
      }
  ]
}
```

- Per i dettagli sull'API, consulta DescribeTags AWS CLICommand Reference.

`detach-load-balancer-from-subnets`

Il seguente esempio di codice mostra come utilizzare`detach-load-balancer-from-subnets`.

AWS CLI

Come scollegare i bilanciatori del carico dalle sottoreti

Questo esempio scollega il bilanciatore del carico specificato dalla sottorete specificata.

Comando:

```
aws elb detach-load-balancer-from-subnets --load-balancer-name my-load-balancer --
subnets subnet-0ecac448
```

Output:

```
{
   "Subnets": [
       "subnet-15aaab61"
   ]
}
```

- Per i dettagli sull'API, consulta DetachLoadBalancerFromSubnets AWS CLICommand Reference.

`disable-availability-zones-for-load-balancer`

Il seguente esempio di codice mostra come utilizzare`disable-availability-zones-for-load-balancer`.

AWS CLI

Come disabilitare le zone di disponibilità per un bilanciatore del carico

Questo esempio rimuove la zona di disponibilità specificata dal set di zone di disponibilità per il bilanciatore del carico specificato.

Comando:

```
aws elb disable-availability-zones-for-load-balancer --load-balancer-name my-load-
balancer --availability-zones us-west-2a
```

Output:

```
{
    "AvailabilityZones": [
        "us-west-2b"
    ]
}
```

- Per i dettagli sull'API, consulta DisableAvailabilityZonesForLoadBalancer AWS CLICommand Reference.

`enable-availability-zones-for-load-balancer`

Il seguente esempio di codice mostra come utilizzare`enable-availability-zones-for-load-balancer`.

AWS CLI

Come abilitare le zone di disponibilità per un bilanciatore del carico

Questo esempio aggiunge la zona di disponibilità specificata al bilanciatore del carico specificato.

Comando:

```
aws elb enable-availability-zones-for-load-balancer --load-balancer-name my-load-balancer --availability-zones us-west-2b
```

Output:

```
{
    "AvailabilityZones": [
        "us-west-2a",
        "us-west-2b"
    ]
}
```

- Per i dettagli sull'API, consulta EnableAvailabilityZonesForLoadBalancer AWS CLICommand Reference.

`modify-load-balancer-attributes`

Il seguente esempio di codice mostra come utilizzare`modify-load-balancer-attributes`.

AWS CLI

Come modificare gli attributi di un bilanciatore del carico

Questo esempio modifica l'attributo `CrossZoneLoadBalancing` del bilanciatore del carico specificato.

Comando:

```
aws elb modify-load-balancer-attributes --load-balancer-name my-load-balancer --
load-balancer-attributes "{\"CrossZoneLoadBalancing\":{\"Enabled\":true}}"
```

Output:

```
{
    "LoadBalancerAttributes": {
        "CrossZoneLoadBalancing": {
            "Enabled": true
        }
    },
    "LoadBalancerName": "my-load-balancer"
}
```

Questo esempio modifica l'attributo `ConnectionDraining` del bilanciatore del carico specificato.

Comando:

```
aws elb modify-load-balancer-attributes --load-balancer-name my-load-balancer
 --load-balancer-attributes "{\"ConnectionDraining\":{\"Enabled\":true,\"Timeout
\":300}}"
```

Output:

```
{
    "LoadBalancerAttributes": {
        "ConnectionDraining": {
            "Enabled": true,
            "Timeout": 300
        }
    },
    "LoadBalancerName": "my-load-balancer"
}
```

- Per i dettagli sull'API, consulta ModifyLoadBalancerAttributes AWS CLICommand Reference.

`register-instances-with-load-balancer`

Il seguente esempio di codice mostra come utilizzare`register-instances-with-load-balancer`.

AWS CLI

Come registrare le istanze con un bilanciatore del carico

Questo esempio registra l'istanza specificata con il bilanciatore del carico specificato.

Comando:

```
aws elb register-instances-with-load-balancer --load-balancer-name my-load-balancer
 --instances i-d6f6fae3
```

Output:

```
{
   "Instances": [
       {
           "InstanceId": "i-d6f6fae3"
       },
       {
           "InstanceId": "i-207d9717"
       },
       {
           "InstanceId": "i-afefb49b"
       }
   ]
}
```

- Per i dettagli sull'API, consulta RegisterInstancesWithLoadBalancer AWS CLICommand Reference.

`remove-tags`

Il seguente esempio di codice mostra come utilizzare`remove-tags`.

AWS CLI

Come rimuovere i tag da un bilanciatore del carico

Questo esempio rimuove un tag dal bilanciatore del carico specificato.

Comando:

```
aws elb remove-tags --load-balancer-name my-load-balancer --tags project
```

- Per i dettagli sull'API, consulta RemoveTags AWS CLICommand Reference.

`set-load-balancer-listener-ssl-certificate`

Il seguente esempio di codice mostra come utilizzare`set-load-balancer-listener-ssl-certificate`.

AWS CLI

Come aggiornare il certificato SSL per un bilanciatore del carico HTTPS

Questo esempio sostituisce il certificato SSL esistente per il bilanciatore del carico HTTPS specificato.

Comando:

```
aws elb set-load-balancer-listener-ssl-certificate --load-balancer-
name my-load-balancer --load-balancer-port 443 --ssl-certificate-
id arn:aws:iam::123456789012:server-certificate/new-server-cert
```

- Per i dettagli sull'API, consulta SetLoadBalancerListenerSslCertificate AWS CLICommand Reference.

`set-load-balancer-policies-for-backend-server`

Il seguente esempio di codice mostra come utilizzare`set-load-balancer-policies-for-backend-server`.

AWS CLI

Come sostituire le policy associate a una porta per un'istanza di backend

Questo esempio sostituisce le policy attualmente associate alla porta specificata.

Comando:

```
aws elb set-load-balancer-policies-for-backend-server --load-balancer-name my-load-
balancer --instance-port 80 --policy-names my-ProxyProtocol-policy
```

Come rimuovere tutte le policy attualmente associate a una porta sull'istanza di backend

Questo esempio rimuove tutte le policy associate alla porta specificata.

Comando:

```
aws elb set-load-balancer-policies-for-backend-server --load-balancer-name my-load-balancer --instance-port 80 --policy-names []
```

Per confermare l'effettiva rimozione delle policy, utilizza il comando `describe-load-balancer-policies`.

- Per i dettagli sull'API, consulta SetLoadBalancerPoliciesForBackendServer AWS CLICommand Reference.

`set-load-balancer-policies-of-listener`

Il seguente esempio di codice mostra come utilizzare`set-load-balancer-policies-of-listener`.

AWS CLI

Come sostituire le policy associate a un listener

Questo esempio sostituisce le policy attualmente associate al listener specificato.

Comando:

```
aws elb set-load-balancer-policies-of-listener --load-balancer-name my-load-balancer --load-balancer-port 443 --policy-names my-SSLNegotiation-policy
```

Come rimuovere tutte le policy associate al listener

Questo esempio rimuove tutte le policy attualmente associate al listener specificato.

Comando:

```
aws elb set-load-balancer-policies-of-listener --load-balancer-name my-load-balancer --load-balancer-port 443 --policy-names []
```

Per confermare l'effettiva rimozione delle policy dal bilanciatore del carico, utilizza il comando `describe-load-balancer-policies`.

- Per i dettagli sull'API, consulta SetLoadBalancerPoliciesOfListener AWS CLICommand Reference.

Elastic Load Balancing - Esempi di utilizzo della versione 2 AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with Elastic Load Balancing - Versione 2.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`add-listener-certificates`

Il seguente esempio di codice mostra come utilizzare`add-listener-certificates`.

AWS CLI

Come aggiungere un certificato a un listener sicuro

Questo esempio aggiunge il certificato specificato al listener sicuro specificato.

Comando:

```
aws elbv2 add-listener-certificates --listener-arn arn:aws:elasticloadbalancing:us-west-2:123456789012:listener/app/my-load-balancer/50dc6c495c0c9188/f2f7dc8efc522ab2 --certificates CertificateArn=arn:aws:acm:us-west-2:123456789012:certificate/5cc54884-f4a3-4072-80be-05b9ba72f705
```

Output:

```
{
```

```
  "Certificates": [
      {
          "CertificateArn": "arn:aws:acm:us-
west-2:123456789012:certificate/5cc54884-f4a3-4072-80be-05b9ba72f705",
          "IsDefault": false
      }
  ]
}
```

- Per i dettagli sull'API, consulta AddListenerCertificates AWS CLICommand Reference.

`add-tags`

Il seguente esempio di codice mostra come utilizzare`add-tags`.

AWS CLI

Come aggiungere tag a un bilanciatore del carico

L'esempio `add-tags` seguente aggiunge i tag `project` e `department` al bilanciatore del carico specificato.

```
aws elbv2 add-tags \
    --resource-arns arn:aws:elasticloadbalancing:us-
west-2:123456789012:loadbalancer/app/my-load-balancer/50dc6c495c0c9188 \
    --tags "Key=project,Value=lima" "Key=department,Value=digital-media"
```

- Per i dettagli sull'API, consulta AddTags AWS CLICommand Reference.

`create-listener`

Il seguente esempio di codice mostra come utilizzare`create-listener`.

AWS CLI

Esempio 1: come creare un listener HTTP

L'esempio `create-listener` seguente crea un listener HTTP per l'Application Load Balancer specificato che inoltra le richieste al gruppo di destinazione specificato.

```
aws elbv2 create-listener \
```

```
    --load-balancer-arn arn:aws:elasticloadbalancing:us-
west-2:123456789012:loadbalancer/app/my-load-balancer/50dc6c495c0c9188 \
    --protocol HTTP \
    --port 80 \
    --default-actions Type=forward,TargetGroupArn=arn:aws:elasticloadbalancing:us-
west-2:123456789012:targetgroup/my-targets/73e2d6bc24d8a067
```

Per ulteriori informazioni, consulta Tutorial: Create an Application Load Balancer usando la AWS CLI nella User Guide for Application Load Balancers.

Esempio 2: come creare un listener HTTPS

L'esempio `create-listener` seguente crea un listener HTTPS per l'Application Load Balancer specificato che inoltra le richieste al gruppo di destinazione specificato. Devi specificare un certificato SSL per un listener HTTPS. È possibile creare e gestire certificati utilizzando AWS Certificate Manager (ACM). In alternativa, puoi creare un certificato utilizzando SSL/TLS strumenti, far firmare il certificato da un'autorità di certificazione (CA) e caricarlo su AWS Identity and Access Management (IAM).

```
aws elbv2 create-listener \
    --load-balancer-arn arn:aws:elasticloadbalancing:us-
west-2:123456789012:loadbalancer/app/my-load-balancer/50dc6c495c0c9188 \
    --protocol HTTPS \
    --port 443 \
    --certificates CertificateArn=arn:aws:acm:us-
west-2:123456789012:certificate/3dcb0a41-bd72-4774-9ad9-756919c40557 \
    --ssl-policy ELBSecurityPolicy-2016-08 \
    --default-actions Type=forward,TargetGroupArn=arn:aws:elasticloadbalancing:us-
west-2:123456789012:targetgroup/my-targets/73e2d6bc24d8a067
```

Per ulteriori informazioni, consulta Aggiunta di un listener HTTPS nella Guida per l'utente di Application Load Balancer.

Esempio 3: come creare un listener TCP

L'esempio `create-listener` seguente crea un listener TCP per il Network Load Balancer specificato che inoltra le richieste al gruppo di destinazione specificato.

```
aws elbv2 create-listener \
    --load-balancer-arn arn:aws:elasticloadbalancing:us-
west-2:123456789012:loadbalancer/net/my-network-load-balancer/5d1b75f4f1cee11e \
    --protocol TCP \
```

```
    --port 80 \
    --default-actions Type=forward,TargetGroupArn=arn:aws:elasticloadbalancing:us-
west-2:123456789012:targetgroup/my-tcp-targets/b6bba954d1361c78
```

Per ulteriori informazioni, consulta Tutorial: Create a Network Load Balancer usando la AWS CLI nella User Guide for Network Load Balancer.

Esempio 4: come creare un listener TLS

L'esempio `create-listener` seguente crea un listener TLS per il Network Load Balancer specificato che inoltra le richieste al gruppo di destinazione specificato. È necessario specificare un certificato SSL per un listener TLS.

```
aws elbv2 create-listener \
    --load-balancer-arn arn:aws:elasticloadbalancing:us-
west-2:123456789012:loadbalancer/app/my-load-balancer/50dc6c495c0c9188 \
    --protocol TLS \
    --port 443 \
    --certificates CertificateArn=arn:aws:acm:us-
west-2:123456789012:certificate/3dcb0a41-bd72-4774-9ad9-756919c40557 \
    --ssl-policy ELBSecurityPolicy-2016-08 \
    --default-actions Type=forward,TargetGroupArn=arn:aws:elasticloadbalancing:us-
west-2:123456789012:targetgroup/my-targets/73e2d6bc24d8a067
```

Per ulteriori informazioni, consulta Listener TLS per il Network Load Balancer nella Guida per l'utente di Network Load Balancer.

Esempio 5: come creare un listener UDP

L'esempio `create-listener` seguente crea un listener UDP per il Network Load Balancer specificato che inoltra le richieste al gruppo di destinazione specificato.

```
aws elbv2 create-listener \
    --load-balancer-arn arn:aws:elasticloadbalancing:us-
west-2:123456789012:loadbalancer/net/my-network-load-balancer/5d1b75f4f1cee11e \
    --protocol UDP \
    --port 53 \
    --default-actions Type=forward,TargetGroupArn=arn:aws:elasticloadbalancing:us-
west-2:123456789012:targetgroup/my-tcp-targets/b6bba954d1361c78
```

Per ulteriori informazioni, consulta Tutorial: Create a Network Load Balancer usando la AWS CLI nella User Guide for Network Load Balancer.

Esempio 6: come creare un listener per il gateway e l'inoltro specificati

L'esempio `create-listener` seguente crea un listener per il Gateway Load Balancer specificato che inoltra le richieste al gruppo target specificato.

```
aws elbv2 create-listener \
    --load-balancer-arn arn:aws:elasticloadbalancing:us-
east-1:850631746142:loadbalancer/gwy/my-gateway-load-balancer/e0f9b3d5c7f7d3d6 \
    --default-actions Type=forward,TargetGroupArn=arn:aws:elasticloadbalancing:us-
east-1:850631746142:targetgroup/my-glb-targets/007ca469fae3bb1615
```

Output:

```
{
    "Listeners": [
        {
            "ListenerArn": "arn:aws:elasticloadbalancing:us-
east-1:850631746142:listener/gwy/my-agw-lb-example2/e0f9b3d5c7f7d3d6/
afc127db15f925de",
            "LoadBalancerArn": "arn:aws:elasticloadbalancing:us-
east-1:850631746142:loadbalancer/gwy/my-agw-lb-example2/e0f9b3d5c7f7d3d6",
            "DefaultActions": [
                {
                    "Type": "forward",
                    "TargetGroupArn": "arn:aws:elasticloadbalancing:us-
east-1:850631746142:targetgroup/test-tg-agw-2/007ca469fae3bb1615",
                    "ForwardConfig": {
                        "TargetGroups": [
                            {
                                "TargetGroupArn": "arn:aws:elasticloadbalancing:us-
east-1:850631746142:targetgroup/test-tg-agw-2/007ca469fae3bb1615"
                            }
                        ]
                    }
                }
            ]
        }
    ]
}
```

Per ulteriori informazioni, consulta Guida introduttiva a Gateway Load Balancers using the AWS CLI nella User Guide for Gateway Load Balancers.

- Per i dettagli sull'API, consulta Command Reference. CreateListenerAWS CLI

`create-load-balancer`

Il seguente esempio di codice mostra come utilizzare`create-load-balancer`.

AWS CLI

Esempio 1: come creare un bilanciatore del carico con connessione a Internet

L'esempio `create-load-balancer` seguente crea un Application Load Balancer con connessione a Internet e abilita le zone di disponibilità per le sottoreti specificate.

```
aws elbv2 create-load-balancer \
    --name my-load-balancer \
    --subnets subnet-b7d581c0 subnet-8360a9e7
```

Output:

```
{
    "LoadBalancers": [
        {
            "Type": "application",
            "Scheme": "internet-facing",
            "IpAddressType": "ipv4",
            "VpcId": "vpc-3ac0fb5f",
            "AvailabilityZones": [
                {
                    "ZoneName": "us-west-2a",
                    "SubnetId": "subnet-8360a9e7"
                },
                {
                    "ZoneName": "us-west-2b",
                    "SubnetId": "subnet-b7d581c0"
                }
            ],
            "CreatedTime": "2017-08-25T21:26:12.920Z",
            "CanonicalHostedZoneId": "Z2P70J7EXAMPLE",
            "DNSName": "my-load-balancer-424835706.us-west-2.elb.amazonaws.com",
            "SecurityGroups": [
                "sg-5943793c"
            ],
```

```
            "LoadBalancerName": "my-load-balancer",
            "State": {
                "Code": "provisioning"
            },
            "LoadBalancerArn": "arn:aws:elasticloadbalancing:us-
west-2:123456789012:loadbalancer/app/my-load-balancer/50dc6c495c0c9188"
        }
    ]
}
```

Per ulteriori informazioni, consulta Tutorial: Create an Application Load Balancer usando la AWS CLI nella User Guide for Application Load Balancers.

Esempio 2: come creare un bilanciatore del carico interno

L'esempio `create-load-balancer` seguente crea un Application Load Balancer interno e abilita le zone di disponibilità per le sottoreti specificate.

```
aws elbv2 create-load-balancer \
    --name my-internal-load-balancer \
    --scheme internal \
    --subnets subnet-b7d581c0 subnet-8360a9e7
```

Output:

```
{
    "LoadBalancers": [
        {
            "Type": "application",
            "Scheme": "internal",
            "IpAddressType": "ipv4",
            "VpcId": "vpc-3ac0fb5f",
            "AvailabilityZones": [
                {
                    "ZoneName": "us-west-2a",
                    "SubnetId": "subnet-8360a9e7"
                },
                {
                    "ZoneName": "us-west-2b",
                    "SubnetId": "subnet-b7d581c0"
                }
            ],
            "CreatedTime": "2016-03-25T21:29:48.850Z",
```

```
            "CanonicalHostedZoneId": "Z2P70J7EXAMPLE",
            "DNSName": "internal-my-internal-load-balancer-1529930873.us-
west-2.elb.amazonaws.com",
            "SecurityGroups": [
                "sg-5943793c"
            ],
            "LoadBalancerName": "my-internal-load-balancer",
            "State": {
                "Code": "provisioning"
            },
            "LoadBalancerArn": "arn:aws:elasticloadbalancing:us-
west-2:123456789012:loadbalancer/app/my-internal-load-balancer/5b49b8d4303115c2"
        }
    ]
}
```

Per ulteriori informazioni, consulta Tutorial: Create an Application Load Balancer usando la AWS CLI nella User Guide for Application Load Balancers.

Esempio 3: come creare un Network Load Balancer

L'esempio `create-load-balancer` seguente crea un Network Load Balancer con connessione a Internet e abilita la zona di disponibilità per la sottorete specificata. Utilizza una mappatura delle sottoreti per associare l'indirizzo IP elastico specificato all'interfaccia di rete utilizzata dai nodi del bilanciatore del carico per la zona di disponibilità.

```
aws elbv2 create-load-balancer \
    --name my-network-load-balancer \
    --type network \
    --subnet-mappings SubnetId=subnet-b7d581c0,AllocationId=eipalloc-64d5890a
```

Output:

```
{
    "LoadBalancers": [
        {
            "Type": "network",
            "Scheme": "internet-facing",
            "IpAddressType": "ipv4",
            "VpcId": "vpc-3ac0fb5f",
            "AvailabilityZones": [
                {
```

```
                    "LoadBalancerAddresses": [
                        {
                            "IpAddress": "35.161.207.171",
                            "AllocationId": "eipalloc-64d5890a"
                        }
                    ],
                    "ZoneName": "us-west-2b",
                    "SubnetId": "subnet-5264e837"
                }
            ],
            "CreatedTime": "2017-10-15T22:41:25.657Z",
            "CanonicalHostedZoneId": "Z2P70J7EXAMPLE",
            "DNSName": "my-network-load-balancer-5d1b75f4f1cee11e.elb.us-
west-2.amazonaws.com",
            "LoadBalancerName": "my-network-load-balancer",
            "State": {
                "Code": "provisioning"
            },
            "LoadBalancerArn": "arn:aws:elasticloadbalancing:us-
west-2:123456789012:loadbalancer/net/my-network-load-balancer/5d1b75f4f1cee11e"
        }
    ]
}
```

Per ulteriori informazioni, consulta Tutorial: Create a Network Load Balancer usando la AWS CLI nella User Guide for Network Load Balancer.

Esempio 4: come creare un Gateway Load Balancer

L'esempio `create-load-balancer` seguente crea un Gateway Load Balancer e abilita le zone di disponibilità per le sottoreti specificate.

```
aws elbv2 create-load-balancer \
    --name my-gateway-load-balancer \
    --type gateway \
    --subnets subnet-dc83f691 subnet-a62583f9
```

Output:

```
{
    "LoadBalancers": [
        {
            "Type": "gateway",
```

```
            "VpcId": "vpc-838475fe",
            "AvailabilityZones": [
                {
                    "ZoneName": "us-east-1b",
                    "SubnetId": "subnet-a62583f9"
                },
            {
                    "ZoneName": "us-east-1a",
                    "SubnetId": "subnet-dc83f691"
                }
            ],
            "CreatedTime": "2021-07-14T19:33:43.324000+00:00",
            "LoadBalancerName": "my-gateway-load-balancer",
            "State": {
                "Code": "provisioning"
            },
            "LoadBalancerArn": "arn:aws:elasticloadbalancing:us-
east-1:850631746142:loadbalancer/gwy/my-gateway-load-balancer/dfbb5a7d32cdee79"
        }
    ]
}
```

Per ulteriori informazioni, consulta Guida introduttiva a Gateway Load Balancers using the AWS CLI nella User Guide for Gateway Load Balancers.

- Per i dettagli sull'API, consulta Command Reference. CreateLoadBalancerAWS CLI

`create-rule`

Il seguente esempio di codice mostra come utilizzare`create-rule`.

AWS CLI

Esempio 1: come creare una regola utilizzando una condizione di percorso e un'azione di inoltro

L'esempio `create-rule` seguente crea una regola che inoltra le richieste al gruppo di destinazione se l'URL contiene lo schema specificato.

```
aws elbv2 create-rule \
    --listener-arn arn:aws:elasticloadbalancing:us-west-2:123456789012:listener/app/
my-load-balancer/50dc6c495c0c9188/f2f7dc8efc522ab2 \
    --priority 5 \
    --conditions file://conditions-pattern.json
```

```
    --actions Type=forward,TargetGroupArn=arn:aws:elasticloadbalancing:us-
west-2:123456789012:targetgroup/my-targets/73e2d6bc24d8a067
```

Contenuto di `conditions-pattern.json`:

```
[
    {
        "Field": "path-pattern",
        "PathPatternConfig": {
            "Values": ["/images/*"]
        }
    }
]
```

Esempio 2: come creare una regola utilizzando una condizione host e una risposta fissa

L'esempio `create-rule` seguente crea una regola che fornisce una risposta fissa se il nome host nell'intestazione host corrisponde al nome host specificato.

```
aws elbv2 create-rule \
    --listener-arn arn:aws:elasticloadbalancing:us-west-2:123456789012:listener/app/
my-load-balancer/50dc6c495c0c9188/f2f7dc8efc522ab2 \
    --priority 10 \
    --conditions file://conditions-host.json \
    --actions file://actions-fixed-response.json
```

Contenuto di `conditions-host.json`

```
[
  {
      "Field": "host-header",
      "HostHeaderConfig": {
          "Values": ["*.example.com"]
      }
  }
]
```

Contenuto di `actions-fixed-response.json`

```
[
    {
```

```
        "Type": "fixed-response",
        "FixedResponseConfig": {
            "MessageBody": "Hello world",
            "StatusCode": "200",
            "ContentType": "text/plain"
        }
    }
]
```

Esempio 3: come creare una regola utilizzando una condizione di indirizzo IP di origine, un'azione di autenticazione e un'azione di inoltro

L'esempio `create-rule` seguente crea una regola che autentica l'utente se l'indirizzo IP di origine corrisponde all'indirizzo IP specificato e inoltra la richiesta al gruppo di destinazione specificato se l'autenticazione ha esito positivo.

```
aws elbv2 create-rule \
    --listener-arn arn:aws:elasticloadbalancing:us-west-2:123456789012:listener/app/
my-load-balancer/50dc6c495c0c9188/f2f7dc8efc522ab2 \
    --priority 20 \
    --conditions file://conditions-source-ip.json \
    --actions file://actions-authenticate.json
```

Contenuto di `conditions-source-ip.json`

```
[
    {
        "Field": "source-ip",
        "SourceIpConfig": {
            "Values": ["192.0.2.0/24", "198.51.100.10/32"]
        }
    }
]
```

Contenuto di `actions-authenticate.json`

```
[
    {
        "Type": "authenticate-oidc",
        "AuthenticateOidcConfig": {
            "Issuer": "https://idp-issuer.com",
            "AuthorizationEndpoint": "https://authorization-endpoint.com",
```

```
            "TokenEndpoint": "https://token-endpoint.com",
            "UserInfoEndpoint": "https://user-info-endpoint.com",
            "ClientId": "abcdefghijklmnopqrstuvwxyz123456789",
            "ClientSecret": "1234567890123456789012345678901234567890",
            "SessionCookieName": "my-cookie",
            "SessionTimeout": 3600,
            "Scope": "email",
            "AuthenticationRequestExtraParams": {
                "display": "page",
                "prompt": "login"
            },
            "OnUnauthenticatedRequest": "deny"
        },
        "Order": 1
    },
    {
        "Type": "forward",
        "TargetGroupArn": "arn:aws:elasticloadbalancing:us-
east-1:880185128111:targetgroup/cli-test/642a97ecb0e0f26b",
        "Order": 2
    }
]
```

- Per i dettagli sull'API, consulta CreateRule AWS CLICommand Reference.

`create-target-group`

Il seguente esempio di codice mostra come utilizzare`create-target-group`.

AWS CLI

Esempio 1: come creare un gruppo di destinazione per l'Application Load Balancer

L'esempio `create-target-group` seguente crea un gruppo di destinazione per un Application Load Balancer in cui si registrano le destinazioni per ID istanza (il tipo di destinazione è `instance`). Questo gruppo di destinazione utilizza il protocollo HTTP, la porta 80 e le impostazioni di controllo dell'integrità predefinite per un gruppo di destinazione HTTP.

```
aws elbv2 create-target-group \
    --name my-targets \
    --protocol HTTP \
    --port 80 \
```

```
    --target-type instance \
    --vpc-id vpc-3ac0fb5f
```

Output:

```
{
    "TargetGroups": [
        {
            "TargetGroupArn": "arn:aws:elasticloadbalancing:us-
west-2:123456789012:targetgroup/my-targets/73e2d6bc24d8a067",
            "TargetGroupName": "my-targets",
            "Protocol": "HTTP",
            "Port": 80,
            "VpcId": "vpc-3ac0fb5f",
            "HealthCheckProtocol": "HTTP",
            "HealthCheckPort": "traffic-port",
            "HealthCheckEnabled": true,
            "HealthCheckIntervalSeconds": 30,
            "HealthCheckTimeoutSeconds": 5,
            "HealthyThresholdCount": 5,
            "UnhealthyThresholdCount": 2,
            "HealthCheckPath": "/",
            "Matcher": {
                "HttpCode": "200"
            },
            "TargetType": "instance",
            "ProtocolVersion": "HTTP1",
            "IpAddressType": "ipv4"
        }
    ]
}
```

Per ulteriori informazioni, consulta Creazione di un gruppo di destinazione nella Guida per l'utente di Application Load Balancer.

Esempio 2: come creare un gruppo di destinazione per indirizzare il traffico da un Application Load Balancer a una funzione Lambda

L'esempio `create-target-group` seguente crea un gruppo di destinazione per un Application Load Balancer in cui la destinazione è una funzione Lambda (il tipo di destinazione è `lambda`). Per impostazione predefinita, i controlli dell'integrità sono disabilitati per il gruppo di destinazione corrente.

```
aws elbv2 create-target-group \
    --name my-lambda-target \
    --target-type lambda
```

Output:

```
{
    "TargetGroups": [
        {
            "TargetGroupArn": "arn:aws:elasticloadbalancing:us-
west-2:123456789012:targetgroup/my-lambda-target/a3003e085dbb8ddc",
            "TargetGroupName": "my-lambda-target",
            "HealthCheckEnabled": false,
            "HealthCheckIntervalSeconds": 35,
            "HealthCheckTimeoutSeconds": 30,
            "HealthyThresholdCount": 5,
            "UnhealthyThresholdCount": 2,
            "HealthCheckPath": "/",
            "Matcher": {
                "HttpCode": "200"
            },
            "TargetType": "lambda",
            "IpAddressType": "ipv4"
        }
    ]
}
```

Per ulteriori informazioni, consulta Funzioni Lambda come destinazioni nella Guida per l'utente di Application Load Balancer.

Esempio 3: come creare un gruppo di destinazione per il Network Load Balancer

L'esempio `create-target-group` seguente crea un gruppo di destinazione per un Network Load Balancer in cui si registrano le destinazioni per indirizzo IP (il tipo di destinazione è `ip`). Questo gruppo di destinazione utilizza il protocollo TCP, la porta 80 e le impostazioni di controllo dell'integrità predefinite per un gruppo di destinazione TCP.

```
aws elbv2 create-target-group \
    --name my-ip-targets \
    --protocol TCP \
    --port 80 \
    --target-type ip \
```

```
    --vpc-id vpc-3ac0fb5f
```

Output:

```
{
    "TargetGroups": [
        {
            "TargetGroupArn": "arn:aws:elasticloadbalancing:us-
west-2:123456789012:targetgroup/my-ip-targets/b6bba954d1361c78",
            "TargetGroupName": "my-ip-targets",
            "Protocol": "TCP",
            "Port": 80,
            "VpcId": "vpc-3ac0fb5f",
            "HealthCheckEnabled": true,
            "HealthCheckProtocol": "TCP",
            "HealthCheckPort": "traffic-port",
            "HealthCheckIntervalSeconds": 30,
            "HealthCheckTimeoutSeconds": 10,
            "HealthyThresholdCount": 5,
            "UnhealthyThresholdCount": 2,
            "TargetType": "ip",
            "IpAddressType": "ipv4"
        }
    ]
}
```

Per ulteriori informazioni, consulta Creazione di un gruppo di destinazione nella Guida per l'utente di Network Load Balancer.

Esempio 4: come creare un gruppo di destinazione per indirizzare il traffico da un Network Load Balancer a un Application Load Balancer

L'esempio `create-target-group` seguente crea un gruppo di destinazione per un Network Load Balancer in cui si registra un Application Load Balancer come destinazione (il tipo di destinazione è `alb`).

aws elbv2 create-target-group --name my-alb-target --protocol TCP --port 80 --target-type alb --vpc-id vpc-3ac0fb5f

Output:

```
{
```

```
    "TargetGroups": [
        {
            "TargetGroupArn": "arn:aws:elasticloadbalancing:us-
west-2:123456789012:targetgroup/my-alb-target/a3003e085dbb8ddc",
            "TargetGroupName": "my-alb-target",
            "Protocol": "TCP",
            "Port": 80,
            "VpcId": "vpc-838475fe",
            "HealthCheckProtocol": "HTTP",
            "HealthCheckPort": "traffic-port",
            "HealthCheckEnabled": true,
            "HealthCheckIntervalSeconds": 30,
            "HealthCheckTimeoutSeconds": 6,
            "HealthyThresholdCount": 5,
            "UnhealthyThresholdCount": 2,
            "HealthCheckPath": "/",
            "Matcher": {
                "HttpCode": "200-399"
            },
            "TargetType": "alb",
            "IpAddressType": "ipv4"
        }
    ]
}
```

Per ulteriori informazioni, consulta Creazione del gruppo di destinazione con Application Load Balancer come destinazione nella Guida per l'utente di Network Load Balancer.

Esempio 5: come creare un gruppo di destinazione per un Gateway Load Balancer

L'esempio `create-target-group` seguente crea un gruppo di destinazione per un Gateway Load Balancer in cui la destinazione è un'istanza e il protocollo del gruppo di destinazione è `GENEVE`.

```
aws elbv2 create-target-group \
    --name my-glb-targetgroup \
    --protocol GENEVE \
    --port 6081 \
    --target-type instance \
    --vpc-id vpc-838475fe
```

Output:

```
{
    "TargetGroups": [
        {
            "TargetGroupArn": "arn:aws:elasticloadbalancing:us-
west-2:123456789012:targetgroup/my-glb-targetgroup/00c3d57eacd6f40b6f",
            "TargetGroupName": "my-glb-targetgroup",
            "Protocol": "GENEVE",
            "Port": 6081,
            "VpcId": "vpc-838475fe",
            "HealthCheckProtocol": "TCP",
            "HealthCheckPort": "80",
            "HealthCheckEnabled": true,
            "HealthCheckIntervalSeconds": 10,
            "HealthCheckTimeoutSeconds": 5,
            "HealthyThresholdCount": 5,
            "UnhealthyThresholdCount": 2,
            "TargetType": "instance"
        }
    ]
}
```

Per ulteriori informazioni, consulta Create a target group < https://docs.aws.amazon.com/elasticloadbalancing/ latest/gateway/create -target-group.html>`__ nella Gateway Load Balancer User Guide.

- Per i CreateTargetGroupdettagli AWS CLI sull'API, consulta Command Reference.

`delete-listener`

Il seguente esempio di codice mostra come utilizzare`delete-listener`.

AWS CLI

Come eliminare un listener

L'esempio `delete-listener` seguente elimina il listener specificato.

```
aws elbv2 delete-listener \
    --listener-arn arn:aws:elasticloadbalancing:ua-west-2:123456789012:listener/app/
my-load-balancer/50dc6c495c0c9188/f2f7dc8efc522ab2
```

- Per i dettagli sull'API, consulta DeleteListener AWS CLICommand Reference.

`delete-load-balancer`

Il seguente esempio di codice mostra come utilizzare`delete-load-balancer`.

AWS CLI

Come eliminare un bilanciatore del carico

L'esempio `delete-load-balancer` seguente elimina il bilanciatore del carico specificato.

```
aws elbv2 delete-load-balancer \
    --load-balancer-arn arn:aws:elasticloadbalancing:us-
west-2:123456789012:loadbalancer/app/my-load-balancer/50dc6c495c0c9188
```

- Per i dettagli sull'API, consulta DeleteLoadBalancer AWS CLICommand Reference.

`delete-rule`

Il seguente esempio di codice mostra come utilizzare`delete-rule`.

AWS CLI

Come eliminare una regola

L'esempio `delete-rule` seguente elimina la regola specificata.

```
aws elbv2 delete-rule \
    --rule-arn arn:aws:elasticloadbalancing:us-west-2:123456789012:listener-rule/
app/my-load-balancer/50dc6c495c0c9188/f2f7dc8efc522ab2/1291d13826f405c3
```

- Per i dettagli sull'API, consulta DeleteRule AWS CLICommand Reference.

`delete-target-group`

Il seguente esempio di codice mostra come utilizzare`delete-target-group`.

AWS CLI

Come eliminare un gruppo di destinazione

L'esempio `delete-target-group` seguente elimina il gruppo di destinazione specificato.

```
aws elbv2 delete-target-group \
```

```
    --target-group-arn arn:aws:elasticloadbalancing:us-
west-2:123456789012:targetgroup/my-targets/73e2d6bc24d8a067
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Eliminare un bilanciatore del carico nella Guida per l'utente di Application Load Balancer.

- Per i dettagli sull'API, consulta DeleteTargetGroup AWS CLICommand Reference.

`deregister-targets`

Il seguente esempio di codice mostra come utilizzare`deregister-targets`.

AWS CLI

Esempio 1: come annullare la registrazione di una destinazione da un gruppo di destinazione

L'esempio `deregister-targets` seguente rimuove l'istanza specificata dal gruppo di destinazione specificato.

```
aws elbv2 deregister-targets \
    --target-group-arn arn:aws:elasticloadbalancing:us-
west-2:123456789012:targetgroup/my-targets/73e2d6bc24d8a067 \
    --targets Id=i-1234567890abcdef0
```

Esempio 2: come annullare la registrazione di una destinazione registrata utilizzando gli override delle porte

L'esempio `deregister-targets` seguente rimuove un'istanza da un gruppo di destinazione registrato mediante gli override delle porte.

```
aws elbv2 deregister-targets \
    --target-group-arn arn:aws:elasticloadbalancing:us-
west-2:123456789012:targetgroup/my-internal-targets/3bb63f11dfb0faf9 \
    --targets Id=i-1234567890abcdef0,Port=80 Id=i-1234567890abcdef0,Port=766
```

- Per i dettagli sull'API, consulta DeregisterTargets AWS CLICommand Reference.

`describe-account-limits`

Il seguente esempio di codice mostra come utilizzare`describe-account-limits`.

AWS CLI

Come descrivere i limiti di Elastic Load Balancing

L'`describe-account-limits`esempio seguente mostra i limiti di Elastic Load Balancing per il tuo AWS account nella regione corrente.

```
aws elbv2 describe-account-limits
```

Output:

```
{
    "Limits": [
        {
            "Name": "target-groups",
            "Max": "3000"
        },
        {
            "Name": "targets-per-application-load-balancer",
            "Max": "1000"
        },
        {
            "Name": "listeners-per-application-load-balancer",
            "Max": "50"
        },
        {
            "Name": "rules-per-application-load-balancer",
            "Max": "100"
        },
        {
            "Name": "network-load-balancers",
            "Max": "50"
        },
        {
            "Name": "targets-per-network-load-balancer",
            "Max": "3000"
        },
        {
            "Name": "targets-per-availability-zone-per-network-load-balancer",
            "Max": "500"
        },
        {
            "Name": "listeners-per-network-load-balancer",
```

```
            "Max": "50"
        },
        {
            "Name": "condition-values-per-alb-rule",
            "Max": "5"
        },
        {
            "Name": "condition-wildcards-per-alb-rule",
            "Max": "5"
        },
        {
            "Name": "target-groups-per-application-load-balancer",
            "Max": "100"
        },
        {
            "Name": "target-groups-per-action-on-application-load-balancer",
            "Max": "5"
        },
        {
            "Name": "target-groups-per-action-on-network-load-balancer",
            "Max": "1"
        },
        {
            "Name": "certificates-per-application-load-balancer",
            "Max": "25"
        },
        {
            "Name": "certificates-per-network-load-balancer",
            "Max": "25"
        },
        {
            "Name": "targets-per-target-group",
            "Max": "1000"
        },
        {
            "Name": "target-id-registrations-per-application-load-balancer",
            "Max": "1000"
        },
        {
            "Name": "network-load-balancer-enis-per-vpc",
            "Max": "1200"
        },
        {
            "Name": "application-load-balancers",
```

```
            "Max": "50"
        },
        {
            "Name": "gateway-load-balancers",
            "Max": "100"
        },
        {
            "Name": "gateway-load-balancers-per-vpc",
            "Max": "100"
        },
        {
            "Name": "geneve-target-groups",
            "Max": "100"
        },
        {
            "Name": "targets-per-availability-zone-per-gateway-load-balancer",
            "Max": "300"
        }
    ]
}
```

Per ulteriori informazioni, consulta Quotas nella documentazione generale di riferimento di AWS .

- Per i dettagli sull'API, consulta DescribeAccountLimits AWS CLICommand Reference.

`describe-listener-certificates`

Il seguente esempio di codice mostra come utilizzare`describe-listener-certificates`.

AWS CLI

Come descrivere i certificati per un listener sicuro

Questo esempio descrive i certificati per il listener sicuro specificato.

Comando:

```
aws elbv2 describe-listener-certificates --listener-
arn arn:aws:elasticloadbalancing:us-west-2:123456789012:listener/app/my-load-
balancer/50dc6c495c0c9188/f2f7dc8efc522ab2
```

Output:

```
{
  "Certificates": [
      {
          "CertificateArn": "arn:aws:acm:us-
west-2:123456789012:certificate/5cc54884-f4a3-4072-80be-05b9ba72f705",
          "IsDefault": false
      },
      {
          "CertificateArn": "arn:aws:acm:us-
west-2:123456789012:certificate/3dcb0a41-bd72-4774-9ad9-756919c40557",
          "IsDefault": false
      },
      {
          "CertificateArn": "arn:aws:acm:us-west-2:123456789012:certificate/
fe59da96-6f58-4a22-8eed-6d0d50477e1d",
          "IsDefault": true
      }
  ]
}
```

- Per i dettagli sull'API, consulta DescribeListenerCertificates AWS CLICommand Reference.

`describe-listeners`

Il seguente esempio di codice mostra come utilizzare`describe-listeners`.

AWS CLI

Come descrivere un listener

Questo esempio descrive il listener specificato.

Comando:

```
aws elbv2 describe-listeners --listener-arns arn:aws:elasticloadbalancing:us-
west-2:123456789012:listener/app/my-load-balancer/50dc6c495c0c9188/f2f7dc8efc522ab2
```

Output:

```
{
  "Listeners": [
```

```
      {
          "Port": 80,
          "Protocol": "HTTP",
          "DefaultActions": [
              {
                  "TargetGroupArn": "arn:aws:elasticloadbalancing:us-
west-2:123456789012:targetgroup/my-targets/73e2d6bc24d8a067",
                  "Type": "forward"
              }
          ],
          "LoadBalancerArn": "arn:aws:elasticloadbalancing:us-
west-2:123456789012:loadbalancer/app/my-load-balancer/50dc6c495c0c9188",
          "ListenerArn": "arn:aws:elasticloadbalancing:us-
west-2:123456789012:listener/app/my-load-balancer/50dc6c495c0c9188/f2f7dc8efc522ab2"
      }
  ]
}
```

Come descrivere i listener per un bilanciatore del carico

Questo esempio descrive i listener per il bilanciatore del carico specificato.

Comando:

```
aws elbv2 describe-listeners --load-balancer-arn arn:aws:elasticloadbalancing:us-
west-2:123456789012:loadbalancer/app/my-load-balancer/50dc6c495c0c9188
```

Output:

```
{
  "Listeners": [
      {
          "Port": 443,
          "Protocol": "HTTPS",
          "DefaultActions": [
              {
                  "TargetGroupArn": "arn:aws:elasticloadbalancing:us-
west-2:123456789012:targetgroup/my-targets/73e2d6bc24d8a067",
                  "Type": "forward"
              }
          ],
          "SslPolicy": "ELBSecurityPolicy-2015-05",
```

```
          "Certificates": [
              {
                  "CertificateArn": "arn:aws:iam::123456789012:server-certificate/
my-server-cert"
              }
          ],
          "LoadBalancerArn": "arn:aws:elasticloadbalancing:us-
west-2:123456789012:loadbalancer/app/my-load-balancer/50dc6c495c0c9188",
          "ListenerArn": "arn:aws:elasticloadbalancing:us-
west-2:123456789012:listener/app/my-load-balancer/50dc6c495c0c9188/0467ef3c8400ae65"
      },
      {
          "Port": 80,
          "Protocol": "HTTP",
          "DefaultActions": [
              {
                  "TargetGroupArn": "arn:aws:elasticloadbalancing:us-
west-2:123456789012:targetgroup/my-targets/73e2d6bc24d8a067",
                  "Type": "forward"
              }
          ],
          "LoadBalancerArn": "arn:aws:elasticloadbalancing:us-
west-2:123456789012:loadbalancer/app/my-load-balancer/50dc6c495c0c9188",
          "ListenerArn": "arn:aws:elasticloadbalancing:us-
west-2:123456789012:listener/app/my-load-balancer/50dc6c495c0c9188/f2f7dc8efc522ab2"
      }
  ]
}
```

- Per i dettagli sull'API, consulta DescribeListeners AWS CLICommand Reference.

`describe-load-balancer-attributes`

Il seguente esempio di codice mostra come utilizzare`describe-load-balancer-attributes`.

AWS CLI

Come descrivere gli attributi del bilanciatore del carico

L'esempio `describe-load-balancer-attributes` seguente visualizza gli attributi del bilanciatore del carico specificato.

```
aws elbv2 describe-load-balancer-attributes \
```

```
    --load-balancer-arn arn:aws:elasticloadbalancing:us-
west-2:123456789012:loadbalancer/app/my-load-balancer/50dc6c495c0c9188
```

L'output dell'esempio seguente mostra gli attributi di un Application Load Balancer.

```
{
    "Attributes": [
        {
            "Value": "false",
            "Key": "access_logs.s3.enabled"
        },
        {
            "Value": "",
            "Key": "access_logs.s3.bucket"
        },
        {
            "Value": "",
            "Key": "access_logs.s3.prefix"
        },
        {
            "Value": "60",
            "Key": "idle_timeout.timeout_seconds"
        },
        {
            "Value": "false",
            "Key": "deletion_protection.enabled"
        },
        {
            "Value": "true",
            "Key": "routing.http2.enabled"
        }
    ]
}
```

L'output dell'esempio seguente include gli attributi di un Network Load Balancer.

```
{
    "Attributes": [
        {
            "Value": "false",
            "Key": "access_logs.s3.enabled"
        },
        {
```

```
            "Value": "",
            "Key": "access_logs.s3.bucket"
        },
        {
            "Value": "",
            "Key": "access_logs.s3.prefix"
        },
        {
            "Value": "false",
            "Key": "deletion_protection.enabled"
        },
        {
            "Value": "false",
            "Key": "load_balancing.cross_zone.enabled"
        }
    ]
}
```

- Per i dettagli sull'API, consulta DescribeLoadBalancerAttributes AWS CLICommand Reference.

`describe-load-balancers`

Il seguente esempio di codice mostra come utilizzare`describe-load-balancers`.

AWS CLI

Come descrivere un bilanciatore del carico

Questo esempio descrive il bilanciatore del carico specificato.

Comando:

```
aws elbv2 describe-load-balancers --load-balancer-
arns arn:aws:elasticloadbalancing:us-west-2:123456789012:loadbalancer/app/my-load-
balancer/50dc6c495c0c9188
```

Output:

```
{
  "LoadBalancers": [
      {
          "Type": "application",
```

```
            "Scheme": "internet-facing",
            "IpAddressType": "ipv4",
            "VpcId": "vpc-3ac0fb5f",
            "AvailabilityZones": [
                {
                    "ZoneName": "us-west-2a",
                    "SubnetId": "subnet-8360a9e7"
                },
                {
                    "ZoneName": "us-west-2b",
                    "SubnetId": "subnet-b7d581c0"
                }
            ],
            "CreatedTime": "2016-03-25T21:26:12.920Z",
            "CanonicalHostedZoneId": "Z2P70J7EXAMPLE",
            "DNSName": "my-load-balancer-424835706.us-west-2.elb.amazonaws.com",
            "SecurityGroups": [
                "sg-5943793c"
            ],
            "LoadBalancerName": "my-load-balancer",
            "State": {
                "Code": "active"
            },
            "LoadBalancerArn": "arn:aws:elasticloadbalancing:us-
west-2:123456789012:loadbalancer/app/my-load-balancer/50dc6c495c0c9188"
        }
    ]
}
```

Come descrivere tutti i bilanciatori del carico

Questo esempio descrive tutti i bilanciatori del carico.

Comando:

```
aws elbv2 describe-load-balancers
```

- Per i dettagli sull'API, consulta DescribeLoadBalancers AWS CLICommand Reference.

`describe-rules`

Il seguente esempio di codice mostra come utilizzare`describe-rules`.

AWS CLI

Esempio 1: come descrivere una regola

L'esempio `describe-rules` seguente visualizza i dettagli della regola specificata.

```
aws elbv2 describe-rules \
    --rule-arns arn:aws:elasticloadbalancing:us-west-2:123456789012:listener-rule/
app/my-load-balancer/50dc6c495c0c9188/f2f7dc8efc522ab2/9683b2d02a6cabee
```

Esempio 2: come descrivere le regole per un listener

L'esempio `describe-rules` seguente visualizza i dettagli delle regole per il listener specificato. L'output include la regola predefinita e tutte le altre regole aggiunte.

```
aws elbv2 describe-rules \
    --listener-arn arn:aws:elasticloadbalancing:us-west-2:123456789012:listener/app/
my-load-balancer/50dc6c495c0c9188/f2f7dc8efc522ab2
```

- Per i dettagli sull'API, consulta DescribeRules AWS CLICommand Reference.

`describe-ssl-policies`

Il seguente esempio di codice mostra come utilizzare`describe-ssl-policies`.

AWS CLI

Esempio 1: come elencare le policy utilizzate per la negoziazione SSL in base al tipo di bilanciatore del carico

L'esempio `describe-ssl-policies` seguente visualizza i nomi delle policy che è possibile utilizzare per la negoziazione SSL con un Application Load Balancer. L'esempio utilizza il parametro `--query` per visualizzare solo i nomi delle policy.

```
aws elbv2 describe-ssl-policies \
    --load-balancer-type application \
    --query SslPolicies[*].Name
```

Output:

```
[
    "ELBSecurityPolicy-2016-08",
    "ELBSecurityPolicy-TLS13-1-2-2021-06",
    "ELBSecurityPolicy-TLS13-1-2-Res-2021-06",
    "ELBSecurityPolicy-TLS13-1-2-Ext1-2021-06",
    "ELBSecurityPolicy-TLS13-1-2-Ext2-2021-06",
    "ELBSecurityPolicy-TLS13-1-1-2021-06",
    "ELBSecurityPolicy-TLS13-1-0-2021-06",
    "ELBSecurityPolicy-TLS13-1-3-2021-06",
    "ELBSecurityPolicy-TLS-1-2-2017-01",
    "ELBSecurityPolicy-TLS-1-1-2017-01",
    "ELBSecurityPolicy-TLS-1-2-Ext-2018-06",
    "ELBSecurityPolicy-FS-2018-06",
    "ELBSecurityPolicy-2015-05",
    "ELBSecurityPolicy-TLS-1-0-2015-04",
    "ELBSecurityPolicy-FS-1-2-Res-2019-08",
    "ELBSecurityPolicy-FS-1-1-2019-08",
    "ELBSecurityPolicy-FS-1-2-2019-08",
    "ELBSecurityPolicy-FS-1-2-Res-2020-10"
]
```

Esempio 2: come elencare le policy che supportano un protocollo specifico

L'esempio `describe-ssl-policies` seguente visualizza i nomi delle policy che supportano il protocollo TLS 1.3. L'esempio utilizza il parametro `--query` per visualizzare solo i nomi delle policy.

```
aws elbv2 describe-ssl-policies \
    --load-balancer-type application \
    --query SslPolicies[?contains(SslProtocols,'TLSv1.3')].Name
```

Output:

```
[
    "ELBSecurityPolicy-TLS13-1-2-2021-06",
    "ELBSecurityPolicy-TLS13-1-2-Res-2021-06",
    "ELBSecurityPolicy-TLS13-1-2-Ext1-2021-06",
    "ELBSecurityPolicy-TLS13-1-2-Ext2-2021-06",
    "ELBSecurityPolicy-TLS13-1-1-2021-06",
    "ELBSecurityPolicy-TLS13-1-0-2021-06",
    "ELBSecurityPolicy-TLS13-1-3-2021-06"
]
```

Esempio 3: come visualizzare i tipi di crittografia di una policy

L'esempio `describe-ssl-policies` seguente visualizza i nomi delle crittografie per la policy specificata. L'esempio utilizza il parametro `--query` per visualizzare solo i nomi delle crittografie. La prima crittografia dell'elenco ha priorità 1 e le crittografie rimanenti sono in ordine di priorità.

```
aws elbv2 describe-ssl-policies \
    --names ELBSecurityPolicy-TLS13-1-2-2021-06 \
    --query SslPolicies[*].Ciphers[*].Name
```

Output:

```
[
    "TLS_AES_128_GCM_SHA256",
    "TLS_AES_256_GCM_SHA384",
    "TLS_CHACHA20_POLY1305_SHA256",
    "ECDHE-ECDSA-AES128-GCM-SHA256",
    "ECDHE-RSA-AES128-GCM-SHA256",
    "ECDHE-ECDSA-AES128-SHA256",
    "ECDHE-RSA-AES128-SHA256",
    "ECDHE-ECDSA-AES256-GCM-SHA384",
    "ECDHE-RSA-AES256-GCM-SHA384",
    "ECDHE-ECDSA-AES256-SHA384",
    "ECDHE-RSA-AES256-SHA384"
]
```

Per ulteriori informazioni, consulta Policy di sicurezza nella Guida per l'utente di Application Load Balancer.

- Per i dettagli sull'API, consulta DescribeSslPolicies AWS CLICommand Reference.

`describe-tags`

Il seguente esempio di codice mostra come utilizzare`describe-tags`.

AWS CLI

Come descrivere i tag assegnati a un bilanciatore del carico

Questo esempio descrive i tag assegnati al bilanciatore del carico specificato.

Comando:

```
aws elbv2 describe-tags --resource-arns arn:aws:elasticloadbalancing:us-
west-2:123456789012:loadbalancer/app/my-load-balancer/50dc6c495c0c9188
```

Output:

```
{
  "TagDescriptions": [
      {
          "ResourceArn": "arn:aws:elasticloadbalancing:us-
west-2:123456789012:loadbalancer/app/my-load-balancer/50dc6c495c0c9188",
          "Tags": [
              {
                  "Value": "lima",
                  "Key": "project"
              },
              {
                  "Value": "digital-media",
                  "Key": "department"
              }
          ]
      }
  ]
}
```

- Per i dettagli sull'API, consulta DescribeTags AWS CLICommand Reference.

`describe-target-group-attributes`

Il seguente esempio di codice mostra come utilizzare`describe-target-group-attributes`.

AWS CLI

Come descrivere gli attributi del gruppo di destinazione

L'esempio `describe-target-group-attributes` seguente visualizza gli attributi del gruppo di destinazione specificato.

```
aws elbv2 describe-target-group-attributes \
    --target-group-arn arn:aws:elasticloadbalancing:us-
west-2:123456789012:targetgroup/my-targets/73e2d6bc24d8a067
```

L'output include gli attributi se il protocollo è HTTP o HTTPS e il tipo di destinazione è `instance` o `ip`.

```
{
    "Attributes": [
        {
            "Value": "false",
            "Key": "stickiness.enabled"
        },
        {
            "Value": "300",
            "Key": "deregistration_delay.timeout_seconds"
        },
        {
            "Value": "lb_cookie",
            "Key": "stickiness.type"
        },
        {
            "Value": "86400",
            "Key": "stickiness.lb_cookie.duration_seconds"
        },
        {
            "Value": "0",
            "Key": "slow_start.duration_seconds"
        }
    ]
}
```

Il seguente output include gli attributi se il protocollo è HTTP o HTTPS e il tipo di destinazione è `lambda`.

```
{
    "Attributes": [
        {
            "Value": "false",
            "Key": "lambda.multi_value_headers.enabled"
        }
    ]
}
```

Il seguente output include gli attributi se il protocollo è TCP, TLS, UDP o TCP_UDP.

```
{
    "Attributes": [
        {
            "Value": "false",
            "Key": "proxy_protocol_v2.enabled"
        },
        {
            "Value": "300",
            "Key": "deregistration_delay.timeout_seconds"
        }
    ]
}
```

- Per i dettagli sull'API, consulta DescribeTargetGroupAttributes AWS CLICommand Reference.

`describe-target-groups`

Il seguente esempio di codice mostra come utilizzare`describe-target-groups`.

AWS CLI

Esempio 1: come descrivere un gruppo di destinazione

L'esempio `describe-target-groups` seguente visualizza i dettagli del gruppo di destinazione specificato.

```
aws elbv2 describe-target-groups \
    --target-group-arns arn:aws:elasticloadbalancing:us-
west-2:123456789012:targetgroup/my-targets/73e2d6bc24d8a067
```

Output:

```
{
    "TargetGroups": [
        {
            "TargetGroupArn": "arn:aws:elasticloadbalancing:us-
west-2:123456789012:targetgroup/my-targets/73e2d6bc24d8a067",
            "TargetGroupName": "my-targets",
            "Protocol": "HTTP",
            "Port": 80,
            "VpcId": "vpc-3ac0fb5f",
            "HealthCheckProtocol": "HTTP",
```

```
            "HealthCheckPort": "traffic-port",
            "HealthCheckEnabled": true,
            "HealthCheckIntervalSeconds": 30,
            "HealthCheckTimeoutSeconds": 5,
            "HealthyThresholdCount": 5,
            "UnhealthyThresholdCount": 2,
            "HealthCheckPath": "/",
            "Matcher": {
                "HttpCode": "200"
            },
            "LoadBalancerArns": [
                "arn:aws:elasticloadbalancing:us-west-2:123456789012:loadbalancer/
app/my-load-balancer/50dc6c495c0c9188"
            ],
            "TargetType": "instance",
            "ProtocolVersion": "HTTP1",
            "IpAddressType": "ipv4"
        }
    ]
}
```

Esempio 2: come descrivere tutti i gruppi di destinazione per un bilanciatore del carico

L'esempio `describe-target-groups` seguente visualizza i dettagli di tutti i gruppi di destinazione per il bilanciatore del carico specificato. L'esempio utilizza il parametro `--query` per visualizzare solo i nomi dei gruppi di destinazione.

```
aws elbv2 describe-target-groups \
    --load-balancer-arn arn:aws:elasticloadbalancing:us-
west-2:123456789012:loadbalancer/app/my-load-balancer/50dc6c495c0c9188 \
    --query TargetGroups[*].TargetGroupName
```

Output:

```
[
    "my-instance-targets",
    "my-ip-targets",
    "my-lambda-target"
]
```

Per ulteriori informazioni, consulta Gruppi di destinazione nella Guida per l'utente di Application Load Balancer.

- Per i dettagli sull'API, consulta DescribeTargetGroups AWS CLICommand Reference.

`describe-target-health`

Il seguente esempio di codice mostra come utilizzare`describe-target-health`.

AWS CLI

Esempio 1: come descrivere l'integrità delle destinazioni per un gruppo di destinazione

L'esempio `describe-target-health` seguente visualizza i dettagli dell'integrità delle destinazioni del gruppo di destinazione specificato. Queste destinazioni sono integre.

```
aws elbv2 describe-target-health \
    --target-group-arn arn:aws:elasticloadbalancing:us-
west-2:123456789012:targetgroup/my-targets/73e2d6bc24d8a067
```

Output:

```
{
    "TargetHealthDescriptions": [
        {
            "HealthCheckPort": "80",
            "Target": {
                "Id": "i-ceddcd4d",
                "Port": 80
            },
            "TargetHealth": {
                "State": "healthy"
            }
        },
        {
            "HealthCheckPort": "80",
            "Target": {
                "Id": "i-0f76fade",
                "Port": 80
            },
            "TargetHealth": {
                "State": "healthy"
            }
        }
    ]
```

```
}
```

Esempio 2: come descrivere l'integrità di una destinazione

L'esempio `describe-target-health` seguente visualizza i dettagli relativi all'integrità della destinazione specificata. La destinazione è integra.

```
aws elbv2 describe-target-health \
    --targets Id=i-0f76fade,Port=80 \
    --target-group-arn arn:aws:elasticloadbalancing:us-
west-2:123456789012:targetgroup/my-targets/73e2d6bc24d8a067
```

Output:

```
{
    "TargetHealthDescriptions": [
        {
            "HealthCheckPort": "80",
            "Target": {
                "Id": "i-0f76fade",
                "Port": 80
            },
            "TargetHealth": {
                "State": "healthy"
            }
        }
    ]
}
```

L'output di esempio seguente si riferisce a una destinazione il cui gruppo di destinazione non è specificato in un'azione per un listener. Questa destinazione non può ricevere traffico dal bilanciatore del carico.

```
{
    "TargetHealthDescriptions": [
    {
        "HealthCheckPort": "80",
        "Target": {
            "Id": "i-0f76fade",
            "Port": 80
        },
            "TargetHealth": {
```

```
                "State": "unused",
                "Reason": "Target.NotInUse",
                "Description": "Target group is not configured to receive traffic
 from the load balancer"
            }
        }
    ]
}
```

L'output di esempio seguente si riferisce a una destinazione il cui gruppo di destinazione è stato specificato in un'azione per un listener. La destinazione è ancora in fase di registrazione.

```
{
    "TargetHealthDescriptions": [
        {
            "HealthCheckPort": "80",
            "Target": {
                "Id": "i-0f76fade",
                "Port": 80
            },
            "TargetHealth": {
                "State": "initial",
                "Reason": "Elb.RegistrationInProgress",
                "Description": "Target registration is in progress"
            }
        }
    ]
}
```

Il seguente output di esempio fa riferimento a una destinazione integra.

```
{
    "TargetHealthDescriptions": [
        {
            "HealthCheckPort": "80",
            "Target": {
                "Id": "i-0f76fade",
                "Port": 80
            },
            "TargetHealth": {
                "State": "unhealthy",
                "Reason": "Target.Timeout",
                "Description": "Connection to target timed out"
```

```
            }
        }
    ]
}
```

L'output di esempio seguente si riferisce a una destinazione che è una funzione Lambda e i controlli dell'integrità sono disabilitati.

```
{
    "TargetHealthDescriptions": [
        {
            "Target": {
                "Id": "arn:aws:lambda:us-west-2:123456789012:function:my-function",
                "AvailabilityZone": "all",
            },
            "TargetHealth": {
                "State": "unavailable",
                "Reason": "Target.HealthCheckDisabled",
                "Description": "Health checks are not enabled for this target"
            }
        }
    ]
}
```

- Per i dettagli sull'API, consulta DescribeTargetHealth AWS CLICommand Reference.

`modify-listener`

Il seguente esempio di codice mostra come utilizzare`modify-listener`.

AWS CLI

Esempio 1: come modificare l'azione predefinita in un'azione di inoltro

L'esempio `modify-listener` seguente modifica l'azione predefinita impostandola su un'azione `forward` per il listener specificato.

```
aws elbv2 modify-listener \
    --listener-arn arn:aws:elasticloadbalancing:us-west-2:123456789012:listener/app/my-load-balancer/50dc6c495c0c9188/f2f7dc8efc522ab2 \
    --default-actions Type=forward,TargetGroupArn=arn:aws:elasticloadbalancing:us-west-2:123456789012:targetgroup/my-new-targets/2453ed029918f21f
```

Output:

```
{
    "Listeners": [
        {
            "ListenerArn": "arn:aws:elasticloadbalancing:us-
west-2:123456789012:listener/app/my-load-balancer/50dc6c495c0c9188/
f2f7dc8efc522ab2",
            "LoadBalancerArn": "arn:aws:elasticloadbalancing:us-
west-2:123456789012:loadbalancer/app/my-load-balancer/50dc6c495c0c9188",
            "Protocol": "HTTP",
            "Port": 80,
            "DefaultActions": [
                {
                    "Type": "forward",
                    "TargetGroupArn": "arn:aws:elasticloadbalancing:us-
west-2:123456789012:targetgroup/my-new-targets/2453ed029918f21f"
                }
            ]
        }
    ]
}
```

Esempio 2: come modificare l'azione predefinita in un'azione di reindirizzamento

L'esempio `modify-listener` seguente modifica l'azione predefinita impostandola su un'azione `redirect` per il listener specificato.

```
aws elbv2 modify-listener \
    --listener-arn arn:aws:elasticloadbalancing:us-west-2:123456789012:listener/app/
my-load-balancer/50dc6c495c0c9188/f2f7dc8efc522ab2 \
    --default-actions Type=redirect,
 RedirectConfig='{Protocol=HTTPS,StatusCode=HTTP_302}'
```

Output:

```
{
    "Listeners": [
        {
            "ListenerArn": "arn:aws:elasticloadbalancing:us-
west-2:123456789012:listener/app/my-load-balancer/50dc6c495c0c9188/
f2f7dc8efc522ab2",
```

```
            "LoadBalancerArn": "arn:aws:elasticloadbalancing:us-
west-2:123456789012:loadbalancer/app/my-load-balancer/50dc6c495c0c9188",
            "Protocol": "HTTP",
            "Port": 80,
            "DefaultActions": [
                {
                    "Type": "redirect",
                    "RedirectConfig": {
                        "Protocol": "HTTPS",
                        "Port": "#{port}",
                        "Host": "#{host}",
                        "Path": "/#{path}",
                        "Query": "#{query}",
                        "StatusCode": "HTTP_302",
                    }
                }
            ]
        }
    ]
}
```

Esempio 3: come modificare il certificato del server

L'esempio `modify-listener` seguente modifica il certificato del server per il listener HTTPS specificato.

```
aws elbv2 modify-listener \
    --listener-arn arn:aws:elasticloadbalancing:us-west-2:123456789012:listener/app/
my-load-balancer/50dc6c495c0c9188/0467ef3c8400ae65 \
    --certificates CertificateArn=arn:aws:iam::123456789012:server-certificate/my-
new-server-cert
```

Output:

```
{
    "Listeners": [
        {
            "ListenerArn": "arn:aws:elasticloadbalancing:us-
west-2:123456789012:listener/app/my-load-
balancer/50dc6c495c0c9188/0467ef3c8400ae65",
            "LoadBalancerArn": "arn:aws:elasticloadbalancing:us-
west-2:123456789012:loadbalancer/app/my-load-balancer/50dc6c495c0c9188",
            "Protocol": "HTTPS",
```

```
            "Port": 443,
            "DefaultActions": [
                {
                    "Type": "forward",
                    "TargetGroupArn": "arn:aws:elasticloadbalancing:us-
west-2:123456789012:targetgroup/my-targets/73e2d6bc24d8a067"
                }
            ],
            "SslPolicy": "ELBSecurityPolicy-2015-05",
            "Certificates": [
                {
                    "CertificateArn": "arn:aws:iam::123456789012:server-certificate/
my-new-server-cert"
                }
            ],
        }
    ]
}
```

Per ulteriori informazioni, consulta Regole dei listener nella Guida per l'utente di Application Load Balancer.

- Per i dettagli sull'API, consulta ModifyListener AWS CLICommand Reference.

`modify-load-balancer-attributes`

Il seguente esempio di codice mostra come utilizzare`modify-load-balancer-attributes`.

AWS CLI

Come abilitare la protezione dall'eliminazione

Questo esempio abilita la protezione dall'eliminazione per il bilanciatore del carico specificato.

Comando:

```
aws elbv2 modify-load-balancer-attributes --load-balancer-
arn arn:aws:elasticloadbalancing:us-west-2:123456789012:loadbalancer/app/my-load-
balancer/50dc6c495c0c9188 --attributes Key=deletion_protection.enabled,Value=true
```

Output:

```
{
```

```
  "Attributes": [
      {
          "Value": "true",
          "Key": "deletion_protection.enabled"
      },
      {
          "Value": "false",
          "Key": "access_logs.s3.enabled"
      },
      {
          "Value": "60",
          "Key": "idle_timeout.timeout_seconds"
      },
      {
          "Value": "",
          "Key": "access_logs.s3.prefix"
      },
      {
          "Value": "",
          "Key": "access_logs.s3.bucket"
      }
  ]
}
```

Come modificare il timeout di inattività

Questo esempio modifica il valore del timeout di inattività per il bilanciatore del carico specificato.

Comando:

```
aws elbv2 modify-load-balancer-attributes --load-balancer-
arn arn:aws:elasticloadbalancing:us-west-2:123456789012:loadbalancer/app/my-load-
balancer/50dc6c495c0c9188 --attributes Key=idle_timeout.timeout_seconds,Value=30
```

Output:

```
{
  "Attributes": [
      {
          "Value": "30",
          "Key": "idle_timeout.timeout_seconds"
      },
      {
```

```
            "Value": "false",
            "Key": "access_logs.s3.enabled"
        },
        {
            "Value": "",
            "Key": "access_logs.s3.prefix"
        },
        {
            "Value": "true",
            "Key": "deletion_protection.enabled"
        },
        {
            "Value": "",
            "Key": "access_logs.s3.bucket"
        }
    ]
}
```

Come abilitare il log degli accessi

Questo esempio abilita il log degli accessi per il bilanciatore del carico specificato. Nota che il bucket S3 deve esistere nella stessa Regione del bilanciatore del carico e deve includere una policy associata che conceda l'accesso al servizio Elastic Load Balancing.

Comando:

```
aws elbv2 modify-load-balancer-attributes --load-balancer-
arn arn:aws:elasticloadbalancing:us-west-2:123456789012:loadbalancer/app/my-load-
balancer/50dc6c495c0c9188 --
attributes Key=access_logs.s3.enabled,Value=true Key=access_logs.s3.bucket,Value=my-
loadbalancer-logs Key=access_logs.s3.prefix,Value=myapp
```

Output:

```
{
    "Attributes": [
        {
            "Value": "true",
            "Key": "access_logs.s3.enabled"
        },
        {
            "Value": "my-load-balancer-logs",
```

```
            "Key": "access_logs.s3.bucket"
        },
        {
            "Value": "myapp",
            "Key": "access_logs.s3.prefix"
        },
        {
            "Value": "60",
            "Key": "idle_timeout.timeout_seconds"
        },
        {
            "Value": "false",
            "Key": "deletion_protection.enabled"
        }
    ]
}
```

- Per i dettagli sull'API, consulta ModifyLoadBalancerAttributes AWS CLICommand Reference.

`modify-rule`

Il seguente esempio di codice mostra come utilizzare`modify-rule`.

AWS CLI

Come modificare una regola

L'esempio `modify-rule` seguente aggiorna le azioni e le condizioni per la regola specificata.

```
aws elbv2 modify-rule \
  --actions Type=forward,TargetGroupArn=arn:aws:elasticloadbalancing:us-
west-2:123456789012:targetgroup/my-targets/73e2d6bc24d8a067 \
  --conditions Field=path-pattern,Values='/images/*'
  --rule-arn arn:aws:elasticloadbalancing:us-west-2:123456789012:listener-rule/app/
my-load-balancer/50dc6c495c0c9188/f2f7dc8efc522ab2/9683b2d02a6cabee
```

Output:

```
{
    "Rules": [
        {
            "Priority": "10",
```

```
            "Conditions": [
                {
                    "Field": "path-pattern",
                    "Values": [
                        "/images/*"
                    ]
                }
            ],
            "RuleArn": "arn:aws:elasticloadbalancing:us-
west-2:123456789012:listener-rule/app/my-load-balancer/50dc6c495c0c9188/
f2f7dc8efc522ab2/9683b2d02a6cabee",
            "IsDefault": false,
            "Actions": [
                {
                    "TargetGroupArn": "arn:aws:elasticloadbalancing:us-
west-2:123456789012:targetgroup/my-targets/73e2d6bc24d8a067",
                    "Type": "forward"
                }
            ]
        }
    ]
}
```

- Per i dettagli sull'API, consulta ModifyRule AWS CLICommand Reference.

`modify-target-group-attributes`

Il seguente esempio di codice mostra come utilizzare`modify-target-group-attributes`.

AWS CLI

Come modificare il timeout del ritardo di annullamento della registrazione

Questo esempio imposta il timeout del ritardo dell'annullamento della registrazione sul valore specificato per il gruppo di destinazione specificato.

Comando:

```
aws elbv2 modify-target-group-attributes --target-group-
arn arn:aws:elasticloadbalancing:us-west-2:123456789012:targetgroup/my-
targets/73e2d6bc24d8a067 --
attributes Key=deregistration_delay.timeout_seconds,Value=600
```

Output:

```
{
  "Attributes": [
      {
          "Value": "false",
          "Key": "stickiness.enabled"
      },
      {
          "Value": "600",
          "Key": "deregistration_delay.timeout_seconds"
      },
      {
          "Value": "lb_cookie",
          "Key": "stickiness.type"
      },
      {
          "Value": "86400",
          "Key": "stickiness.lb_cookie.duration_seconds"
      }
  ]
}
```

- Per i dettagli sull'API, consulta ModifyTargetGroupAttributes AWS CLICommand Reference.

`modify-target-group`

Il seguente esempio di codice mostra come utilizzare`modify-target-group`.

AWS CLI

Come modificare le impostazioni di controllo dell'integrità per un gruppo di destinazione

L'esempio `modify-target-group` seguente modifica la configurazione dei controlli dell'integrità utilizzati per valutare lo stato delle destinazioni per il gruppo di destinazione specificato. A causa del modo in cui la CLI analizza le virgole, è necessario racchiudere l'intervallo dell'opzione `--matcher` tra virgolette singole anziché virgolette doppie.

```
aws elbv2 modify-target-group \
    --target-group-arn arn:aws:elasticloadbalancing:us-
west-2:123456789012:targetgroup/my-https-targets/2453ed029918f21f \
    --health-check-protocol HTTPS \
```

```
    --health-check-port 443 \
    --matcher HttpCode='200,299'
```

Output:

```
{
    "TargetGroups": [
        {
            "TargetGroupArn": "arn:aws:elasticloadbalancing:us-
west-2:123456789012:targetgroup/my-https-targets/2453ed029918f21f",
            "TargetGroupName": "my-https-targets",
            "Protocol": "HTTPS",
            "Port": 443,
            "VpcId": "vpc-3ac0fb5f",
            "HealthCheckProtocol": "HTTPS",
            "HealthCheckPort": "443",
            "HealthCheckEnabled": true,
            "HealthCheckIntervalSeconds": 30,
            "HealthCheckTimeoutSeconds": 5,
            "HealthyThresholdCount": 5,
            "UnhealthyThresholdCount": 2,
            "Matcher": {
                "HttpCode": "200,299"
            },
            "LoadBalancerArns": [
                "arn:aws:elasticloadbalancing:us-west-2:123456789012:loadbalancer/
app/my-load-balancer/50dc6c495c0c9188"
            ],
            "TargetType": "instance",
            "ProtocolVersion": "HTTP1",
            "IpAddressType": "ipv4"
        }
    ]
}
```

Per ulteriori informazioni, consulta Gruppi di destinazione nella Guida per l'utente di Application Load Balancer.

- Per i dettagli sull'API, consulta ModifyTargetGroup AWS CLICommand Reference.

`register-targets`

Il seguente esempio di codice mostra come utilizzare`register-targets`.

AWS CLI

Esempio 1: come registrare le destinazioni con un gruppo di destinazione in base all'ID dell'istanza

L'esempio `register-targets` seguente registra le istanze specificate con un gruppo di destinazione. Il tipo di destinazione del gruppo di destinazione deve essere `instance`.

```
aws elbv2 register-targets \
    --target-group-arn arn:aws:elasticloadbalancing:us-
west-2:123456789012:targetgroup/my-targets/73e2d6bc24d8a067 \
    --targets Id=i-1234567890abcdef0 Id=i-0abcdef1234567890
```

Esempio 2: come registrare le destinazioni con un gruppo di destinazione utilizzando gli override delle porte

L'esempio `register-targets` seguente registra l'istanza specificata con un gruppo di destinazione utilizzando più porte. Ciò consente di registrare i container sulla stessa istanza delle destinazioni nel gruppo di destinazione.

```
aws elbv2 register-targets \
    --target-group-arn arn:aws:elasticloadbalancing:us-
west-2:123456789012:targetgroup/my-internal-targets/3bb63f11dfb0faf9 \
    --targets Id=i-0598c7d356eba48d7,Port=80 Id=i-0598c7d356eba48d7,Port=766
```

Esempio 3: come registrare le destinazioni con un gruppo di destinazione in base all'indirizzo IP

L'esempio `register-targets` seguente registra gli indirizzi IP specificati con un gruppo di destinazione. Il tipo di destinazione del gruppo di destinazione deve essere `ip`.

```
aws elbv2 register-targets \
    --target-group-arn arn:aws:elasticloadbalancing:us-
west-2:123456789012:targetgroup/my-tcp-ip-targets/8518e899d173178f \
    --targets Id=10.0.1.15 Id=10.0.1.23
```

Esempio 4: come registrare una funzione Lambda come destinazione

L'esempio `register-targets` seguente registra gli indirizzi IP specificati con un gruppo di destinazione. Il tipo di destinazione del gruppo di destinazione deve essere `lambda`. Per invocare la funzione Lambda, devi concedere l'autorizzazione Elastic Load Balancing.

```
aws elbv2 register-targets \
    --target-group-arn arn:aws:elasticloadbalancing:us-
west-2:123456789012:targetgroup/my-tcp-ip-targets/8518e899d173178f \
    --targets Id=arn:aws:lambda:us-west-2:123456789012:function:my-function
```

- Per i dettagli sull'API, consulta RegisterTargets AWS CLICommand Reference.

`remove-listener-certificates`

Il seguente esempio di codice mostra come utilizzare`remove-listener-certificates`.

AWS CLI

Come rimuovere un certificato da un listener sicuro

Questo esempio rimuove il certificato specificato dal listener sicuro specificato.

Comando:

```
aws elbv2 remove-listener-certificates --listener-
arn arn:aws:elasticloadbalancing:us-west-2:123456789012:listener/
app/my-load-balancer/50dc6c495c0c9188/f2f7dc8efc522ab2 --
certificates CertificateArn=arn:aws:acm:us-west-2:123456789012:certificate/5cc54884-
f4a3-4072-80be-05b9ba72f705
```

- Per i dettagli sull'API, consulta RemoveListenerCertificates AWS CLICommand Reference.

`remove-tags`

Il seguente esempio di codice mostra come utilizzare`remove-tags`.

AWS CLI

Come rimuovere i tag da un bilanciatore del carico

L'esempio `remove-tags` seguente rimuove i tag `project` e `department` dal bilanciatore del carico specificato.

```
aws elbv2 remove-tags \
    --resource-arns arn:aws:elasticloadbalancing:us-
west-2:123456789012:loadbalancer/app/my-load-balancer/50dc6c495c0c9188 \
```

```
    --tag-keys project department
```

- Per i dettagli sull'API, consulta RemoveTags AWS CLICommand Reference.

`set-ip-address-type`

Il seguente esempio di codice mostra come utilizzare`set-ip-address-type`.

AWS CLI

Come impostare il tipo di indirizzo di un bilanciatore del carico

Questo esempio imposta il tipo di indirizzo del bilanciatore del carico specificato su `dualstack`. Le sottoreti di bilanciamento del carico devono avere blocchi CIDR associati IPv6 .

Comando:

```
aws elbv2 set-ip-address-type --load-balancer-arn arn:aws:elasticloadbalancing:us-west-2:123456789012:loadbalancer/app/my-load-balancer/50dc6c495c0c9188 --ip-address-type dualstack
```

Output:

```
{
    "IpAddressType": "dualstack"
}
```

- Per i dettagli sull'API, vedere SetIpAddressTypein Command Reference.AWS CLI

`set-rule-priorities`

Il seguente esempio di codice mostra come utilizzare`set-rule-priorities`.

AWS CLI

Come impostare la priorità della regola

Questo esempio imposta la priorità della regola specificata.

Comando:

```
aws elbv2 set-rule-priorities --rule-
priorities RuleArn=arn:aws:elasticloadbalancing:us-west-2:123456789012:listener-
rule/app/my-load-balancer/50dc6c495c0c9188/
f2f7dc8efc522ab2/1291d13826f405c3,Priority=5
```

Output:

```
{
  "Rules": [
      {
          "Priority": "5",
          "Conditions": [
              {
                  "Field": "path-pattern",
                  "Values": [
                      "/img/*"
                  ]
              }
          ],
          "RuleArn": "arn:aws:elasticloadbalancing:us-west-2:123456789012:listener-
rule/app/my-load-balancer/50dc6c495c0c9188/f2f7dc8efc522ab2/1291d13826f405c3",
          "IsDefault": false,
          "Actions": [
              {
                  "TargetGroupArn": "arn:aws:elasticloadbalancing:us-
west-2:123456789012:targetgroup/my-targets/73e2d6bc24d8a067",
                  "Type": "forward"
              }
          ]
      }
  ]
}
```

- Per i dettagli sull'API, consulta SetRulePriorities AWS CLICommand Reference.

`set-security-groups`

Il seguente esempio di codice mostra come utilizzare`set-security-groups`.

AWS CLI

Come associare un gruppo di sicurezza a un bilanciatore del carico

Questo esempio associa il gruppo di sicurezza specificato al bilanciatore del carico specificato.

Comando:

```
aws elbv2 set-security-groups --load-balancer-arn arn:aws:elasticloadbalancing:us-west-2:123456789012:loadbalancer/app/my-load-balancer/50dc6c495c0c9188 --security-groups sg-5943793c
```

Output:

```
{
  "SecurityGroupIds": [
      "sg-5943793c"
  ]
}
```

- Per i dettagli sull'API, consulta SetSecurityGroups AWS CLICommand Reference.

`set-subnets`

Il seguente esempio di codice mostra come utilizzare`set-subnets`.

AWS CLI

Come abilitare le zone di disponibilità per un bilanciatore del carico

Questo esempio abilita la zona di disponibilità per la sottorete specificata per il bilanciatore del carico specificato.

Comando:

```
aws elbv2 set-subnets --load-balancer-arn arn:aws:elasticloadbalancing:us-west-2:123456789012:loadbalancer/app/my-load-balancer/50dc6c495c0c9188 --subnets subnet-8360a9e7 subnet-b7d581c0
```

Output:

```
{
  "AvailabilityZones": [
      {
          "SubnetId": "subnet-8360a9e7",
```

```
            "ZoneName": "us-west-2a"
        },
        {
            "SubnetId": "subnet-b7d581c0",
            "ZoneName": "us-west-2b"
        }
    ]
}
```

- Per i dettagli sull'API, consulta SetSubnets AWS CLICommand Reference.

ElastiCache esempi utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with ElastiCache.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`add-tags-to-resource`

Il seguente esempio di codice mostra come utilizzare`add-tags-to-resource`.

AWS CLI

Come aggiungere tag a una risorsa

L'esempio `add-tags-to-resource` seguente aggiunge un massimo di 10 tag, coppie chiave-valore, a una risorsa cluster o snapshot.

```
aws elasticache add-tags-to-resource \
```

```
    --resource-name "arn:aws:elasticache:us-east-1:1234567890:cluster:my-mem-
cluster" \
    --tags '{"20150202":15, "ElastiCache":"Service"}'
```

Output:

```
{
    "TagList": [
        {
            "Value": "20150202",
            "Key": "APIVersion"
        },
        {
            "Value": "ElastiCache",
            "Key": "Service"
        }
    ]
}
```

Per ulteriori informazioni, consulta Monitoraggio dei costi con i tag di allocazione dei costi nella Guida per l'utente di ElastiCache.

- Per i dettagli sull'API, consulta AddTagsToResource AWS CLICommand Reference.

`authorize-cache-security-group-ingress`

Il seguente esempio di codice mostra come utilizzare`authorize-cache-security-group-ingress`.

AWS CLI

Come autorizzare l'ingresso del gruppo di sicurezza cache

L'esempio `authorize-cache-security-group-ingress` seguente consente l'ingresso di rete a un gruppo di sicurezza cache.

```
aws elasticache authorize-cache-security-group-ingress \
     --cache-security-group-name  "my-sec-grp" \
     --ec2-security-group-name "my-ec2-sec-grp" \
     --ec2-security-group-owner-id "1234567890"
```

Il comando non produce alcun output.

Per ulteriori informazioni, consulta Self-Service Updates in Amazon ElastiCache nella Elasticache User Guide.

- Per i dettagli sull'API, consulta Command AuthorizeCacheSecurityGroupIngressReference AWS CLI .

`batch-apply-update-action`

Il seguente esempio di codice mostra come utilizzare`batch-apply-update-action`.

AWS CLI

Come applicare un aggiornamento del servizio

L'esempio `batch-apply-update-action` seguente applica un aggiornamento del servizio a un cluster Redis.

```
aws elasticache batch-apply-update-action \
    --service-update-name elc-xxxxx406-xxx \
    --replication-group-ids test-cluster
```

Output:

```
{
    "ProcessedUpdateActions": [
        {
            "ReplicationGroupId": "pat-cluster",
            "ServiceUpdateName": "elc-xxxxx406-xxx",
            "UpdateActionStatus": "waiting-to-start"
        }
    ],
    "UnprocessedUpdateActions": []
}
```

Per ulteriori informazioni, consulta Self-Service Updates in Amazon ElastiCache nella Elasticache User Guide.

- Per i dettagli sull'API, consulta Command BatchApplyUpdateActionReference AWS CLI .

`batch-stop-update-action`

Il seguente esempio di codice mostra come utilizzare`batch-stop-update-action`.

AWS CLI

Come arrestare un aggiornamento del servizio

L'esempio `batch-stop-update-action` seguente applica un aggiornamento del servizio a un cluster Redis.

```
aws elasticache batch-stop-update-action \
    --service-update-name elc-xxxxx406-xxx \
    --replication-group-ids test-cluster
```

Output:

```
{
    "ProcessedUpdateActions": [
        {
            "ReplicationGroupId": "pat-cluster",
            "ServiceUpdateName": "elc-xxxxx406-xxx",
            "UpdateActionStatus": "stopping"
        }
    ],
    "UnprocessedUpdateActions": []
}
```

Per ulteriori informazioni, consulta Self-Service Updates in Amazon ElastiCache nella Elasticache User Guide.

- Per i dettagli sull'API, consulta Command BatchStopUpdateActionReference AWS CLI .

`copy-snapshot`

Il seguente esempio di codice mostra come utilizzare`copy-snapshot`.

AWS CLI

Come copiare una snapshot

L'esempio `copy-snapshot` seguente crea una copia di uno snapshot esistente.

```
aws elasticache copy-snapshot \
    --source-snapshot-name "my-snapshot" \
```

```
    --target-snapshot-name "my-snapshot-copy"
```

Output:

```
{
    "Snapshot":{
        "Engine": "redis",
        "CacheParameterGroupName": "default.redis3.2",
        "VpcId": "vpc-3820329f3",
        "CacheClusterId": "my-redis4",
        "SnapshotRetentionLimit": 7,
        "NumCacheNodes": 1,
        "SnapshotName": "my-snapshot-copy",
        "CacheClusterCreateTime": "2016-12-21T22:24:04.955Z",
        "AutoMinorVersionUpgrade": true,
        "PreferredAvailabilityZone": "us-east-1c",
        "SnapshotStatus": "creating",
        "SnapshotSource": "manual",
        "SnapshotWindow": "07:00-08:00",
        "EngineVersion": "3.2.4",
        "NodeSnapshots": [
            {
                "CacheSize": "3 MB",
                "SnapshotCreateTime": "2016-12-28T07:00:52Z",
                "CacheNodeId": "0001",
                "CacheNodeCreateTime": "2016-12-21T22:24:04.955Z"
            }
        ],
        "CacheSubnetGroupName": "default",
        "Port": 6379,
        "PreferredMaintenanceWindow": "tue:09:30-tue:10:30",
        "CacheNodeType": "cache.m3.large"
    }
}
```

Per ulteriori informazioni, consulta Esportazione di un backup nella Guida per l'utente di ElastiCache.

- Per i dettagli sull'API, consulta CopySnapshot AWS CLICommand Reference.

`create-cache-cluster`

Il seguente esempio di codice mostra come utilizzare`create-cache-cluster`.

AWS CLI

Come creare un cluster di cache

L'esempio `create-cache-cluster` seguente crea un cluster di cache utilizzando il motore Redis.

```
aws elasticache create-cache-cluster \
    --cache-cluster-id "cluster-test" \
    --engine redis \
    --cache-node-type cache.m5.large \
    --num-cache-nodes 1
```

Output:

```
{
    "CacheCluster": {
        "CacheClusterId": "cluster-test",
        "ClientDownloadLandingPage": "https://console.aws.amazon.com/elasticache/
home#client-download:",
        "CacheNodeType": "cache.m5.large",
        "Engine": "redis",
        "EngineVersion": "5.0.5",
        "CacheClusterStatus": "creating",
        "NumCacheNodes": 1,
        "PreferredMaintenanceWindow": "sat:13:00-sat:14:00",
        "PendingModifiedValues": {},
        "CacheSecurityGroups": [],
        "CacheParameterGroup": {
            "CacheParameterGroupName": "default.redis5.0",
            "ParameterApplyStatus": "in-sync",
            "CacheNodeIdsToReboot": []
        },
        "CacheSubnetGroupName": "default",
        "AutoMinorVersionUpgrade": true,
        "SnapshotRetentionLimit": 0,
        "SnapshotWindow": "06:30-07:30",
        "TransitEncryptionEnabled": false,
        "AtRestEncryptionEnabled": false
    }
}
```

Per ulteriori informazioni, consulta Creazione di un cluster nella Guida per l'utente di ElastiCache.

- Per i dettagli sull'API, consulta CreateCacheCluster AWS CLICommand Reference.

`create-cache-parameter-group`

Il seguente esempio di codice mostra come utilizzare`create-cache-parameter-group`.

AWS CLI

Come creare un gruppo di parametri cache

L'`create-cache-parameter-group`esempio seguente crea un nuovo gruppo di parametri Amazon ElastiCache cache.

```
aws elasticache create-cache-parameter-group \
    --cache-parameter-group-family "redis5.0" \
    --cache-parameter-group-name "mygroup" \
    --description "mygroup"
```

Output:

```
{
    "CacheParameterGroup": {
        "CacheParameterGroupName": "mygroup",
        "CacheParameterGroupFamily": "redis5.0",
        "Description": "my group"
    }
}
```

Per ulteriori informazioni, consulta Creazione di un gruppo di parametri nella Guida per l'utente di ElastiCache.

- Per i dettagli sull'API, consulta CreateCacheParameterGroup AWS CLICommand Reference.

`create-cache-subnet-group`

Il seguente esempio di codice mostra come utilizzare`create-cache-subnet-group`.

AWS CLI

Come creare un gruppo di sottoreti della cache

L'esempio `create-cache-subnet-group` seguente crea un nuovo gruppo di sottoreti della cache.

```
aws elasticache create-cache-subnet-group \
    --cache-subnet-group-name "mygroup" \
    --cache-subnet-group-description "my subnet group" \
    --subnet-ids "subnet-xxxxec4f"
```

Output:

```
{
    "CacheSubnetGroup": {
        "CacheSubnetGroupName": "mygroup",
        "CacheSubnetGroupDescription": "my subnet group",
        "VpcId": "vpc-a3e97cdb",
        "Subnets": [
            {
                "SubnetIdentifier": "subnet-xxxxec4f",
                "SubnetAvailabilityZone": {
                    "Name": "us-west-2d"
                }
            }
        ]
    }
}
```

Per ulteriori informazioni, consulta Creazione di un gruppo di sottoreti della cache nella Guida per l'utente di ElastiCache.

- Per i dettagli sull'API, consulta CreateCacheSubnetGroup AWS CLICommand Reference.

`create-global-replication-group`

Il seguente esempio di codice mostra come utilizzare`create-global-replication-group`.

AWS CLI

Come creare un gruppo di replica globale

L'esempio `create-global-replication-group` seguente crea un'attività di replica.

```
aws elasticache create-global-replication-group \
```

```
    --global-replication-group-id-suffix my-global-replication-group \
    --primary-replication-group-id my-primary-cluster
```

Output:

```
{
    "GlobalReplicationGroup": {
        "GlobalReplicationGroupId": "sgaui-my-global-replication-group",
        "GlobalReplicationGroupDescription": " ",
        "Status": "creating",
        "CacheNodeType": "cache.r5.large",
        "Engine": "redis",
        "EngineVersion": "5.0.6",
        "Members": [
            {
                "ReplicationGroupId": "my-primary-cluster",
                "ReplicationGroupRegion": "us-west-2",
                "Role": "PRIMARY",
                "AutomaticFailover": "enabled",
                "Status": "associating"
            }
        ],
        "ClusterEnabled": true,
        "GlobalNodeGroups": [
            {
                "GlobalNodeGroupId": "sgaui-my-global-replication-group-0001",
                "Slots": "0-16383"
            }
        ],
        "AuthTokenEnabled": false,
        "TransitEncryptionEnabled": false,
        "AtRestEncryptionEnabled": false
    }
}
```

Per ulteriori informazioni, consulta Replica tra AWS regioni utilizzando Global Datastore nella Guida per l'utente di Elasticache.

- Per i dettagli sull'API, consulta Command Reference. CreateGlobalReplicationGroupAWS CLI

`create-replication-group`

Il seguente esempio di codice mostra come utilizzare`create-replication-group`.

AWS CLI

Come creare un gruppo di replica

L'esempio `create-replication-group` seguente crea un gruppo di replica Redis (modalità cluster disabilitata) o Redis (modalità cluster abilitata). Questa operazione è valida solo per Redis.

```
aws elasticache create-replication-group \
    --replication-group-id "mygroup" \
    --replication-group-description "my group" \
    --engine "redis" \
    --cache-node-type "cache.m5.large"
```

Output:

```
{
    "ReplicationGroup": {
        "ReplicationGroupId": "mygroup",
        "Description": "my group",
        "Status": "creating",
        "PendingModifiedValues": {},
        "MemberClusters": [
            "mygroup-001"
        ],
        "AutomaticFailover": "disabled",
        "SnapshotRetentionLimit": 0,
        "SnapshotWindow": "06:00-07:00",
        "ClusterEnabled": false,
        "CacheNodeType": "cache.m5.large",
        "TransitEncryptionEnabled": false,
        "AtRestEncryptionEnabled": false
    }
}
```

Per ulteriori informazioni, consulta Creazione di un gruppo di replica Redis nella Guida per l'utente di ElastiCache.

- Per i dettagli sull'API, consulta CreateReplicationGroup AWS CLICommand Reference.

`create-snapshot`

Il seguente esempio di codice mostra come utilizzare`create-snapshot`.

AWS CLI

Come creare uno snapshot

L'esempio `create-snapshot` seguente crea uno snapshot utilizzando il motore Redis.

```
aws elasticache create-snapshot \
    --snapshot-name mysnapshot \
    --cache-cluster-id cluster-test
```

Output:

```
{
    "Snapshot": {
        "SnapshotName": "mysnapshot",
        "CacheClusterId": "cluster-test",
        "SnapshotStatus": "creating",
        "SnapshotSource": "manual",
        "CacheNodeType": "cache.m5.large",
        "Engine": "redis",
        "EngineVersion": "5.0.5",
        "NumCacheNodes": 1,
        "PreferredAvailabilityZone": "us-west-2b",
        "CacheClusterCreateTime": "2020-03-19T03:12:01.483Z",
        "PreferredMaintenanceWindow": "sat:13:00-sat:14:00",
        "Port": 6379,
        "CacheParameterGroupName": "default.redis5.0",
        "CacheSubnetGroupName": "default",
        "VpcId": "vpc-a3e97cdb",
        "AutoMinorVersionUpgrade": true,
        "SnapshotRetentionLimit": 0,
        "SnapshotWindow": "06:30-07:30",
        "NodeSnapshots": [
            {
                "CacheNodeId": "0001",
                "CacheSize": "",
                "CacheNodeCreateTime": "2020-03-19T03:12:01.483Z"
            }
        ]
    }
}
```

Per ulteriori informazioni, consulta Backup and Restore ElastiCache for Redis nella Guida per l'utente di Elasticache.

- Per i dettagli sull'API, consulta Command CreateSnapshotReference AWS CLI .

`create-user-group`

Il seguente esempio di codice mostra come utilizzare`create-user-group`.

AWS CLI

Come creare un gruppo di utenti

L'esempio `create-user-group` seguente crea un'attività di replica.

```
aws elasticache create-user-group \
    --user-group-id myusergroup \
    --engine redis \
    --user-ids default
```

Output:

```
{
    "UserGroupId": "myusergroup",
    "Status": "creating",
    "Engine": "redis",
    "UserIds": [
        "default"
    ],
    "ReplicationGroups": [],
    "ARN": "arn:aws:elasticache:us-west-2:xxxxxxxxxx52:usergroup:myusergroup"
}
```

Per ulteriori informazioni, consulta Autenticazione degli utenti con il controllo degli accessi basato sui ruoli (RBAC) nella Guida per l'utente di ElastiCache.

- Per i dettagli sull'API, consulta CreateUserGroup AWS CLICommand Reference.

`create-user`

Il seguente esempio di codice mostra come utilizzare`create-user`.

AWS CLI

Come creare un utente

L'esempio `create-user` seguente crea un'attività di replica.

```
aws elasticache create-user \
    --user-id user1 \
    --user-name myUser \
    --passwords mYnuUzrpAxXw2rdzx \
    --engine redis \
    --access-string "on ~app::* -@all +@read"
```

Output:

```
{
    "UserId": "user2",
    "UserName": "myUser",
    "Status": "active",
    "Engine": "redis",
    "AccessString": "on ~app::* -@all +@read +@hash +@bitmap +@geo -setbit -bitfield
 -hset -hsetnx -hmset -hincrby -hincrbyfloat -hdel -bitop -geoadd -georadius -
georadiusbymember",
    "UserGroupIds": [],
    "Authentication": {
        "Type": "password",
        "PasswordCount": 1
    },
    "ARN": "arn:aws:elasticache:us-west-2:xxxxxxxxxx52:user:user2"
}
```

Per ulteriori informazioni, consulta Autenticazione degli utenti con il controllo degli accessi basato sui ruoli (RBAC) nella Guida per l'utente di ElastiCache.

- Per i dettagli sull'API, consulta CreateUser AWS CLICommand Reference.

`decrease-node-groups-in-global-replication-group`

Il seguente esempio di codice mostra come utilizzare`decrease-node-groups-in-global-replication-group`.

AWS CLI

Come ridurre il numero dei gruppi di nodi in un gruppo di replica globale

Il comando `decrease-node-groups-in-global-replication-group` seguente riduce il numero dei gruppi di nodi utilizzando il motore Redis.

```
aws elasticache decrease-node-groups-in-global-replication-group \
    --global-replication-group-id sgaui-test \
    --node-group-count 1 \
    --apply-immediately \
    --global-node-groups-to-retain sgaui-test-0003
```

Output:

```
{
    "GlobalReplicationGroup":
    {
        "GlobalReplicationGroupId": "sgaui-test",
        "GlobalReplicationGroupDescription": "test",
        "Status": "modifying",
        "CacheNodeType": "cache.r5.large",
        "Engine": "redis",
        "EngineVersion": "5.0.6",
        "Members": [
            {
                "ReplicationGroupId": "test-2",
                "ReplicationGroupRegion": "us-east-1",
                "Role": "SECONDARY",
                "AutomaticFailover": "enabled",
                "Status": "associated"
            },
            {
                "ReplicationGroupId": "test-1",
                "ReplicationGroupRegion": "us-west-2",
                "Role": "PRIMARY",
                "AutomaticFailover": "enabled",
                "Status": "associated"
            }
        ],
        "ClusterEnabled": true,
        "GlobalNodeGroups": [
            {
```

```
                "GlobalNodeGroupId": "sgaui-test-0001",
                "Slots": "0-449,1816-5461"
            },
            {
                "GlobalNodeGroupId": "sgaui-test-0002",
                "Slots": "6827-10922"
            },
            {
                "GlobalNodeGroupId": "sgaui-test-0003",
                "Slots": "10923-14052,15418-16383"
            },
            {
                "GlobalNodeGroupId": "sgaui-test-0004",
                "Slots": "450-1815,5462-6826,14053-15417"
            }
        ],
        "AuthTokenEnabled": false,
        "TransitEncryptionEnabled": false,
        "AtRestEncryptionEnabled": false
    }
}
```

Per ulteriori informazioni, consulta Replica tra AWS regioni utilizzando Global Datastore nella Guida per l'utente di Elasticache.

- Per i dettagli sull'API, consulta Command Reference. DecreaseNodeGroupsInGlobalReplicationGroupAWS CLI

`decrease-replica-count`

Il seguente esempio di codice mostra come utilizzare`decrease-replica-count`.

AWS CLI

Come ridurre il numero di repliche

L'esempio `decrease-replica-count` seguente riduce dinamicamente il numero di repliche in un gruppo di replica Redis (modalità cluster disabilitata) o il numero di nodi di replica in uno o più gruppi di nodi (shard) di un gruppo di replica Redis (modalità cluster abilitata). Questa operazione viene eseguita senza causare alcun tempo di inattività del cluster.

```
aws elasticache decrease-replica-count \
```

```
    --replication-group-id my-cluster \
    --apply-immediately  \
    --new-replica-count 2
```

Output:

```
{
    "ReplicationGroup": {
        "ReplicationGroupId": "my-cluster",
        "Description": " ",
        "Status": "modifying",
        "PendingModifiedValues": {},
        "MemberClusters": [
            "myrepliace",
            "my-cluster-001",
            "my-cluster-002",
            "my-cluster-003"
        ],
        "NodeGroups": [
            {
                "NodeGroupId": "0001",
                "Status": "modifying",
                "PrimaryEndpoint": {
                    "Address": "my-cluster.xxxxx.ng.0001.usw2.cache.amazonaws.com",
                    "Port": 6379
                },
                "ReaderEndpoint": {
                    "Address": "my-cluster-
ro.xxxxx.ng.0001.usw2.cache.amazonaws.com",
                    "Port": 6379
                },
                "NodeGroupMembers": [
                    {
                        "CacheClusterId": "myrepliace",
                        "CacheNodeId": "0001",
                        "ReadEndpoint": {
                            "Address":
 "myrepliace.xxxxx.0001.usw2.cache.amazonaws.com",
                            "Port": 6379
                        },
                        "PreferredAvailabilityZone": "us-west-2a",
                        "CurrentRole": "replica"
                    },
```

```
                    {
                        "CacheClusterId": "my-cluster-001",
                        "CacheNodeId": "0001",
                        "ReadEndpoint": {
                            "Address": "my-
cluster-001.xxxxx.0001.usw2.cache.amazonaws.com",
                            "Port": 6379
                        },
                        "PreferredAvailabilityZone": "us-west-2a",
                        "CurrentRole": "primary"
                    },
                    {
                        "CacheClusterId": "my-cluster-002",
                        "CacheNodeId": "0001",
                        "ReadEndpoint": {
                            "Address": "my-
cluster-002.xxxxx.0001.usw2.cache.amazonaws.com",
                            "Port": 6379
                        },
                        "PreferredAvailabilityZone": "us-west-2a",
                        "CurrentRole": "replica"
                    },
                    {
                        "CacheClusterId": "my-cluster-003",
                        "CacheNodeId": "0001",
                        "ReadEndpoint": {
                            "Address": "my-
cluster-003.xxxxx.0001.usw2.cache.amazonaws.com",
                            "Port": 6379
                        },
                        "PreferredAvailabilityZone": "us-west-2a",
                        "CurrentRole": "replica"
                    }
                ]
            }
        ],
        "AutomaticFailover": "disabled",
        "SnapshotRetentionLimit": 0,
        "SnapshotWindow": "07:30-08:30",
        "ClusterEnabled": false,
        "CacheNodeType": "cache.r5.xlarge",
        "TransitEncryptionEnabled": false,
        "AtRestEncryptionEnabled": false
    }
```

```
}
```

Per ulteriori informazioni, consulta Modifica del numero di repliche nella Guida per l'utente di ElastiCache.

- Per i dettagli sull'API, consulta DecreaseReplicaCount AWS CLICommand Reference.

`delete-cache-cluster`

Il seguente esempio di codice mostra come utilizzare`delete-cache-cluster`.

AWS CLI

Come eliminare un cluster di cache

L'esempio `delete-cache-cluster` seguente elimina il cluster DAX specificato. Il comando elimina tutti i nodi di cache associati, gli endpoint dei nodi e il cluster stesso. Quando ricevi una risposta corretta da questa operazione, Amazon inizia ElastiCache immediatamente a eliminare il cluster; non puoi annullare o annullare questa operazione.

Questa operazione non è valida per quanto segue:

Cluster Redis (modalità cluster abilitata); un cluster che costituisce l'ultima replica di lettura di un gruppo di replica; un gruppo di nodi (shard) con modalità Multi-AZ abilitata; un cluster di un gruppo di replica Redis (modalità cluster abilitata); un cluster con stato non disponibile

```
aws elasticache delete-cache-cluster \
    --cache-cluster-id "my-cluster-002"
```

Output:

```
{
    "CacheCluster": {
        "CacheClusterId": "my-cluster-002",
        "ClientDownloadLandingPage": "https://console.aws.amazon.com/elasticache/
home#client-download:",
        "CacheNodeType": "cache.r5.xlarge",
        "Engine": "redis",
        "EngineVersion": "5.0.5",
        "CacheClusterStatus": "deleting",
```

```
        "NumCacheNodes": 1,
        "PreferredAvailabilityZone": "us-west-2a",
        "CacheClusterCreateTime": "2019-11-26T03:35:04.546Z",
        "PreferredMaintenanceWindow": "mon:04:05-mon:05:05",
        "PendingModifiedValues": {},
        "NotificationConfiguration": {
            "TopicArn": "arn:aws:sns:us-west-x:xxxxxxx4152:My_Topic",
            "TopicStatus": "active"
        },
        "CacheSecurityGroups": [],
        "CacheParameterGroup": {
            "CacheParameterGroupName": "mygroup",
            "ParameterApplyStatus": "in-sync",
            "CacheNodeIdsToReboot": []
        },
        "CacheSubnetGroupName": "kxkxk",
        "AutoMinorVersionUpgrade": true,
        "SecurityGroups": [
            {
                "SecurityGroupId": "sg-xxxxxxxxxx9836",
                "Status": "active"
            },
            {
                "SecurityGroupId": "sg-xxxxxxxxxxxx7b",
                "Status": "active"
            }
        ],
        "ReplicationGroupId": "my-cluster",
        "SnapshotRetentionLimit": 0,
        "SnapshotWindow": "07:30-08:30",
        "TransitEncryptionEnabled": false,
        "AtRestEncryptionEnabled": false
    }
}
```

Per ulteriori informazioni, consulta Eliminazione di un cluster nella Guida per l'utente di ElastiCache.

- Per i dettagli sull'API, consulta AWS CLI Command DeleteCacheClusterReference.

`delete-cache-parameter-group`

Il seguente esempio di codice mostra come utilizzare`delete-cache-parameter-group`.

AWS CLI

Come eliminare un gruppo di parametri cache

L'esempio `delete-cache-parameter-group` seguente elimina il gruppo di parametri cache specificato. Un gruppo di parametri cache non può essere eliminato se è associato a cluster di cache.

```
aws elasticache delete-cache-parameter-group \
    --cache-parameter-group-name myparamgroup
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Eliminazione di un gruppo di parametri nella Guida per l'utente di ElastiCache.

- Per i dettagli sull'API, consulta DeleteCacheParameterGroup AWS CLICommand Reference.

`delete-cache-subnet-group`

Il seguente esempio di codice mostra come utilizzare`delete-cache-subnet-group`.

AWS CLI

Come eliminare un gruppo di sottoreti della cache

L'esempio `delete-cache-subnet-group` seguente elimina il gruppo di sottoreti della cache specificato. Un gruppo di sottoreti della cache non può essere eliminato se è associato a cluster di cache.

```
aws elasticache delete-cache-subnet-group \
    --cache-subnet-group-name "mygroup"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Eliminazione di un gruppo di sottoreti nella Guida per l'utente di ElastiCache.

- Per i dettagli sull'API, consulta DeleteCacheSubnetGroup AWS CLICommand Reference.

`delete-global-replication-group`

Il seguente esempio di codice mostra come utilizzare`delete-global-replication-group`.

AWS CLI

Come eliminare un gruppo di replica globale

L'esempio `delete-global-replication-group` seguente crea un'attività di replica.

```
aws elasticache delete-global-replication-group \
    --global-replication-group-id my-global-replication-group \
    --retain-primary-replication-group
```

Output:

```
{
    "GlobalReplicationGroup": {
        "GlobalReplicationGroupId": "sgaui-my-grg",
        "GlobalReplicationGroupDescription": "my-grg",
        "Status": "deleting",
        "CacheNodeType": "cache.r5.large",
        "Engine": "redis",
        "EngineVersion": "5.0.6",
        "Members": [
            {
                "ReplicationGroupId": "my-cluster-grg",
                "ReplicationGroupRegion": "us-west-2",
                "Role": "PRIMARY",
                "AutomaticFailover": "enabled",
                "Status": "associated"
            }
        ],
        "ClusterEnabled": false,
        "AuthTokenEnabled": false,
        "TransitEncryptionEnabled": false,
        "AtRestEncryptionEnabled": false
    }
}
```

Per ulteriori informazioni, consulta Replica tra AWS regioni utilizzando Global Datastore nella Guida per l'utente di Elasticache.

- Per i dettagli sull'API, consulta Command Reference. DeleteGlobalReplicationGroupAWS CLI

`delete-replication-group`

Il seguente esempio di codice mostra come utilizzare`delete-replication-group`.

AWS CLI

Come eliminare un gruppo di replica

L'esempio `delete-replication-group` seguente elimina un gruppo di replica esistente. Per impostazione predefinita, questa operazione elimina l'intero gruppo di replica, incluse tutte le primary/primaries repliche lette. Se il gruppo di replica ha un solo primario, è possibile eliminare facoltativamente solo le repliche di lettura, mantenendo la principale impostando =true. RetainPrimaryCluster

Quando ricevi una risposta corretta da questa operazione, Amazon inizia ElastiCache immediatamente a eliminare le risorse selezionate; non puoi annullare o annullare questa operazione. Operazione valida solo per Redis.

```
aws elasticache delete-replication-group \
    --replication-group-id "mygroup"
```

Output:

```
{
     "ReplicationGroup": {
         "ReplicationGroupId": "mygroup",
         "Description": "my group",
         "Status": "deleting",
         "PendingModifiedValues": {},
         "AutomaticFailover": "disabled",
         "SnapshotRetentionLimit": 0,
         "SnapshotWindow": "06:00-07:00",
         "TransitEncryptionEnabled": false,
         "AtRestEncryptionEnabled": false
     }
 }
```

- Per i dettagli sull'API, consulta AWS CLI Command DeleteReplicationGroupReference.

`delete-snapshot`

Il seguente esempio di codice mostra come utilizzare`delete-snapshot`.

AWS CLI

Come eliminare uno snapshot

L'esempio `delete-snapshot` seguente elimina uno snapshot utilizzando il motore Redis.

```
aws elasticache delete-snapshot \
    --snapshot-name mysnapshot
```

Output:

```
{
    "Snapshot": {
        "SnapshotName": "my-cluster-snapshot",
        "ReplicationGroupId": "mycluster",
        "ReplicationGroupDescription": "mycluster",
        "SnapshotStatus": "deleting",
        "SnapshotSource": "manual",
        "CacheNodeType": "cache.r5.xlarge",
        "Engine": "redis",
        "EngineVersion": "5.0.5",
        "PreferredMaintenanceWindow": "thu:12:00-thu:13:00",
        "TopicArn": "arn:aws:sns:us-west-2:xxxxxxxxxxxxx152:My_Topic",
        "Port": 6379,
        "CacheParameterGroupName": "default.redis5.0.cluster.on",
        "CacheSubnetGroupName": "default",
        "VpcId": "vpc-a3e97cdb",
        "AutoMinorVersionUpgrade": true,
        "SnapshotRetentionLimit": 1,
        "SnapshotWindow": "13:00-14:00",
        "NumNodeGroups": 4,
        "AutomaticFailover": "enabled",
        "NodeSnapshots": [
            {
                "CacheClusterId": "mycluster-0002-003",
                "NodeGroupId": "0002",
                "CacheNodeId": "0001",
                "CacheSize": "6 MB",
                "CacheNodeCreateTime": "2020-06-18T00:05:44.719000+00:00",
                "SnapshotCreateTime": "2020-06-25T20:34:30+00:00"
            },
            {
                "CacheClusterId": "mycluster-0003-003",
```

```
                "NodeGroupId": "0003",
                "CacheNodeId": "0001",
                "CacheSize": "6 MB",
                "CacheNodeCreateTime": "2019-12-05T19:13:15.912000+00:00",
                "SnapshotCreateTime": "2020-06-25T20:34:30+00:00"
            },
            {
                "CacheClusterId": "mycluster-0004-002",
                "NodeGroupId": "0004",
                "CacheNodeId": "0001",
                "CacheSize": "6 MB",
                "CacheNodeCreateTime": "2019-12-09T19:44:34.324000+00:00",
                "SnapshotCreateTime": "2020-06-25T20:34:30+00:00"
            },
            {
                "CacheClusterId": "mycluster-0005-003",
                "NodeGroupId": "0005",
                "CacheNodeId": "0001",
                "CacheSize": "6 MB",
                "CacheNodeCreateTime": "2020-06-18T00:05:44.775000+00:00",
                "SnapshotCreateTime": "2020-06-25T20:34:30+00:00"
            }
        ]
    }
}
```

Per ulteriori informazioni, consulta Backup and Restore ElastiCache for Redis nella Guida per l'utente di Elasticache.

- Per i dettagli sull'API, consulta Command DeleteSnapshotReference AWS CLI .

`delete-user-group`

Il seguente esempio di codice mostra come utilizzare`delete-user-group`.

AWS CLI

Come eliminare un gruppo di utenti

L'esempio `delete-user-group` seguente elimina un sink.

```
aws elasticache delete-user-group \
    --user-group-id myusergroup
```

Output:

```
{
    "UserGroupId": "myusergroup",
    "Status": "deleting",
    "Engine": "redis",
    "UserIds": [
        "default"
    ],
    "ReplicationGroups": [],
    "ARN": "arn:aws:elasticache:us-west-2:xxxxxxxxxx52:usergroup:myusergroup"
}
```

Per ulteriori informazioni, consulta Autenticazione degli utenti con il controllo degli accessi basato sui ruoli (RBAC) nella Guida per l'utente di ElastiCache.

- Per i dettagli sull'API, consulta DeleteUserGroup AWS CLICommand Reference.

`delete-user`

Il seguente esempio di codice mostra come utilizzare`delete-user`.

AWS CLI

Come eliminare un utente

L'esempio `delete-user` seguente elimina un sink.

```
aws elasticache delete-user \
    --user-id user2
```

Output:

```
{
    "UserId": "user1",
    "UserName": "myUser",
    "Status": "deleting",
    "Engine": "redis",
    "AccessString": "on ~* +@all",
    "UserGroupIds": [
        "myusergroup"
    ],
    "Authentication": {
```

```
        "Type": "password",
        "PasswordCount": 1
    },
    "ARN": "arn:aws:elasticache:us-west-2:xxxxxxxxxx52:user:user1"
}
```

Per ulteriori informazioni, consulta Autenticazione degli utenti con il controllo degli accessi basato sui ruoli (RBAC) nella Guida per l'utente di ElastiCache.

- Per i dettagli sull'API, consulta DeleteUser AWS CLICommand Reference.

`describe-cache-clusters`

Il seguente esempio di codice mostra come utilizzare`describe-cache-clusters`.

AWS CLI

Come descrivere un cluster di cache

L'esempio `describe-cache-clusters` seguente descrive un cluster di cache.

```
aws elasticache describe-cache-clusters
```

Output:

```
{
    "CacheClusters": [
    {
            "CacheClusterId": "my-cluster-003",
            "ClientDownloadLandingPage": "https://console.aws.amazon.com/
elasticache/home#client-download:",
            "CacheNodeType": "cache.r5.large",
            "Engine": "redis",
            "EngineVersion": "5.0.5",
            "CacheClusterStatus": "available",
            "NumCacheNodes": 1,
            "PreferredAvailabilityZone": "us-west-2a",
            "CacheClusterCreateTime": "2019-11-26T01:22:52.396Z",
            "PreferredMaintenanceWindow": "mon:17:30-mon:18:30",
            "PendingModifiedValues": {},
            "NotificationConfiguration": {
                "TopicArn": "arn:aws:sns:us-west-2:xxxxxxxxxxx152:My_Topic",
                "TopicStatus": "active"
```

```
            },
            "CacheSecurityGroups": [],
            "CacheParameterGroup": {
                "CacheParameterGroupName": "default.redis5.0",
                "ParameterApplyStatus": "in-sync",
                "CacheNodeIdsToReboot": []
            },
            "CacheSubnetGroupName": "kxkxk",
            "AutoMinorVersionUpgrade": true,
            "SecurityGroups": [
                {
                    "SecurityGroupId": "sg-xxxxxd7b",
                    "Status": "active"
                }
            ],
            "ReplicationGroupId": "my-cluster",
            "SnapshotRetentionLimit": 0,
            "SnapshotWindow": "06:30-07:30",
            "AuthTokenEnabled": false,
            "TransitEncryptionEnabled": false,
            "AtRestEncryptionEnabled": false,
            "ARN": "arn:aws:elasticache:us-west-2:xxxxxxxxxxx152:cluster:my-cache-
cluster",
            "ReplicationGroupLogDeliveryEnabled": false,
            "LogDeliveryConfigurations": [
                {
                    "LogType": "slow-log",
                    "DestinationType": "cloudwatch-logs",
                    "DestinationDetails": {
                        "CloudWatchLogsDetails": {
                            "LogGroup": "test-log"
                        }
                    },
                    "LogFormat": "text",
                    "Status": "active"
                }
            ]
        }
    ]
}
```

Per ulteriori informazioni, consulta Gestione dei cluster nella Guida per l'utente di ElastiCache.

- Per i dettagli sull'API, consulta DescribeCacheClusters AWS CLICommand Reference.

`describe-cache-engine-versions`

Il seguente esempio di codice mostra come utilizzare`describe-cache-engine-versions`.

AWS CLI

Come descrivere la versione di un motore di cache

L'esempio `describe-cache-engine-versions` seguente restituisce un elenco dei motori di cache disponibili con le relative versioni.

```
aws elasticache describe-cache-engine-versions \
    --engine "Redis"
```

Output:

```
{
    "CacheEngineVersions": [
        {
            "Engine": "redis",
            "EngineVersion": "2.6.13",
            "CacheParameterGroupFamily": "redis2.6",
            "CacheEngineDescription": "Redis",
            "CacheEngineVersionDescription": "redis version 2.6.13"
        },
        {
            "Engine": "redis",
            "EngineVersion": "2.8.19",
            "CacheParameterGroupFamily": "redis2.8",
            "CacheEngineDescription": "Redis",
            "CacheEngineVersionDescription": "redis version 2.8.19"
        },
        {
            "Engine": "redis",
            "EngineVersion": "2.8.21",
            "CacheParameterGroupFamily": "redis2.8",
            "CacheEngineDescription": "Redis",
            "CacheEngineVersionDescription": "redis version 2.8.21"
        },
        {
            "Engine": "redis",
            "EngineVersion": "2.8.22",
            "CacheParameterGroupFamily": "redis2.8",
```

```
            "CacheEngineDescription": "Redis",
            "CacheEngineVersionDescription": "redis version 2.8.22"
        },
        {
            "Engine": "redis",
            "EngineVersion": "2.8.23",
            "CacheParameterGroupFamily": "redis2.8",
            "CacheEngineDescription": "Redis",
            "CacheEngineVersionDescription": "redis version 2.8.23"
        },
        {
            "Engine": "redis",
            "EngineVersion": "2.8.24",
            "CacheParameterGroupFamily": "redis2.8",
            "CacheEngineDescription": "Redis",
            "CacheEngineVersionDescription": "redis version 2.8.24"
        },
        {
            "Engine": "redis",
            "EngineVersion": "2.8.6",
            "CacheParameterGroupFamily": "redis2.8",
            "CacheEngineDescription": "Redis",
            "CacheEngineVersionDescription": "redis version 2.8.6"
        },
        {
            "Engine": "redis",
            "EngineVersion": "3.2.10",
            "CacheParameterGroupFamily": "redis3.2",
            "CacheEngineDescription": "Redis",
            "CacheEngineVersionDescription": "redis version 3.2.10"
        },
        {
            "Engine": "redis",
            "EngineVersion": "3.2.4",
            "CacheParameterGroupFamily": "redis3.2",
            "CacheEngineDescription": "Redis",
            "CacheEngineVersionDescription": "redis version 3.2.4"
        },
        {
            "Engine": "redis",
            "EngineVersion": "3.2.6",
            "CacheParameterGroupFamily": "redis3.2",
            "CacheEngineDescription": "Redis",
            "CacheEngineVersionDescription": "redis version 3.2.6"
```

```
        },
        {
            "Engine": "redis",
            "EngineVersion": "4.0.10",
            "CacheParameterGroupFamily": "redis4.0",
            "CacheEngineDescription": "Redis",
            "CacheEngineVersionDescription": "redis version 4.0.10"
        },
        {
            "Engine": "redis",
            "EngineVersion": "5.0.0",
            "CacheParameterGroupFamily": "redis5.0",
            "CacheEngineDescription": "Redis",
            "CacheEngineVersionDescription": "redis version 5.0.0"
        },
        {
            "Engine": "redis",
            "EngineVersion": "5.0.3",
            "CacheParameterGroupFamily": "redis5.0",
            "CacheEngineDescription": "Redis",
            "CacheEngineVersionDescription": "redis version 5.0.3"
        },
        {
            "Engine": "redis",
            "EngineVersion": "5.0.4",
            "CacheParameterGroupFamily": "redis5.0",
            "CacheEngineDescription": "Redis",
            "CacheEngineVersionDescription": "redis version 5.0.4"
        },
        {
            "Engine": "redis",
            "EngineVersion": "5.0.5",
            "CacheParameterGroupFamily": "redis5.0",
            "CacheEngineDescription": "Redis",
            "CacheEngineVersionDescription": "redis version 5.0.5"
        }
    ]
}
```

- Per i dettagli sull'API, consulta DescribeCacheEngineVersions AWS CLICommand Reference.

`describe-cache-parameter-groups`

Il seguente esempio di codice mostra come utilizzare`describe-cache-parameter-groups`.

AWS CLI

Come descrivere un gruppo di parametri cache

L'esempio `describe-cache-parameter-groups` seguente restituisce un elenco di descrizioni del gruppo di parametri cache.

```
aws elasticache describe-cache-parameter-groups \
    --cache-parameter-group-name "mygroup"
```

Output:

```
{
    "CacheParameterGroups": [
        {
            "CacheParameterGroupName": "mygroup",
            "CacheParameterGroupFamily": "redis5.0",
            "Description": " "
        }
    ]
}
```

Per ulteriori informazioni, consulta Configurazione dei parametri di motore con i gruppi di parametri nella Guida per l'utente di ElastiCache.

- Per i dettagli sull'API, consulta DescribeCacheParameterGroups AWS CLICommand Reference.

`describe-cache-parameters`

Il seguente esempio di codice mostra come utilizzare`describe-cache-parameters`.

AWS CLI

Come descrivere i parametri cache

Il seguente esempio "describe-cache-parameters" restituisce l'elenco dettagliato dei parametri per il gruppo di parametri di cache specificato.

```
aws elasticache describe-cache-parameters \
    --cache-parameter-group-name "myparamgroup"
```

Output:


```
{
    "Parameters": [
        {
            "ParameterName": "activedefrag",
            "ParameterValue": "yes",
            "Description": "Enabled active memory defragmentation",
            "Source": "user",
            "DataType": "string",
            "AllowedValues": "yes,no",
            "IsModifiable": true,
            "MinimumEngineVersion": "5.0.0",
            "ChangeType": "immediate"
        },
        {
            "ParameterName": "active-defrag-cycle-max",
            "ParameterValue": "75",
            "Description": "Maximal effort for defrag in CPU percentage",
            "Source": "user",
            "DataType": "integer",
            "AllowedValues": "1-75",
            "IsModifiable": true,
            "MinimumEngineVersion": "5.0.0",
            "ChangeType": "immediate"
        },
        {
            "ParameterName": "active-defrag-cycle-min",
            "ParameterValue": "5",
            "Description": "Minimal effort for defrag in CPU percentage",
            "Source": "user",
            "DataType": "integer",
            "AllowedValues": "1-75",
            "IsModifiable": true,
            "MinimumEngineVersion": "5.0.0",
            "ChangeType": "immediate"
        },
        {
            "ParameterName": "active-defrag-ignore-bytes",
            "ParameterValue": "104857600",
```

```
            "Description": "Minimum amount of fragmentation waste to start active
defrag",
            "Source": "user",
            "DataType": "integer",
            "AllowedValues": "1048576-",
            "IsModifiable": true,
            "MinimumEngineVersion": "5.0.0",
            "ChangeType": "immediate"
        },
        {
            "ParameterName": "active-defrag-max-scan-fields",
            "ParameterValue": "1000",
            "Description": "Maximum number of set/hash/zset/list fields that will be
processed from the main dictionary scan",
            "Source": "user",
            "DataType": "integer",
            "AllowedValues": "1-1000000",
            "IsModifiable": true,
            "MinimumEngineVersion": "5.0.0",
            "ChangeType": "immediate"
        },
        {
            "ParameterName": "active-defrag-threshold-lower",
            "ParameterValue": "10",
            "Description": "Minimum percentage of fragmentation to start active
defrag",
            "Source": "user",
            "DataType": "integer",
            "AllowedValues": "1-100",
            "IsModifiable": true,
            "MinimumEngineVersion": "5.0.0",
            "ChangeType": "immediate"
        },
        {
            "ParameterName": "active-defrag-threshold-upper",
            "ParameterValue": "100",
            "Description": "Maximum percentage of fragmentation at which we use
maximum effort",
            "Source": "user",
            "DataType": "integer",
            "AllowedValues": "1-100",
            "IsModifiable": true,
            "MinimumEngineVersion": "5.0.0",
            "ChangeType": "immediate"
```

```
        },
        {
            "ParameterName": "activerehashing",
            "ParameterValue": "yes",
            "Description": "Apply rehashing or not.",
            "Source": "user",
            "DataType": "string",
            "AllowedValues": "yes,no",
            "IsModifiable": true,
            "MinimumEngineVersion": "5.0.0",
            "ChangeType": "requires-reboot"
        },
        {
            "ParameterName": "appendfsync",
            "ParameterValue": "everysec",
            "Description": "fsync policy for AOF persistence",
            "Source": "system",
            "DataType": "string",
            "AllowedValues": "always,everysec,no",
            "IsModifiable": false,
            "MinimumEngineVersion": "5.0.0",
            "ChangeType": "immediate"
        },
        {
            "ParameterName": "appendonly",
            "ParameterValue": "no",
            "Description": "Enable Redis persistence.",
            "Source": "system",
            "DataType": "string",
            "AllowedValues": "yes,no",
            "IsModifiable": false,
            "MinimumEngineVersion": "5.0.0",
            "ChangeType": "immediate"
        },
        {
            "ParameterName": "client-output-buffer-limit-normal-hard-limit",
            "ParameterValue": "0",
            "Description": "Normal client output buffer hard limit in bytes.",
            "Source": "user",
            "DataType": "integer",
            "AllowedValues": "0-",
            "IsModifiable": true,
            "MinimumEngineVersion": "5.0.0",
            "ChangeType": "immediate"
```

```
        },
        {
            "ParameterName": "client-output-buffer-limit-normal-soft-limit",
            "ParameterValue": "0",
            "Description": "Normal client output buffer soft limit in bytes.",
            "Source": "user",
            "DataType": "integer",
            "AllowedValues": "0-",
            "IsModifiable": true,
            "MinimumEngineVersion": "5.0.0",
            "ChangeType": "immediate"
        },
        {
            "ParameterName": "client-output-buffer-limit-normal-soft-seconds",
            "ParameterValue": "0",
            "Description": "Normal client output buffer soft limit in seconds.",
            "Source": "user",
            "DataType": "integer",
            "AllowedValues": "0-",
            "IsModifiable": true,
            "MinimumEngineVersion": "5.0.0",
            "ChangeType": "immediate"
        },
        {
            "ParameterName": "client-output-buffer-limit-pubsub-hard-limit",
            "ParameterValue": "33554432",
            "Description": "Pubsub client output buffer hard limit in bytes.",
            "Source": "user",
            "DataType": "integer",
            "AllowedValues": "0-",
            "IsModifiable": true,
            "MinimumEngineVersion": "5.0.0",
            "ChangeType": "immediate"
        },
        {
            "ParameterName": "client-output-buffer-limit-pubsub-soft-limit",
            "ParameterValue": "8388608",
            "Description": "Pubsub client output buffer soft limit in bytes.",
            "Source": "user",
            "DataType": "integer",
            "AllowedValues": "0-",
            "IsModifiable": true,
            "MinimumEngineVersion": "5.0.0",
            "ChangeType": "immediate"
```

```
        },
        {
            "ParameterName": "client-output-buffer-limit-pubsub-soft-seconds",
            "ParameterValue": "60",
            "Description": "Pubsub client output buffer soft limit in seconds.",
            "Source": "user",
            "DataType": "integer",
            "AllowedValues": "0-",
            "IsModifiable": true,
            "MinimumEngineVersion": "5.0.0",
            "ChangeType": "immediate"
        },
        {
            "ParameterName": "client-output-buffer-limit-replica-soft-seconds",
            "ParameterValue": "60",
            "Description": "Replica client output buffer soft limit in seconds.",
            "Source": "system",
            "DataType": "integer",
            "AllowedValues": "0-",
            "IsModifiable": false,
            "MinimumEngineVersion": "5.0.0",
            "ChangeType": "immediate"
        },
        {
            "ParameterName": "client-query-buffer-limit",
            "ParameterValue": "1073741824",
            "Description": "Max size of a single client query buffer",
            "Source": "user",
            "DataType": "integer",
            "AllowedValues": "1048576-1073741824",
            "IsModifiable": true,
            "MinimumEngineVersion": "5.0.0",
            "ChangeType": "immediate"
        },
        {
            "ParameterName": "close-on-replica-write",
            "ParameterValue": "yes",
            "Description": "If enabled, clients who attempt to write to a read-only
replica will be disconnected. Applicable to 2.8.23 and higher.",
            "Source": "user",
            "DataType": "string",
            "AllowedValues": "yes,no",
            "IsModifiable": true,
            "MinimumEngineVersion": "5.0.0",
```

```
            "ChangeType": "immediate"
        },
        {
            "ParameterName": "cluster-enabled",
            "ParameterValue": "no",
            "Description": "Enable cluster mode",
            "Source": "user",
            "DataType": "string",
            "AllowedValues": "yes,no",
            "IsModifiable": true,
            "MinimumEngineVersion": "5.0.0",
            "ChangeType": "requires-reboot"
        },
        {
            "ParameterName": "cluster-require-full-coverage",
            "ParameterValue": "no",
            "Description": "Whether cluster becomes unavailable if one or more slots
are not covered",
            "Source": "user",
            "DataType": "string",
            "AllowedValues": "yes,no",
            "IsModifiable": true,
            "MinimumEngineVersion": "5.0.0",
            "ChangeType": "immediate"
        },
        {
            "ParameterName": "databases",
            "ParameterValue": "16",
            "Description": "Set the number of databases.",
            "Source": "user",
            "DataType": "integer",
            "AllowedValues": "1-1200000",
            "IsModifiable": true,
            "MinimumEngineVersion": "5.0.0",
            "ChangeType": "requires-reboot"
        },
        {
            "ParameterName": "hash-max-ziplist-entries",
            "ParameterValue": "512",
            "Description": "The maximum number of hash entries in order for the
dataset to be compressed.",
            "Source": "user",
            "DataType": "integer",
            "AllowedValues": "0-",
```

```
            "IsModifiable": true,
            "MinimumEngineVersion": "5.0.0",
            "ChangeType": "immediate"
        },
        {
            "ParameterName": "hash-max-ziplist-value",
            "ParameterValue": "64",
            "Description": "The threshold of biggest hash entries in order for the
dataset to be compressed.",
            "Source": "user",
            "DataType": "integer",
            "AllowedValues": "0-",
            "IsModifiable": true,
            "MinimumEngineVersion": "5.0.0",
            "ChangeType": "immediate"
        },
        {
            "ParameterName": "hll-sparse-max-bytes",
            "ParameterValue": "3000",
            "Description": "HyperLogLog sparse representation bytes limit",
            "Source": "user",
            "DataType": "integer",
            "AllowedValues": "1-16000",
            "IsModifiable": true,
            "MinimumEngineVersion": "5.0.0",
            "ChangeType": "immediate"
        },
        {
            "ParameterName": "lazyfree-lazy-eviction",
            "ParameterValue": "no",
            "Description": "Perform an asynchronous delete on evictions",
            "Source": "user",
            "DataType": "string",
            "AllowedValues": "yes,no",
            "IsModifiable": true,
            "MinimumEngineVersion": "5.0.0",
            "ChangeType": "immediate"
        },
        {
            "ParameterName": "lazyfree-lazy-expire",
            "ParameterValue": "no",
            "Description": "Perform an asynchronous delete on expired keys",
            "Source": "user",
            "DataType": "string",
```

```
            "AllowedValues": "yes,no",
            "IsModifiable": true,
            "MinimumEngineVersion": "5.0.0",
            "ChangeType": "immediate"
        },
        {
            "ParameterName": "lazyfree-lazy-server-del",
            "ParameterValue": "no",
            "Description": "Perform an asynchronous delete on key updates",
            "Source": "user",
            "DataType": "string",
            "AllowedValues": "yes,no",
            "IsModifiable": true,
            "MinimumEngineVersion": "5.0.0",
            "ChangeType": "immediate"
        },
        {
            "ParameterName": "lfu-decay-time",
            "ParameterValue": "1",
            "Description": "The amount of time in minutes to decrement the key
counter for LFU eviction policy",
            "Source": "user",
            "DataType": "integer",
            "AllowedValues": "0-",
            "IsModifiable": true,
            "MinimumEngineVersion": "5.0.0",
            "ChangeType": "immediate"
        },
        {
            "ParameterName": "lfu-log-factor",
            "ParameterValue": "10",
            "Description": "The log factor for incrementing key counter for LFU
eviction policy",
            "Source": "user",
            "DataType": "integer",
            "AllowedValues": "1-",
            "IsModifiable": true,
            "MinimumEngineVersion": "5.0.0",
            "ChangeType": "immediate"
        },
        {
            "ParameterName": "list-compress-depth",
            "ParameterValue": "0",
```

```
            "Description": "Number of quicklist ziplist nodes from each side of
the list to exclude from compression. The head and tail of the list are always
uncompressed for fast push/pop operations",
            "Source": "user",
            "DataType": "integer",
            "AllowedValues": "0-",
            "IsModifiable": true,
            "MinimumEngineVersion": "5.0.0",
            "ChangeType": "immediate"
        },
        {
            "ParameterName": "list-max-ziplist-size",
            "ParameterValue": "-2",
            "Description": "The number of entries allowed per internal list node can
be specified as a fixed maximum size or a maximum number of elements",
            "Source": "system",
            "DataType": "integer",
            "AllowedValues": "-5,-4,-3,-2,-1,1-",
            "IsModifiable": false,
            "MinimumEngineVersion": "5.0.0",
            "ChangeType": "immediate"
        },
        {
            "ParameterName": "lua-replicate-commands",
            "ParameterValue": "yes",
            "Description": "Always enable Lua effect replication or not",
            "Source": "user",
            "DataType": "string",
            "AllowedValues": "yes,no",
            "IsModifiable": true,
            "MinimumEngineVersion": "5.0.0",
            "ChangeType": "immediate"
        },
        {
            "ParameterName": "lua-time-limit",
            "ParameterValue": "5000",
            "Description": "Max execution time of a Lua script in milliseconds. 0
for unlimited execution without warnings.",
            "Source": "system",
            "DataType": "integer",
            "AllowedValues": "5000",
            "IsModifiable": false,
            "MinimumEngineVersion": "5.0.0",
            "ChangeType": "immediate"
```

```
        },
        {
            "ParameterName": "maxclients",
            "ParameterValue": "65000",
            "Description": "The maximum number of Redis clients.",
            "Source": "system",
            "DataType": "integer",
            "AllowedValues": "1-65000",
            "IsModifiable": false,
            "MinimumEngineVersion": "5.0.0",
            "ChangeType": "requires-reboot"
        },
        {
            "ParameterName": "maxmemory-policy",
            "ParameterValue": "volatile-lru",
            "Description": "Max memory policy.",
            "Source": "user",
            "DataType": "string",
            "AllowedValues": "volatile-lru,allkeys-lru,volatile-lfu,allkeys-
lfu,volatile-random,allkeys-random,volatile-ttl,noeviction",
            "IsModifiable": true,
            "MinimumEngineVersion": "5.0.0",
            "ChangeType": "immediate"
        },
        {
            "ParameterName": "maxmemory-samples",
            "ParameterValue": "3",
            "Description": "Max memory samples.",
            "Source": "user",
            "DataType": "integer",
            "AllowedValues": "1-",
            "IsModifiable": true,
            "MinimumEngineVersion": "5.0.0",
            "ChangeType": "immediate"
        },
        {
            "ParameterName": "min-replicas-max-lag",
            "ParameterValue": "10",
            "Description": "The maximum amount of replica lag in seconds beyond
 which the master would stop taking writes. A value of 0 means the master always
 takes writes.",
            "Source": "user",
            "DataType": "integer",
            "AllowedValues": "0-",
```

```
            "IsModifiable": true,
            "MinimumEngineVersion": "5.0.0",
            "ChangeType": "immediate"
        },
        {
            "ParameterName": "min-replicas-to-write",
            "ParameterValue": "0",
            "Description": "The minimum number of replicas that must be present with
lag no greater than min-replicas-max-lag for master to take writes. Setting this to
0 means the master always takes writes.",
            "Source": "user",
            "DataType": "integer",
            "AllowedValues": "0-",
            "IsModifiable": true,
            "MinimumEngineVersion": "5.0.0",
            "ChangeType": "immediate"
        },
        {
            "ParameterName": "notify-keyspace-events",
            "Description": "The keyspace events for Redis to notify Pub/Sub clients
about. By default all notifications are disabled",
            "Source": "user",
            "DataType": "string",
            "IsModifiable": true,
            "MinimumEngineVersion": "5.0.0",
            "ChangeType": "immediate"
        },
        {
            "ParameterName": "proto-max-bulk-len",
            "ParameterValue": "536870912",
            "Description": "Max size of a single element request",
            "Source": "user",
            "DataType": "integer",
            "AllowedValues": "1048576-536870912",
            "IsModifiable": true,
            "MinimumEngineVersion": "5.0.0",
            "ChangeType": "immediate"
        },
        {
            "ParameterName": "rename-commands",
            "ParameterValue": "",
            "Description": "Redis commands that can be dynamically renamed by the
customer",
            "Source": "user",
```

```
            "DataType": "string",
            "AllowedValues":
"APPEND,BITCOUNT,BITFIELD,BITOP,BITPOS,BLPOP,BRPOP,BRPOPLPUSH,BZPOPMIN,BZPOPMAX,CLIENT,COMM
            "IsModifiable": true,
            "MinimumEngineVersion": "5.0.3",
            "ChangeType": "immediate"
        },
        {
            "ParameterName": "repl-backlog-size",
            "ParameterValue": "1048576",
            "Description": "The replication backlog size in bytes for PSYNC. This is
the size of the buffer which accumulates slave data when slave is disconnected for
some time, so that when slave reconnects again, only transfer the portion of data
which the slave missed. Minimum value is 16K.",
            "Source": "user",
            "DataType": "integer",
            "AllowedValues": "16384-",
            "IsModifiable": true,
            "MinimumEngineVersion": "5.0.0",
            "ChangeType": "immediate"
        },
        {
            "ParameterName": "repl-backlog-ttl",
            "ParameterValue": "3600",
            "Description": "The amount of time in seconds after the master no longer
have any slaves connected for the master to free the replication backlog. A value
of 0 means to never release the backlog.",
            "Source": "user",
            "DataType": "integer",
            "AllowedValues": "0-",
            "IsModifiable": true,
            "MinimumEngineVersion": "5.0.0",
            "ChangeType": "immediate"
        },
        {
            "ParameterName": "replica-allow-chaining",
            "ParameterValue": "no",
            "Description": "Configures if chaining of replicas is allowed",
            "Source": "system",
            "DataType": "string",
            "AllowedValues": "yes,no",
            "IsModifiable": false,
            "MinimumEngineVersion": "5.0.0",
            "ChangeType": "immediate"
```

```
        },
        {
            "ParameterName": "replica-ignore-maxmemory",
            "ParameterValue": "yes",
            "Description": "Determines if replica ignores maxmemory setting by not
evicting items independent from the master",
            "Source": "system",
            "DataType": "string",
            "AllowedValues": "yes,no",
            "IsModifiable": false,
            "MinimumEngineVersion": "5.0.0",
            "ChangeType": "immediate"
        },
        {
            "ParameterName": "replica-lazy-flush",
            "ParameterValue": "no",
            "Description": "Perform an asynchronous flushDB during replica sync",
            "Source": "system",
            "DataType": "string",
            "AllowedValues": "yes,no",
            "IsModifiable": false,
            "MinimumEngineVersion": "5.0.0",
            "ChangeType": "immediate"
        },
        {
            "ParameterName": "reserved-memory-percent",
            "ParameterValue": "25",
            "Description": "The percent of memory reserved for non-cache memory
usage. You may want to increase this parameter for nodes with read replicas, AOF
enabled, etc, to reduce swap usage.",
            "Source": "user",
            "DataType": "integer",
            "AllowedValues": "0-100",
            "IsModifiable": true,
            "MinimumEngineVersion": "5.0.0",
            "ChangeType": "immediate"
        },
        {
            "ParameterName": "set-max-intset-entries",
            "ParameterValue": "512",
            "Description": "The limit in the size of the set in order for the
dataset to be compressed.",
            "Source": "user",
            "DataType": "integer",
```

```
            "AllowedValues": "0-",
            "IsModifiable": true,
            "MinimumEngineVersion": "5.0.0",
            "ChangeType": "immediate"
        },
        {
            "ParameterName": "slowlog-log-slower-than",
            "ParameterValue": "10000",
            "Description": "The execution time, in microseconds, to exceed in order
for the command to get logged. Note that a negative number disables the slow log,
while a value of zero forces the logging of every command.",
            "Source": "user",
            "DataType": "integer",
            "AllowedValues": "-",
            "IsModifiable": true,
            "MinimumEngineVersion": "5.0.0",
            "ChangeType": "immediate"
        },
        {
            "ParameterName": "slowlog-max-len",
            "ParameterValue": "128",
            "Description": "The length of the slow log. There is no limit to this
length. Just be aware that it will consume memory. You can reclaim memory used by
the slow log with SLOWLOG RESET.",
            "Source": "user",
            "DataType": "integer",
            "AllowedValues": "0-",
            "IsModifiable": true,
            "MinimumEngineVersion": "5.0.0",
            "ChangeType": "immediate"
        },
        {
            "ParameterName": "stream-node-max-bytes",
            "ParameterValue": "4096",
            "Description": "The maximum size of a single node in a stream in bytes",
            "Source": "user",
            "DataType": "integer",
            "AllowedValues": "0-",
            "IsModifiable": true,
            "MinimumEngineVersion": "5.0.0",
            "ChangeType": "immediate"
        },
        {
            "ParameterName": "stream-node-max-entries",
```

```
            "ParameterValue": "100",
            "Description": "The maximum number of items a single node in a stream
can contain",
            "Source": "user",
            "DataType": "integer",
            "AllowedValues": "0-",
            "IsModifiable": true,
            "MinimumEngineVersion": "5.0.0",
            "ChangeType": "immediate"
        },
        {
            "ParameterName": "tcp-keepalive",
            "ParameterValue": "300",
            "Description": "If non-zero, send ACKs every given number of seconds.",
            "Source": "user",
            "DataType": "integer",
            "AllowedValues": "0-",
            "IsModifiable": true,
            "MinimumEngineVersion": "5.0.0",
            "ChangeType": "immediate"
        },
        {
            "ParameterName": "timeout",
            "ParameterValue": "0",
            "Description": "Close connection if client is idle for a given number of
seconds, or never if 0.",
            "Source": "user",
            "DataType": "integer",
            "AllowedValues": "0,20-",
            "IsModifiable": true,
            "MinimumEngineVersion": "5.0.0",
            "ChangeType": "immediate"
        },
        {
            "ParameterName": "zset-max-ziplist-entries",
            "ParameterValue": "128",
            "Description": "The maximum number of sorted set entries in order for
the dataset to be compressed.",
            "Source": "user",
            "DataType": "integer",
            "AllowedValues": "0-",
            "IsModifiable": true,
            "MinimumEngineVersion": "5.0.0",
            "ChangeType": "immediate"
```

```
        },
        {
            "ParameterName": "zset-max-ziplist-value",
            "ParameterValue": "64",
            "Description": "The threshold of biggest sorted set entries in order for
 the dataset to be compressed.",
            "Source": "user",
            "DataType": "integer",
            "AllowedValues": "0-",
            "IsModifiable": true,
            "MinimumEngineVersion": "5.0.0",
            "ChangeType": "immediate"
        }
    ]
}
```

Per ulteriori informazioni, consulta Gestione dei parametri nella Guida per l'utente di ElastiCache.

- Per i dettagli sull'API, consulta DescribeCacheParameters AWS CLICommand Reference.

`describe-cache-subnet-groups`

Il seguente esempio di codice mostra come utilizzare`describe-cache-subnet-groups`.

AWS CLI

Come descrivere i gruppi di sottoreti della cache

L'esempio `describe-cache-subnet-groups` seguente restituisce un elenco di gruppi di sottoreti.

```
aws elasticache describe-cache-subnet-groups
```

Output:

```
{
    "CacheSubnetGroups": [
        {
            "CacheSubnetGroupName": "default",
            "CacheSubnetGroupDescription": "Default CacheSubnetGroup",
            "VpcId": "vpc-a3e97cdb",
            "Subnets": [
                {
```

```
                    "SubnetIdentifier": "subnet-8d4bacf5",
                    "SubnetAvailabilityZone": {
                        "Name": "us-west-2b"
                    }
                },
                {
                    "SubnetIdentifier": "subnet-dde21380",
                    "SubnetAvailabilityZone": {
                        "Name": "us-west-2c"
                    }
                },
                {
                    "SubnetIdentifier": "subnet-6485ec4f",
                    "SubnetAvailabilityZone": {
                        "Name": "us-west-2d"
                    }
                },
                {
                    "SubnetIdentifier": "subnet-b4ebebff",
                    "SubnetAvailabilityZone": {
                        "Name": "us-west-2a"
                    }
                }
            ]
        },
        {
            "CacheSubnetGroupName": "kxkxk",
            "CacheSubnetGroupDescription": "mygroup",
            "VpcId": "vpc-a3e97cdb",
            "Subnets": [
                {
                    "SubnetIdentifier": "subnet-b4ebebff",
                    "SubnetAvailabilityZone": {
                        "Name": "us-west-2a"
                    }
                }
            ]
        },
        {
            "CacheSubnetGroupName": "test",
            "CacheSubnetGroupDescription": "test",
            "VpcId": "vpc-a3e97cdb",
            "Subnets": [
                {
```

```
                    "SubnetIdentifier": "subnet-b4ebebff",
                    "SubnetAvailabilityZone": {
                        "Name": "us-west-2a"
                    }
                }
            ]
        }
    ]
}
```

Per ulteriori informazioni, consulta Subnet and Subnet Groups nella Elasticache User Guide o Subnet and Subnet Groups nella for Memcached User Guide. ElastiCache

- Per DescribeCacheSubnetGroups AWS CLIi dettagli sull'API, consulta Command Reference.

`describe-engine-default-parameters`

Il seguente esempio di codice mostra come utilizzare`describe-engine-default-parameters`.

AWS CLI

Come descrivere i parametri predefiniti del motore

L'esempio `describe-engine-default-parameters` seguente restituisce le informazioni sui parametri predefiniti del motore e del sistema per il motore di cache specificato.

```
aws elasticache describe-engine-default-parameters \
    --cache-parameter-group-family "redis5.0"
```

Output:

```
{
    "EngineDefaults": {
        "Parameters": [
            {
                "ParameterName": "activedefrag",
                "ParameterValue": "no",
                "Description": "Enabled active memory defragmentation",
                "Source": "system",
                "DataType": "string",
                "AllowedValues": "yes,no",
                "IsModifiable": true,
                "MinimumEngineVersion": "5.0.0",
```

```
                "ChangeType": "immediate"
            },
            {
                "ParameterName": "active-defrag-cycle-max",
                "ParameterValue": "75",
                "Description": "Maximal effort for defrag in CPU percentage",
                "Source": "system",
                "DataType": "integer",
                "AllowedValues": "1-75",
                "IsModifiable": true,
                "MinimumEngineVersion": "5.0.0",
                "ChangeType": "immediate"
            },
            {
                "ParameterName": "active-defrag-cycle-min",
                "ParameterValue": "5",
                "Description": "Minimal effort for defrag in CPU percentage",
                "Source": "system",
                "DataType": "integer",
                "AllowedValues": "1-75",
                "IsModifiable": true,
                "MinimumEngineVersion": "5.0.0",
                "ChangeType": "immediate"
            },
            {
                "ParameterName": "active-defrag-ignore-bytes",
                "ParameterValue": "104857600",
                "Description": "Minimum amount of fragmentation waste to start
active defrag",
                "Source": "system",
                "DataType": "integer",
                "AllowedValues": "1048576-",
                "IsModifiable": true,
                "MinimumEngineVersion": "5.0.0",
                "ChangeType": "immediate"
            },
            {
                "ParameterName": "active-defrag-max-scan-fields",
                "ParameterValue": "1000",
                "Description": "Maximum number of set/hash/zset/list fields that
will be processed from the main dictionary scan",
                "Source": "system",
                "DataType": "integer",
                "AllowedValues": "1-1000000",
```

```
                "IsModifiable": true,
                "MinimumEngineVersion": "5.0.0",
                "ChangeType": "immediate"
            },
            {
                "ParameterName": "active-defrag-threshold-lower",
                "ParameterValue": "10",
                "Description": "Minimum percentage of fragmentation to start active
defrag",
                "Source": "system",
                "DataType": "integer",
                "AllowedValues": "1-100",
                "IsModifiable": true,
                "MinimumEngineVersion": "5.0.0",
                "ChangeType": "immediate"
            },
            {
                "ParameterName": "active-defrag-threshold-upper",
                "ParameterValue": "100",
                "Description": "Maximum percentage of fragmentation at which we use
maximum effort",
                "Source": "system",
                "DataType": "integer",
                "AllowedValues": "1-100",
                "IsModifiable": true,
                "MinimumEngineVersion": "5.0.0",
                "ChangeType": "immediate"
            },
            {
                "ParameterName": "activerehashing",
                "ParameterValue": "yes",
                "Description": "Apply rehashing or not.",
                "Source": "system",
                "DataType": "string",
                "AllowedValues": "yes,no",
                "IsModifiable": false,
                "MinimumEngineVersion": "5.0.0",
                "ChangeType": "requires-reboot"
            },
            {
                "ParameterName": "appendfsync",
                "ParameterValue": "everysec",
                "Description": "fsync policy for AOF persistence",
                "Source": "system",
```

```
                "DataType": "string",
                "AllowedValues": "always,everysec,no",
                "IsModifiable": false,
                "MinimumEngineVersion": "5.0.0",
                "ChangeType": "immediate"
            },
            {
                "ParameterName": "appendonly",
                "ParameterValue": "no",
                "Description": "Enable Redis persistence.",
                "Source": "system",
                "DataType": "string",
                "AllowedValues": "yes,no",
                "IsModifiable": false,
                "MinimumEngineVersion": "5.0.0",
                "ChangeType": "immediate"
            },
            {
                "ParameterName": "client-output-buffer-limit-normal-hard-limit",
                "ParameterValue": "0",
                "Description": "Normal client output buffer hard limit in bytes.",
                "Source": "system",
                "DataType": "integer",
                "AllowedValues": "0-",
                "IsModifiable": true,
                "MinimumEngineVersion": "5.0.0",
                "ChangeType": "immediate"
            },
            {
                "ParameterName": "client-output-buffer-limit-normal-soft-limit",
                "ParameterValue": "0",
                "Description": "Normal client output buffer soft limit in bytes.",
                "Source": "system",
                "DataType": "integer",
                "AllowedValues": "0-",
                "IsModifiable": true,
                "MinimumEngineVersion": "5.0.0",
                "ChangeType": "immediate"
            },
            {
                "ParameterName": "client-output-buffer-limit-normal-soft-seconds",
                "ParameterValue": "0",
                "Description": "Normal client output buffer soft limit in seconds.",
                "Source": "system",
```

```
                "DataType": "integer",
                "AllowedValues": "0-",
                "IsModifiable": true,
                "MinimumEngineVersion": "5.0.0",
                "ChangeType": "immediate"
            },
            {
                "ParameterName": "client-output-buffer-limit-pubsub-hard-limit",
                "ParameterValue": "33554432",
                "Description": "Pubsub client output buffer hard limit in bytes.",
                "Source": "system",
                "DataType": "integer",
                "AllowedValues": "0-",
                "IsModifiable": true,
                "MinimumEngineVersion": "5.0.0",
                "ChangeType": "immediate"
            },
            {
                "ParameterName": "client-output-buffer-limit-pubsub-soft-limit",
                "ParameterValue": "8388608",
                "Description": "Pubsub client output buffer soft limit in bytes.",
                "Source": "system",
                "DataType": "integer",
                "AllowedValues": "0-",
                "IsModifiable": true,
                "MinimumEngineVersion": "5.0.0",
                "ChangeType": "immediate"
            },
            {
                "ParameterName": "client-output-buffer-limit-pubsub-soft-seconds",
                "ParameterValue": "60",
                "Description": "Pubsub client output buffer soft limit in seconds.",
                "Source": "system",
                "DataType": "integer",
                "AllowedValues": "0-",
                "IsModifiable": true,
                "MinimumEngineVersion": "5.0.0",
                "ChangeType": "immediate"
            },
            {
                "ParameterName": "client-output-buffer-limit-replica-soft-seconds",
                "ParameterValue": "60",
                "Description": "Replica client output buffer soft limit in
seconds.",
```

```
                "Source": "system",
                "DataType": "integer",
                "AllowedValues": "0-",
                "IsModifiable": false,
                "MinimumEngineVersion": "5.0.0",
                "ChangeType": "immediate"
            },
            {
                "ParameterName": "client-query-buffer-limit",
                "ParameterValue": "1073741824",
                "Description": "Max size of a single client query buffer",
                "Source": "system",
                "DataType": "integer",
                "AllowedValues": "1048576-1073741824",
                "IsModifiable": true,
                "MinimumEngineVersion": "5.0.0",
                "ChangeType": "immediate"
            },
            {
                "ParameterName": "close-on-replica-write",
                "ParameterValue": "yes",
                "Description": "If enabled, clients who attempt to write to a read-
only replica will be disconnected. Applicable to 2.8.23 and higher.",
                "Source": "system",
                "DataType": "string",
                "AllowedValues": "yes,no",
                "IsModifiable": true,
                "MinimumEngineVersion": "5.0.0",
                "ChangeType": "immediate"
            },
            {
                "ParameterName": "cluster-enabled",
                "ParameterValue": "no",
                "Description": "Enable cluster mode",
                "Source": "system",
                "DataType": "string",
                "AllowedValues": "yes,no",
                "IsModifiable": false,
                "MinimumEngineVersion": "5.0.0",
                "ChangeType": "requires-reboot"
            },
            {
                "ParameterName": "cluster-require-full-coverage",
                "ParameterValue": "no",
```

```
                "Description": "Whether cluster becomes unavailable if one or more
slots are not covered",
                "Source": "system",
                "DataType": "string",
                "AllowedValues": "yes,no",
                "IsModifiable": true,
                "MinimumEngineVersion": "5.0.0",
                "ChangeType": "immediate"
            },
            {
                "ParameterName": "databases",
                "ParameterValue": "16",
                "Description": "Set the number of databases.",
                "Source": "system",
                "DataType": "integer",
                "AllowedValues": "1-1200000",
                "IsModifiable": false,
                "MinimumEngineVersion": "5.0.0",
                "ChangeType": "requires-reboot"
            },
            {
                "ParameterName": "hash-max-ziplist-entries",
                "ParameterValue": "512",
                "Description": "The maximum number of hash entries in order for the
dataset to be compressed.",
                "Source": "system",
                "DataType": "integer",
                "AllowedValues": "0-",
                "IsModifiable": true,
                "MinimumEngineVersion": "5.0.0",
                "ChangeType": "immediate"
            },
            {
                "ParameterName": "hash-max-ziplist-value",
                "ParameterValue": "64",
                "Description": "The threshold of biggest hash entries in order for
the dataset to be compressed.",
                "Source": "system",
                "DataType": "integer",
                "AllowedValues": "0-",
                "IsModifiable": true,
                "MinimumEngineVersion": "5.0.0",
                "ChangeType": "immediate"
            },
```

```
            {
                "ParameterName": "hll-sparse-max-bytes",
                "ParameterValue": "3000",
                "Description": "HyperLogLog sparse representation bytes limit",
                "Source": "system",
                "DataType": "integer",
                "AllowedValues": "1-16000",
                "IsModifiable": true,
                "MinimumEngineVersion": "5.0.0",
                "ChangeType": "immediate"
            },
            {
                "ParameterName": "lazyfree-lazy-eviction",
                "ParameterValue": "no",
                "Description": "Perform an asynchronous delete on evictions",
                "Source": "system",
                "DataType": "string",
                "AllowedValues": "yes,no",
                "IsModifiable": true,
                "MinimumEngineVersion": "5.0.0",
                "ChangeType": "immediate"
            },
            {
                "ParameterName": "lazyfree-lazy-expire",
                "ParameterValue": "no",
                "Description": "Perform an asynchronous delete on expired keys",
                "Source": "system",
                "DataType": "string",
                "AllowedValues": "yes,no",
                "IsModifiable": true,
                "MinimumEngineVersion": "5.0.0",
                "ChangeType": "immediate"
            },
            {
                "ParameterName": "lazyfree-lazy-server-del",
                "ParameterValue": "no",
                "Description": "Perform an asynchronous delete on key updates",
                "Source": "system",
                "DataType": "string",
                "AllowedValues": "yes,no",
                "IsModifiable": true,
                "MinimumEngineVersion": "5.0.0",
                "ChangeType": "immediate"
            },
```

```
            {
                "ParameterName": "lfu-decay-time",
                "ParameterValue": "1",
                "Description": "The amount of time in minutes to decrement the key
counter for LFU eviction policy",
                "Source": "system",
                "DataType": "integer",
                "AllowedValues": "0-",
                "IsModifiable": true,
                "MinimumEngineVersion": "5.0.0",
                "ChangeType": "immediate"
            },
            {
                "ParameterName": "lfu-log-factor",
                "ParameterValue": "10",
                "Description": "The log factor for incrementing key counter for LFU
eviction policy",
                "Source": "system",
                "DataType": "integer",
                "AllowedValues": "1-",
                "IsModifiable": true,
                "MinimumEngineVersion": "5.0.0",
                "ChangeType": "immediate"
            },
            {
                "ParameterName": "list-compress-depth",
                "ParameterValue": "0",
                "Description": "Number of quicklist ziplist nodes from each side
of the list to exclude from compression. The head and tail of the list are always
uncompressed for fast push/pop operations",
                "Source": "system",
                "DataType": "integer",
                "AllowedValues": "0-",
                "IsModifiable": true,
                "MinimumEngineVersion": "5.0.0",
                "ChangeType": "immediate"
            },
            {
                "ParameterName": "list-max-ziplist-size",
                "ParameterValue": "-2",
                "Description": "The number of entries allowed per internal list node
can be specified as a fixed maximum size or a maximum number of elements",
                "Source": "system",
                "DataType": "integer",
```

```
                "AllowedValues": "-5,-4,-3,-2,-1,1-",
                "IsModifiable": false,
                "MinimumEngineVersion": "5.0.0",
                "ChangeType": "immediate"
            },
            {
                "ParameterName": "lua-replicate-commands",
                "ParameterValue": "yes",
                "Description": "Always enable Lua effect replication or not",
                "Source": "system",
                "DataType": "string",
                "AllowedValues": "yes,no",
                "IsModifiable": true,
                "MinimumEngineVersion": "5.0.0",
                "ChangeType": "immediate"
            },
            {
                "ParameterName": "lua-time-limit",
                "ParameterValue": "5000",
                "Description": "Max execution time of a Lua script in milliseconds.
0 for unlimited execution without warnings.",
                "Source": "system",
                "DataType": "integer",
                "AllowedValues": "5000",
                "IsModifiable": false,
                "MinimumEngineVersion": "5.0.0",
                "ChangeType": "immediate"
            },
            {
                "ParameterName": "maxclients",
                "ParameterValue": "65000",
                "Description": "The maximum number of Redis clients.",
                "Source": "system",
                "DataType": "integer",
                "AllowedValues": "1-65000",
                "IsModifiable": false,
                "MinimumEngineVersion": "5.0.0",
                "ChangeType": "requires-reboot"
            },
            {
                "ParameterName": "maxmemory-policy",
                "ParameterValue": "volatile-lru",
                "Description": "Max memory policy.",
                "Source": "system",
```

```
                "DataType": "string",
                "AllowedValues": "volatile-lru,allkeys-lru,volatile-lfu,allkeys-
lfu,volatile-random,allkeys-random,volatile-ttl,noeviction",
                "IsModifiable": true,
                "MinimumEngineVersion": "5.0.0",
                "ChangeType": "immediate"
            },
            {
                "ParameterName": "maxmemory-samples",
                "ParameterValue": "3",
                "Description": "Max memory samples.",
                "Source": "system",
                "DataType": "integer",
                "AllowedValues": "1-",
                "IsModifiable": true,
                "MinimumEngineVersion": "5.0.0",
                "ChangeType": "immediate"
            },
            {
                "ParameterName": "min-replicas-max-lag",
                "ParameterValue": "10",
                "Description": "The maximum amount of replica lag in seconds beyond
 which the master would stop taking writes. A value of 0 means the master always
 takes writes.",
                "Source": "system",
                "DataType": "integer",
                "AllowedValues": "0-",
                "IsModifiable": true,
                "MinimumEngineVersion": "5.0.0",
                "ChangeType": "immediate"
            },
            {
                "ParameterName": "min-replicas-to-write",
                "ParameterValue": "0",
                "Description": "The minimum number of replicas that must be present
 with lag no greater than min-replicas-max-lag for master to take writes. Setting
 this to 0 means the master always takes writes.",
                "Source": "system",
                "DataType": "integer",
                "AllowedValues": "0-",
                "IsModifiable": true,
                "MinimumEngineVersion": "5.0.0",
                "ChangeType": "immediate"
            },
```

```
            {
                "ParameterName": "notify-keyspace-events",
                "Description": "The keyspace events for Redis to notify Pub/Sub
clients about. By default all notifications are disabled",
                "Source": "system",
                "DataType": "string",
                "IsModifiable": true,
                "MinimumEngineVersion": "5.0.0",
                "ChangeType": "immediate"
            },
            {
                "ParameterName": "proto-max-bulk-len",
                "ParameterValue": "536870912",
                "Description": "Max size of a single element request",
                "Source": "system",
                "DataType": "integer",
                "AllowedValues": "1048576-536870912",
                "IsModifiable": true,
                "MinimumEngineVersion": "5.0.0",
                "ChangeType": "immediate"
            },
            {
                "ParameterName": "rename-commands",
                "ParameterValue": "",
                "Description": "Redis commands that can be dynamically renamed by
the customer",
                "Source": "system",
                "DataType": "string",
                "AllowedValues":
"APPEND,BITCOUNT,BITFIELD,BITOP,BITPOS,BLPOP,BRPOP,BRPOPLPUSH,BZPOPMIN,BZPOPMAX,CLIENT,COMM
                "IsModifiable": true,
                "MinimumEngineVersion": "5.0.3",
                "ChangeType": "immediate"
            },
            {
                "ParameterName": "repl-backlog-size",
                "ParameterValue": "1048576",
                "Description": "The replication backlog size in bytes for PSYNC.
This is the size of the buffer which accumulates slave data when slave is
disconnected for some time, so that when slave reconnects again, only transfer the
portion of data which the slave missed. Minimum value is 16K.",
                "Source": "system",
                "DataType": "integer",
                "AllowedValues": "16384-",
```

```
                "IsModifiable": true,
                "MinimumEngineVersion": "5.0.0",
                "ChangeType": "immediate"
            },
            {
                "ParameterName": "repl-backlog-ttl",
                "ParameterValue": "3600",
                "Description": "The amount of time in seconds after the master no
longer have any slaves connected for the master to free the replication backlog. A
value of 0 means to never release the backlog.",
                "Source": "system",
                "DataType": "integer",
                "AllowedValues": "0-",
                "IsModifiable": true,
                "MinimumEngineVersion": "5.0.0",
                "ChangeType": "immediate"
            },
            {
                "ParameterName": "replica-allow-chaining",
                "ParameterValue": "no",
                "Description": "Configures if chaining of replicas is allowed",
                "Source": "system",
                "DataType": "string",
                "AllowedValues": "yes,no",
                "IsModifiable": false,
                "MinimumEngineVersion": "5.0.0",
                "ChangeType": "immediate"
            },
            {
                "ParameterName": "replica-ignore-maxmemory",
                "ParameterValue": "yes",
                "Description": "Determines if replica ignores maxmemory setting by
not evicting items independent from the master",
                "Source": "system",
                "DataType": "string",
                "AllowedValues": "yes,no",
                "IsModifiable": false,
                "MinimumEngineVersion": "5.0.0",
                "ChangeType": "immediate"
            },
            {
                "ParameterName": "replica-lazy-flush",
                "ParameterValue": "no",
```

```
                "Description": "Perform an asynchronous flushDB during replica
sync",
                "Source": "system",
                "DataType": "string",
                "AllowedValues": "yes,no",
                "IsModifiable": false,
                "MinimumEngineVersion": "5.0.0",
                "ChangeType": "immediate"
            },
            {
                "ParameterName": "reserved-memory-percent",
                "ParameterValue": "25",
                "Description": "The percent of memory reserved for non-cache memory
usage. You may want to increase this parameter for nodes with read replicas, AOF
enabled, etc, to reduce swap usage.",
                "Source": "system",
                "DataType": "integer",
                "AllowedValues": "0-100",
                "IsModifiable": true,
                "MinimumEngineVersion": "5.0.0",
                "ChangeType": "immediate"
            },
            {
                "ParameterName": "set-max-intset-entries",
                "ParameterValue": "512",
                "Description": "The limit in the size of the set in order for the
dataset to be compressed.",
                "Source": "system",
                "DataType": "integer",
                "AllowedValues": "0-",
                "IsModifiable": true,
                "MinimumEngineVersion": "5.0.0",
                "ChangeType": "immediate"
            },
            {
                "ParameterName": "slowlog-log-slower-than",
                "ParameterValue": "10000",
                "Description": "The execution time, in microseconds, to exceed in
order for the command to get logged. Note that a negative number disables the slow
log, while a value of zero forces the logging of every command.",
                "Source": "system",
                "DataType": "integer",
                "AllowedValues": "-",
                "IsModifiable": true,
```

```
                "MinimumEngineVersion": "5.0.0",
                "ChangeType": "immediate"
            },
            {
                "ParameterName": "slowlog-max-len",
                "ParameterValue": "128",
                "Description": "The length of the slow log. There is no limit to
this length. Just be aware that it will consume memory. You can reclaim memory used
by the slow log with SLOWLOG RESET.",
                "Source": "system",
                "DataType": "integer",
                "AllowedValues": "0-",
                "IsModifiable": true,
                "MinimumEngineVersion": "5.0.0",
                "ChangeType": "immediate"
            },
            {
                "ParameterName": "stream-node-max-bytes",
                "ParameterValue": "4096",
                "Description": "The maximum size of a single node in a stream in
bytes",
                "Source": "system",
                "DataType": "integer",
                "AllowedValues": "0-",
                "IsModifiable": true,
                "MinimumEngineVersion": "5.0.0",
                "ChangeType": "immediate"
            },
            {
                "ParameterName": "stream-node-max-entries",
                "ParameterValue": "100",
                "Description": "The maximum number of items a single node in a
stream can contain",
                "Source": "system",
                "DataType": "integer",
                "AllowedValues": "0-",
                "IsModifiable": true,
                "MinimumEngineVersion": "5.0.0",
                "ChangeType": "immediate"
            },
            {
                "ParameterName": "tcp-keepalive",
                "ParameterValue": "300",
```

```
                "Description": "If non-zero, send ACKs every given number of
seconds.",
                "Source": "system",
                "DataType": "integer",
                "AllowedValues": "0-",
                "IsModifiable": true,
                "MinimumEngineVersion": "5.0.0",
                "ChangeType": "immediate"
            },
            {
                "ParameterName": "timeout",
                "ParameterValue": "0",
                "Description": "Close connection if client is idle for a given
number of seconds, or never if 0.",
                "Source": "system",
                "DataType": "integer",
                "AllowedValues": "0,20-",
                "IsModifiable": true,
                "MinimumEngineVersion": "5.0.0",
                "ChangeType": "immediate"
            },
            {
                "ParameterName": "zset-max-ziplist-entries",
                "ParameterValue": "128",
                "Description": "The maximum number of sorted set entries in order
for the dataset to be compressed.",
                "Source": "system",
                "DataType": "integer",
                "AllowedValues": "0-",
                "IsModifiable": true,
                "MinimumEngineVersion": "5.0.0",
                "ChangeType": "immediate"
            },
            {
                "ParameterName": "zset-max-ziplist-value",
                "ParameterValue": "64",
                "Description": "The threshold of biggest sorted set entries in order
for the dataset to be compressed.",
                "Source": "system",
                "DataType": "integer",
                "AllowedValues": "0-",
                "IsModifiable": true,
                "MinimumEngineVersion": "5.0.0",
                "ChangeType": "immediate"
```

```
            }
        ]
    }
}
```

- Per i dettagli sull'API, consulta DescribeEngineDefaultParameters AWS CLICommand Reference.

`describe-events`

Il seguente esempio di codice mostra come utilizzare`describe-events`.

AWS CLI

Come descrivere gli eventi di un gruppo di replica

L'esempio `describe-events` seguente restituisce un elenco di eventi per un gruppo di replica.

```
aws elasticache describe-events \
    --source-identifier test-cluster \
    --source-type replication-group
```

Output:

```
{
    "Events": [
        {
            "SourceIdentifier": "test-cluster",
            "SourceType": "replication-group",
            "Message": "Automatic failover has been turned on for replication group
 test-cluster",
            "Date": "2020-03-18T23:51:34.457Z"
        },
        {
            "SourceIdentifier": "test-cluster",
            "SourceType": "replication-group",
            "Message": "Replication group test-cluster created",
            "Date": "2020-03-18T23:50:31.378Z"
        }
    ]
}
```

Per ulteriori informazioni, consulta Monitoraggio degli eventi nella Guida per l'utente di ElastiCache.

- Per i dettagli sull'API, consulta DescribeEvents AWS CLICommand Reference.

`describe-global-replication-groups`

Il seguente esempio di codice mostra come utilizzare`describe-global-replication-groups`.

AWS CLI

Come descrivere i gruppi di replica globale

L'esempio `describe-global-replication-groups` seguente restituisce i dettagli relativi a un datastore globale.

```
aws elasticache describe-global-replication-groups \
    --global-replication-group-id my-grg
```

Output:

```
{
    "GlobalReplicationGroups": [
        {
            "GlobalReplicationGroupId": "my-grg",
            "GlobalReplicationGroupDescription": "my-grg",
            "Status": "creating",
            "CacheNodeType": "cache.r5.large",
            "Engine": "redis",
            "EngineVersion": "5.0.6",
            "ClusterEnabled": false,
            "AuthTokenEnabled": false,
            "TransitEncryptionEnabled": false,
            "AtRestEncryptionEnabled": false
        }
    ]
}
```

Per ulteriori informazioni, consulta Replica tra AWS regioni utilizzando Global Datastore nella Guida per l'utente di Elasticache.

- Per i dettagli sull'API, consulta Command Reference. DescribeGlobalReplicationGroupsAWS CLI

`describe-replication-groups`

Il seguente esempio di codice mostra come utilizzare`describe-replication-groups`.

AWS CLI

Come restituire un elenco di dettagli del gruppo di replica

L'esempio `describe-replication-groups` seguente restituisce i gruppi di replica.

```
aws elasticache describe-replication-groups
```

Output:

```
{
    "ReplicationGroups": [
        {
            "ReplicationGroupId": "my-cluster",
            "Description": "mycluster",
            "Status": "available",
            "PendingModifiedValues": {},
            "MemberClusters": [
                "pat-cluster-001",
                "pat-cluster-002",
                "pat-cluster-003",
                "pat-cluster-004"
            ],
            "NodeGroups": [
                {
                    "NodeGroupId": "0001",
                    "Status": "available",
                    "PrimaryEndpoint": {
                        "Address": "my-
cluster.xxxxih.ng.0001.usw2.cache.amazonaws.com",
                        "Port": 6379
                    },
                    "ReaderEndpoint": {
                        "Address": "my-cluster-
ro.xxxxih.ng.0001.usw2.cache.amazonaws.com",
                        "Port": 6379
                    },
                    "NodeGroupMembers": [
                        {
                            "CacheClusterId": "my-cluster-001",
```

```
                            "CacheNodeId": "0001",
                            "ReadEndpoint": {
                                "Address": "pat-
cluster-001.xxxih.0001.usw2.cache.amazonaws.com",
                                "Port": 6379
                            },
                            "PreferredAvailabilityZone": "us-west-2a",
                            "CurrentRole": "primary"
                        },
                        {
                            "CacheClusterId": "my-cluster-002",
                            "CacheNodeId": "0001",
                            "ReadEndpoint": {
                                "Address": "pat-
cluster-002.xxxxih.0001.usw2.cache.amazonaws.com",
                                "Port": 6379
                            },
                            "PreferredAvailabilityZone": "us-west-2a",
                            "CurrentRole": "replica"
                        },
                        {
                            "CacheClusterId": "my-cluster-003",
                            "CacheNodeId": "0001",
                            "ReadEndpoint": {
                                "Address": "pat-
cluster-003.xxxxih.0001.usw2.cache.amazonaws.com",
                                "Port": 6379
                            },
                            "PreferredAvailabilityZone": "us-west-2a",
                            "CurrentRole": "replica"
                        },
                        {
                            "CacheClusterId": "my-cluster-004",
                            "CacheNodeId": "0001",
                            "ReadEndpoint": {
                                "Address": "pat-
cluster-004.xxxih.0001.usw2.cache.amazonaws.com",
                                "Port": 6379
                            },
                            "PreferredAvailabilityZone": "us-west-2a",
                            "CurrentRole": "replica"
                        }
                    ]
                }
```

```
            ],
            "AutomaticFailover": "disabled",
            "SnapshotRetentionLimit": 0,
            "SnapshotWindow": "07:30-08:30",
            "ClusterEnabled": false,
            "CacheNodeType": "cache.r5.xlarge",
            "AuthTokenEnabled": false,
            "TransitEncryptionEnabled": false,
            "AtRestEncryptionEnabled": false,
            "ARN": "arn:aws:elasticache:us-
west-2:xxxxxxxxxxx152:replicationgroup:my-cluster",
            "LogDeliveryConfigurations": [
                {
                    "LogType": "slow-log",
                    "DestinationType": "cloudwatch-logs",
                    "DestinationDetails": {
                        "CloudWatchLogsDetails": {
                            "LogGroup": "test-log"
                        }
                    },
                    "LogFormat": "json",
                    "Status": "active"
                }
            ]
        }
    ]
}
```

Per ulteriori informazioni, consulta Gestione dei cluster nella Guida per l'utente di ElastiCache.

- Per i dettagli sull'API, consulta DescribeReplicationGroups AWS CLICommand Reference.

`describe-reserved-cache-nodes-offerings`

Il seguente esempio di codice mostra come utilizzare`describe-reserved-cache-nodes-offerings`.

AWS CLI

Descrivere reserved-cache-nodes-offerings

L'`describe-reserved-cache-nodes-offerings`esempio seguente restituisce i dettagli di un' reserved-cache-nodeopzione.

```
aws elasticache describe-reserved-cache-nodes-offerings
```

Output:

```
{
    "ReservedCacheNodesOfferings": [
        {
            "ReservedCacheNodesOfferingId": "01ce0a19-a476-41cb-8aee-48eacbcdc8e5",
            "CacheNodeType": "cache.t3.small",
            "Duration": 31536000,
            "FixedPrice": 97.0,
            "UsagePrice": 0.0,
            "ProductDescription": "memcached",
            "OfferingType": "Partial Upfront",
            "RecurringCharges": [
                {
                    "RecurringChargeAmount": 0.011,
                    "RecurringChargeFrequency": "Hourly"
                }
            ]
        },
        {
            "ReservedCacheNodesOfferingId": "0443a27b-4da5-4b90-b92d-929fbd7abed2",
            "CacheNodeType": "cache.m3.2xlarge",
            "Duration": 31536000,
            "FixedPrice": 1772.0,
            "UsagePrice": 0.0,
            "ProductDescription": "redis",
            "OfferingType": "Heavy Utilization",
            "RecurringCharges": [
                {
                    "RecurringChargeAmount": 0.25,
                    "RecurringChargeFrequency": "Hourly"
                }
            ]
        },

        ...

    ]
}
```

Per ulteriori informazioni, consulta Recupero di informazioni sulle offerte di nodi riservati nella Guida per l'utente di ElastiCache Redis o Recupero di informazioni sulle offerte di nodi riservati nella Guida per l'utente di ElastiCache Memcached.

- Per i dettagli sull'API, vedere DescribeReservedCacheNodesOfferingsin AWS CLI Command Reference.

`describe-reserved-cache-nodes`

Il seguente esempio di codice mostra come utilizzare`describe-reserved-cache-nodes`.

AWS CLI

Come descrivere i nodi di cache riservati

L'esempio `describe-reserved-cache-nodes` seguente restituisce le informazioni sui nodi di cache riservati per l'account oppure su un determinato nodo di cache riservato.

come elasticache describe-reserved-cache-nodes

Output:

```
{
    "ReservedCacheNodes": [
        {
            "ReservedCacheNodeId": "mynode",
            "ReservedCacheNodesOfferingId": "xxxxxxxxx-xxxxx-xxxxx-xxxx-xxxxxxxx71",
            "CacheNodeType": "cache.t3.small",
            "StartTime": "2019-12-06T02:50:44.003Z",
            "Duration": 31536000,
            "FixedPrice": 0.0,
            "UsagePrice": 0.0,
            "CacheNodeCount": 1,
            "ProductDescription": "redis",
            "OfferingType": "No Upfront",
            "State": "payment-pending",
            "RecurringCharges": [
                {
                    "RecurringChargeAmount": 0.023,
                    "RecurringChargeFrequency": "Hourly"
                }
            ],
```

```
            "ReservationARN": "arn:aws:elasticache:us-
west-2:xxxxxxxxxxx52:reserved-instance:mynode"
        }
    ]
}
```

Per ulteriori informazioni, consulta Gestione dei costi con nodi riservati nella Guida per l'utente di ElastiCache.

- Per i dettagli sull'API, consulta AWS CLI Command DescribeReservedCacheNodesReference.

`describe-service-updates`

Il seguente esempio di codice mostra come utilizzare`describe-service-updates`.

AWS CLI

Come descrivere gli aggiornamenti del servizio

L'esempio `describe-service-updates` seguente restituisce i dettagli relativi agli aggiornamenti del servizio.

```
aws elasticache describe-service-updates
```

Output:

```
{
     "ServiceUpdates": [
         {
             "ServiceUpdateName": "elc-xxxxxxxx7-001",
             "ServiceUpdateReleaseDate": "2019-10-09T16:00:00Z",
             "ServiceUpdateEndDate": "2020-02-09T15:59:59Z",
             "ServiceUpdateSeverity": "important",
             "ServiceUpdateRecommendedApplyByDate": "2019-11-08T15:59:59Z",
             "ServiceUpdateStatus": "available",
             "ServiceUpdateDescription": "Upgrades to improve the security,
 reliability, and operational performance of your ElastiCache nodes",
             "ServiceUpdateType": "security-update",
             "Engine": "redis, memcached",
             "EngineVersion": "redis 2.6.13 and onwards, memcached 1.4.5 and
 onwards",
             "AutoUpdateAfterRecommendedApplyByDate": false,
             "EstimatedUpdateTime": "30 minutes per node"
```

```
        },
        {
            "ServiceUpdateName": "elc-xxxxxxxx4-001",
            "ServiceUpdateReleaseDate": "2019-06-11T15:00:00Z",
            "ServiceUpdateEndDate": "2019-10-01T09:24:00Z",
            "ServiceUpdateSeverity": "important",
            "ServiceUpdateRecommendedApplyByDate": "2019-07-11T14:59:59Z",
            "ServiceUpdateStatus": "expired",
            "ServiceUpdateDescription": "Upgrades to improve the security,
reliability, and operational performance of your ElastiCache nodes",
            "ServiceUpdateType": "security-update",
            "Engine": "redis",
            "EngineVersion": "redis 3.2.6, redis 4.0 and onwards",
            "AutoUpdateAfterRecommendedApplyByDate": false,
            "EstimatedUpdateTime": "30 minutes per node"
        }
    ]
}
```

- Per i dettagli sull'API, consulta DescribeServiceUpdates AWS CLICommand Reference.

`describe-snapshots`

Il seguente esempio di codice mostra come utilizzare`describe-snapshots`.

AWS CLI

Come descrivere gli snapshot

L'esempio "describe-snapshots" seguente restituisce le informazioni relative agli snapshot del cluster o del gruppo di replica.

```
aws elasticache describe-snapshots
```

Output:

```
{
    "Snapshots": [
        {
            "SnapshotName": "automatic.my-cluster2-002-2019-12-05-06-38",
            "CacheClusterId": "my-cluster2-002",
            "SnapshotStatus": "available",
```

```
            "SnapshotSource": "automated",
            "CacheNodeType": "cache.r5.large",
            "Engine": "redis",
            "EngineVersion": "5.0.5",
            "NumCacheNodes": 1,
            "PreferredAvailabilityZone": "us-west-2a",
            "CacheClusterCreateTime": "2019-11-26T01:22:52.396Z",
            "PreferredMaintenanceWindow": "mon:17:30-mon:18:30",
            "TopicArn": "arn:aws:sns:us-west-2:xxxxxxxxx52:My_Topic",
            "Port": 6379,
            "CacheParameterGroupName": "default.redis5.0",
            "CacheSubnetGroupName": "kxkxk",
            "VpcId": "vpc-a3e97cdb",
            "AutoMinorVersionUpgrade": true,
            "SnapshotRetentionLimit": 1,
            "SnapshotWindow": "06:30-07:30",
            "NodeSnapshots": [
                {
                    "CacheNodeId": "0001",
                    "CacheSize": "5 MB",
                    "CacheNodeCreateTime": "2019-11-26T01:22:52.396Z",
                    "SnapshotCreateTime": "2019-12-05T06:38:23Z"
                }
            ]
        },
        {
            "SnapshotName": "myreplica-backup",
            "CacheClusterId": "myreplica",
            "SnapshotStatus": "available",
            "SnapshotSource": "manual",
            "CacheNodeType": "cache.r5.large",
            "Engine": "redis",
            "EngineVersion": "5.0.5",
            "NumCacheNodes": 1,
            "PreferredAvailabilityZone": "us-west-2a",
            "CacheClusterCreateTime": "2019-11-26T00:14:52.439Z",
            "PreferredMaintenanceWindow": "sat:10:00-sat:11:00",
            "TopicArn": "arn:aws:sns:us-west-2:xxxxxxxxxx152:My_Topic",
            "Port": 6379,
            "CacheParameterGroupName": "default.redis5.0",
            "CacheSubnetGroupName": "kxkxk",
            "VpcId": "vpc-a3e97cdb",
            "AutoMinorVersionUpgrade": true,
            "SnapshotRetentionLimit": 0,
```

```
            "SnapshotWindow": "09:00-10:00",
            "NodeSnapshots": [
                {
                    "CacheNodeId": "0001",
                    "CacheSize": "5 MB",
                    "CacheNodeCreateTime": "2019-11-26T00:14:52.439Z",
                    "SnapshotCreateTime": "2019-11-26T00:25:01Z"
                }
            ]
        },
        {
            "SnapshotName": "my-cluster",
            "CacheClusterId": "my-cluster-003",
            "SnapshotStatus": "available",
            "SnapshotSource": "manual",
            "CacheNodeType": "cache.r5.large",
            "Engine": "redis",
            "EngineVersion": "5.0.5",
            "NumCacheNodes": 1,
            "PreferredAvailabilityZone": "us-west-2a",
            "CacheClusterCreateTime": "2019-11-25T23:56:17.186Z",
            "PreferredMaintenanceWindow": "sat:10:00-sat:11:00",
            "TopicArn": "arn:aws:sns:us-west-2:xxxxxxxxxx152:My_Topic",
            "Port": 6379,
            "CacheParameterGroupName": "default.redis5.0",
            "CacheSubnetGroupName": "kxkxk",
            "VpcId": "vpc-a3e97cdb",
            "AutoMinorVersionUpgrade": true,
            "SnapshotRetentionLimit": 0,
            "SnapshotWindow": "09:00-10:00",
            "NodeSnapshots": [
                {
                    "CacheNodeId": "0001",
                    "CacheSize": "5 MB",
                    "CacheNodeCreateTime": "2019-11-25T23:56:17.186Z",
                    "SnapshotCreateTime": "2019-11-26T03:08:33Z"
                }
            ]
        }
    ]
}
```

Per ulteriori informazioni, consulta Backup and Restore ElastiCache for Redis nella Guida per l'utente di Elasticache.

- Per i dettagli sull'API, consulta Command DescribeSnapshotsReference AWS CLI .

`describe-update-actions`

Il seguente esempio di codice mostra come utilizzare`describe-update-actions`.

AWS CLI

Come descrivere le azioni di aggiornamento

L'esempio `describe-update-actions` seguente restituisce i dettagli relativi alle azioni di aggiornamento.

```
aws elasticache describe-update-actions
```

Output:

```
{
    "UpdateActions": [
        {
            "ReplicationGroupId": "mycluster",
            "ServiceUpdateName": "elc-20191007-001",
            "ServiceUpdateReleaseDate": "2019-10-09T16:00:00Z",
            "ServiceUpdateSeverity": "important",
            "ServiceUpdateStatus": "available",
            "ServiceUpdateRecommendedApplyByDate": "2019-11-08T15:59:59Z",
            "ServiceUpdateType": "security-update",
            "UpdateActionAvailableDate": "2019-12-05T19:15:19.995Z",
            "UpdateActionStatus": "complete",
            "NodesUpdated": "9/9",
            "UpdateActionStatusModifiedDate": "2019-12-05T19:15:20.461Z",
            "SlaMet": "n/a",
            "Engine": "redis"
        },
        {
            "CacheClusterId": "my-memcached-cluster",
            "ServiceUpdateName": "elc-20191007-001",
            "ServiceUpdateReleaseDate": "2019-10-09T16:00:00Z",
            "ServiceUpdateSeverity": "important",
            "ServiceUpdateStatus": "available",
```

```
            "ServiceUpdateRecommendedApplyByDate": "2019-11-08T15:59:59Z",
            "ServiceUpdateType": "security-update",
            "UpdateActionAvailableDate": "2019-12-04T18:26:05.349Z",
            "UpdateActionStatus": "complete",
            "NodesUpdated": "1/1",
            "UpdateActionStatusModifiedDate": "2019-12-04T18:26:05.352Z",
            "SlaMet": "n/a",
            "Engine": "redis"
        },
        {
            "ReplicationGroupId": "my-cluster",
            "ServiceUpdateName": "elc-20191007-001",
            "ServiceUpdateReleaseDate": "2019-10-09T16:00:00Z",
            "ServiceUpdateSeverity": "important",
            "ServiceUpdateStatus": "available",
            "ServiceUpdateRecommendedApplyByDate": "2019-11-08T15:59:59Z",
            "ServiceUpdateType": "security-update",
            "UpdateActionAvailableDate": "2019-11-26T03:36:26.320Z",
            "UpdateActionStatus": "complete",
            "NodesUpdated": "4/4",
            "UpdateActionStatusModifiedDate": "2019-12-04T22:11:12.664Z",
            "SlaMet": "n/a",
            "Engine": "redis"
        },
        {
            "ReplicationGroupId": "my-cluster2",
            "ServiceUpdateName": "elc-20191007-001",
            "ServiceUpdateReleaseDate": "2019-10-09T16:00:00Z",
            "ServiceUpdateSeverity": "important",
            "ServiceUpdateStatus": "available",
            "ServiceUpdateRecommendedApplyByDate": "2019-11-08T15:59:59Z",
            "ServiceUpdateType": "security-update",
            "UpdateActionAvailableDate": "2019-11-26T01:26:01.617Z",
            "UpdateActionStatus": "complete",
            "NodesUpdated": "3/3",
            "UpdateActionStatusModifiedDate": "2019-11-26T01:26:01.753Z",
            "SlaMet": "n/a",
            "Engine": "redis"
        }
    ]
}
```

Per ulteriori informazioni, consulta Self-Service Updates in Amazon ElastiCache nella Elasticache User Guide.

- Per i dettagli sull'API, consulta Command DescribeUpdateActionsReference AWS CLI .

`describe-user-groups`

Il seguente esempio di codice mostra come utilizzare`describe-user-groups`.

AWS CLI

Come descrivere i gruppi di utenti

L'esempio `describe-user-groups` seguente restituisce un elenco di gruppi di utenti.

```
aws elasticache describe-user-groups
```

Output:

```
{
    "UserGroups": [
        {
            "UserGroupId": "myusergroup",
            "Status": "active",
            "Engine": "redis",
            "UserIds": [
                "default"
            ],
            "ReplicationGroups": [],
            "ARN": "arn:aws:elasticache:us-
west-2:xxxxxxxxxx52:usergroup:myusergroup"
        }
    ]
}
```

Per ulteriori informazioni, consulta Autenticazione degli utenti con il controllo degli accessi basato sui ruoli (RBAC) nella Guida per l'utente di ElastiCache.

- Per i dettagli sull'API, consulta DescribeUserGroups AWS CLICommand Reference.

`describe-users`

Il seguente esempio di codice mostra come utilizzare`describe-users`.

AWS CLI

Come descrivere gli utenti

L'esempio `describe-users` seguente restituisce un elenco di utenti.

```
aws elasticache describe-users
```

Output:

```
{
    "Users": [
        {
            "UserId": "default",
            "UserName": "default",
            "Status": "active",
            "Engine": "redis",
            "AccessString": "on ~* +@all",
            "UserGroupIds": [
                "myusergroup"
            ],
            "Authentication": {
                "Type": "no-password"
            },
            "ARN": "arn:aws:elasticache:us-west-2:xxxxxxxxxx52:user:default"
        },
        {
            "UserId": "user1",
            "UserName": "myUser",
            "Status": "active",
            "Engine": "redis",
            "AccessString": "on ~* +@all",
            "UserGroupIds": [],
            "Authentication": {
                "Type": "password",
                "PasswordCount": 1
            },
            "ARN": "arn:aws:elasticache:us-west-2:xxxxxxxxxx52:user:user1"
        },
        {
```

```
            "UserId": "user2",
            "UserName": "myUser",
            "Status": "active",
            "Engine": "redis",
            "AccessString": "on ~app::* -@all +@read +@hash +@bitmap +@geo -setbit -
bitfield -hset -hsetnx -hmset -hincrby -hincrbyfloat -hdel -bitop -geoadd -georadius
 -georadiusbymember",
            "UserGroupIds": [],
            "Authentication": {
                "Type": "password",
                "PasswordCount": 1
            },
            "ARN": "arn:aws:elasticache:us-west-2:xxxxxxxxxx52:user:user2"
        }
    ]
}
```

Per ulteriori informazioni, consulta Autenticazione degli utenti con il controllo degli accessi basato sui ruoli (RBAC) nella Guida per l'utente di ElastiCache.

- Per i dettagli sull'API, consulta DescribeUsers AWS CLICommand Reference.

`disassociate-global-replication-group`

Il seguente esempio di codice mostra come utilizzare`disassociate-global-replication-group`.

AWS CLI

Come dissociare un cluster secondario da un gruppo di replica globale

L'esempio `disassociate-global-replication-group` seguente rimuove un cluster secondario da un datastore globale.

```
aws elasticache disassociate-global-replication-group \
    --global-replication-group-id my-grg \
    --replication-group-id my-cluster-grg-secondary \
    --replication-group-region us-east-1
```

Output:

```
{
    "GlobalReplicationGroup": {
```

```
        "GlobalReplicationGroupId": "my-grg",
        "GlobalReplicationGroupDescription": "my-grg",
        "Status": "modifying",
        "CacheNodeType": "cache.r5.large",
        "Engine": "redis",
        "EngineVersion": "5.0.6",
        "Members": [
            {
                "ReplicationGroupId": "my-cluster-grg-secondary",
                "ReplicationGroupRegion": "us-east-1",
                "Role": "SECONDARY",
                "AutomaticFailover": "enabled",
                "Status": "associated"
            },
            {
                "ReplicationGroupId": "my-cluster-grg",
                "ReplicationGroupRegion": "us-west-2",
                "Role": "PRIMARY",
                "AutomaticFailover": "enabled",
                "Status": "associated"
            }
        ],
        "ClusterEnabled": false,
        "AuthTokenEnabled": false,
        "TransitEncryptionEnabled": false,
        "AtRestEncryptionEnabled": false
    }
}
```

Per ulteriori informazioni, consulta Replica tra AWS regioni utilizzando Global Datastore nella Guida per l'utente di Elasticache.

- Per i dettagli sull'API, consulta Command Reference. DisassociateGlobalReplicationGroupAWS CLI

`increase-node-groups-in-global-replication-group`

Il seguente esempio di codice mostra come utilizzare`increase-node-groups-in-global-replication-group`.

AWS CLI

Come aumentare il numero dei gruppi di nodi in un gruppo di replica globale

L'esempio `increase-node-groups-in-global-replication-group` seguente aumenta il numero dei gruppi di nodi utilizzando il motore Redis.

```
aws elasticache increase-node-groups-in-global-replication-group \
    --global-replication-group-id sgaui-pat-test-4 \
    --node-group-count 6 \
    --apply-immediately
```

Output:

```
{
    "GlobalReplicationGroup": {
        "GlobalReplicationGroupId": "sgaui-test-4",
        "GlobalReplicationGroupDescription": "test-4",
        "Status": "modifying",
        "CacheNodeType": "cache.r5.large",
        "Engine": "redis",
        "EngineVersion": "5.0.6",
        "Members": [
            {
                "ReplicationGroupId": "my-cluster-b",
                "ReplicationGroupRegion": "us-east-1",
                "Role": "SECONDARY",
                "AutomaticFailover": "enabled",
                "Status": "associated"
            },
            {
                "ReplicationGroupId": "my-cluster-a",
                "ReplicationGroupRegion": "us-west-2",
                "Role": "PRIMARY",
                "AutomaticFailover": "enabled",
                "Status": "associated"
            }
        ],
        "ClusterEnabled": true,
        "GlobalNodeGroups": [
            {
                "GlobalNodeGroupId": "sgaui-test-4-0001",
                "Slots": "0-234,2420-5461"
            },
            {
                "GlobalNodeGroupId": "sgaui-test-4-0002",
                "Slots": "5462-5904,6997-9830"
```

```
            },
            {
                "GlobalNodeGroupId": "sgaui-test-4-0003",
                "Slots": "10923-11190,13375-16383"
            },
            {
                "GlobalNodeGroupId": "sgaui-test-4-0004",
                "Slots": "235-2419,5905-6996"
            },
            {
                "GlobalNodeGroupId": "sgaui-test-4-0005",
                "Slots": "9831-10922,11191-13374"
            }
        ],
        "AuthTokenEnabled": false,
        "TransitEncryptionEnabled": false,
        "AtRestEncryptionEnabled": false
    }
}
```

Per ulteriori informazioni, consulta Replica tra AWS regioni utilizzando Global Datastore nella Guida per l'utente di Elasticache.

- Per i dettagli sull'API, consulta Command Reference. IncreaseNodeGroupsInGlobalReplicationGroupAWS CLI

`increase-replica-count`

Il seguente esempio di codice mostra come utilizzare`increase-replica-count`.

AWS CLI

Come aumentare il numero di repliche

L'esempio `increase-replica-count` seguente esegue una di queste due operazioni. Può aumentare dinamicamente il numero di repliche in un gruppo di replica Redis (modalità cluster disabilitata). In alternativa, può aumentare dinamicamente il numero di nodi di replica in uno o più gruppi di nodi (shard) di un gruppo di replica Redis (modalità cluster abilitata). Questa operazione viene eseguita senza causare alcun tempo di inattività del cluster.

```
aws elasticache increase-replica-count \
    --replication-group-id "my-cluster" \
```

```
    --apply-immediately \
    --new-replica-count 3
```

Output:

```
{
     "ReplicationGroup": {
         "ReplicationGroupId": "my-cluster",
         "Description": " ",
         "Status": "modifying",
         "PendingModifiedValues": {},
         "MemberClusters": [
             "my-cluster-001",
             "my-cluster-002",
             "my-cluster-003",
             "my-cluster-004"
         ],
         "NodeGroups": [
             {
                 "NodeGroupId": "0001",
                 "Status": "modifying",
                 "PrimaryEndpoint": {
                     "Address": "my-
cluster.xxxxxih.ng.0001.usw2.cache.amazonaws.com",
                     "Port": 6379
                 },
                 "ReaderEndpoint": {
                     "Address": "my-cluster-
ro.xxxxxxih.ng.0001.usw2.cache.amazonaws.com",
                     "Port": 6379
                 },
                 "NodeGroupMembers": [
                     {
                         "CacheClusterId": "my-cluster-001",
                         "CacheNodeId": "0001",
                         "ReadEndpoint": {
                             "Address": "my-
cluster-001.xxxxxih.0001.usw2.cache.amazonaws.com",
                             "Port": 6379
                         },
                         "PreferredAvailabilityZone": "us-west-2a",
                         "CurrentRole": "primary"
                     },
```

```
                    {
                        "CacheClusterId": "my-cluster-003",
                        "CacheNodeId": "0001",
                        "ReadEndpoint": {
                            "Address": "my-
cluster-003.xxxxxih.0001.usw2.cache.amazonaws.com",
                            "Port": 6379
                        },
                        "PreferredAvailabilityZone": "us-west-2a",
                        "CurrentRole": "replica"
                    }
                ]
            }
        ],
        "AutomaticFailover": "disabled",
        "SnapshotRetentionLimit": 0,
        "SnapshotWindow": "07:30-08:30",
        "ClusterEnabled": false,
        "CacheNodeType": "cache.r5.xlarge",
        "TransitEncryptionEnabled": false,
        "AtRestEncryptionEnabled": false
    }
}
```

Per ulteriori informazioni, consulta Incremento del numero di repliche in uno shard nella Guida per l'utente di ElastiCache.

- Per i dettagli sull'API, consulta IncreaseReplicaCount AWS CLICommand Reference.

`list-allowed-node-type-modifications`

Il seguente esempio di codice mostra come utilizzare`list-allowed-node-type-modifications`.

AWS CLI

Come elencare le modifiche ai nodi consentite

L'esempio `list-allowed-node-type-modifications` seguente elenca tutti i tipi di nodi disponibili su cui è possibile scalare il tipo di nodo corrente del gruppo di replica o del cluster Redis.

```
aws elasticache list-allowed-node-type-modifications \
```

```
    --replication-group-id "my-replication-group"
```

Output:

```
{
    "ScaleUpModifications": [
        "cache.m5.12xlarge",
        "cache.m5.24xlarge",
        "cache.m5.4xlarge",
        "cache.r5.12xlarge",
        "cache.r5.24xlarge",
        "cache.r5.2xlarge",
        "cache.r5.4xlarge"
    ],
    "ScaleDownModifications": [
        "cache.m3.large",
        "cache.m3.medium",
        "cache.m3.xlarge",
        "cache.m4.large",
        "cache.m4.xlarge",
        "cache.m5.2xlarge",
        "cache.m5.large",
        "cache.m5.xlarge",
        "cache.r3.large",
        "cache.r4.large",
        "cache.r4.xlarge",
        "cache.r5.large",
        "cache.t2.medium",
        "cache.t2.micro",
        "cache.t2.small",
        "cache.t3.medium",
        "cache.t3.micro",
        "cache.t3.small"
    ]
}
```

Per ulteriori informazioni, consulta Scaling ElastiCache for Redis Clusters nella Guida per l'utente di Elasticache.

- Per i dettagli sull'API, consulta Command Reference. ListAllowedNodeTypeModificationsAWS CLI

`list-tags-for-resource`

Il seguente esempio di codice mostra come utilizzare`list-tags-for-resource`.

AWS CLI

Come elencare i tag per una risorsa

L'esempio `list-tags-for-resource` seguente elenca i tag per una risorsa.

```
aws elasticache list-tags-for-resource \
    --resource-name "arn:aws:elasticache:us-east-1:123456789012:cluster:my-cluster"
```

Output:

```
{
    "TagList": [
        {
            "Key": "Project",
            "Value": "querySpeedUp"
        },
        {
            "Key": "Environment",
            "Value": "PROD"
        }
    ]
}
```

Per ulteriori informazioni, consulta Elencare i tag utilizzando la AWS CLI nella Guida per l'utente di Elasticache.

- Per i dettagli sull'API, consulta Command ListTagsForResourceReference AWS CLI .

`modify-cache-cluster`

Il seguente esempio di codice mostra come utilizzare`modify-cache-cluster`.

AWS CLI

Come modificare i cluster di cache

L'esempio `modify-cache-cluster` seguente modifica le impostazioni per il cluster specificato.

```
aws elasticache modify-cache-cluster \
    --cache-cluster-id "my-cluster" \
    --num-cache-nodes 1
```

Output:

```
{
    "CacheCluster": {
        "CacheClusterId": "my-cluster",
        "ClientDownloadLandingPage": "https://console.aws.amazon.com/elasticache/
home#client-download:",
        "CacheNodeType": "cache.m5.large",
        "Engine": "redis",
        "EngineVersion": "5.0.5",
        "CacheClusterStatus": "available",
        "NumCacheNodes": 1,
        "PreferredAvailabilityZone": "us-west-2c",
        "CacheClusterCreateTime": "2019-12-04T18:24:56.652Z",
        "PreferredMaintenanceWindow": "sat:10:00-sat:11:00",
        "PendingModifiedValues": {},
        "CacheSecurityGroups": [],
        "CacheParameterGroup": {
            "CacheParameterGroupName": "default.redis5.0",
            "ParameterApplyStatus": "in-sync",
            "CacheNodeIdsToReboot": []
        },
        "CacheSubnetGroupName": "default",
        "AutoMinorVersionUpgrade": true,
        "SnapshotRetentionLimit": 0,
        "SnapshotWindow": "07:00-08:00",
        "TransitEncryptionEnabled": false,
        "AtRestEncryptionEnabled": false
    }
}
```

Per ulteriori informazioni, consulta Modificare un ElastiCache cluster nella Guida per l'utente di Elasticache.

- Per i dettagli sull'API, consulta Command ModifyCacheClusterReference AWS CLI .

`modify-cache-parameter-group`

Il seguente esempio di codice mostra come utilizzare`modify-cache-parameter-group`.

AWS CLI

Come modificare un gruppo di parametri cache

L'esempio `modify-cache-parameter-group` seguente modifica i parametri del gruppo di parametri cache specificato.

```
aws elasticache modify-cache-parameter-group \
    --cache-parameter-group-name "mygroup" \
    --parameter-name-values "ParameterName=activedefrag, ParameterValue=no"
```

Output:

```
{
    "CacheParameterGroupName": "mygroup"
}
```

Per ulteriori informazioni, consulta Modifica di un gruppo di parametri nella Guida per l'utente di ElastiCache.

- Per i dettagli sull'API, consulta ModifyCacheParameterGroup AWS CLICommand Reference.

`modify-cache-subnet-group`

Il seguente esempio di codice mostra come utilizzare`modify-cache-subnet-group`.

AWS CLI

Come modificare un gruppo di sottoreti della cache

L'esempio `modify-cache-subnet-group` seguente modifica il gruppo di sottoreti della cache specificato.

```
aws elasticache modify-cache-subnet-group \
    --cache-subnet-group-name kxkxk \
    --cache-subnet-group-description "mygroup"
```

Output:

```
{
    "CacheSubnetGroup": {
        "CacheSubnetGroupName": "kxkxk",
        "CacheSubnetGroupDescription": "mygroup",
        "VpcId": "vpc-xxxxcdb",
        "Subnets": [
            {
                "SubnetIdentifier": "subnet-xxxxbff",
                "SubnetAvailabilityZone": {
                    "Name": "us-west-2a"
                }
            }
        ]
    }
}
```

Per ulteriori informazioni, consulta Modifica di un gruppo di sottoreti nella Guida per l'utente di ElastiCache.

- Per i dettagli sull'API, consulta ModifyCacheSubnetGroup AWS CLICommand Reference.

`modify-global-replication-group`

Il seguente esempio di codice mostra come utilizzare`modify-global-replication-group`.

AWS CLI

Come modificare un gruppo di replica globale

L'esempio `modify-global-replication-group` seguente modifica le proprietà di un gruppo di replica globale, con disabilitazione del failover automatico, utilizzando il motore Redis.

```
aws elasticache modify-global-replication-group \
    --global-replication-group-id sgaui-pat-group \
    --apply-immediately \
    --no-automatic-failover-enabled
```

Output

```
{
    "GlobalReplicationGroup": {
        "GlobalReplicationGroupId": "sgaui-test-group",
```

```
        "GlobalReplicationGroupDescription": " ",
        "Status": "modifying",
        "CacheNodeType": "cache.r5.large",
        "Engine": "redis",
        "EngineVersion": "5.0.6",
        "ClusterEnabled": false,
        "AuthTokenEnabled": false,
        "TransitEncryptionEnabled": false,
        "AtRestEncryptionEnabled": false
    }
}
```

Per ulteriori informazioni, consulta Replica tra AWS regioni utilizzando Global Datastore nella Guida per l'utente di Elasticache.

- Per i dettagli sull'API, consulta Command Reference. ModifyGlobalReplicationGroupAWS CLI

`modify-replication-group-shard-configuration`

Il seguente esempio di codice mostra come utilizzare`modify-replication-group-shard-configuration`.

AWS CLI

Come modificare la configurazione di uno shard di un gruppo di replica

Il comando `modify-replication-group-shard-configuration` seguente riduce il numero dei gruppi di nodi utilizzando il motore Redis.

```
aws elasticache modify-replication-group-shard-configuration \
    --replication-group-id mycluster \
    --node-group-count 3 \
    --apply-immediately \
    --node-groups-to-remove 0002
```

Output

```
{
    "ReplicationGroup": {
        "ReplicationGroupId": "mycluster",
        "Description": "mycluster",
        "GlobalReplicationGroupInfo": {},
        "Status": "modifying",
```

```
        "PendingModifiedValues": {},
        "MemberClusters": [
            "mycluster-0002-001",
            "mycluster-0002-002",
            "mycluster-0002-003",
            "mycluster-0003-001",
            "mycluster-0003-002",
            "mycluster-0003-003",
            "mycluster-0003-004",
            "mycluster-0004-001",
            "mycluster-0004-002",
            "mycluster-0004-003",
            "mycluster-0005-001",
            "mycluster-0005-002",
            "mycluster-0005-003"
        ],
        "NodeGroups": [
            {
                "NodeGroupId": "0002",
                "Status": "modifying",
                "Slots": "894-1767,3134-4443,5149-5461,6827-7332,12570-13662",
                "NodeGroupMembers": [
                    {
                        "CacheClusterId": "mycluster-0002-001",
                        "CacheNodeId": "0001",
                        "PreferredAvailabilityZone": "us-west-2c"
                    },
                    {
                        "CacheClusterId": "mycluster-0002-002",
                        "CacheNodeId": "0001",
                        "PreferredAvailabilityZone": "us-west-2a"
                    },
                    {
                        "CacheClusterId": "mycluster-0002-003",
                        "CacheNodeId": "0001",
                        "PreferredAvailabilityZone": "us-west-2b"
                    }
                ]
            },
            {
                "NodeGroupId": "0003",
                "Status": "modifying",
                "Slots":
 "0-324,5462-5692,6784-6826,7698-8191,10923-11075,12441-12569,13663-16383",
```

```
                "NodeGroupMembers": [
                    {
                        "CacheClusterId": "mycluster-0003-001",
                        "CacheNodeId": "0001",
                        "PreferredAvailabilityZone": "us-west-2c"
                    },
                    {
                        "CacheClusterId": "mycluster-0003-002",
                        "CacheNodeId": "0001",
                        "PreferredAvailabilityZone": "us-west-2b"
                    },
                    {
                        "CacheClusterId": "mycluster-0003-003",
                        "CacheNodeId": "0001",
                        "PreferredAvailabilityZone": "us-west-2a"
                    },
                    {
                        "CacheClusterId": "mycluster-0003-004",
                        "CacheNodeId": "0001",
                        "PreferredAvailabilityZone": "us-west-2c"
                    }
                ]
            },
            {
                "NodeGroupId": "0004",
                "Status": "modifying",
                "Slots": "325-336,4706-5148,7333-7697,9012-10922,11076-12440",
                "NodeGroupMembers": [
                    {
                        "CacheClusterId": "mycluster-0004-001",
                        "CacheNodeId": "0001",
                        "PreferredAvailabilityZone": "us-west-2b"
                    },
                    {
                        "CacheClusterId": "mycluster-0004-002",
                        "CacheNodeId": "0001",
                        "PreferredAvailabilityZone": "us-west-2a"
                    },
                    {
                        "CacheClusterId": "mycluster-0004-003",
                        "CacheNodeId": "0001",
                        "PreferredAvailabilityZone": "us-west-2c"
                    }
                ]
```

```
            },
            {
                "NodeGroupId": "0005",
                "Status": "modifying",
                "Slots": "337-893,1768-3133,4444-4705,5693-6783,8192-9011",
                "NodeGroupMembers": [
                    {
                        "CacheClusterId": "mycluster-0005-001",
                        "CacheNodeId": "0001",
                        "PreferredAvailabilityZone": "us-west-2a"
                    },
                    {
                        "CacheClusterId": "mycluster-0005-002",
                        "CacheNodeId": "0001",
                        "PreferredAvailabilityZone": "us-west-2c"
                    },
                    {
                        "CacheClusterId": "mycluster-0005-003",
                        "CacheNodeId": "0001",
                        "PreferredAvailabilityZone": "us-west-2b"
                    }
                ]
            }
        ],
        "AutomaticFailover": "enabled",
        "MultiAZ": "enabled",
        "ConfigurationEndpoint": {
            "Address": "mycluster.g2xbih.clustercfg.usw2.cache.amazonaws.com",
            "Port": 6379
        },
        "SnapshotRetentionLimit": 1,
        "SnapshotWindow": "13:00-14:00",
        "ClusterEnabled": true,
        "CacheNodeType": "cache.r5.xlarge",
        "TransitEncryptionEnabled": false,
        "AtRestEncryptionEnabled": false
    }
}
```

Per ulteriori informazioni, consulta Scaling ElastiCache for Redis Clusters nella Guida per l'utente di Elasticache.

- Per i dettagli sull'API, consulta Command Reference. ModifyReplicationGroupShardConfigurationAWS CLI

`modify-replication-group`

Il seguente esempio di codice mostra come utilizzare`modify-replication-group`.

AWS CLI

Come modificare un gruppo di replica

Il comando `modify-replication-group` seguente disabilita Multi-AZ utilizzando il motore Redis.

```
aws elasticache modify-replication-group \
    --replication-group-id test-cluster \
    --no-multi-az-enabled \
    --apply-immediately
```

Output

```
{
    "ReplicationGroup": {
        "ReplicationGroupId": "test-cluster",
        "Description": "test-cluster",
        "GlobalReplicationGroupInfo": {
            "GlobalReplicationGroupId": "sgaui-pat-group",
            "GlobalReplicationGroupMemberRole": "PRIMARY"
        },
        "Status": "available",
        "PendingModifiedValues": {},
        "MemberClusters": [
            "test-cluster-001",
            "test-cluster-002",
            "test-cluster-003"
        ],
        "NodeGroups": [
            {
                "NodeGroupId": "0001",
                "Status": "available",
                "PrimaryEndpoint": {
                    "Address": "test-
cluster.g2xbih.ng.0001.usw2.cache.amazonaws.com",
                    "Port": 6379
                },
                "ReaderEndpoint": {
```

```
                    "Address": "test-cluster-
ro.g2xbih.ng.0001.usw2.cache.amazonaws.com",
                    "Port": 6379
                },
                "NodeGroupMembers": [
                    {
                        "CacheClusterId": "test-cluster-001",
                        "CacheNodeId": "0001",
                        "ReadEndpoint": {
                            "Address": "test-
cluster-001.g2xbih.0001.usw2.cache.amazonaws.com",
                            "Port": 6379
                        },
                        "PreferredAvailabilityZone": "us-west-2c",
                        "CurrentRole": "primary"
                    },
                    {
                        "CacheClusterId": "test-cluster-002",
                        "CacheNodeId": "0001",
                        "ReadEndpoint": {
                            "Address": "test-
cluster-002.g2xbih.0001.usw2.cache.amazonaws.com",
                            "Port": 6379
                        },
                        "PreferredAvailabilityZone": "us-west-2b",
                        "CurrentRole": "replica"
                    },
                    {
                        "CacheClusterId": "test-cluster-003",
                        "CacheNodeId": "0001",
                        "ReadEndpoint": {
                            "Address": "test-
cluster-003.g2xbih.0001.usw2.cache.amazonaws.com",
                            "Port": 6379
                        },
                        "PreferredAvailabilityZone": "us-west-2a",
                        "CurrentRole": "replica"
                    }
                ]
            }
        ],
        "SnapshottingClusterId": "test-cluster-002",
        "AutomaticFailover": "enabled",
        "MultiAZ": "disabled",
```

```
        "SnapshotRetentionLimit": 1,
        "SnapshotWindow": "08:00-09:00",
        "ClusterEnabled": false,
        "CacheNodeType": "cache.r5.large",
        "TransitEncryptionEnabled": false,
        "AtRestEncryptionEnabled": false
    }
}
```

Per ulteriori informazioni, consulta Modifica di un gruppo di replica nella Guida per l'utente di ElastiCache.

- Per i dettagli sull'API, consulta ModifyReplicationGroup AWS CLICommand Reference.

`modify-user-group`

Il seguente esempio di codice mostra come utilizzare`modify-user-group`.

AWS CLI

Come modificare un gruppo di utenti

L'esempio `modify-user-group` seguente aggiunge un utente al gruppo di utenti.

```
aws elasticache modify-user-group \
    --user-group-id myusergroup \
    --user-ids-to-add user1
```

Output:

```
{
    "UserGroupId": "myusergroup",
    "Status": "modifying",
    "Engine": "redis",
    "UserIds": [
        "default"
    ],
    "PendingChanges": {
        "UserIdsToAdd": [
            "user1"
        ]
    },
    "ReplicationGroups": [],
```

```
    "ARN": "arn:aws:elasticache:us-west-2:xxxxxxxxxx52:usergroup:myusergroup"
}
```

Per ulteriori informazioni, consulta Autenticazione degli utenti con il controllo degli accessi basato sui ruoli (RBAC) nella Guida per l'utente di ElastiCache.

- Per i dettagli sull'API, consulta ModifyUserGroup AWS CLICommand Reference.

`modify-user`

Il seguente esempio di codice mostra come utilizzare`modify-user`.

AWS CLI

Come modificare un utente

L'esempio `modify-user` seguente modifica la stringa di accesso di un utente.

```
aws elasticache modify-user \
    --user-id user2 \
    --append-access-string "on ~* +@all"
```

Output:

```
{
    "UserId": "user2",
    "UserName": "myUser",
    "Status": "modifying",
    "Engine": "redis",
    "AccessString": "on ~* +@all",
    "UserGroupIds": [],
    "Authentication": {
        "Type": "password",
        "PasswordCount": 1
    },
    "ARN": "arn:aws:elasticache:us-west-2:xxxxxxxxxx52:user:user2"
}
```

Per ulteriori informazioni, consulta Autenticazione degli utenti con il controllo degli accessi basato sui ruoli (RBAC) nella Guida per l'utente di ElastiCache.

- Per i dettagli sull'API, consulta ModifyUser AWS CLICommand Reference.

`purchase-reserved-cache-nodes-offering`

Il seguente esempio di codice mostra come utilizzare`purchase-reserved-cache-nodes-offering`.

AWS CLI

Per acquistare un reserved-cache-node-offering

L'esempio `purchase-reserved-cache-nodes-offering` seguente consente di acquistare un'offerta di nodi di cache riservati.

```
aws elasticache purchase-reserved-cache-nodes-offering \
    --reserved-cache-nodes-offering-id xxxxxxx-4da5-4b90-b92d-929fbd7abed2
```

Output

```
{
    "ReservedCacheNode": {
        "ReservedCacheNodeId": "ri-2020-06-30-17-59-40-474",
        "ReservedCacheNodesOfferingId": "xxxxxxx-4da5-4b90-b92d-929fbd7abed2",
        "CacheNodeType": "cache.m3.2xlarge",
        "StartTime": "2020-06-30T17:59:40.474000+00:00",
        "Duration": 31536000,
        "FixedPrice": 1772.0,
        "UsagePrice": 0.0,
        "CacheNodeCount": 1,
        "ProductDescription": "redis",
        "OfferingType": "Heavy Utilization",
        "State": "payment-pending",
        "RecurringCharges": [
            {
                "RecurringChargeAmount": 0.25,
                "RecurringChargeFrequency": "Hourly"
            }
        ]
    }
}
```

Per ulteriori informazioni, consulta Recupero di informazioni sulle offerte di nodi riservati nella Guida per l'utente di ElastiCache Redis o Recupero di informazioni sulle offerte di nodi riservati nella Guida per l'utente di ElastiCache Memcached.

- Per i dettagli sull'API, consulta PurchaseReservedCacheNodesOffering AWS CLICommand Reference.

`reboot-cache-cluster`

Il seguente esempio di codice mostra come utilizzare`reboot-cache-cluster`.

AWS CLI

Come riavviare un cluster di cache

L'esempio `reboot-cache-cluster` seguente riavvia alcuni o tutti i nodi di cache all'interno di un cluster sottoposto a provisioning. Questa operazione applica tutti i gruppi di parametri cache modificati al cluster. L'operazione di riavvio avviene il prima possibile e provoca un'interruzione momentanea del cluster. Durante il riavvio, lo stato del cluster è impostato su `REBOOTING`.

```
aws elasticache reboot-cache-cluster \
    --cache-cluster-id "my-cluster-001" \
    --cache-node-ids-to-reboot "0001"
```

Output:

```
{
    "CacheCluster": {
        "CacheClusterId": "my-cluster-001",
        "ClientDownloadLandingPage": "https://console.aws.amazon.com/elasticache/
home#client-download:",
        "CacheNodeType": "cache.r5.xlarge",
        "Engine": "redis",
        "EngineVersion": "5.0.5",
        "CacheClusterStatus": "rebooting cache cluster nodes",
        "NumCacheNodes": 1,
        "PreferredAvailabilityZone": "us-west-2a",
        "CacheClusterCreateTime": "2019-11-26T03:35:04.546Z",
        "PreferredMaintenanceWindow": "mon:04:05-mon:05:05",
        "PendingModifiedValues": {},
        "NotificationConfiguration": {
            "TopicArn": "arn:aws:sns:us-west-2:xxxxxxxxxx152:My_Topic",
            "TopicStatus": "active"
        },
        "CacheSecurityGroups": [],
        "CacheParameterGroup": {
```

```
            "CacheParameterGroupName": "mygroup",
            "ParameterApplyStatus": "in-sync",
            "CacheNodeIdsToReboot": []
        },
        "CacheSubnetGroupName": "kxkxk",
        "AutoMinorVersionUpgrade": true,
        "SecurityGroups": [
            {
                "SecurityGroupId": "sg-xxxxxxxxxxxxx836",
                "Status": "active"
            },
            {
                "SecurityGroupId": "sg-xxxxxxxx7b",
                "Status": "active"
            }
        ],
        "ReplicationGroupId": "my-cluster",
        "SnapshotRetentionLimit": 0,
        "SnapshotWindow": "07:30-08:30",
        "TransitEncryptionEnabled": false,
        "AtRestEncryptionEnabled": false
    }
}
```

Per ulteriori informazioni, consulta Rebooting a Cluster < https://docs.aws.amazon.com/AmazonElastiCache/ latest/red-ug/Clusters .rebooting.html nella Guida per l'utente di Elasticache.

- Per i dettagli sull'API, consulta Command Reference. RebootCacheClusterAWS CLI

`reset-cache-parameter-group`

Il seguente esempio di codice mostra come utilizzare`reset-cache-parameter-group`.

AWS CLI

Come reimpostare un gruppo di parametri cache

L'esempio `reset-cache-parameter-group` seguente modifica i parametri di un gruppo di parametri cache reimpostando il valore predefinito del motore o del sistema. È possibile reimpostare parametri specifici specificando un elenco di nomi di parametri. Per reimpostare l'intero gruppo di parametri cache, specifica i parametri `--reset-all-parameters` e `--cache-parameter-group-name`.

```
aws elasticache reset-cache-parameter-group \
    --cache-parameter-group-name "mygroup" \
    --reset-all-parameters
```

Output:

```
{
    "CacheParameterGroupName": "mygroup"
}
```

- Per i dettagli sull'API, consulta ResetCacheParameterGroup AWS CLICommand Reference.

`start-migration`

Il seguente esempio di codice mostra come utilizzare`start-migration`.

AWS CLI

Come avviare una migrazione

Quanto segue `start-migration` migra i tuoi dati da Redis self-hosted su Amazon ad EC2 Amazon ElastiCache, utilizzando il motore Redis.

```
aws elasticache start-migration \
   --replication-group-id test \
   --customer-node-endpoint-
list "Address='test.g2xbih.ng.0001.usw2.cache.amazonaws.com',Port=6379"
```

Output

```
{
    "ReplicationGroup": {
        "ReplicationGroupId": "test",
        "Description": "test",
        "GlobalReplicationGroupInfo": {},
        "Status": "modifying",
        "PendingModifiedValues": {},
        "MemberClusters": [
            "test-001",
            "test-002",
            "test-003"
```

```
        ],
        "NodeGroups": [
            {
                "NodeGroupId": "0001",
                "Status": "available",
                "PrimaryEndpoint": {
                    "Address": "test.g2xbih.ng.0001.usw2.cache.amazonaws.com",
                    "Port": 6379
                },
                "ReaderEndpoint": {
                    "Address": "test-ro.g2xbih.ng.0001.usw2.cache.amazonaws.com",
                    "Port": 6379
                },
                "NodeGroupMembers": [
                    {
                        "CacheClusterId": "test-001",
                        "CacheNodeId": "0001",
                        "ReadEndpoint": {
                            "Address":
"test-001.g2xbih.0001.usw2.cache.amazonaws.com",
                            "Port": 6379
                        },
                        "PreferredAvailabilityZone": "us-west-2a",
                        "CurrentRole": "primary"
                    },
                    {
                        "CacheClusterId": "test-002",
                        "CacheNodeId": "0001",
                        "ReadEndpoint": {
                            "Address":
"test-002.g2xbih.0001.usw2.cache.amazonaws.com",
                            "Port": 6379
                        },
                        "PreferredAvailabilityZone": "us-west-2c",
                        "CurrentRole": "replica"
                    },
                    {
                        "CacheClusterId": "test-003",
                        "CacheNodeId": "0001",
                        "ReadEndpoint": {
                            "Address":
"test-003.g2xbih.0001.usw2.cache.amazonaws.com",
                            "Port": 6379
                        },
```

```
                        "PreferredAvailabilityZone": "us-west-2b",
                        "CurrentRole": "replica"
                    }
                ]
            }
        ],
        "SnapshottingClusterId": "test-002",
        "AutomaticFailover": "enabled",
        "MultiAZ": "enabled",
        "SnapshotRetentionLimit": 1,
        "SnapshotWindow": "07:30-08:30",
        "ClusterEnabled": false,
        "CacheNodeType": "cache.r5.large",
        "TransitEncryptionEnabled": false,
        "AtRestEncryptionEnabled": false
    }
}
```

Per ulteriori informazioni, consulta la sezione Migrazione online a ElastiCache nella Guida per l'utente di Elasticache.

- Per i dettagli sull'API, consulta AWS CLI Command StartMigrationReference.

`test-failover`

Il seguente esempio di codice mostra come utilizzare`test-failover`.

AWS CLI

Come testare il failover di un gruppo di nodi

L'esempio `test-failover` seguente esegue il test del failover automatico sul gruppo di nodi specificato (chiamato shard nella console) in un gruppo di replica (chiamato cluster nella console).

```
aws elasticache test-failover /
    --replication-group-id "mycluster" /
    --node-group-id "0001"
```

Output:

```
{
    "ReplicationGroup": {
        "ReplicationGroupId": "mycluster",
```

```
        "Description": "My Cluster",
        "Status": "available",
        "PendingModifiedValues": {},
        "MemberClusters": [
            "mycluster-0001-001",
            "mycluster-0001-002",
            "mycluster-0001-003",
            "mycluster-0002-001",
            "mycluster-0002-002",
            "mycluster-0002-003",
            "mycluster-0003-001",
            "mycluster-0003-002",
            "mycluster-0003-003"
        ],
        "NodeGroups": [
            {
                "NodeGroupId": "0001",
                "Status": "available",
                "Slots": "0-5461",
                "NodeGroupMembers": [
                    {
                        "CacheClusterId": "mycluster-0001-001",
                        "CacheNodeId": "0001",
                        "PreferredAvailabilityZone": "us-west-2b"
                    },
                    {
                        "CacheClusterId": "mycluster-0001-002",
                        "CacheNodeId": "0001",
                        "PreferredAvailabilityZone": "us-west-2a"
                    },
                    {
                        "CacheClusterId": "mycluster-0001-003",
                        "CacheNodeId": "0001",
                        "PreferredAvailabilityZone": "us-west-2c"
                    }
                ]
            },
            {
                "NodeGroupId": "0002",
                "Status": "available",
                "Slots": "5462-10922",
                "NodeGroupMembers": [
                    {
                        "CacheClusterId": "mycluster-0002-001",
```

```
                    "CacheNodeId": "0001",
                    "PreferredAvailabilityZone": "us-west-2a"
                },
                {
                    "CacheClusterId": "mycluster-0002-002",
                    "CacheNodeId": "0001",
                    "PreferredAvailabilityZone": "us-west-2b"
                },
                {
                    "CacheClusterId": "mycluster-0002-003",
                    "CacheNodeId": "0001",
                    "PreferredAvailabilityZone": "us-west-2c"
                }
            ]
        },
        {
            "NodeGroupId": "0003",
            "Status": "available",
            "Slots": "10923-16383",
            "NodeGroupMembers": [
                {
                    "CacheClusterId": "mycluster-0003-001",
                    "CacheNodeId": "0001",
                    "PreferredAvailabilityZone": "us-west-2c"
                },
                {
                    "CacheClusterId": "mycluster-0003-002",
                    "CacheNodeId": "0001",
                    "PreferredAvailabilityZone": "us-west-2b"
                },
                {
                    "CacheClusterId": "mycluster-0003-003",
                    "CacheNodeId": "0001",
                    "PreferredAvailabilityZone": "us-west-2a"
                }
            ]
        }
    ],
    "AutomaticFailover": "enabled",
    "ConfigurationEndpoint": {
        "Address": "mycluster.xxxxih.clustercfg.usw2.cache.amazonaws.com",
        "Port": 6379
    },
    "SnapshotRetentionLimit": 1,
```

```
        "SnapshotWindow": "13:00-14:00",
        "ClusterEnabled": true,
        "CacheNodeType": "cache.r5.large",
        "TransitEncryptionEnabled": false,
        "AtRestEncryptionEnabled": false
    }
}
```

- Per i dettagli sull'API, consulta TestFailover AWS CLICommand Reference.

MediaStore esempi utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with MediaStore.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`create-container`

Il seguente esempio di codice mostra come utilizzare`create-container`.

AWS CLI

Come creare un container

L'esempio `create-container` seguente crea un nuovo container vuoto.

```
aws mediastore create-container --container-name ExampleContainer
```

Output:

```
{
    "Container": {
        "AccessLoggingEnabled": false,
        "CreationTime": 1563557265,
        "Name": "ExampleContainer",
        "Status": "CREATING",
        "ARN": "arn:aws:mediastore:us-west-2:111122223333:container/
ExampleContainer"
    }
}
```

Per ulteriori informazioni, consulta Creazione di un contenitore nella Guida MediaStore utente di AWS Elemental.

- Per i dettagli sull'API, consulta CreateContainer AWS CLICommand Reference.

`delete-container-policy`

Il seguente esempio di codice mostra come utilizzare`delete-container-policy`.

AWS CLI

Come eliminare una policy di container

L'esempio `delete-container-policy` seguente elimina la policy assegnata al container specificato. Quando la policy viene eliminata, AWS Elemental assegna MediaStore automaticamente la policy predefinita al contenitore.

```
aws mediastore delete-container-policy \
    --container-name LiveEvents
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta il riferimento DeleteContainerPolicyall'API AWS MediaStore Elemental.

- Per i dettagli sull'API, consulta DeleteContainerPolicy AWS CLICommand Reference.

`delete-container`

Il seguente esempio di codice mostra come utilizzare`delete-container`.

AWS CLI

Come eliminare un container

L'esempio `delete-container` seguente elimina il container specificato. Puoi eliminare un container solo se non contiene oggetti.

```
aws mediastore delete-container \
    --container-name=ExampleLiveDemo
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Eliminazione di un contenitore nella Guida per l' MediaStore utente di AWS Elemental.

- Per i dettagli sull'API, consulta AWS CLI Command DeleteContainerReference.

`delete-cors-policy`

Il seguente esempio di codice mostra come utilizzare`delete-cors-policy`.

AWS CLI

Come eliminare una policy CORS

L'esempio `delete-cors-policy` seguente elimina la policy Cross-Origin Resource Sharing (CORS) assegnata al container specificato.

```
aws mediastore delete-cors-policy \
    --container-name ExampleContainer
```

Questo comando non produce alcun output.

Per ulteriori informazioni, vedete Eliminazione di una policy CORS nella Guida per l'utente di AWS MediaStore Elemental.

- Per i dettagli sull'API, consultate Command DeleteCorsPolicyReference AWS CLI .

`delete-lifecycle-policy`

Il seguente esempio di codice mostra come utilizzare`delete-lifecycle-policy`.

AWS CLI

Come eliminare una policy del ciclo di vita degli oggetti

L'esempio `delete-lifecycle-policy` seguente elimina la policy del ciclo di vita degli oggetti allegata al container specificato. Sono necessari fino a 20 minuti per rendere effettiva la modifica.

```
aws mediastore delete-lifecycle-policy \
    --container-name LiveEvents
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Eliminazione di una policy sul ciclo di vita degli oggetti nella Guida per l'utente di AWS MediaStore Elemental.

- Per i dettagli sull'API, consulta DeleteLifecyclePolicyCommand Reference.AWS CLI

`describe-container`

Il seguente esempio di codice mostra come utilizzare`describe-container`.

AWS CLI

Come visualizzare i dettagli di un container

L'esempio `describe-container` seguente visualizza i dettagli del container specificato.

```
aws mediastore describe-container \
    --container-name ExampleContainer
```

Output:

```
{
    "Container": {
        "CreationTime": 1563558086,
        "AccessLoggingEnabled": false,
        "ARN": "arn:aws:mediastore:us-west-2:111122223333:container/
ExampleContainer",
        "Status": "ACTIVE",
        "Name": "ExampleContainer",
        "Endpoint": "https://aaabbbcccdddee.data.mediastore.us-west-2.amazonaws.com"
```

```
    }
}
```

Per ulteriori informazioni, consulta Visualizzazione dei dettagli di un contenitore nella Guida per l' MediaStore utente di AWS Elemental.

- Per i dettagli sull'API, consulta DescribeContainer AWS CLICommand Reference.

`describe-object`

Il seguente esempio di codice mostra come utilizzare`describe-object`.

AWS CLI

Come visualizzare un elenco di oggetti e cartelle in un determinato container

L'esempio `describe-object` seguente visualizza degli elementi (oggetti e cartelle) memorizzati in un container specificato.

```
aws mediastore-data describe-object \
    --endpoint https://aaabbbcccdddee.data.mediastore.us-west-2.amazonaws.com \
    --path /folder_name/file1234.jpg
```

Output:

```
{
    "ContentType": "image/jpeg",
    "LastModified": "Fri, 19 Jul 2019 21:32:20 GMT",
    "ContentLength": "2307346",
    "ETag": "2aa333bbcc8d8d22d777e999c88d4aa9eeeeee4dd89ff7f555555555555da6d3"
}
```

Per ulteriori informazioni, consulta Visualizzazione dei dettagli di un oggetto nella Guida per l' MediaStore utente di AWS Elemental.

- Per i dettagli sull'API, consulta DescribeObject AWS CLICommand Reference.

`get-container-policy`

Il seguente esempio di codice mostra come utilizzare`get-container-policy`.

AWS CLI

Come visualizzare una policy di container

L'esempio `get-container-policy` seguente visualizza la policy basata sulle risorse del container specificato.

```
aws mediastore get-container-policy \
    --container-name ExampleLiveDemo
```

Output:

```
{
    "Policy": {
        "Version":"2012-10-17",
        "Statement": [
            {
                "Sid": "PublicReadOverHttps",
                "Effect": "Allow",
                "Principal": {
                    "AWS": "arn:aws:iam::111122223333:root"
                },
                "Action": [
                    "mediastore:GetObject",
                    "mediastore:DescribeObject"
                ],
                "Resource": "arn:aws:mediastore:us-west-2:111122223333:container/
ExampleLiveDemo/",
                "Condition": {
                    "Bool": {
                        "aws:SecureTransport": "true"
                    }
                }
            }
        ]
    }
}
```

Per ulteriori informazioni, consulta Visualizzazione di una politica sui contenitori nella Guida per l' MediaStore utente di AWS Elemental.

- Per i dettagli sull'API, consulta GetContainerPolicy AWS CLICommand Reference.

`get-cors-policy`

Il seguente esempio di codice mostra come utilizzare`get-cors-policy`.

AWS CLI

Come visualizzare una policy CORS

L'esempio `get-cors-policy` seguente visualizza la policy Cross-Origin Resource Sharing (CORS) assegnata al container specificato.

```
aws mediastore get-cors-policy \
    --container-name ExampleContainer \
    --region us-west-2
```

Output:

```
{
    "CorsPolicy": [
        {
            "AllowedMethods": [
                "GET",
                "HEAD"
            ],
            "MaxAgeSeconds": 3000,
            "AllowedOrigins": [
                ""
            ],
            "AllowedHeaders": [
                ""
            ]
        }
    ]
}
```

Per ulteriori informazioni, vedere Visualizzazione di una politica CORS nella Guida per l' MediaStore utente di AWS Elemental.

- Per i dettagli sulle API, consultate AWS CLI Command GetCorsPolicyReference.

`get-lifecycle-policy`

Il seguente esempio di codice mostra come utilizzare`get-lifecycle-policy`.

AWS CLI

Come visualizzare una policy del ciclo di vita degli oggetti

L'esempio `get-lifecycle-policy` seguente elimina la policy del ciclo di vita degli oggetti associata al container specificato.

```
aws mediastore get-lifecycle-policy \
    --container-name LiveEvents
```

Output:

```
{
    "LifecyclePolicy": {
        "rules": [
            {
                "definition": {
                    "path": [
                        {
                            "prefix": "Football/"
                        },
                        {
                            "prefix": "Baseball/"
                        }
                    ],
                    "days_since_create": [
                        {
                            "numeric": [
                                ">",
                                28
                            ]
                        }
                    ]
                },
                "action": "EXPIRE"
            }
        ]
    }
}
```

Per ulteriori informazioni, consulta Visualizzazione di una politica sul ciclo di vita degli oggetti nella Guida per l'utente di AWS MediaStore Elemental.

- Per i dettagli sull'API, consulta Command GetLifecyclePolicyReference AWS CLI .

`get-object`

Il seguente esempio di codice mostra come utilizzare`get-object`.

AWS CLI

Come scaricare un oggetto

L'esempio `get-object` seguente scarica un oggetto nell'endpoint specificato.

```
aws mediastore-data get-object \
    --endpoint https://aaabbbcccdddee.data.mediastore.us-west-2.amazonaws.com \
    --path=/folder_name/README.md README.md
```

Output:

```
{
    "ContentLength": "2307346",
    "ContentType": "image/jpeg",
    "LastModified": "Fri, 19 Jul 2019 21:32:20 GMT",
    "ETag": "2aa333bbcc8d8d22d777e999c88d4aa9eeeeee4dd89ff7f555555555555da6d3",
    "StatusCode": 200
}
```

Come scaricare parte di un oggetto

L'esempio `get-object` seguente scarica una parte di un oggetto nell'endpoint specificato.

```
aws mediastore-data get-object \
    --endpoint https://aaabbbcccdddee.data.mediastore.us-west-2.amazonaws.com \
    --path /folder_name/README.md \
    --range="bytes=0-100" README2.md
```

Output:

```
{
    "StatusCode": 206,
    "ContentRange": "bytes 0-100/2307346",
    "ContentLength": "101",
    "LastModified": "Fri, 19 Jul 2019 21:32:20 GMT",
```

```
    "ContentType": "image/jpeg",
    "ETag": "2aa333bbcc8d8d22d777e999c88d4aa9eeeeee4dd89ff7f555555555555da6d3"
}
```

Per ulteriori informazioni, consulta Downloading an Object nella AWS Elemental MediaStore User Guide.

- Per i dettagli sull'API, consulta GetObject AWS CLICommand Reference.

`list-containers`

Il seguente esempio di codice mostra come utilizzare`list-containers`.

AWS CLI

Come visualizzare un elenco di container

L'esempio `list-containers` seguente visualizza un elenco di tutti i container associati all'account in uso.

```
aws mediastore list-containers
```

Output:

```
{
    "Containers": [
        {
            "CreationTime": 1505317931,
            "Endpoint": "https://aaabbbcccdddee.data.mediastore.us-
west-2.amazonaws.com",
            "Status": "ACTIVE",
            "ARN": "arn:aws:mediastore:us-west-2:111122223333:container/
ExampleLiveDemo",
            "AccessLoggingEnabled": false,
            "Name": "ExampleLiveDemo"
        },
        {
            "CreationTime": 1506528818,
            "Endpoint": "https://fffgggghhhiiijj.data.mediastore.us-
west-2.amazonaws.com",
            "Status": "ACTIVE",
            "ARN": "arn:aws:mediastore:us-west-2:111122223333:container/
ExampleContainer",
```

```
            "AccessLoggingEnabled": false,
            "Name": "ExampleContainer"
        }
    ]
}
```

Per ulteriori informazioni, consulta Visualizzazione di un elenco di contenitori nella Guida per l' MediaStore utente di AWS Elemental.

- Per i dettagli sull'API, consulta ListContainers AWS CLICommand Reference.

`list-items`

Il seguente esempio di codice mostra come utilizzare`list-items`.

AWS CLI

Esempio 1: come visualizzare un elenco di oggetti e cartelle in un determinato container

L'esempio `list-items` seguente visualizza degli elementi (oggetti e cartelle) archiviati in un container specificato.

```
aws mediastore-data list-items \
    --endpoint https://aaabbbcccdddee.data.mediastore.us-west-2.amazonaws.com
```

Output:

```
{
    "Items": [
        {
            "ContentType": "image/jpeg",
            "LastModified": 1563571859.379,
            "Name": "filename.jpg",
            "Type": "OBJECT",
            "ETag":
 "543ab21abcd1a234ab123456a1a2b12345ab12abc12a1234abc1a2bc12345a12",
            "ContentLength": 3784
        },
        {
            "Type": "FOLDER",
            "Name": "ExampleLiveDemo"
        }
    ]
```

```
}
```

Esempio 2: come visualizzare un elenco di oggetti e cartelle in una determinata cartella

L'esempio `list-items` seguente visualizza degli elementi (oggetti e cartelle) archiviati in una cartella specificata.

```
aws mediastore-data list-items \
    --endpoint https://aaabbbcccdddee.data.mediastore.us-west-2.amazonaws.com
```

Output:

```
{
    "Items": [
        {
            "ContentType": "image/jpeg",
            "LastModified": 1563571859.379,
            "Name": "filename.jpg",
            "Type": "OBJECT",
            "ETag":
 "543ab21abcd1a234ab123456a1a2b12345ab12abc12a1234abc1a2bc12345a12",
            "ContentLength": 3784
        },
        {
            "Type": "FOLDER",
            "Name": "ExampleLiveDemo"
        }
    ]
}
```

Per ulteriori informazioni, consulta Visualizzazione di un elenco di oggetti nella Guida MediaStore utente di AWS Elemental.

- Per i dettagli sull'API, consulta ListItems AWS CLICommand Reference.

`list-tags-for-resource`

Il seguente esempio di codice mostra come utilizzare`list-tags-for-resource`.

AWS CLI

Come elencare i tag per un container

L'esempio `list-tags-for-resource` seguente visualizza le chiavi e i valori dei tag assegnati al container specificato.

```
aws mediastore list-tags-for-resource \
    --resource arn:aws:mediastore:us-west-2:1213456789012:container/ExampleContainer
```

Output:

```
{
    "Tags": [
        {
            "Value": "Test",
            "Key": "Environment"
        },
        {
            "Value": "West",
            "Key": "Region"
        }
    ]
}
```

Per ulteriori informazioni, consulta ListTagsForResource AWSElemental MediaStore API Reference.

- Per i dettagli sull'API, consulta ListTagsForResource AWS CLICommand Reference.

`put-container-policy`

Il seguente esempio di codice mostra come utilizzare`put-container-policy`.

AWS CLI

Come modificare una policy di container

L'esempio `put-container-policy` seguente assegna una policy diversa al container specificato. In questo esempio, la policy aggiornata viene definita in un file denominato `LiveEventsContainerPolicy.json`.

```
aws mediastore put-container-policy \
    --container-name LiveEvents \
    --policy file://LiveEventsContainerPolicy.json
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Modifica di una politica sui contenitori nella Guida per l' MediaStore utente di AWS Elemental.

- Per i dettagli sull'API, consulta PutContainerPolicy AWS CLICommand Reference.

`put-cors-policy`

Il seguente esempio di codice mostra come utilizzare`put-cors-policy`.

AWS CLI

Esempio 1: come aggiungere una policy CORS

L'esempio `put-cors-policy` seguente aggiunge una policy Cross-Origin Resource Sharing (CORS) assegnata al container specificato. Il contenuto della policy CORS si trova nel file denominato `corsPolicy.json`.

```
aws mediastore put-cors-policy \
    --container-name ExampleContainer \
    --cors-policy file://corsPolicy.json
```

Questo comando non produce alcun output.

Per ulteriori informazioni, vedete Aggiungere una politica CORS a un contenitore nella Guida per l' MediaStore utente di AWS Elemental.

Esempio 2: come modificare una policy CORS

L'esempio `put-cors-policy` seguente aggiorna la policy Cross-Origin Resource Sharing (CORS) assegnata al container specificato. Il contenuto della policy CORS aggiornata si trova nel file denominato `corsPolicy2.json`.

Per ulteriori informazioni, vedete Modifica di una politica CORS nella Guida per l'utente di AWS MediaStore Elemental.

- Per i dettagli sulle API, consultate AWS CLI Command PutCorsPolicyReference.

`put-lifecycle-policy`

Il seguente esempio di codice mostra come utilizzare`put-lifecycle-policy`.

AWS CLI

Come creare una policy del ciclo di vita degli oggetti

L'esempio `put-lifecycle-policy` seguente associa una policy del ciclo di vita degli oggetti al container specificato. Ciò consente di specificare per quanto tempo il servizio deve archiviare gli oggetti nel contenitore. MediaStore elimina gli oggetti nel contenitore una volta raggiunta la data di scadenza, come indicato nella policy, che si trova nel file denominato`LiveEventsLifecyclePolicy.json`.

```
aws mediastore put-lifecycle-policy \
    --container-name ExampleContainer \
    --lifecycle-policy file://ExampleLifecyclePolicy.json
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Aggiungere una policy sul ciclo di vita degli oggetti a un contenitore nella Guida per l'utente di AWS MediaStore Elemental.

- Per i dettagli sull'API, consulta Command PutLifecyclePolicyReference AWS CLI .

`put-object`

Il seguente esempio di codice mostra come utilizzare`put-object`.

AWS CLI

Come caricare un oggetto

L'esempio `put-object` seguente carica un oggetto nel container specificato. Puoi specificare un percorso di cartella in cui salvare l'oggetto all'interno del container. Se la cartella esiste già, AWS Elemental MediaStore memorizza l'oggetto nella cartella. Se la cartella non esiste, il servizio la crea e quindi archivia l'oggetto nella cartella.

```
aws mediastore-data put-object \
    --endpoint https://aaabbbcccdddee.data.mediastore.us-west-2.amazonaws.com \
    --body README.md \
    --path /folder_name/README.md \
    --cache-control "max-age=6, public" \
    --content-type binary/octet-stream
```

Output:

```
{
    "ContentSHA256":
 "74b5fdb517f423ed750ef214c44adfe2be36e37d861eafe9c842cbe1bf387a9d",
    "StorageClass": "TEMPORAL",
    "ETag": "af3e4731af032167a106015d1f2fe934e68b32ed1aa297a9e325f5c64979277b"
}
```

Per ulteriori informazioni, consulta Caricamento di un oggetto nella Guida utente di AWS MediaStore Elemental.

- Per i dettagli sull'API, consulta AWS CLI Command PutObjectReference.

`start-access-logging`

Il seguente esempio di codice mostra come utilizzare`start-access-logging`.

AWS CLI

Come abilitare la registrazione di log su un container

L'esempio `start-access-logging` seguente abilita la registrazione di log degli accessi sul container specificato.

```
aws mediastore start-access-logging \
    --container-name LiveEvents
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Enabling Access Logging for a Container nella Guida per l' MediaStore utente di AWS Elemental.

- Per i dettagli sull'API, consulta AWS CLI Command StartAccessLoggingReference.

`stop-access-logging`

Il seguente esempio di codice mostra come utilizzare`stop-access-logging`.

AWS CLI

Come disabilitare la registrazione di log su un container

L'esempio `stop-access-logging` seguente disabilita la registrazione di log sul container specificato.

```
aws mediastore stop-access-logging \
    --container-name LiveEvents
```

Questo comando non produce alcun output.

Per ulteriori informazioni, vedete Disabling Access Logging for a Container nella Guida per l'utente di AWS MediaStore Elemental.

- Per i dettagli sull'API, consultate Command StopAccessLoggingReference AWS CLI .

`tag-resource`

Il seguente esempio di codice mostra come utilizzare`tag-resource`.

AWS CLI

Come aggiungere tag a un container

L'esempio `tag-resource` seguente aggiunge chiavi e valori di tag al container specificato.

```
aws mediastore tag-resource \
    --resource arn:aws:mediastore:us-west-2:123456789012:container/ExampleContainer
 \
    --tags '[{"Key": "Region", "Value": "West"}, {"Key": "Environment", "Value":
 "Test"}]'
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta TagResource AWSElemental MediaStore API Reference.

- Per i dettagli sull'API, consulta TagResource AWS CLICommand Reference.

`untag-resource`

Il seguente esempio di codice mostra come utilizzare`untag-resource`.

AWS CLI

Come rimuovere i tag da un container

L'esempio `untag-resource` seguente rimuove la chiave di tag specificata e il valore associato da un container.

```
aws mediastore untag-resource \
    --resource arn:aws:mediastore:us-west-2:123456789012:container/ExampleContainer
 \
    --tag-keys Region
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta UntagResource AWSElemental MediaStore API Reference. .

- Per i dettagli sull'API, consulta UntagResource AWS CLICommand Reference.

Esempi di utilizzo di Amazon EMR AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando Amazon EMR. AWS Command Line Interface

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`add-instance-fleet`

Il seguente esempio di codice mostra come usare`add-instance-fleet`.

AWS CLI

Come aggiungere un parco istanze di attività a un cluster

Questo esempio aggiunge un nuovo parco istanze di attività al cluster specificato.

Comando:

```
aws emr add-instance-fleet --cluster-id 'j-12ABCDEFGHI34JK' --instance-fleet
 InstanceFleetType=TASK,TargetSpotCapacity=1,LaunchSpecifications={SpotSpecification='{Timeo
```

Output:

```
{
   "ClusterId": "j-12ABCDEFGHI34JK",
   "InstanceFleetId": "if-23ABCDEFGHI45JJ"
}
```

- Per i dettagli sull'API, consulta AddInstanceFleet AWS CLICommand Reference.

`add-steps`

Il seguente esempio di codice mostra come utilizzare`add-steps`.

AWS CLI

1. Come aggiungere fasi JAR personalizzate a un cluster

Comando:

```
aws emr add-steps --cluster-id j-XXXXXXXX --steps
 Type=CUSTOM_JAR,Name=CustomJAR,ActionOnFailure=CONTINUE,Jar=s3://amzn-s3-demo-
bucket/mytest.jar,Args=arg1,arg2,arg3
 Type=CUSTOM_JAR,Name=CustomJAR,ActionOnFailure=CONTINUE,Jar=s3://amzn-s3-demo-
bucket/mytest.jar,MainClass=mymainclass,Args=arg1,arg2,arg3
```

Parametri obbligatori:

```
Jar
```

Parametri facoltativi:

```
Type, Name, ActionOnFailure, Args
```

Output:

```
{
    "StepIds":[
        "s-XXXXXXXX",
        "s-YYYYYYYY"
    ]
}
```

2. Come aggiungere fasi di streaming a un cluster

Comando:

```
aws emr add-steps --cluster-id j-XXXXXXXX --steps Type=STREAMING,Name='Streaming
 Program',ActionOnFailure=CONTINUE,Args=[-files,s3://elasticmapreduce/samples/
wordcount/wordSplitter.py,-mapper,wordSplitter.py,-reducer,aggregate,-input,s3://
elasticmapreduce/samples/wordcount/input,-output,s3://amzn-s3-demo-bucket/wordcount/
output]
```

Parametri obbligatori:

```
Type, Args
```

Parametri facoltativi:

```
Name, ActionOnFailure
```

Equivalente a JSON (contenuto di step.json):

```
 [
  {
    "Name": "JSON Streaming Step",
    "Args": ["-files","s3://elasticmapreduce/samples/wordcount/wordSplitter.py","-
mapper","wordSplitter.py","-reducer","aggregate","-input","s3://elasticmapreduce/
samples/wordcount/input","-output","s3://amzn-s3-demo-bucket/wordcount/output"],
    "ActionOnFailure": "CONTINUE",
    "Type": "STREAMING"
  }
]
```

NOTA: gli argomenti JSON devono includere opzioni e valori come elementi propri nell'elenco.

Comando (utilizzando step.json):

```
aws emr add-steps --cluster-id j-XXXXXXXX --steps file://./step.json
```

Output:

```
{
    "StepIds":[
        "s-XXXXXXXX",
        "s-YYYYYYYY"
    ]
}
```

3. Come aggiungere una fase di streaming con più file a un cluster (solo JSON)

JSON (multiplefiles.json):

```
[
  {
     "Name": "JSON Streaming Step",
     "Type": "STREAMING",
     "ActionOnFailure": "CONTINUE",
     "Args": [
         "-files",
         "s3://amzn-s3-demo-bucket/mapper.py,s3://amzn-s3-demo-bucket/reducer.py",
         "-mapper",
         "mapper.py",
         "-reducer",
         "reducer.py",
         "-input",
         "s3://amzn-s3-demo-bucket/input",
         "-output",
         "s3://amzn-s3-demo-bucket/output"]
  }
]
```

Comando:

```
aws emr add-steps --cluster-id j-XXXXXXXX  --steps file://./multiplefiles.json
```

Parametri obbligatori:

```
Type, Args
```

Parametri facoltativi:

```
Name, ActionOnFailure
```

Output:

```
{
    "StepIds":[
        "s-XXXXXXXX",
    ]
}
```

4. Come aggiungere fasi Hive a un cluster

Comando:

```
aws emr add-steps --cluster-id j-XXXXXXXX --steps Type=HIVE,Name='Hive
 program',ActionOnFailure=CONTINUE,Args=[-f,s3://amzn-s3-demo-bucket/
myhivescript.q,-d,INPUT=s3://amzn-s3-demo-bucket/myhiveinput,-d,OUTPUT=s3://
amzn-s3-demo-bucket/myhiveoutput,arg1,arg2] Type=HIVE,Name='Hive
 steps',ActionOnFailure=TERMINATE_CLUSTER,Args=[-f,s3://elasticmapreduce/samples/
hive-ads/libs/model-build.q,-d,INPUT=s3://elasticmapreduce/samples/hive-ads/tables,-
d,OUTPUT=s3://amzn-s3-demo-bucket/hive-ads/output/2014-04-18/11-07-32,-d,LIBS=s3://
elasticmapreduce/samples/hive-ads/libs]
```

Parametri obbligatori:

```
Type, Args
```

Parametri facoltativi:

```
Name, ActionOnFailure
```

Output:

```
{
    "StepIds":[
        "s-XXXXXXXX",
        "s-YYYYYYYY"
    ]
```

```
}
```

5. Come aggiungere fasi Pig a un cluster

Comando:

```
aws emr add-steps --cluster-id j-XXXXXXXX --steps Type=PIG,Name='Pig
 program',ActionOnFailure=CONTINUE,Args=[-f,s3://amzn-s3-demo-bucket/
mypigscript.pig,-p,INPUT=s3://amzn-s3-demo-bucket/mypiginput,-p,OUTPUT=s3://
amzn-s3-demo-bucket/mypigoutput,arg1,arg2] Type=PIG,Name='Pig program',Args=[-
f,s3://elasticmapreduce/samples/pig-apache/do-reports2.pig,-p,INPUT=s3://
elasticmapreduce/samples/pig-apache/input,-p,OUTPUT=s3://amzn-s3-demo-bucket/pig-
apache/output,arg1,arg2]
```

Parametri obbligatori:

```
Type, Args
```

Parametri facoltativi:

```
Name, ActionOnFailure
```

Output:

```
{
    "StepIds":[
        "s-XXXXXXXX",
        "s-YYYYYYYY"
    ]
}
```

6. Come aggiungere fasi Impala a un cluster

Comando:

```
aws emr add-steps --cluster-id j-XXXXXXXX --steps Type=IMPALA,Name='Impala
 program',ActionOnFailure=CONTINUE,Args=--impala-script,s3://myimpala/input,--
console-output-path,s3://myimpala/output
```

Parametri obbligatori:

```
Type, Args
```

Parametri facoltativi:

```
Name, ActionOnFailure
```

Output:

```
{
    "StepIds":[
        "s-XXXXXXXX",
        "s-YYYYYYYY"
    ]
}
```

- Per i dettagli sull'API, vedere AddStepsin AWS CLI Command Reference.

`add-tags`

Il seguente esempio di codice mostra come utilizzare`add-tags`.

AWS CLI

1. Come aggiungere tag a un cluster

Comando:

```
aws emr add-tags --resource-id j-xxxxxxx --tags name="John Doe" age=29 sex=male
 address="123 East NW Seattle"
```

Output:

```
None
```

2. Come elencare i tag di un cluster

Comando:

```
aws emr describe-cluster --cluster-id j-XXXXXXYY --query Cluster.Tags
```

Output:

```
[
    {
        "Value": "male",
        "Key": "sex"
    },
    {
        "Value": "123 East NW Seattle",
        "Key": "address"
    },
    {
        "Value": "John Doe",
        "Key": "name"
    },
    {
        "Value": "29",
        "Key": "age"
    }
]
```

- Per i dettagli sull'API, consulta AddTags AWS CLICommand Reference.

`create-cluster-examples`

Il seguente esempio di codice mostra come utilizzare`create-cluster-examples`.

AWS CLI

La maggior parte degli esempi seguenti presuppone che tu abbia specificato il ruolo del servizio Amazon EMR e il profilo dell' EC2 istanza Amazon. In caso contrario, devi specificare ogni ruolo IAM richiesto o utilizzare il parametro `--use-default-roles` durante la creazione del cluster. Per ulteriori informazioni sulla specificazione dei ruoli IAM, consulta Configure IAM Roles for Amazon EMR Permissions AWS to Services nella Amazon EMR Management Guide.

Esempio 1: come creare un cluster

L'esempio `create-cluster` seguente crea un cluster EMR semplice.

```
aws emr create-cluster \
    --release-label emr-5.14.0 \
```

```
    --instance-type m4.large \
    --instance-count 2
```

Questo comando non produce alcun output.

Esempio 2: creare un cluster Amazon EMR con impostazioni predefinite ServiceRole e ruoli InstanceProfile

L'esempio `create-cluster` seguente crea un cluster Amazon EMR che utilizza la configurazione `--instance-groups` e include una policy di dimensionamento gestito.

```
aws emr create-cluster \
    --release-label emr-5.14.0 \
    --service-role EMR_DefaultRole \
    --ec2-attributes InstanceProfile=EMR_EC2_DefaultRole \
    --instance-
groups InstanceGroupType=MASTER,InstanceCount=1,InstanceType=m4.large InstanceGroupType=CORE
```

Esempio 3: come creare un cluster Amazon EMR che utilizza un parco istanze

L'`create-cluster`esempio seguente crea un cluster Amazon EMR che utilizza la `--instance-fleets` configurazione, specificando due tipi di istanze per ogni flotta e due sottoreti. EC2

```
aws emr create-cluster \
    --release-label emr-5.14.0 \
    --service-role EMR_DefaultRole \
    --ec2-attributes InstanceProfile=EMR_EC2_DefaultRole,SubnetIds=['subnet-
ab12345c','subnet-de67890f'] \
    --instance-fleets
 InstanceFleetType=MASTER,TargetOnDemandCapacity=1,InstanceTypeConfigs=['{InstanceType=m4.la
 InstanceFleetType=CORE,TargetSpotCapacity=11,InstanceTypeConfigs=['{InstanceType=m4.large,B
```

Esempio 4: come creare un cluster con ruoli predefiniti

L'esempio `create-cluster` seguente utilizza il parametro `--use-default-roles` per specificare il ruolo di servizio e il profilo dell'istanza predefiniti.

```
aws emr create-cluster \
    --release-label emr-5.9.0 \
    --use-default-roles \
```

```
    --instance-
groups InstanceGroupType=MASTER,InstanceCount=1,InstanceType=m4.large InstanceGroupType=CORE
 \
    --auto-terminate
```

Esempio 5: come creare un cluster e specificare le applicazioni da installare

L'esempio `create-cluster` seguente utilizza il parametro `--applications` per specificare le applicazioni installate da Amazon EMR. Questo esempio installa Hadoop, Hive e Pig.

```
aws emr create-cluster \
    --applications Name=Hadoop Name=Hive Name=Pig \
    --release-label emr-5.9.0 \
    --instance-
groups InstanceGroupType=MASTER,InstanceCount=1,InstanceType=m4.large InstanceGroupType=CORE
 \
    --auto-terminate
```

Esempio 6: come creare un cluster che include Spark

L'esempio seguente installa Spark.

```
aws emr create-cluster \
    --release-label emr-5.9.0 \
    --applications Name=Spark \
    --ec2-attributes KeyName=myKey \
    --instance-
groups InstanceGroupType=MASTER,InstanceCount=1,InstanceType=m4.large InstanceGroupType=CORE
 \
    --auto-terminate
```

Esempio 7: come specificare un'AMI personalizzata da utilizzare per le istanze del cluster

L'esempio `create-cluster` seguente crea un'istanza del cluster basata sull'AMI Amazon Linux con ID `ami-a518e6df`.

```
aws emr create-cluster \
    --name "Cluster with My Custom AMI" \
    --custom-ami-id ami-a518e6df \
    --ebs-root-volume-size 20 \
    --release-label emr-5.9.0 \
    --use-default-roles \
```

```
    --instance-count 2 \
    --instance-type m4.large
```

Esempio 8: come personalizzare le configurazioni delle applicazioni

Gli esempi seguenti utilizzano il parametro `--configurations` per specificare un file di configurazione JSON che contiene personalizzazioni delle applicazioni per Hadoop. Per ulteriori informazioni, consulta Configurazione delle applicazioni nella Guida alle versioni di Amazon EMR.

Contenuto di `configurations.json`:

```
[
    {
       "Classification": "mapred-site",
       "Properties": {
           "mapred.tasktracker.map.tasks.maximum": 2
       }
    },
    {
        "Classification": "hadoop-env",
        "Properties": {},
        "Configurations": [
            {
                "Classification": "export",
                "Properties": {
                    "HADOOP_DATANODE_HEAPSIZE": 2048,
                    "HADOOP_NAMENODE_OPTS": "-XX:GCTimeRatio=19"
                }
            }
        ]
    }
]
```

L'esempio seguente fa riferimento a `configurations.json` come file locale.

```
aws emr create-cluster \
    --configurations file://configurations.json \
    --release-label emr-5.9.0 \
    --instance-
groups InstanceGroupType=MASTER,InstanceCount=1,InstanceType=m4.large InstanceGroupType=CORE
 \
    --auto-terminate
```

L'esempio seguente fa riferimento a `configurations.json` come file in Amazon S3.

```
aws emr create-cluster \
    --configurations https://s3.amazonaws.com/amzn-s3-demo-bucket/
configurations.json \
    --release-label emr-5.9.0 \
    --instance-
groups InstanceGroupType=MASTER,InstanceCount=1,InstanceType=m4.large InstanceGroupType=CORE
 \
    --auto-terminate
```

Esempio 9: come creare un cluster con gruppi di istanze master, principali e di attività

L'`create-cluster`esempio seguente specifica il `--instance-groups` tipo e il numero di istanze da utilizzare per EC2 i gruppi di istanze master, core e task.

```
aws emr create-cluster \
    --release-label emr-5.9.0 \
    --instance-
groups Name=Master,InstanceGroupType=MASTER,InstanceType=m4.large,InstanceCount=1 Name=Core,
```

Esempio 10: come specificare che un cluster deve essere terminato dopo il completamento di tutte le fasi

L'esempio `create-cluster` seguente utilizza `--auto-terminate` per specificare che il cluster deve arrestarsi automaticamente dopo il completamento di tutte le fasi.

```
aws emr create-cluster \
    --release-label emr-5.9.0 \
    --instance-groups InstanceGroupType=MASTER,InstanceCount=1,InstanceType=m4.large
  InstanceGroupType=CORE,InstanceCount=2,InstanceType=m4.large \
    --auto-terminate
```

Esempio 11: specificare i dettagli della configurazione del cluster come la coppia di EC2 chiavi Amazon, la configurazione di rete e i gruppi di sicurezza

L'`create-cluster`esempio seguente crea un cluster con la coppia di EC2 chiavi Amazon denominata `myKey` e un profilo di istanza personalizzato denominato`myProfile`. Le coppie di chiavi vengono utilizzate per autorizzare connessioni SSH ai nodi del cluster, molto spesso al nodo principale. Per ulteriori informazioni, consulta Use an Amazon EC2 Key Pair for SSH Credentials nella Amazon EMR Management Guide.

```
aws emr create-cluster \
    --ec2-attributes KeyName=myKey,InstanceProfile=myProfile \
    --release-label emr-5.9.0 \
    --instance-
groups InstanceGroupType=MASTER,InstanceCount=1,InstanceType=m4.large InstanceGroupType=CORE
 \
    --auto-terminate
```

L'esempio seguente crea un cluster in una sottorete Amazon VPC.

```
aws emr create-cluster \
    --ec2-attributes SubnetId=subnet-xxxxx \
    --release-label emr-5.9.0 \
    --instance-
groups InstanceGroupType=MASTER,InstanceCount=1,InstanceType=m4.large InstanceGroupType=CORE
 \
    --auto-terminate
```

L'esempio seguente crea un cluster nella zona di disponibilità `us-east-1b`.

```
aws emr create-cluster \
    --ec2-attributes AvailabilityZone=us-east-1b \
    --release-label emr-5.9.0 \
    --instance-
groups InstanceGroupType=MASTER,InstanceCount=1,InstanceType=m4.large InstanceGroupType=CORE
```

L'esempio seguente crea un cluster e specifica solo i gruppi di sicurezza gestiti da Amazon EMR.

```
aws emr create-cluster \
    --release-label emr-5.9.0 \
    --service-role myServiceRole \
    --ec2-attributes InstanceProfile=myRole,EmrManagedMasterSecurityGroup=sg-
master1,EmrManagedSlaveSecurityGroup=sg-slave1 \
    --instance-
groups InstanceGroupType=MASTER,InstanceCount=1,InstanceType=m4.large InstanceGroupType=CORE
```

L'esempio seguente crea un cluster e specifica solo gruppi di EC2 sicurezza Amazon aggiuntivi.

```
aws emr create-cluster \
    --release-label emr-5.9.0 \
    --service-role myServiceRole \
```

```
    --ec2-attributes InstanceProfile=myRole,AdditionalMasterSecurityGroups=[sg-
addMaster1,sg-addMaster2,sg-addMaster3,sg-
addMaster4],AdditionalSlaveSecurityGroups=[sg-addSlave1,sg-addSlave2,sg-
addSlave3,sg-addSlave4] \
    --instance-
groups InstanceGroupType=MASTER,InstanceCount=1,InstanceType=m4.large InstanceGroupType=CORE
```

L'esempio seguente crea un cluster e specifica i gruppi di sicurezza gestiti da EMR, insieme a gruppi di sicurezza aggiuntivi.

```
aws emr create-cluster \
    --release-label emr-5.9.0 \
    --service-role myServiceRole \
    --ec2-attributes InstanceProfile=myRole,EmrManagedMasterSecurityGroup=sg-
master1,EmrManagedSlaveSecurityGroup=sg-slave1,AdditionalMasterSecurityGroups=[sg-
addMaster1,sg-addMaster2,sg-addMaster3,sg-
addMaster4],AdditionalSlaveSecurityGroups=[sg-addSlave1,sg-addSlave2,sg-
addSlave3,sg-addSlave4] \
    --instance-
groups InstanceGroupType=MASTER,InstanceCount=1,InstanceType=m4.large InstanceGroupType=CORE
```

L'esempio seguente crea un cluster in una sottorete privata VPC e utilizza un gruppo di EC2 sicurezza Amazon specifico per abilitare l'accesso al servizio Amazon EMR, necessario per i cluster in sottoreti private.

```
aws emr create-cluster \
    --release-label emr-5.9.0 \
    --service-role myServiceRole \
    --ec2-attributes InstanceProfile=myRole,ServiceAccessSecurityGroup=sg-service-
access,EmrManagedMasterSecurityGroup=sg-master,EmrManagedSlaveSecurityGroup=sg-slave
 \
    --instance-
groups InstanceGroupType=MASTER,InstanceCount=1,InstanceType=m4.large InstanceGroupType=CORE
```

L'esempio seguente specifica i parametri di configurazione del gruppo di sicurezza utilizzando un file JSON denominato `ec2_attributes.json` archiviato in locale. NOTA: gli argomenti JSON devono includere opzioni e valori come elementi propri nell'elenco.

```
aws emr create-cluster \
    --release-label emr-5.9.0 \
    --service-role myServiceRole \
```

```
    --ec2-attributes file://ec2_attributes.json  \
    --instance-
groups InstanceGroupType=MASTER,InstanceCount=1,InstanceType=m4.large InstanceGroupType=CORE
```

Contenuto di `ec2_attributes.json`:

```
[
    {
        "SubnetId": "subnet-xxxxx",
        "KeyName": "myKey",
        "InstanceProfile":"myRole",
        "EmrManagedMasterSecurityGroup": "sg-master1",
        "EmrManagedSlaveSecurityGroup": "sg-slave1",
        "ServiceAccessSecurityGroup": "sg-service-access",
        "AdditionalMasterSecurityGroups": ["sg-addMaster1","sg-addMaster2","sg-
addMaster3","sg-addMaster4"],
        "AdditionalSlaveSecurityGroups": ["sg-addSlave1","sg-addSlave2","sg-
addSlave3","sg-addSlave4"]
    }
]
```

Esempio 12: come abilitare il debug e specificare un URI di log

L'esempio `create-cluster` seguente utilizza il parametro `--enable-debugging`, che permette di visualizzare i file di log più facilmente utilizzando lo strumento di debug nella console Amazon EMR. Il parametro `--log-uri` è obbligatorio con `--enable-debugging`.

```
aws emr create-cluster \
    --enable-debugging \
    --log-uri s3://amzn-s3-demo-bucket/myLog \
    --release-label emr-5.9.0 \
    --instance-
groups InstanceGroupType=MASTER,InstanceCount=1,InstanceType=m4.large InstanceGroupType=CORE
 \
    --auto-terminate
```

Esempio 13: come aggiungere tag durante la creazione di un cluster

I tag sono coppie chiave-valore che permettono di individuare e gestire i cluster. L'esempio `create-cluster` seguente utilizza il parametro `--tags` per creare tre tag per un cluster, uno con il nome di chiave `name` e il valore `Shirley Rodriguez`, un secondo con il nome di chiave `age` e il valore 29 e un terzo tag con il nome di chiave `department` e il valore `Analytics`.

```
aws emr create-cluster \
    --tags name="Shirley Rodriguez" age=29 department="Analytics" \
    --release-label emr-5.32.0 \
    --instance-type m5.xlarge \
    --instance-count 3 \
    --use-default-roles
```

L'esempio seguente elenca i tag applicati a un cluster.

```
aws emr describe-cluster \
    --cluster-id j-XXXXXXYY \
    --query Cluster.Tags
```

Esempio 14: come utilizzare una configurazione di sicurezza che abilita la crittografia e altre funzionalità di sicurezza

L'esempio `create-cluster` seguente utilizza il parametro `--security-configuration` per specificare una configurazione di sicurezza per un cluster EMR. Puoi utilizzare configurazioni di sicurezza con Amazon EMR versione 4.8.0 o successive.

```
aws emr create-cluster \
    --instance-type m4.large \
    --release-label emr-5.9.0 \
    --security-configuration mySecurityConfiguration
```

Esempio 15: come creare un cluster con volumi di archiviazione EBS aggiuntivi configurati per i gruppi di istanze

Per specificare volumi EBS aggiuntivi, sono necessari questi argomenti: `VolumeType` e `SizeInGB` se è specificato `EbsBlockDeviceConfigs`.

L'`create-cluster`esempio seguente crea un cluster con più volumi EBS collegati alle EC2 istanze del gruppo di istanze principale.

```
aws emr create-cluster \
    --release-label emr-5.9.0  \
    --use-default-roles \
    --instance-
groups InstanceGroupType=MASTER,InstanceCount=1,InstanceType=d2.xlarge
 'InstanceGroupType=CORE,InstanceCount=2,InstanceType=d2.xlarge,EbsConfiguration={EbsOptimiz
```

```
{VolumeSpecification={VolumeType=io1,SizeInGB=100,Iops=100},VolumesPerInstance=4}]}'
 \
    --auto-terminate
```

L'esempio seguente crea un cluster con più volumi EBS collegati alle EC2 istanze del gruppo di istanze principale.

```
aws emr create-cluster \
    --release-label emr-5.9.0 \
    --use-default-roles \
    --instance-groups 'InstanceGroupType=MASTER, InstanceCount=1,
 InstanceType=d2.xlarge, EbsConfiguration={EbsOptimized=true,
 EbsBlockDeviceConfigs=[{VolumeSpecification={VolumeType=io1, SizeInGB=100,
 Iops=100}},
{VolumeSpecification={VolumeType=standard,SizeInGB=50},VolumesPerInstance=3}]}' InstanceGrou
 \
    --auto-terminate
```

Esempio 16: come creare un cluster con una policy di dimensionamento automatico

Puoi collegare policy di dimensionamento automatico a gruppi di istanze principali e di attività utilizzando Amazon EMR versione 4.0 e successive. La policy di scalabilità automatica aggiunge e rimuove dinamicamente EC2 le istanze in risposta a un parametro Amazon. CloudWatch Per ulteriori informazioni, consulta Using Automatic Scaling in Amazon EMR https://docs.aws.amazon.com/emr/ latest/ManagementGuide/emr < -automatic-scaling.html>`_ nella Amazon EMR Management Guide.

Per collegare una policy di dimensionamento automatico, devi specificare anche il ruolo predefinito per il dimensionamento automatico utilizzando `--auto-scaling-role EMR_AutoScaling_DefaultRole`.

L'esempio `create-cluster` seguente specifica la policy di dimensionamento automatico per il gruppo di istanze `CORE` utilizzando l'argomento `AutoScalingPolicy` con una struttura JSON incorporata, che specifica la configurazione della policy di dimensionamento. Per i gruppi di istanze con una struttura JSON incorporata, l'intera raccolta di argomenti deve essere racchiusa tra virgolette singole. L'utilizzo delle virgolette singole è facoltativo per i gruppi di istanze senza una struttura JSON incorporata.

```
aws emr create-cluster
    --release-label emr-5.9.0 \
```

```
    --use-default-roles --auto-scaling-role EMR_AutoScaling_DefaultRole \
    --instance-
groups InstanceGroupType=MASTER,InstanceType=d2.xlarge,InstanceCount=1
 'InstanceGroupType=CORE,InstanceType=d2.xlarge,InstanceCount=2,AutoScalingPolicy={Constrain
```

L'esempio seguente utilizza un file JSON denominato `instancegroupconfig.json` per specificare la configurazione di tutti i gruppi di istanze in un cluster. Il file JSON specifica la configurazione della policy di dimensionamento automatico per il gruppo di istanze principali.

```
aws emr create-cluster \
    --release-label emr-5.9.0 \
    --service-role EMR_DefaultRole \
    --ec2-attributes InstanceProfile=EMR_EC2_DefaultRole \
    --instance-groups file://myfolder/instancegroupconfig.json \
    --auto-scaling-role EMR_AutoScaling_DefaultRole
```

Contenuto di `instancegroupconfig.json`:

```
[
    {
        "InstanceCount": 1,
        "Name": "MyMasterIG",
        "InstanceGroupType": "MASTER",
        "InstanceType": "m4.large"
    },
    {
        "InstanceCount": 2,
        "Name": "MyCoreIG",
        "InstanceGroupType": "CORE",
        "InstanceType": "m4.large",
        "AutoScalingPolicy": {
            "Constraints": {
                "MinCapacity": 2,
                "MaxCapacity": 10
            },
            "Rules": [
                {
                    "Name": "Default-scale-out",
                    "Description": "Replicates the default scale-out rule in the
 console for YARN memory.",
                    "Action": {
                        "SimpleScalingPolicyConfiguration": {
```

```
                            "AdjustmentType": "CHANGE_IN_CAPACITY",
                            "ScalingAdjustment": 1,
                            "CoolDown": 300
                        }
                    },
                    "Trigger": {
                        "CloudWatchAlarmDefinition": {
                            "ComparisonOperator": "LESS_THAN",
                            "EvaluationPeriods": 1,
                            "MetricName": "YARNMemoryAvailablePercentage",
                            "Namespace": "AWS/ElasticMapReduce",
                            "Period": 300,
                            "Threshold": 15,
                            "Statistic": "AVERAGE",
                            "Unit": "PERCENT",
                            "Dimensions": [
                                {
                                    "Key": "JobFlowId",
                                    "Value": "${emr.clusterId}"
                                }
                            ]
                        }
                    }
                }
            ]
        }
    }
]
```

Esempio 17: aggiunge fasi JAR personalizzate durante la creazione di un cluster

L'esempio `create-cluster` seguente aggiunge fasi specificando un file JAR archiviato in Amazon S3. Le fasi inviano il lavoro a un cluster. La funzione principale definita nel file JAR viene eseguita dopo il provisioning delle istanze, l'esecuzione di eventuali azioni di bootstrap e l' EC2 installazione delle applicazioni. Le fasi vengono specificate utilizzando `Type=CUSTOM_JAR`.

Le fasi JAR personalizzate richiedono il parametro `Jar=`, che specifica il percorso e il nome del file JAR. I parametri facoltativi sono `Type`, `Name`, `ActionOnFailure`, `Args` e `MainClass`. Se non è indicata la classe principale, il file JAR deve specificare `Main-Class` nel proprio file manifesto.

```
aws emr create-cluster \
```

```
    --steps Type=CUSTOM_JAR,Name=CustomJAR,ActionOnFailure=CONTINUE,Jar=s3://amzn-
s3-demo-bucket/
mytest.jar,Args=arg1,arg2,arg3 Type=CUSTOM_JAR,Name=CustomJAR,ActionOnFailure=CONTINUE,Jar=s
amzn-s3-demo-bucket/mytest.jar,MainClass=mymainclass,Args=arg1,arg2,arg3  \
    --release-label emr-5.3.1 \
    --instance-
groups InstanceGroupType=MASTER,InstanceCount=1,InstanceType=m4.large InstanceGroupType=CORE
 \
    --auto-terminate
```

Esempio 18: come aggiungere fasi di streaming durante la creazione di un cluster

Gli esempi `create-cluster` seguenti aggiungono una fase di streaming a un cluster che viene terminato dopo l'esecuzione di tutte le fasi. Le fasi di streaming richiedono i parametri `Type` e `Args`. I parametri facoltativi per le fasi di streaming sono `Name` e `ActionOnFailure`.

L'esempio seguente specifica la fase inline.

```
aws emr create-cluster \
    --steps Type=STREAMING,Name='Streaming Program',ActionOnFailure=CONTINUE,Args=[-
files,s3://elasticmapreduce/samples/wordcount/wordSplitter.py,-
mapper,wordSplitter.py,-reducer,aggregate,-input,s3://elasticmapreduce/samples/
wordcount/input,-output,s3://amzn-s3-demo-bucket/wordcount/output] \
    --release-label emr-5.3.1 \
    --instance-
groups InstanceGroupType=MASTER,InstanceCount=1,InstanceType=m4.large InstanceGroupType=CORE
 \
    --auto-terminate
```

L'esempio seguente utilizza un file di configurazione JSON archiviato in locale denominato `multiplefiles.json`. La configurazione JSON specifica più file. Per specificare più file all'interno di una fase, devi utilizzare un file di configurazione JSON per specificare la fase. Gli argomenti JSON devono includere opzioni e valori come elementi propri nell'elenco.

```
aws emr create-cluster \
    --steps file://./multiplefiles.json \
    --release-label emr-5.9.0  \
    --instance-
groups InstanceGroupType=MASTER,InstanceCount=1,InstanceType=m4.large InstanceGroupType=CORE
 \
    --auto-terminate
```

Contenuto di `multiplefiles.json`:

```
[
    {
        "Name": "JSON Streaming Step",
        "Args": [
            "-files",
            "s3://elasticmapreduce/samples/wordcount/wordSplitter.py",
            "-mapper",
            "wordSplitter.py",
            "-reducer",
            "aggregate",
            "-input",
            "s3://elasticmapreduce/samples/wordcount/input",
            "-output",
            "s3://amzn-s3-demo-bucket/wordcount/output"
        ],
        "ActionOnFailure": "CONTINUE",
        "Type": "STREAMING"
    }
]
```

Esempio 19: come aggiungere fasi Hive durante la creazione di un cluster

L'esempio seguente aggiunge fasi Hive durante la creazione di un cluster. Le fasi Hive richiedono i parametri `Type` e `Args`. I parametri facoltativi per le fasi Hive sono `Name` e `ActionOnFailure`.

```
aws emr create-cluster \
    --steps Type=HIVE,Name='Hive
 program',ActionOnFailure=CONTINUE,ActionOnFailure=TERMINATE_CLUSTER,Args=[-
f,s3://elasticmapreduce/samples/hive-ads/libs/model-build.q,-d,INPUT=s3://
elasticmapreduce/samples/hive-ads/tables,-d,OUTPUT=s3://amzn-s3-demo-bucket/hive-
ads/output/2014-04-18/11-07-32,-d,LIBS=s3://elasticmapreduce/samples/hive-ads/libs]
 \
    --applications Name=Hive \
    --release-label emr-5.3.1 \
    --instance-
groups InstanceGroupType=MASTER,InstanceCount=1,InstanceType=m4.large InstanceGroupType=CORE
```

Esempio 20: come aggiungere fasi Pig durante la creazione di un cluster

L'esempio seguente aggiunge fasi Pig durante la creazione di un cluster. I parametri obbligatori per le fasi Pig sono `Type` e `Args`. I parametri facoltativi per le fasi Pig sono `Name` e `ActionOnFailure`.

```
aws emr create-cluster \
    --steps Type=PIG,Name='Pig program',ActionOnFailure=CONTINUE,Args=[-f,s3://
elasticmapreduce/samples/pig-apache/do-reports2.pig,-p,INPUT=s3://elasticmapreduce/
samples/pig-apache/input,-p,OUTPUT=s3://amzn-s3-demo-bucket/pig-apache/output] \
    --applications Name=Pig \
    --release-label emr-5.3.1 \
    --instance-
groups InstanceGroupType=MASTER,InstanceCount=1,InstanceType=m4.large InstanceGroupType=CORE
```

Esempio 21: come aggiungere azioni di bootstrap

L'esempio `create-cluster` seguente esegue due azioni di bootstrap definite come script archiviati in Amazon S3.

```
aws emr create-cluster \
    --bootstrap-actions Path=s3://amzn-s3-demo-bucket/
myscript1,Name=BootstrapAction1,Args=[arg1,arg2] Path=s3://amzn-s3-demo-bucket/
myscript2,Name=BootstrapAction2,Args=[arg1,arg2] \
    --release-label emr-5.3.1 \
    --instance-
groups InstanceGroupType=MASTER,InstanceCount=1,InstanceType=m4.large InstanceGroupType=CORE
 \
    --auto-terminate
```

Esempio 22: per abilitare la visualizzazione coerente di EMRFS e personalizzare le impostazioni e RetryCount RetryPeriod

L'esempio `create-cluster` seguente specifica il numero e il periodo di nuovi tentativi per la visualizzazione EMRFS coerente. L'argomento `Consistent=true` è obbligatorio.

```
aws emr create-cluster \
    --instance-type m4.large \
    --release-label emr-5.9.0 \
    --emrfs Consistent=true,RetryCount=6,RetryPeriod=30
```

L'esempio seguente specifica la stessa configurazione EMRFS dell'esempio precedente, utilizzando un file di configurazione JSON archiviato in locale denominato `emrfsconfig.json`.

```
aws emr create-cluster \
    --instance-type m4.large \
    --release-label emr-5.9.0 \
    --emrfs file://emrfsconfig.json
```

Contenuto di `emrfsconfig.json`:

```
{
    "Consistent": true,
    "RetryCount": 6,
    "RetryPeriod": 30
}
```

Esempio 23: come creare un cluster con Kerberos configurato

Gli esempi `create-cluster` seguenti creano un cluster utilizzando una configurazione di sicurezza con Kerberos abilitato e definiscono i parametri Kerberos per il cluster utilizzando `--kerberos-attributes`.

Il comando seguente specifica gli attributi Kerberos per il cluster inline.

```
aws emr create-cluster \
    --instance-type m3.xlarge \
    --release-label emr-5.10.0 \
    --service-role EMR_DefaultRole \
    --ec2-attributes InstanceProfile=EMR_EC2_DefaultRole \
    --security-configuration mySecurityConfiguration \
    --kerberos-
attributes Realm=EC2.INTERNAL,KdcAdminPassword=123,CrossRealmTrustPrincipalPassword=123
```

Il comando seguente specifica gli stessi attributi, ma fa riferimento a un file JSON archiviato in locale denominato `kerberos_attributes.json`. In questo esempio il file viene salvato nella stessa directory in cui viene eseguito il comando. Puoi fare riferimento anche a un file di configurazione salvato in Amazon S3.

```
aws emr create-cluster \
    --instance-type m3.xlarge \
    --release-label emr-5.10.0 \
    --service-role EMR_DefaultRole \
    --ec2-attributes InstanceProfile=EMR_EC2_DefaultRole \
    --security-configuration mySecurityConfiguration \
```

```
    --kerberos-attributes file://kerberos_attributes.json
```

Contenuto di `kerberos_attributes.json`:

```
{
    "Realm": "EC2.INTERNAL",
    "KdcAdminPassword": "123",
    "CrossRealmTrustPrincipalPassword": "123",
}
```

L'esempio `create-cluster` seguente crea un cluster Amazon EMR che utilizza la configurazione `--instance-groups` e include una policy di dimensionamento gestito.

```
aws emr create-cluster \
    --release-label emr-5.30.0 \
    --service-role EMR_DefaultRole \
    --ec2-attributes InstanceProfile=EMR_EC2_DefaultRole \
    --instance-
groups InstanceGroupType=MASTER,InstanceCount=1,InstanceType=m4.large InstanceGroupType=CORE
    --managed-scaling-policy
 ComputeLimits='{MinimumCapacityUnits=2,MaximumCapacityUnits=4,UnitType=Instances}'
```

L'`create-cluster`esempio seguente crea un cluster Amazon EMR che utilizza «-- log-encryption-kms-key -id» per definire l'ID della chiave KMS utilizzato per la crittografia dei log.

```
aws emr create-cluster \
    --release-label emr-5.30.0 \
    --log-uri s3://amzn-s3-demo-bucket/myLog \
    --log-encryption-kms-key-id arn:aws:kms:us-east-1:110302272565:key/
dd559181-283e-45d7-99d1-66da348c4d33 \
    --instance-
groups InstanceGroupType=MASTER,InstanceCount=1,InstanceType=m4.large InstanceGroupType=CORE
```

L'`create-cluster`esempio seguente crea un cluster Amazon EMR che utilizza la configurazione «--placement-group-configs" per posizionare i nodi master in un cluster ad alta disponibilità (HA) all'interno di un gruppo di collocamento utilizzando `SPREAD` la strategia di EC2 posizionamento.

```
aws emr create-cluster \
    --release-label emr-5.30.0 \
    --service-role EMR_DefaultRole \
```

```
    --ec2-attributes InstanceProfile=EMR_EC2_DefaultRole \
    --instance-
groups InstanceGroupType=MASTER,InstanceCount=3,InstanceType=m4.largeInstanceGroupType=CORE,
 \
    --placement-group-configs InstanceRole=MASTER
```

L'`create-cluster`esempio seguente crea un cluster Amazon EMR che utilizza la configurazione «--auto-termination-policy" per impostare una soglia di terminazione automatica dei periodi di inattività per il cluster.

```
aws emr create-cluster \
    --release-label emr-5.34.0 \
    --service-role EMR_DefaultRole \
    --ec2-attributes InstanceProfile=EMR_EC2_DefaultRole \
    --instance-
groups InstanceGroupType=MASTER,InstanceCount=1,InstanceType=m4.large InstanceGroupType=CORE
 \
    --auto-termination-policy IdleTimeout=100
```

L'`create-cluster`esempio seguente crea un cluster Amazon EMR che utilizza il «--os-release-label" per definire una versione di Amazon Linux per il lancio del cluster

```
aws emr create-cluster \
    --release-label emr-6.6.0 \
    --os-release-label 2.0.20220406.1 \
    --service-role EMR_DefaultRole \
    --ec2-attributes InstanceProfile=EMR_EC2_DefaultRole \
    --instance-
groups InstanceGroupType=MASTER,InstanceCount=1,InstanceType=m4.large InstanceGroupType=CORE
```

Esempio 24: come specificare gli attributi di un volume root EBS size, iops e throughput per le istanze del cluster create con EMR versione 6.15.0 e successive

L'`create-cluster`esempio seguente crea un cluster Amazon EMR che utilizza gli attributi del volume root per configurare le specifiche dei volumi root per le EC2 istanze.

```
aws emr create-cluster \
    --name "Cluster with My Custom AMI" \
    --custom-ami-id ami-a518e6df \
    --ebs-root-volume-size 20 \
    --ebs-root-volume-iops 3000 \
    --ebs-root-volume-throughput 125 \
```

```
    --release-label emr-6.15.0 \
    --use-default-roles \
    --instance-count 2 \
    --instance-type m4.large
```

- Per i dettagli sull'API, consulta AWS CLI Command CreateClusterExamplesReference.

`create-default-roles`

Il seguente esempio di codice mostra come utilizzare`create-default-roles`.

AWS CLI

1. Per creare il ruolo IAM predefinito per EC2

   Comando:

   ```
   aws emr create-default-roles
   ```

   Output:

   ```
   If the role already exists then the command returns nothing.

   If the role does not exist then the output will be:

   [
       {
           "RolePolicy": {
               "Version":"2012-10-17",
               "Statement": [
                   {
                       "Action": [
                           "cloudwatch:*",
                           "dynamodb:*",
                           "ec2:Describe*",
                           "elasticmapreduce:Describe*",
                           "elasticmapreduce:ListBootstrapActions",
                           "elasticmapreduce:ListClusters",
                           "elasticmapreduce:ListInstanceGroups",
                           "elasticmapreduce:ListInstances",
                           "elasticmapreduce:ListSteps",
                           "kinesis:CreateStream",
                           "kinesis:DeleteStream",
   ```

```
                    "kinesis:DescribeStream",
                    "kinesis:GetRecords",
                    "kinesis:GetShardIterator",
                    "kinesis:MergeShards",
                    "kinesis:PutRecord",
                    "kinesis:SplitShard",
                    "rds:Describe*",
                    "s3:*",
                    "sdb:*",
                    "sns:*",
                    "sqs:*"
                ],
                "Resource": "*",
                "Effect": "Allow"
            }
        ]
    },
    "Role": {
        "AssumeRolePolicyDocument": {
            "Version":"2012-10-17",
            "Statement": [
                {
                    "Action": "sts:AssumeRole",
                    "Sid": "",
                    "Effect": "Allow",
                    "Principal": {
                        "Service": "ec2.amazonaws.com"
                    }
                }
            ]
        },
        "RoleId": "AROAIQ5SIQUGL5KMYBJX6",
        "CreateDate": "2015-06-09T17:09:04.602Z",
        "RoleName": "EMR_EC2_DefaultRole",
        "Path": "/",
        "Arn": "arn:aws:iam::176430881729:role/EMR_EC2_DefaultRole"
    }
},
{
    "RolePolicy": {
        "Version":"2012-10-17",
        "Statement": [
            {
                "Action": [
```

```
                    "ec2:AuthorizeSecurityGroupIngress",
                    "ec2:CancelSpotInstanceRequests",
                    "ec2:CreateSecurityGroup",
                    "ec2:CreateTags",
                    "ec2:DeleteTags",
                    "ec2:DescribeAvailabilityZones",
                    "ec2:DescribeAccountAttributes",
                    "ec2:DescribeInstances",
                    "ec2:DescribeInstanceStatus",
                    "ec2:DescribeKeyPairs",
                    "ec2:DescribePrefixLists",
                    "ec2:DescribeRouteTables",
                    "ec2:DescribeSecurityGroups",
                    "ec2:DescribeSpotInstanceRequests",
                    "ec2:DescribeSpotPriceHistory",
                    "ec2:DescribeSubnets",
                    "ec2:DescribeVpcAttribute",
                    "ec2:DescribeVpcEndpoints",
                    "ec2:DescribeVpcEndpointServices",
                    "ec2:DescribeVpcs",
                    "ec2:ModifyImageAttribute",
                    "ec2:ModifyInstanceAttribute",
                    "ec2:RequestSpotInstances",
                    "ec2:RunInstances",
                    "ec2:TerminateInstances",
                    "iam:GetRole",
                    "iam:GetRolePolicy",
                    "iam:ListInstanceProfiles",
                    "iam:ListRolePolicies",
                    "iam:PassRole",
                    "s3:CreateBucket",
                    "s3:Get*",
                    "s3:List*",
                    "sdb:BatchPutAttributes",
                    "sdb:Select",
                    "sqs:CreateQueue",
                    "sqs:Delete*",
                    "sqs:GetQueue*",
                    "sqs:ReceiveMessage"
                ],
                "Resource": "*",
                "Effect": "Allow"
            }
        ]
```

```
        },
        "Role": {
            "AssumeRolePolicyDocument": {
                "Version":"2012-10-17",
                "Statement": [
                    {
                        "Action": "sts:AssumeRole",
                        "Sid": "",
                        "Effect": "Allow",
                        "Principal": {
                            "Service": "elasticmapreduce.amazonaws.com"
                        }
                    }
                ]
            },
            "RoleId": "AROAI3SRVPPVSRDLARBPY",
            "CreateDate": "2015-06-09T17:09:10.401Z",
            "RoleName": "EMR_DefaultRole",
            "Path": "/",
            "Arn": "arn:aws:iam::176430881729:role/EMR_DefaultRole"
        }
    }
]
```

- Per i dettagli sull'API, consulta CreateDefaultRoles AWS CLICommand Reference.

`create-security-configuration`

Il seguente esempio di codice mostra come utilizzare`create-security-configuration`.

AWS CLI

1. Per creare una configurazione di sicurezza con crittografia in transito abilitata con PEM per il fornitore di certificati e crittografia a riposo abilitata con SSE-S3 per la crittografia S3 e -KMS per il provider di chiavi del disco locale AWS

   Comando:

```
 aws emr create-security-configuration --name MySecurityConfig --security-
configuration '{
        "EncryptionConfiguration": {
                "EnableInTransitEncryption" : true,
                "EnableAtRestEncryption" : true,
```

```
                "InTransitEncryptionConfiguration" : {
                        "TLSCertificateConfiguration" : {
                                "CertificateProviderType" : "PEM",
                                "S3Object" : "s3://mycertstore/artifacts/
MyCerts.zip"
                        }
                },
                "AtRestEncryptionConfiguration" : {
                        "S3EncryptionConfiguration" : {
                                "EncryptionMode" : "SSE-S3"
                        },
                        "LocalDiskEncryptionConfiguration" : {
                                "EncryptionKeyProviderType" : "AwsKms",
                                "AwsKmsKey" : "arn:aws:kms:us-
east-1:123456789012:key/12345678-1234-1234-1234-123456789012"
                        }
                }
        }
}'
```

Output:

```
{
"CreationDateTime": 1474070889.129,
"Name": "MySecurityConfig"
}
```

Equivalente JSON (contenuto di security_configuration.json):

```
{
    "EncryptionConfiguration": {
        "EnableInTransitEncryption": true,
        "EnableAtRestEncryption": true,
        "InTransitEncryptionConfiguration": {
            "TLSCertificateConfiguration": {
                "CertificateProviderType": "PEM",
                "S3Object": "s3://mycertstore/artifacts/MyCerts.zip"
            }
        },
        "AtRestEncryptionConfiguration": {
            "S3EncryptionConfiguration": {
                "EncryptionMode": "SSE-S3"
            },
```

```
            "LocalDiskEncryptionConfiguration": {
                "EncryptionKeyProviderType": "AwsKms",
                "AwsKmsKey": "arn:aws:kms:us-
east-1:123456789012:key/12345678-1234-1234-1234-123456789012"
            }
        }
    }
}
```

Comando (utilizzando security_configuration.json):

```
aws emr create-security-configuration --name "MySecurityConfig" --security-
configuration file://./security_configuration.json
```

Output:

```
{
"CreationDateTime": 1474070889.129,
"Name": "MySecurityConfig"
}
```

2. Come creare una configurazione di sicurezza con Kerberos abilitato utilizzando KDC dedicato per il cluster e attendibilità tra domini

Comando:

```
 aws emr create-security-configuration --name MySecurityConfig --security-
configuration '{
     "AuthenticationConfiguration": {
         "KerberosConfiguration": {
             "Provider": "ClusterDedicatedKdc",
             "ClusterDedicatedKdcConfiguration": {
                 "TicketLifetimeInHours": 24,
                 "CrossRealmTrustConfiguration": {
                   "Realm": "AD.DOMAIN.COM",
                   "Domain": "ad.domain.com",
                   "AdminServer": "ad.domain.com",
                   "KdcServer": "ad.domain.com"
                 }
             }
         }
     }
```

```
}'
```

Output:

```
{
"CreationDateTime": 1490225558.982,
"Name": "MySecurityConfig"
}
```

Equivalente JSON (contenuto di security_configuration.json):

```
{
    "AuthenticationConfiguration": {
        "KerberosConfiguration": {
            "Provider": "ClusterDedicatedKdc",
            "ClusterDedicatedKdcConfiguration": {
                "TicketLifetimeInHours": 24,
                "CrossRealmTrustConfiguration": {
                    "Realm": "AD.DOMAIN.COM",
                    "Domain": "ad.domain.com",
                    "AdminServer": "ad.domain.com",
                    "KdcServer": "ad.domain.com"
                }
            }
        }
    }
}
```

Comando (utilizzando security_configuration.json):

```
aws emr create-security-configuration --name "MySecurityConfig" --security-
configuration file://./security_configuration.json
```

Output:

```
{
"CreationDateTime": 1490225558.982,
"Name": "MySecurityConfig"
}
```

- Per i dettagli sull'API, consulta CreateSecurityConfiguration AWS CLICommand Reference.

`delete-security-configuration`

Il seguente esempio di codice mostra come utilizzare`delete-security-configuration`.

AWS CLI

Come eliminare una configurazione di sicurezza nella Regione corrente

Comando:

```
aws emr delete-security-configuration --name MySecurityConfig
```

Output:

```
None
```

- Per i dettagli sull'API, consulta DeleteSecurityConfiguration AWS CLICommand Reference.

`describe-cluster`

Il seguente esempio di codice mostra come utilizzare`describe-cluster`.

AWS CLI

Comando:

```
aws emr describe-cluster --cluster-id j-XXXXXXXX
```

Output:

```
For release-label based uniform instance groups cluster:

        {
            "Cluster": {
                "Status": {
                    "Timeline": {
                        "ReadyDateTime": 1436475075.199,
                        "CreationDateTime": 1436474656.563,
                    },
                    "State": "WAITING",
                    "StateChangeReason": {
                        "Message": "Waiting for steps to run"
```

```
            }
        },
        "Ec2InstanceAttributes": {
            "ServiceAccessSecurityGroup": "sg-xxxxxxxx",
            "EmrManagedMasterSecurityGroup": "sg-xxxxxxxx",
            "IamInstanceProfile": "EMR_EC2_DefaultRole",
            "Ec2KeyName": "myKey",
            "Ec2AvailabilityZone": "us-east-1c",
            "EmrManagedSlaveSecurityGroup": "sg-yyyyyyyyy"
        },
        "Name": "My Cluster",
        "ServiceRole": "EMR_DefaultRole",
        "Tags": [],
        "TerminationProtected": true,
        "UnhealthyNodeReplacement": true,
        "ReleaseLabel": "emr-4.0.0",
        "NormalizedInstanceHours": 96,
        "InstanceGroups": [
            {
                "RequestedInstanceCount": 2,
                "Status": {
                    "Timeline": {
                        "ReadyDateTime": 1436475074.245,
                        "CreationDateTime": 1436474656.564,
                        "EndDateTime": 1436638158.387
                    },
                    "State": "RUNNING",
                    "StateChangeReason": {
                        "Message": "",
                    }
                },
                "Name": "CORE",
                "InstanceGroupType": "CORE",
                "Id": "ig-YYYYYYY",
                "Configurations": [],
                "InstanceType": "m3.large",
                "Market": "ON_DEMAND",
                "RunningInstanceCount": 2
            },
            {
                "RequestedInstanceCount": 1,
                "Status": {
                    "Timeline": {
                        "ReadyDateTime": 1436475074.245,
```

```
                            "CreationDateTime": 1436474656.564,
                            "EndDateTime": 1436638158.387
                        },
                        "State": "RUNNING",
                        "StateChangeReason": {
                            "Message": "",
                        }
                    },
                    "Name": "MASTER",
                    "InstanceGroupType": "MASTER",
                    "Id": "ig-XXXXXXXXX",
                    "Configurations": [],
                    "InstanceType": "m3.large",
                    "Market": "ON_DEMAND",
                    "RunningInstanceCount": 1
                }
            ],
            "Applications": [
                {
                    "Name": "Hadoop"
                }
            ],
            "VisibleToAllUsers": true,
            "BootstrapActions": [],
            "MasterPublicDnsName": "ec2-54-147-144-78.compute-1.amazonaws.com",
            "AutoTerminate": false,
            "Id": "j-XXXXXXXX",
            "Configurations": [
                {
                    "Properties": {
                        "fs.s3.consistent.retryPeriodSeconds": "20",
                        "fs.s3.enableServerSideEncryption": "true",
                        "fs.s3.consistent": "false",
                        "fs.s3.consistent.retryCount": "2"
                    },
                    "Classification": "emrfs-site"
                }
            ]
        }
    }


For release-label based instance fleet cluster:
{
```

```
    "Cluster": {
        "Status": {
            "Timeline": {
                "ReadyDateTime": 1487897289.705,
                "CreationDateTime": 1487896933.942
            },
            "State": "WAITING",
            "StateChangeReason": {
                "Message": "Waiting for steps to run"
            }
        },
        "Ec2InstanceAttributes": {
            "EmrManagedMasterSecurityGroup": "sg-xxxxx",
            "RequestedEc2AvailabilityZones": [],
            "RequestedEc2SubnetIds": [],
            "IamInstanceProfile": "EMR_EC2_DefaultRole",
            "Ec2AvailabilityZone": "us-east-1a",
            "EmrManagedSlaveSecurityGroup": "sg-xxxxx"
        },
        "Name": "My Cluster",
        "ServiceRole": "EMR_DefaultRole",
        "Tags": [],
        "TerminationProtected": false,
        "UnhealthyNodeReplacement": false,
        "ReleaseLabel": "emr-5.2.0",
        "NormalizedInstanceHours": 472,
        "InstanceCollectionType": "INSTANCE_FLEET",
        "InstanceFleets": [
            {
                "Status": {
                    "Timeline": {
                        "ReadyDateTime": 1487897212.74,
                        "CreationDateTime": 1487896933.948
                    },
                    "State": "RUNNING",
                    "StateChangeReason": {
                        "Message": ""
                    }
                },
                "ProvisionedSpotCapacity": 1,
                "Name": "MASTER",
                "InstanceFleetType": "MASTER",
                "LaunchSpecifications": {
                    "SpotSpecification": {
```

```
                        "TimeoutDurationMinutes": 60,
                        "TimeoutAction": "TERMINATE_CLUSTER"
                    }
                },
                "TargetSpotCapacity": 1,
                "ProvisionedOnDemandCapacity": 0,
                "InstanceTypeSpecifications": [
                    {
                        "BidPrice": "0.5",
                        "InstanceType": "m3.xlarge",
                        "WeightedCapacity": 1
                    }
                ],
                "Id": "if-xxxxxxx",
                "TargetOnDemandCapacity": 0
            }
        ],
        "Applications": [
            {
                "Version": "2.7.3",
                "Name": "Hadoop"
            }
        ],
        "ScaleDownBehavior": "TERMINATE_AT_INSTANCE_HOUR",
        "VisibleToAllUsers": true,
        "BootstrapActions": [],
        "MasterPublicDnsName": "ec2-xxx-xx-xxx-xx.compute-1.amazonaws.com",
        "AutoTerminate": false,
        "Id": "j-xxxxx",
        "Configurations": []
    }
}

For ami based uniform instance group cluster:

    {
        "Cluster": {
            "Status": {
                "Timeline": {
                    "ReadyDateTime": 1399400564.432,
                    "CreationDateTime": 1399400268.62
                },
                "State": "WAITING",
                "StateChangeReason": {
```

```
                "Message": "Waiting for steps to run"
            }
        },
        "Ec2InstanceAttributes": {
            "IamInstanceProfile": "EMR_EC2_DefaultRole",
            "Ec2AvailabilityZone": "us-east-1c"
        },
        "Name": "My Cluster",
        "Tags": [],
        "TerminationProtected": true,
        "UnhealthyNodeReplacement": true,
        "RunningAmiVersion": "2.5.4",
        "InstanceGroups": [
            {
                "RequestedInstanceCount": 1,
                "Status": {
                    "Timeline": {
                        "ReadyDateTime": 1399400558.848,
                        "CreationDateTime": 1399400268.621
                    },
                    "State": "RUNNING",
                    "StateChangeReason": {
                        "Message": ""
                    }
                },
                "Name": "Master instance group",
                "InstanceGroupType": "MASTER",
                "InstanceType": "m1.small",
                "Id": "ig-ABCD",
                "Market": "ON_DEMAND",
                "RunningInstanceCount": 1
            },
            {
                "RequestedInstanceCount": 2,
                "Status": {
                    "Timeline": {
                        "ReadyDateTime": 1399400564.439,
                        "CreationDateTime": 1399400268.621
                    },
                    "State": "RUNNING",
                    "StateChangeReason": {
                        "Message": ""
                    }
                },
```

```
                    "Name": "Core instance group",
                    "InstanceGroupType": "CORE",
                    "InstanceType": "m1.small",
                    "Id": "ig-DEF",
                    "Market": "ON_DEMAND",
                    "RunningInstanceCount": 2
                }
            ],
            "Applications": [
                {
                    "Version": "1.0.3",
                    "Name": "hadoop"
                }
            ],
            "BootstrapActions": [],
            "VisibleToAllUsers": false,
            "RequestedAmiVersion": "2.4.2",
            "LogUri": "s3://myLogUri/",
            "AutoTerminate": false,
            "Id": "j-XXXXXXXX"
        }
    }
```

- Per i dettagli sull'API, consulta DescribeCluster AWS CLICommand Reference.

`describe-step`

Il seguente esempio di codice mostra come utilizzare`describe-step`.

AWS CLI

Il comando seguente descrive una fase con ID fase `s-3LZC0QUT43AM` in un cluster con ID cluster `j-3SD91U2E1L2QX`.

```
aws emr describe-step --cluster-id j-3SD91U2E1L2QX --step-id s-3LZC0QUT43AM
```

Output:

```
{
    "Step": {
        "Status": {
            "Timeline": {
```

```
                "EndDateTime": 1433200470.481,
                "CreationDateTime": 1433199926.597,
                "StartDateTime": 1433200404.959
            },
            "State": "COMPLETED",
            "StateChangeReason": {}
        },
        "Config": {
            "Args": [
                "s3://us-west-2.elasticmapreduce/libs/hive/hive-script",
                "--base-path",
                "s3://us-west-2.elasticmapreduce/libs/hive/",
                "--install-hive",
                "--hive-versions",
                "0.13.1"
            ],
            "Jar": "s3://us-west-2.elasticmapreduce/libs/script-runner/script-
runner.jar",
            "Properties": {}
        },
        "Id": "s-3LZC0QUT43AM",
        "ActionOnFailure": "TERMINATE_CLUSTER",
        "Name": "Setup hive"
    }
}
```

- Per i dettagli sull'API, consulta DescribeStep AWS CLICommand Reference.

`get`

Il seguente esempio di codice mostra come utilizzare`get`.

AWS CLI

Il comando seguente scarica l'archivio `hadoop-examples.jar` dall'istanza master in un cluster con ID cluster `j-3SD91U2E1L2QX`:

```
aws emr get --cluster-id j-3SD91U2E1L2QX --key-pair-file ~/.ssh/mykey.pem --src /
home/hadoop-examples.jar --dest ~
```

- Per informazioni dettagliate sull'API, consulta Get in AWS CLI Command Reference.

`list-clusters`

Il seguente esempio di codice mostra come utilizzare`list-clusters`.

AWS CLI

Il comando seguente elenca tutti i cluster EMR attivi nella Regione corrente:

```
aws emr list-clusters --active
```

Output:

```
{
    "Clusters": [
        {
            "Status": {
                "Timeline": {
                    "ReadyDateTime": 1433200405.353,
                    "CreationDateTime": 1433199926.596
                },
                "State": "WAITING",
                "StateChangeReason": {
                    "Message": "Waiting after step completed"
                }
            },
            "NormalizedInstanceHours": 6,
            "Id": "j-3SD91U2E1L2QX",
            "Name": "my-cluster"
        }
    ]
}
```

- Per i dettagli sull'API, consulta ListClusters AWS CLICommand Reference.

`list-instance-fleets`

Il seguente esempio di codice mostra come utilizzare`list-instance-fleets`.

AWS CLI

Come ottenere i dettagli di configurazione dei parchi istanze in un cluster

Questo esempio elenca i dettagli dei parchi istanze nel cluster specificato.

Comando:

```
list-instance-fleets --cluster-id 'j-12ABCDEFGHI34JK'
```

Output:

```
{
  "InstanceFleets": [
      {
          "Status": {
              "Timeline": {
                  "ReadyDateTime": 1488759094.637,
                  "CreationDateTime": 1488758719.817
              },
              "State": "RUNNING",
              "StateChangeReason": {
                  "Message": ""
              }
          },
          "ProvisionedSpotCapacity": 6,
          "Name": "CORE",
          "InstanceFleetType": "CORE",
          "LaunchSpecifications": {
              "SpotSpecification": {
                  "TimeoutDurationMinutes": 60,
                  "TimeoutAction": "TERMINATE_CLUSTER"
              }
          },
          "ProvisionedOnDemandCapacity": 2,
          "InstanceTypeSpecifications": [
              {
                  "BidPrice": "0.5",
                  "InstanceType": "m3.xlarge",
                  "WeightedCapacity": 2
              }
          ],
          "Id": "if-1ABC2DEFGHIJ3"
      },
      {
          "Status": {
              "Timeline": {
                  "ReadyDateTime": 1488759058.598,
                  "CreationDateTime": 1488758719.811
```

```
            },
            "State": "RUNNING",
            "StateChangeReason": {
                "Message": ""
            }
        },
        "ProvisionedSpotCapacity": 0,
        "Name": "MASTER",
        "InstanceFleetType": "MASTER",
        "ProvisionedOnDemandCapacity": 1,
        "InstanceTypeSpecifications": [
            {
                "BidPriceAsPercentageOfOnDemandPrice": 100.0,
                "InstanceType": "m3.xlarge",
                "WeightedCapacity": 1
            }
        ],
       "Id": "if-2ABC4DEFGHIJ4"
    }
  ]
}
```

- Per i dettagli sull'API, consulta ListInstanceFleets AWS CLICommand Reference.

`list-instances`

Il seguente esempio di codice mostra come utilizzare`list-instances`.

AWS CLI

Il comando seguente elenca tutte le istanze in un cluster con ID cluster `j-3C6XNQ39VR9WL`:

```
aws emr list-instances --cluster-id j-3C6XNQ39VR9WL
```

Output:

```
For a uniform instance group based cluster
  {
    "Instances": [
         {
            "Status": {
                "Timeline": {
                    "ReadyDateTime": 1433200400.03,
```

```
                "CreationDateTime": 1433199960.152
            },
            "State": "RUNNING",
            "StateChangeReason": {}
        },
        "Ec2InstanceId": "i-f19ecfee",
        "PublicDnsName": "ec2-52-52-41-150.us-west-2.compute.amazonaws.com",
        "PrivateDnsName": "ip-172-21-11-216.us-west-2.compute.internal",
        "PublicIpAddress": "52.52.41.150",
        "Id": "ci-3NNHQUQ2TWB6Y",
        "PrivateIpAddress": "172.21.11.216"
    },
    {
        "Status": {
            "Timeline": {
                "ReadyDateTime": 1433200400.031,
                "CreationDateTime": 1433199949.102
            },
            "State": "RUNNING",
            "StateChangeReason": {}
        },
        "Ec2InstanceId": "i-1feee4c2",
        "PublicDnsName": "ec2-52-63-246-32.us-west-2.compute.amazonaws.com",
        "PrivateDnsName": "ip-172-31-24-130.us-west-2.compute.internal",
        "PublicIpAddress": "52.63.246.32",
        "Id": "ci-GAOCMKNKDCV7",
        "PrivateIpAddress": "172.21.11.215"
    },
    {
        "Status": {
            "Timeline": {
                "ReadyDateTime": 1433200400.031,
                "CreationDateTime": 1433199949.102
            },
            "State": "RUNNING",
            "StateChangeReason": {}
        },
        "Ec2InstanceId": "i-15cfeee3",
        "PublicDnsName": "ec2-52-25-246-63.us-west-2.compute.amazonaws.com",
        "PrivateDnsName": "ip-172-31-24-129.us-west-2.compute.internal",
        "PublicIpAddress": "52.25.246.63",
        "Id": "ci-2W3TDFFB47UAD",
        "PrivateIpAddress": "172.21.11.214"
    }
```

```
    ]
  }


For a fleet based cluster:
   {
      "Instances": [
          {
              "Status": {
                  "Timeline": {
                      "ReadyDateTime": 1487810810.878,
                      "CreationDateTime": 1487810588.367,
                      "EndDateTime": 1488022990.924
                  },
                  "State": "TERMINATED",
                  "StateChangeReason": {
                      "Message": "Instance was terminated."
                  }
              },
              "Ec2InstanceId": "i-xxxxx",
              "InstanceFleetId": "if-xxxxx",
              "EbsVolumes": [],
              "PublicDnsName": "ec2-xx-xxx-xxx-xxx.compute-1.amazonaws.com",
              "InstanceType": "m3.xlarge",
              "PrivateDnsName": "ip-xx-xx-xxx-xx.ec2.internal",
              "Market": "SPOT",
              "PublicIpAddress": "xx.xx.xxx.xxx",
              "Id": "ci-xxxxx",
              "PrivateIpAddress": "10.47.191.80"
          }
      ]
  }
```

- Per i dettagli sull'API, consulta ListInstances AWS CLICommand Reference.

`list-security-configurations`

Il seguente esempio di codice mostra come utilizzare`list-security-configurations`.

AWS CLI

Come elencare le configurazioni di sicurezza nella Regione corrente

Comando:

```
aws emr list-security-configurations
```

Output:

```
{
    "SecurityConfigurations": [
        {
            "CreationDateTime": 1473889697.417,
            "Name": "MySecurityConfig-1"
        },
        {
            "CreationDateTime": 1473889697.417,
            "Name": "MySecurityConfig-2"
        }
    ]
}
```

- Per i dettagli sull'API, consulta ListSecurityConfigurations AWS CLICommand Reference.

`list-steps`

Il seguente esempio di codice mostra come utilizzare`list-steps`.

AWS CLI

Il comando seguente elenca tutte le fasi di un cluster con ID cluster `j-3SD91U2E1L2QX`:

```
aws emr list-steps --cluster-id j-3SD91U2E1L2QX
```

- Per i dettagli sull'API, consulta ListSteps AWS CLICommand Reference.

`modify-cluster-attributes`

Il seguente esempio di codice mostra come utilizzare`modify-cluster-attributes`.

AWS CLI

Il comando seguente imposta la visibilità di un cluster EMR con ID `j-301CDNY0J5XM4` per tutti gli utenti:

```
aws emr modify-cluster-attributes --cluster-id j-301CDNY0J5XM4 --visible-to-all-
users
```

- Per i dettagli sull'API, consulta ModifyClusterAttributes AWS CLICommand Reference.

`modify-instance-fleet`

Il seguente esempio di codice mostra come utilizzare`modify-instance-fleet`.

AWS CLI

Come modificare le capacità delle destinazioni di un parco istanze

Questo esempio modifica a 1 le capacità delle destinazioni on-demand e spot per il parco istanze specificato.

Comando:

```
aws emr modify-instance-fleet --cluster-id 'j-12ABCDEFGHI34JK' --instance-fleet
 InstanceFleetId='if-2ABC4DEFGHIJ4',TargetOnDemandCapacity=1,TargetSpotCapacity=1
```

- Per i dettagli sull'API, consulta ModifyInstanceFleet AWS CLICommand Reference.

`put`

Il seguente esempio di codice mostra come utilizzare`put`.

AWS CLI

Il comando seguente carica un file denominato `healthcheck.sh` nell'istanza master in un cluster con ID cluster `j-3SD91U2E1L2QX`:

```
aws emr put --cluster-id j-3SD91U2E1L2QX --key-pair-file ~/.ssh/mykey.pem --src ~/
scripts/healthcheck.sh --dest /home/hadoop/bin/healthcheck.sh
```

- Per informazioni dettagliate sull'API, consulta Put in AWS CLI Command Reference.

`remove-tags`

Il seguente esempio di codice mostra come utilizzare`remove-tags`.

AWS CLI

Il comando seguente rimuove un tag con la chiave `prod` da un cluster con ID cluster `j-3SD91U2E1L2QX`:

```
aws emr remove-tags --resource-id j-3SD91U2E1L2QX --tag-keys prod
```

- Per i dettagli sull'API, consulta RemoveTags AWS CLICommand Reference.

`schedule-hbase-backup`

Il seguente esempio di codice mostra come utilizzare`schedule-hbase-backup`.

AWS CLI

Nota: questo comando può essere utilizzato solo con le HBase versioni AMI 2.x e 3.x

1. Per pianificare un HBase backup completo **>>>>>>** 06ab6d6e13564b5733d75abaf3b599f93cf39a23

Comando:

```
aws emr schedule-hbase-backup --cluster-id j-XXXXXXYY --type full --dir
s3://amzn-s3-demo-bucket/backup --interval 10 --unit hours --start-time
2014-04-21T05:26:10Z --consistent
```

Output:

```
None
```

2. Per pianificare un backup incrementale HBase

Comando:

```
aws emr schedule-hbase-backup --cluster-id j-XXXXXXYY --type incremental
 --dir s3://amzn-s3-demo-bucket/backup --interval 30 --unit minutes --start-time
2014-04-21T05:26:10Z --consistent
```

Output:

```
None
```

- Per i dettagli sull'API, consulta ScheduleHbaseBackup AWS CLICommand Reference.

`socks`

Il seguente esempio di codice mostra come utilizzare`socks`.

AWS CLI

Il comando seguente apre una connessione SOCKS con l'istanza master in un cluster con ID cluster `j-3SD91U2E1L2QX`:

```
aws emr socks --cluster-id j-3SD91U2E1L2QX --key-pair-file ~/.ssh/mykey.pem
```

L'opzione key pair file richiede un percorso locale a un file di chiave privata.

- Per informazioni dettagliate sull'API, consulta Socks in AWS CLI Command Reference.

`ssh`

Il seguente esempio di codice mostra come utilizzare`ssh`.

AWS CLI

Il comando seguente apre una connessione SSH con l'istanza master in un cluster con ID cluster `j-3SD91U2E1L2QX`:

```
aws emr ssh --cluster-id j-3SD91U2E1L2QX --key-pair-file ~/.ssh/mykey.pem
```

L'opzione key pair file richiede un percorso locale a un file di chiave privata.

Output:

```
ssh -o StrictHostKeyChecking=no -o ServerAliveInterval=10 -i /home/local/user/.ssh/
mykey.pem hadoop@ec2-52-52-41-150.us-west-2.compute.amazonaws.com
Warning: Permanently added 'ec2-52-52-41-150.us-
west-2.compute.amazonaws.com,52.52.41.150' (ECDSA) to the list of known hosts.
Last login: Mon Jun  1 23:15:38 2015
```

```
      __|  __|_  )
       _|  (     /   Amazon Linux AMI
      ___|\___|___|

https://aws.amazon.com/amazon-linux-ami/2015.03-release-notes/
26 package(s) needed for security, out of 39 available
Run "sudo yum update" to apply all updates.

--------------------------------------------------------------------------------

Welcome to Amazon Elastic MapReduce running Hadoop and Amazon Linux.

Hadoop is installed in /home/hadoop. Log files are in /mnt/var/log/hadoop. Check
/mnt/var/log/hadoop/steps for diagnosing step failures.

The Hadoop UI can be accessed via the following commands:

  ResourceManager    lynx http://ip-172-21-11-216:9026/
  NameNode           lynx http://ip-172-21-11-216:9101/

--------------------------------------------------------------------------------

[hadoop@ip-172-31-16-216 ~]$
```

- Per informazioni dettagliate sull'API, consulta Ssh in AWS CLI Command Reference.

Esempi di utilizzo di Amazon EMR su EKS AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando Amazon EMR su EKS. AWS Command Line Interface

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`update-role-trust-policy`

Il seguente esempio di codice mostra come usare`update-role-trust-policy`.

AWS CLI

Come aggiornare la policy di attendibilità di un ruolo IAM da utilizzare con Amazon EMR su EKS

Questo comando di esempio aggiorna la policy di affidabilità di un ruolo denominato example_iam_role in modo che possa essere utilizzato con Amazon EMR su EKS con il namespace example_namespace da un cluster EKS denominato example_cluster.

Comando:

```
aws emr-containers update-role-trust-policy \
    --cluster example_cluster \
    --namespace example_namespace \
    --role-name example_iam_role
```

Output:

```
If the trust policy has already been updated, then the output will be:
Trust policy statement already exists for role example_iam_role. No
changes were made!

If the trust policy has not been updated yet, then the output will be:
Successfully updated trust policy of role example_iam_role.
```

- Per i dettagli sull'API, consulta UpdateRoleTrustPolicy AWS CLICommand Reference.

EventBridge esempi utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with EventBridge.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`delete-rule`

Il seguente esempio di codice mostra come utilizzare`delete-rule`.

AWS CLI

Per eliminare una regola CloudWatch Events

Questo esempio elimina la regola denominata EC2InstanceStateChanges:

```
aws events delete-rule --name "EC2InstanceStateChanges"
```

- Per i dettagli sull'API, consulta DeleteRule AWS CLICommand Reference.

`describe-rule`

Il seguente esempio di codice mostra come utilizzare`describe-rule`.

AWS CLI

Per visualizzare informazioni su una regola CloudWatch Events

Questo esempio visualizza informazioni sulla regola denominata DailyLambdaFunction:

```
aws events describe-rule --name "DailyLambdaFunction"
```

- Per i dettagli sull'API, vedere DescribeRulein AWS CLI Command Reference.

`disable-rule`

Il seguente esempio di codice mostra come utilizzare`disable-rule`.

AWS CLI

Per disabilitare una regola CloudWatch Events

Questo esempio disabilita la regola denominata DailyLambdaFunction. La regola non viene eliminata:

```
aws events disable-rule --name "DailyLambdaFunction"
```

- Per i dettagli sull'API, vedere DisableRulein AWS CLI Command Reference.

`enable-rule`

Il seguente esempio di codice mostra come utilizzare`enable-rule`.

AWS CLI

Per abilitare una regola CloudWatch Events

Questo esempio abilita la regola denominata DailyLambdaFunction, che era stata precedentemente disabilitata:

```
aws events enable-rule --name "DailyLambdaFunction"
```

- Per i dettagli sull'API, consulta EnableRule AWS CLICommand Reference.

`list-rule-names-by-target`

Il seguente esempio di codice mostra come utilizzare`list-rule-names-by-target`.

AWS CLI

Come visualizzare tutte le regole che hanno un obiettivo specificato

Questo esempio visualizza tutte le regole che hanno come destinazione la funzione Lambda denominata MyFunctionName "":

```
aws events list-rule-names-by-target --target-arn "arn:aws:lambda:us-
east-1:123456789012:function:MyFunctionName"
```

- Per i dettagli sull'API, consulta ListRuleNamesByTarget AWS CLICommand Reference.

`list-rules`

Il seguente esempio di codice mostra come utilizzare`list-rules`.

AWS CLI

Per visualizzare un elenco di tutte le regole CloudWatch degli eventi

Questo esempio visualizza tutte le regole CloudWatch Events della regione:

```
aws events list-rules
```

Per visualizzare un elenco di regole CloudWatch Events che iniziano con una determinata stringa.

Questo esempio visualizza tutte le regole CloudWatch Events nella regione il cui nome inizia con «Daily»:

```
aws events list-rules --name-prefix "Daily"
```

- Per i dettagli sull'API, consulta ListRules AWS CLICommand Reference.

`list-targets-by-rule`

Il seguente esempio di codice mostra come utilizzare`list-targets-by-rule`.

AWS CLI

Per visualizzare tutti gli obiettivi di una regola CloudWatch Events

Questo esempio visualizza tutti gli obiettivi della regola denominata DailyLambdaFunction:

```
aws events list-targets-by-rule --rule  "DailyLambdaFunction"
```

- Per i dettagli sull'API, consulta ListTargetsByRule AWS CLICommand Reference.

`put-events`

Il seguente esempio di codice mostra come utilizzare`put-events`.

AWS CLI

Per inviare un evento personalizzato a CloudWatch Events

Questo esempio invia un evento personalizzato a CloudWatch Events. L'evento è contenuto nel file putevents.json:

```
aws events put-events --entries file://putevents.json
```

Visualizzare il contenuto del file putevents.json:

```
[
  {
    "Source": "com.mycompany.myapp",
    "Detail": "{ \"key1\": \"value1\", \"key2\": \"value2\" }",
    "Resources": [
      "resource1",
      "resource2"
    ],
    "DetailType": "myDetailType"
  },
  {
    "Source": "com.mycompany.myapp",
    "Detail": "{ \"key1\": \"value3\", \"key2\": \"value4\" }",
    "Resources": [
      "resource1",
      "resource2"
    ],
    "DetailType": "myDetailType"
   }
]
```

- Per i dettagli sull'API, consulta PutEvents AWS CLICommand Reference.

put-rule

Il seguente esempio di codice mostra come utilizzare`put-rule`.

AWS CLI

Per creare regole relative CloudWatch agli eventi

In questo esempio viene creata una regola attivata ogni giorno alle 9:00 UTC. Se usi put-targets per aggiungere una funzione Lambda come destinazione di questa regola, puoi eseguire la funzione Lambda ogni giorno all'ora specificata:

```
aws events put-rule --name "DailyLambdaFunction" --schedule-expression "cron(0 9 *
 * ? *)"
```

Questo esempio crea una regola che si attiva quando EC2 un'istanza nella regione cambia stato:

```
aws events put-rule --name "EC2InstanceStateChanges" --event-pattern "{\"source\":
[\"aws.ec2\"],\"detail-type\":[\"EC2 Instance State-change Notification\"]}"  --
role-arn "arn:aws:iam::123456789012:role/MyRoleForThisRule"
```

Questo esempio crea una regola che si attiva quando un' EC2 istanza nella regione viene interrotta o terminata:

```
aws events put-rule --name "EC2InstanceStateChangeStopOrTerminate" --event-
pattern "{\"source\":[\"aws.ec2\"],\"detail-type\":[\"EC2 Instance State-change
 Notification\"],\"detail\":{\"state\":[\"stopped\",\"terminated\"]}}" --role-
arn "arn:aws:iam::123456789012:role/MyRoleForThisRule"
```

- Per i dettagli sull'API, consulta AWS CLI Command PutRuleReference.

`put-targets`

Il seguente esempio di codice mostra come utilizzare`put-targets`.

AWS CLI

Per aggiungere obiettivi per le regole CloudWatch Events

Nell'esempio seguente viene aggiunta una funzione Lambda come destinazione di una regola:

```
aws events put-targets --rule DailyLambdaFunction --targets
 "Id"="1","Arn"="arn:aws:lambda:us-east-1:123456789012:function:MyFunctionName"
```

Questo esempio imposta un flusso Amazon Kinesis come destinazione, in modo che gli eventi rilevati da questa regola vengano inoltrati allo stream:

```
aws events put-targets --rule EC2InstanceStateChanges --targets
 "Id"="1","Arn"="arn:aws:kinesis:us-east-1:123456789012:stream/
MyStream","RoleArn"="arn:aws:iam::123456789012:role/MyRoleForThisRule"
```

Questo esempio imposta due flussi Amazon Kinesis come destinazione per una regola:

```
aws events put-targets --rule DailyLambdaFunction --targets
 "Id"="Target1","Arn"="arn:aws:kinesis:us-east-1:379642911888:stream/
MyStream1","RoleArn"="arn:aws:iam::379642911888:role/ MyRoleToAccessLambda"
  "Id"="Target2"," Arn"="arn:aws:kinesis:us-east-1:379642911888:stream/
MyStream2","RoleArn"="arn:aws:iam::379642911888:role/MyRoleToAccessLambda"
```

- Per i dettagli sull'API, consulta PutTargets AWS CLICommand Reference.

`remove-targets`

Il seguente esempio di codice mostra come utilizzare`remove-targets`.

AWS CLI

Come rimuovere una destinazione per un evento

Questo esempio rimuove lo stream Amazon Kinesis denominato MyStream 1 dall'obiettivo della regola. DailyLambdaFunction Quando DailyLambdaFunction è stato creato, questo flusso è stato impostato come destinazione con un ID Target1:

```
aws events remove-targets --rule "DailyLambdaFunction" --ids "Target1"
```

- Per i dettagli sull'API, consulta AWS CLI Command RemoveTargetsReference.

`test-event-pattern`

Il seguente esempio di codice mostra come utilizzare`test-event-pattern`.

AWS CLI

Come verificare se un modello di evento corrisponde a un evento specificato

Questo esempio verifica se il modello “source:com.mycompany.myapp” corrisponde all'evento specificato. In questo esempio, l'output sarebbe “true”:

```
aws events test-event-pattern --event-pattern "{\"source\":[\"com.mycompany.myapp
\"]}" --event "{\"id\":\"1\",\"source\":\"com.mycompany.myapp\",\"detail-type\":
\"myDetailType\",\"account\":\"123456789012\",\"region\":\"us-east-1\",\"time\":
\"2017-04-11T20:11:04Z\"}"
```

- Per i dettagli sull'API, consulta TestEventPattern AWS CLICommand Reference.

EventBridge Esempi di tubi che utilizzano AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with EventBridge Pipes.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`create-pipe`

Il seguente esempio di codice mostra come utilizzare`create-pipe`.

AWS CLI

Come creare una pipe

L'`create-pipe`esempio seguente crea una Pipe denominata `Demo_Pipe` con SQS come origine e CloudWatch Log Group come destinazione per la Pipe.

```
aws pipes create-pipe \
    --name Demo_Pipe \
    --desired-state RUNNING \
    --role-arn arn:aws:iam::123456789012:role/service-role/
Amazon_EventBridge_Pipe_Demo_Pipe_28b3aa4f \
    --source arn:aws:sqs:us-east-1:123456789012:Demo_Queue \
    --target arn:aws:logs:us-east-1:123456789012:log-group:/aws/pipes/Demo_LogGroup
```

Output:

```
{
    "Arn": "arn:aws:pipes:us-east-1:123456789012:pipe/Demo_Pipe",
```

```
    "Name": "Demo_Pipe",
    "DesiredState": "RUNNING",
    "CurrentState": "CREATING",
    "CreationTime": "2024-10-08T12:33:59-05:00",
    "LastModifiedTime": "2024-10-08T12:33:59.684839-05:00"
}
```

Per ulteriori informazioni, consulta EventBridge i concetti di Amazon Pipes nella Amazon EventBridge User Guide.

- Per i dettagli sull'API, consulta CreatePipe AWS CLICommand Reference.

`delete-pipe`

Il seguente esempio di codice mostra come utilizzare`delete-pipe`.

AWS CLI

Come eliminare una pipe esistente

L'esempio `delete-pipe` seguente elimina una pipe denominata `Demo_Pipe` nell'account specificato.

```
aws pipes delete-pipe \
    --name Demo_Pipe
```

Output:

```
{
    "Arn": "arn:aws:pipes:us-east-1:123456789012:pipe/Demo_Pipe",
    "Name": "Demo_Pipe",
    "DesiredState": "STOPPED",
    "CurrentState": "DELETING",
    "CreationTime": "2024-10-08T09:29:10-05:00",
    "LastModifiedTime": "2024-10-08T11:57:22-05:00"
}
```

Per ulteriori informazioni, consulta EventBridge i concetti di Amazon Pipes nella Amazon EventBridge User Guide.

- Per i dettagli sull'API, consulta DeletePipe AWS CLICommand Reference.

`describe-pipe`

Il seguente esempio di codice mostra come utilizzare`describe-pipe`.

AWS CLI

Come recuperare le informazioni relative a una pipe

L'esempio `describe-pipe` seguente visualizza informazioni sulla pipe `Demo_Pipe` nell'account specificato.

```
aws pipes describe-pipe \
    --name Demo_Pipe
```

Output:

```
{
    "Arn": "arn:aws:pipes:us-east-1:123456789012:pipe/Demo_Pipe",
    "Name": "Demo_Pipe",
    "DesiredState": "RUNNING",
    "CurrentState": "RUNNING",
    "StateReason": "User initiated",
    "Source": "arn:aws:sqs:us-east-1:123456789012:Demo_Queue",
    "SourceParameters": {
        "SqsQueueParameters": {
            "BatchSize": 1
        }
    },
    "EnrichmentParameters": {},
    "Target": "arn:aws:logs:us-east-1:123456789012:log-group:/aws/pipes/
Demo_LogGroup",
    "TargetParameters": {},
    "RoleArn": "arn:aws:iam::123456789012:role/service-role/
Amazon_EventBridge_Pipe_Demo_Pipe_28b3aa4f",
    "Tags": {},
    "CreationTime": "2024-10-08T09:29:10-05:00",
    "LastModifiedTime": "2024-10-08T10:23:47-05:00",
    "LogConfiguration": {
        "CloudwatchLogsLogDestination": {
            "LogGroupArn": "arn:aws:logs:us-east-1:123456789012:log-group:/aws/
vendedlogs/pipes/Demo_Pipe"
        },
        "Level": "ERROR"
```

```
    }
}
```

Per ulteriori informazioni, consulta EventBridge i concetti di Amazon Pipes nella Amazon EventBridge User Guide.

- Per i dettagli sull'API, consulta DescribePipe AWS CLICommand Reference.

`list-pipes`

Il seguente esempio di codice mostra come utilizzare`list-pipes`.

AWS CLI

Come recuperare un elenco di pipe

L'esempio `list-pipes` seguente mostra tutte le pipe nell'account specificato.

```
aws pipes list-pipes
```

Output:

```
{
    "Pipes": [
        {
            "Name": "Demo_Pipe",
            "Arn": "arn:aws:pipes:us-east-1:123456789012:pipe/Demo_Pipe",
            "DesiredState": "RUNNING",
            "CurrentState": "RUNNING",
            "StateReason": "User initiated",
            "CreationTime": "2024-10-08T09:29:10-05:00",
            "LastModifiedTime": "2024-10-08T10:23:47-05:00",
            "Source": "arn:aws:sqs:us-east-1:123456789012:Demo_Queue",
            "Target": "arn:aws:logs:us-east-1:123456789012:log-group:/aws/pipes/
Demo_LogGroup"
        }
    ]
}
```

Per ulteriori informazioni, consulta EventBridge i concetti di Amazon Pipes nella Amazon EventBridge User Guide.

- Per i dettagli sull'API, consulta ListPipes AWS CLICommand Reference.

`list-tags-for-resource`

Il seguente esempio di codice mostra come utilizzare`list-tags-for-resource`.

AWS CLI

Come elencare i tag associati a una pipe esistente

L'esempio `list-tags-for-resource` seguente elenca tutti i tag associati a una pipe denominata `Demo_Pipe` nell'account specificato.

```
aws pipes list-tags-for-resource \
    --resource-arn arn:aws:pipes:us-east-1:123456789012:pipe/Demo_Pipe
```

Output:

```
{
    "tags": {
        "stack": "Production",
        "team": "DevOps"
    }
}
```

Per ulteriori informazioni, consulta EventBridge i concetti di Amazon Pipes nella Amazon EventBridge User Guide.

- Per i dettagli sull'API, consulta ListTagsForResource AWS CLICommand Reference.

`start-pipe`

Il seguente esempio di codice mostra come utilizzare`start-pipe`.

AWS CLI

Come avviare una pipe esistente

L'esempio `start-pipe` seguente avvia una pipe denominata `Demo_Pipe` nell'account specificato.

```
aws pipes start-pipe \
    --name Demo_Pipe
```

Output:

```
{
    "Arn": "arn:aws:pipes:us-east-1:123456789012:pipe/Demo_Pipe",
    "Name": "Demo_Pipe",
    "DesiredState": "RUNNING",
    "CurrentState": "STARTING",
    "CreationTime": "2024-10-08T09:29:10-05:00",
    "LastModifiedTime": "2024-10-08T10:17:24-05:00"
}
```

Per ulteriori informazioni, consulta Avvio o arresto di una EventBridge pipa Amazon nella Amazon EventBridge User Guide.

- Per i dettagli sull'API, consulta StartPipe AWS CLICommand Reference.

`stop-pipe`

Il seguente esempio di codice mostra come utilizzare`stop-pipe`.

AWS CLI

Come arrestare una pipe esistente

L'esempio `stop-pipe` seguente arresta una pipe denominata `Demo_Pipe` nell'account specificato.

```
aws pipes stop-pipe \
    --name Demo_Pipe
```

Output:

```
{
    "Arn": "arn:aws:pipes:us-east-1:123456789012:pipe/Demo_Pipe",
    "Name": "Demo_Pipe",
    "DesiredState": "STOPPED",
    "CurrentState": "STOPPING",
```

```
    "CreationTime": "2024-10-08T09:29:10-05:00",
    "LastModifiedTime": "2024-10-08T09:29:49-05:00"
}
```

Per ulteriori informazioni, consulta Avvio o arresto di una EventBridge pipa Amazon nella Amazon EventBridge User Guide.

- Per i dettagli sull'API, consulta StopPipe AWS CLICommand Reference.

`tag-resource`

Il seguente esempio di codice mostra come utilizzare`tag-resource`.

AWS CLI

Come taggare una pipe esistente

L'esempio `tag-resource` seguente tagga una pipe denominata `Demo_Pipe`. Se il comando va a buon fine, non viene restituito alcun output.

```
aws pipes tag-resource \
    --resource-arn arn:aws:pipes:us-east-1:123456789012:pipe/Demo_Pipe \
    --tags stack=Production
```

Per ulteriori informazioni, consulta EventBridge i concetti di Amazon Pipes nella Amazon EventBridge User Guide.

- Per i dettagli sull'API, consulta TagResource AWS CLICommand Reference.

`untag-resource`

Il seguente esempio di codice mostra come utilizzare`untag-resource`.

AWS CLI

Come rimuovere un tag da una pipe esistente

L'esempio `untag-resource` seguente rimuove il tag con la chiave `stack` per il nome della pipe denominata `Demo_Pipe`. Se il comando va a buon fine, non viene restituito alcun output.

```
aws pipes untag-resource \
```

```
    --resource-arn arn:aws:pipes:us-east-1:123456789012:pipe/Demo_Pipe \
    --tags stack
```

Per ulteriori informazioni, consulta EventBridge i concetti di Amazon Pipes nella Amazon EventBridge User Guide.

- Per i dettagli sull'API, consulta UntagResource AWS CLICommand Reference.

`update-pipe`

Il seguente esempio di codice mostra come utilizzare`update-pipe`.

AWS CLI

Come aggiornare una pipe esistente

L'`update-pipe`esempio seguente aggiorna la Pipe denominata `Demo_Pipe` aggiungendo un parametro di configurazione CloudWatch Log, assicuratevi di aggiornare il ruolo di esecuzione della pipe in modo che abbia le autorizzazioni corrette per la destinazione Log.

```
aws pipes update-pipe \
    --name Demo_Pipe \
    --desired-state RUNNING \
    --log-configuration CloudwatchLogsLogDestination={LogGroupArn=arn:aws:logs:us-
east-1:123456789012:log-group:/aws/vendedlogs/pipes/Demo_Pipe},Level=TRACE \
    --role-arn arn:aws:iam::123456789012:role/service-role/
Amazon_EventBridge_Pipe_Demo_Pipe_28b3aa4f
```

Output:

```
{
    "Arn": "arn:aws:pipes:us-east-1:123456789012:pipe/Demo_Pipe",
    "Name": "Demo_Pipe",
    "DesiredState": "RUNNING",
    "CurrentState": "UPDATING",
    "CreationTime": "2024-10-08T09:29:10-05:00",
    "LastModifiedTime": "2024-10-08T11:35:48-05:00"
}
```

Per ulteriori informazioni, consulta EventBridge i concetti di Amazon Pipes nella Amazon EventBridge User Guide.

- Per i dettagli sull'API, consulta UpdatePipe AWS CLICommand Reference.

Esempi di Firewall Manager che utilizzano AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface con Firewall Manager.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`associate-admin-account`

Il seguente esempio di codice mostra come utilizzare`associate-admin-account`.

AWS CLI

Come impostare l'account amministratore Firewall Manager

L'esempio `associate-admin-account` seguente imposta l'account amministratore per Firewall Manager.

```
aws fms associate-admin-account \
    --admin-account 123456789012
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Impostare l'account amministratore di AWS Firewall Manager nella AWS WAF, AWS Firewall Manager e AWS Shield Advanced Developer Guide.

- Per i dettagli sull'API, consulta AssociateAdminAccount AWS CLICommand Reference.

`delete-notification-channel`

Il seguente esempio di codice mostra come utilizzare`delete-notification-channel`.

AWS CLI

Come rimuovere le informazioni sull'argomento SNS per i log di Firewall Manager

L'esempio `delete-notification-channel` seguente rimuove le informazioni sull'argomento SNS.

```
aws fms delete-notification-channel
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Configurare Amazon SNS Notifications and Amazon CloudWatch Alarms nella AWS WAF, AWS Firewall Manager e AWS Shield Advanced Developer Guide.

- Per i dettagli sull'API, consulta Command DeleteNotificationChannelReference AWS CLI .

`delete-policy`

Il seguente esempio di codice mostra come utilizzare`delete-policy`.

AWS CLI

Come eliminare una policy di Firewall Manager

L'esempio `delete-policy` seguente rimuove la policy con l'ID specificato, insieme a tutte le relative risorse.

```
aws fms delete-policy \
    --policy-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 \
    --delete-all-policy-resources
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Working with AWS Firewall Manager Policies nella AWS WAF, AWS Firewall Manager and AWS Shield Advanced Developer Guide.

- Per i dettagli sulle API, consulta DeletePolicy AWS CLICommand Reference.

`disassociate-admin-account`

Il seguente esempio di codice mostra come utilizzare`disassociate-admin-account`.

AWS CLI

Come rimuovere l'account amministratore Firewall Manager

L'esempio `disassociate-admin-account` seguente rimuove l'associazione corrente degli account di amministratore da Firewall Manager.

```
aws fms disassociate-admin-account
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Impostare l'account amministratore di AWS Firewall Manager nella AWS WAF, AWS Firewall Manager e AWS Shield Advanced Developer Guide.

- Per i dettagli sull'API, consulta DisassociateAdminAccount AWS CLICommand Reference.

`get-admin-account`

Il seguente esempio di codice mostra come utilizzare`get-admin-account`.

AWS CLI

Come recuperare l'account amministratore Firewall Manager

L'esempio `get-admin-account` seguente recupera l'account amministratore.

```
aws fms get-admin-account
```

Output:

```
{
    "AdminAccount": "123456789012",
    "RoleStatus": "READY"
}
```

Per ulteriori informazioni, consulta la sezione Prerequisiti di AWS Firewall Manager nella AWS WAF, AWS Firewall Manager and AWS Shield Advanced Developer Guide.

- Per i dettagli sulle API, consulta AWS CLI Command GetAdminAccountReference.

`get-compliance-detail`

Il seguente esempio di codice mostra come utilizzare`get-compliance-detail`.

AWS CLI

Come recuperare le informazioni sulla conformità di un account

L'esempio `get-compliance-detail` seguente recupera le informazioni sulla conformità per la policy e l'account membro specificati.

```
aws fms get-compliance-detail \
    --policy-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 \
    --member-account 123456789012
```

Output:

```
{
    "PolicyComplianceDetail": {
    "EvaluationLimitExceeded": false,
    "IssueInfoMap": {},
    "MemberAccount": "123456789012",
    "PolicyId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "PolicyOwner": "123456789012",
    "Violators": []
}
```

Per ulteriori informazioni, vedere Visualizzazione della conformità delle risorse con una politica nella AWS WAF, AWS Firewall Manager, and AWS Shield Advanced Developer Guide.

- Per i dettagli sull'API, consulta GetComplianceDetail AWS CLICommand Reference.

`get-notification-channel`

Il seguente esempio di codice mostra come utilizzare`get-notification-channel`.

AWS CLI

Come recuperare le informazioni sull'argomento SNS per i log di Firewall Manager

L'esempio `get-notification-channel` seguente recupera le informazioni sull'argomento SNS.

```
aws fms get-notification-channel
```

Output:

```
{
    "SnsTopicArn": "arn:aws:sns:us-west-2:123456789012:us-west-2-fms",
    "SnsRoleName": "arn:aws:iam::123456789012:role/aws-service-role/
fms.amazonaws.com/AWSServiceRoleForFMS"
}
```

Per ulteriori informazioni, consulta Configurare Amazon SNS Notifications and Amazon CloudWatch Alarms nella AWS WAF, AWS Firewall Manager e AWS Shield Advanced Developer Guide.

- Per i dettagli sull'API, consulta Command GetNotificationChannelReference AWS CLI .

`get-policy`

Il seguente esempio di codice mostra come utilizzare`get-policy`.

AWS CLI

Come recuperare una policy di Firewall Manager

L'esempio `get-policy` seguente recupera la policy con l'ID specificato.

```
aws fms get-policy \
    --policy-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Output:

```
{
    "Policy": {
        "PolicyId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
        "PolicyName": "test",
        "PolicyUpdateToken": "1:p+2RpKR4wPFx7mcrL1UOQQ==",
        "SecurityServicePolicyData": {
            "Type": "SECURITY_GROUPS_COMMON",
            "ManagedServiceData": "{\"type\":\"SECURITY_GROUPS_COMMON\",
\"revertManualSecurityGroupChanges\":true,\"exclusiveResourceSecurityGroupManagement
\":false,\"securityGroups\":[{\"id\":\"sg-045c43ccc9724e63e\"}]}"
        },
```

```
        "ResourceType": "AWS::EC2::Instance",
        "ResourceTags": [],
        "ExcludeResourceTags": false,
        "RemediationEnabled": false
    },
    "PolicyArn": "arn:aws:fms:us-west-2:123456789012:policy/d1ac59b8-938e-42b3-
b2e0-7c620422ddc2"
}
```

Per ulteriori informazioni, consulta Working with AWS Firewall Manager Policies nella AWS WAF, AWS Firewall Manager and AWS Shield Advanced Developer Guide.

- Per i dettagli sulle API, consulta GetPolicy AWS CLICommand Reference.

`list-compliance-status`

Il seguente esempio di codice mostra come utilizzare`list-compliance-status`.

AWS CLI

Come recuperare le informazioni sulla conformità delle policy per gli account membri

L'esempio `list-compliance-status` seguente recupera le informazioni sulla conformità degli account membri per la policy specificata.

```
aws fms list-compliance-status \
    --policy-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Output:

```
{
    "PolicyComplianceStatusList": [
        {
            "PolicyOwner": "123456789012",
            "PolicyId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
            "PolicyName": "test",
            "MemberAccount": "123456789012",
            "EvaluationResults": [
                {
                    "ComplianceStatus": "COMPLIANT",
                    "ViolatorCount": 0,
                    "EvaluationLimitExceeded": false
                },
```

```
                {
                    "ComplianceStatus": "NON_COMPLIANT",
                    "ViolatorCount": 2,
                    "EvaluationLimitExceeded": false
                }
            ],
            "LastUpdated": 1576283774.0,
            "IssueInfoMap": {}
        }
    ]
}
```

Per ulteriori informazioni, vedere Visualizzazione della conformità delle risorse con una politica nella AWS WAF, AWS Firewall Manager, and AWS Shield Advanced Developer Guide.

- Per i dettagli sull'API, consulta ListComplianceStatus AWS CLICommand Reference.

`list-member-accounts`

Il seguente esempio di codice mostra come utilizzare`list-member-accounts`.

AWS CLI

Come recuperare gli account membri dell'organizzazione

L'esempio `list-member-accounts` seguente elenca tutti gli account membri che fanno parte dell'organizzazione dell'amministratore di Firewall Manager.

```
aws fms list-member-accounts
```

Output:

```
{
    "MemberAccounts": [
        "222222222222",
        "333333333333",
        "444444444444"
    ]
}
```

Per ulteriori informazioni, consulta AWS Firewall Manager nella AWS WAF, AWS Firewall Manager and AWS Shield Advanced Developer Guide.

- Per i dettagli sull'API, consulta ListMemberAccounts AWS CLICommand Reference.

`list-policies`

Il seguente esempio di codice mostra come utilizzare`list-policies`.

AWS CLI

Come recuperare tutte le policy di Firewall Manager

L'esempio `list-policies` seguente recupera l'elenco delle policy per l'account. In questo esempio, l'output è limitato a due risultati per richiesta. Ogni chiamata restituisce un `NextToken` che può essere utilizzato come valore per il parametro `--starting-token` nella chiamata `list-policies` successiva per ottenere il prossimo set di risultati per l'elenco.

```
aws fms list-policies \
    --max-items 2
```

Output:

```
{
    "PolicyList": [
        {
            "PolicyArn": "arn:aws:fms:us-west-2:123456789012:policy/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
            "PolicyId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
            "PolicyName": "test",
            "ResourceType": "AWS::EC2::Instance",
            "SecurityServiceType": "SECURITY_GROUPS_COMMON",
            "RemediationEnabled": false
        },
        {
            "PolicyArn": "arn:aws:fms:us-west-2:123456789012:policy/
a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
            "PolicyId": "457c9b21-fc94-406c-ae63-21217395ba72",
            "PolicyName": "test",
            "ResourceType": "AWS::EC2::Instance",
            "SecurityServiceType": "SECURITY_GROUPS_COMMON",
            "RemediationEnabled": false
        }
    ],
    "NextToken": "eyJOZXh0VG9rZW4iOiBudWxsLCAiYm90b190cnVuY2F0ZV9hbW91bnQiOiAyfQ=="
```

```
}
```

Per ulteriori informazioni, consulta Working with AWS Firewall Manager Policies nella AWS WAF, AWS Firewall Manager and AWS Shield Advanced Developer Guide.

- Per i dettagli sulle API, consulta ListPolicies AWS CLICommand Reference.

`put-notification-channel`

Il seguente esempio di codice mostra come utilizzare`put-notification-channel`.

AWS CLI

Come impostare le informazioni sull'argomento SNS per i log di Firewall Manager

L'esempio `put-notification-channel` seguente imposta le informazioni sull'argomento SNS.

```
aws fms put-notification-channel \
    --sns-topic-arn arn:aws:sns:us-west-2:123456789012:us-west-2-fms \
    --sns-role-name arn:aws:iam::123456789012:role/aws-service-role/
fms.amazonaws.com/AWSServiceRoleForFMS
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Configurare Amazon SNS Notifications and Amazon CloudWatch Alarms nella AWS WAF, AWS Firewall Manager e AWS Shield Advanced Developer Guide.

- Per i dettagli sull'API, consulta Command PutNotificationChannelReference AWS CLI .

`put-policy`

Il seguente esempio di codice mostra come utilizzare`put-policy`.

AWS CLI

Come creare una policy di Firewall Manager

L'esempio `put-policy` seguente crea una policy del gruppo di sicurezza di Firewall Manager.

```
aws fms put-policy \
```

```
    --cli-input-json file://policy.json
```

Contenuto di `policy.json`:

```
{
    "Policy": {
        "PolicyName": "test",
        "SecurityServicePolicyData": {
            "Type": "SECURITY_GROUPS_USAGE_AUDIT",
            "ManagedServiceData": "{\"type\":\"SECURITY_GROUPS_USAGE_AUDIT\",
\"deleteUnusedSecurityGroups\":false,\"coalesceRedundantSecurityGroups\":true}"
        },
        "ResourceType": "AWS::EC2::SecurityGroup",
        "ResourceTags": [],
        "ExcludeResourceTags": false,
        "RemediationEnabled": false
    },
    "TagList": [
        {
            "Key": "foo",
            "Value": "foo"
        }
    ]
}
```

Output:

```
{
    "Policy": {
        "PolicyId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
        "PolicyName": "test",
        "PolicyUpdateToken": "1:X9QGexP7HASDlsFp+G31Iw==",
        "SecurityServicePolicyData": {
            "Type": "SECURITY_GROUPS_USAGE_AUDIT",
            "ManagedServiceData": "{\"type\":\"SECURITY_GROUPS_USAGE_AUDIT\",
\"deleteUnusedSecurityGroups\":false,\"coalesceRedundantSecurityGroups\":true,
\"optionalDelayForUnusedInMinutes\":null}"
        },
        "ResourceType": "AWS::EC2::SecurityGroup",
        "ResourceTags": [],
        "ExcludeResourceTags": false,
        "RemediationEnabled": false
    },
```

```
    "PolicyArn": "arn:aws:fms:us-west-2:123456789012:policy/a1b2c3d4-5678-90ab-cdef-
EXAMPLE11111"
}
```

Per ulteriori informazioni, consulta Working with AWS Firewall Manager Policies nella AWS WAF, AWS Firewall Manager and AWS Shield Advanced Developer Guide.

- Per i dettagli sulle API, consulta PutPolicy AWS CLICommand Reference.

AWS FIS esempi utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with AWS FIS.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`create-experiment-template`

Il seguente esempio di codice mostra come utilizzare`create-experiment-template`.

AWS CLI

Come creare un modello di esperimento

L'`create-experiment-template`esempio seguente crea un modello di esperimento nel tuo account AWS FIS.

```
aws fis create-experiment-template \
    --cli-input-json file://myfile.json
```

Contenuto di `myfile.json`:

```
{
    "description": "experimentTemplate",
    "stopConditions": [
        {
            "source": "aws:cloudwatch:alarm",
            "value": "arn:aws:cloudwatch:us-west-2:123456789012:alarm:alarmName"
        }
    ],
    "targets": {
        "Instances-Target-1": {
            "resourceType": "aws:ec2:instance",
            "resourceArns": [
                "arn:aws:ec2:us-west-2:123456789012:instance/i-12a3b4c56d78e9012"
            ],
            "selectionMode": "ALL"
        }
    },
    "actions": {
        "reboot": {
            "actionId": "aws:ec2:reboot-instances",
            "description": "reboot",
            "parameters": {},
            "targets": {
                "Instances": "Instances-Target-1"
            }
        }
    },
    "roleArn": "arn:aws:iam::123456789012:role/myRole"
}
```

Output:

```
{
    "experimentTemplate": {
        "id": "ABCDE1fgHIJkLmNop",
        "description": "experimentTemplate",
        "targets": {
            "Instances-Target-1": {
                "resourceType": "aws:ec2:instance",
                "resourceArns": [
                    "arn:aws:ec2:us-west-2:123456789012:instance/
i-12a3b4c56d78e9012"
                ],
```

```
                "selectionMode": "ALL"
            }
        },
        "actions": {
            "reboot": {
                "actionId": "aws:ec2:reboot-instances",
                "description": "reboot",
                "parameters": {},
                "targets": {
                    "Instances": "Instances-Target-1"
                }
            }
        },
        "stopConditions": [
            {
                "source": "aws:cloudwatch:alarm",
                "value": "arn:aws:cloudwatch:us-west-2:123456789012:alarm:alarmName"
            }
        ],
        "creationTime": 1616434850.659,
        "lastUpdateTime": 1616434850.659,
        "roleArn": "arn:aws:iam::123456789012:role/myRole",
        "tags": {}
    }
}
```

Per ulteriori informazioni, consulta Creare un modello di esperimento nella Guida per l'utente di AWS Fault Injection Simulator.

- Per i dettagli sull'API, vedere CreateExperimentTemplatein AWS CLI Command Reference.

`delete-experiment-template`

Il seguente esempio di codice mostra come utilizzare`delete-experiment-template`.

AWS CLI

Come eliminare un modello di esperimento

L'esempio `delete-experiment-template` seguente elimina il modello di esperimento specificato.

```
aws fis delete-experiment-template \
```

```
    --id ABCDE1fgHIJkLmNop
```

Output:

```
{
    "experimentTemplate": {
        "id": "ABCDE1fgHIJkLmNop",
        "description": "myExperimentTemplate",
        "targets": {
            "Instances-Target-1": {
                "resourceType": "aws:ec2:instance",
                "resourceArns": [
                    "arn:aws:ec2:us-west-2:123456789012:instance/
i-12a3b4c56d78e9012"
                ],
                "selectionMode": "ALL"
            }
        },
        "actions": {
            "testaction": {
                "actionId": "aws:ec2:stop-instances",
                "parameters": {},
                "targets": {
                    "Instances": "Instances-Target-1"
                }
            }
        },
        "stopConditions": [
            {
                "source": "none"
            }
        ],
        "creationTime": 1616017191.124,
        "lastUpdateTime": 1616017859.607,
        "roleArn": "arn:aws:iam::123456789012:role/FISRole"
    }
}
```

Per ulteriori informazioni, consulta Eliminare un modello di esperimento nella Guida per l'utente di AWS Fault Injection Simulator.

- Per i dettagli sull'API, consulta DeleteExperimentTemplate AWS CLICommand Reference.

`get-action`

Il seguente esempio di codice mostra come utilizzare`get-action`.

AWS CLI

Come ottenere dettagli sull'azione

L'esempio `get-action` seguente ottiene i dettagli dell'azione specificata.

```
aws fis get-action \
    --id aws:ec2:stop-instances
```

Output:

```
{
    "action": {
        "id": "aws:ec2:stop-instances",
        "description": "Stop the specified EC2 instances.",
        "parameters": {
            "startInstancesAfterDuration": {
                "description": "The time to wait before restarting the instances
 (ISO 8601 duration).",
                "required": false
            }
        },
        "targets": {
            "Instances": {
                "resourceType": "aws:ec2:instance"
            }
        },
        "tags": {}
    }
}
```

Per ulteriori informazioni, consulta Azioni nella Guida per l'utente di AWS Fault Injection Simulator.

- Per i dettagli sull'API, consulta GetAction AWS CLICommand Reference.

`get-experiment-template`

Il seguente esempio di codice mostra come utilizzare`get-experiment-template`.

AWS CLI

Come ottenere i dettagli del modello di esperimento

L'esempio `get-experiment-template` seguente ottiene i dettagli del modello di esperimento specificato.

```
aws fis get-experiment-template \
    --id ABCDE1fgHIJkLmNop
```

Output:

```
{
    "experimentTemplate": {
        "id": "ABCDE1fgHIJkLmNop",
        "description": "myExperimentTemplate",
        "targets": {
            "Instances-Target-1": {
                "resourceType": "aws:ec2:instance",
                "resourceArns": [
                    "arn:aws:ec2:us-west-2:123456789012:instance/
i-12a3b4c56d78e9012"
                ],
                "selectionMode": "ALL"
            }
        },
        "actions": {
            "testaction": {
                "actionId": "aws:ec2:stop-instances",
                "parameters": {},
                "targets": {
                    "Instances": "Instances-Target-1"
                }
            }
        },
        "stopConditions": [
            {
                "source": "none"
            }
        ],
        "creationTime": 1616017191.124,
        "lastUpdateTime": 1616017331.51,
        "roleArn": "arn:aws:iam::123456789012:role/FISRole",
```

```
        "tags": {
        "key: "value"
        }
    }
}
```

Per ulteriori informazioni, consulta Modelli di esperimenti nella Guida per l'utente di AWS Fault Injection Simulator.

- Per i dettagli sull'API, consulta GetExperimentTemplate AWS CLICommand Reference.

`get-experiment`

Il seguente esempio di codice mostra come utilizzare`get-experiment`.

AWS CLI

Come ottenere i dettagli dell'esperimento

L'esempio `get-experiment` seguente ottiene i dettagli dell'esperimento specificato.

```
aws fis get-experiment \
    --id ABC12DeFGhI3jKLMNOP
```

Output:

```
{
    "experiment": {
        "id": "ABC12DeFGhI3jKLMNOP",
        "experimentTemplateId": "ABCDE1fgHIJkLmNop",
        "roleArn": "arn:aws:iam::123456789012:role/myRole",
        "state": {
            "status": "completed",
            "reason": "Experiment completed."
        },
        "targets": {
            "Instances-Target-1": {
                "resourceType": "aws:ec2:instance",
                "resourceArns": [
                    "arn:aws:ec2:us-west-2:123456789012:instance/
i-12a3b4c56d78e9012"
                ],
                "selectionMode": "ALL"
```

```
            }
        },
        "actions": {
            "reboot": {
                "actionId": "aws:ec2:reboot-instances",
                "parameters": {},
                "targets": {
                    "Instances": "Instances-Target-1"
                },
                "state": {
                    "status": "completed",
                    "reason": "Action was completed."
                }
            }
        },
        "stopConditions": [
            {
                "source": "none"
            }
        ],
        "creationTime": 1616432509.662,
        "startTime": 1616432509.962,
        "endTime": 1616432522.307,
        "tags": {}
    }
}
```

Per ulteriori informazioni, vedere Experiments for AWS FIS nella Guida per l'utente di AWS Fault Injection Simulator.

- Per i dettagli sull'API, vedere GetExperimentin AWS CLI Command Reference.

`list-actions`

Il seguente esempio di codice mostra come utilizzare`list-actions`.

AWS CLI

Come elencare le azioni

L'esempio `list-actions` seguente elenca le azioni disponibili.

```
aws fis list-actions
```

Output:


```
{
    "actions": [
        {
            "id": "aws:ec2:reboot-instances",
            "description": "Reboot the specified EC2 instances.",
            "targets": {
                "Instances": {
                    "resourceType": "aws:ec2:instance"
                }
            },
            "tags": {}
        },
        {
            "id": "aws:ec2:stop-instances",
            "description": "Stop the specified EC2 instances.",
            "targets": {
                "Instances": {
                    "resourceType": "aws:ec2:instance"
                }
            },
            "tags": {}
        },
        {
            "id": "aws:ec2:terminate-instances",
            "description": "Terminate the specified EC2 instances.",
            "targets": {
                "Instances": {
                    "resourceType": "aws:ec2:instance"
                }
            },
            "tags": {}
        },
        {
            "id": "aws:ecs:drain-container-instances",
            "description": "Drain percentage of underlying EC2 instances on an ECS
 cluster.",
            "targets": {
                "Clusters": {
                    "resourceType": "aws:ecs:cluster"
                }
            },
            "tags": {}
```

```
        },
        {
            "id": "aws:eks:terminate-nodegroup-instances",
            "description": "Terminates a percentage of the underlying EC2 instances
in an EKS cluster.",
            "targets": {
                "Nodegroups": {
                    "resourceType": "aws:eks:nodegroup"
                }
            },
            "tags": {}
        },
        {
            "id": "aws:fis:inject-api-internal-error",
            "description": "Cause an AWS service to return internal error responses
for specific callers and operations.",
            "targets": {
                "Roles": {
                    "resourceType": "aws:iam:role"
                }
            },
            "tags": {}
        },
        {
            "id": "aws:fis:inject-api-throttle-error",
            "description": "Cause an AWS service to return throttled responses for
specific callers and operations.",
            "targets": {
                "Roles": {
                    "resourceType": "aws:iam:role"
                }
            },
            "tags": {}
        },
        {
        "id": "aws:fis:inject-api-unavailable-error",
            "description": "Cause an AWS service to return unavailable error
responses for specific callers and operations.",
            "targets": {
                "Roles": {
                    "resourceType": "aws:iam:role"
                }
            },
            "tags": {}
```

```
        },
        {
            "id": "aws:fis:wait",
            "description": "Wait for the specified duration. Stop condition
 monitoring will continue during this time.",
            "tags": {}
        },
        {
            "id": "aws:rds:failover-db-cluster",
            "description": "Failover a DB Cluster to one of the replicas.",
            "targets": {
                "Clusters": {
                    "resourceType": "aws:rds:cluster"
                }
            },
            "tags": {}
        },
        {
            "id": "aws:rds:reboot-db-instances",
            "description": "Reboot the specified DB instances.",
            "targets": {
                "DBInstances": {
                    "resourceType": "aws:rds:db"
                }
            },
            "tags": {}
        },
        {
            "id": "aws:ssm:send-command",
            "description": "Run the specified SSM document.",
            "targets": {
                "Instances": {
                    "resourceType": "aws:ec2:instance"
                }
            },
            "tags": {}
        }
    ]
}
```

Per ulteriori informazioni, consulta Azioni nella Guida per l'utente di AWS Fault Injection Simulator.

- Per i dettagli sull'API, consulta ListActions AWS CLICommand Reference.

`list-experiment-templates`

Il seguente esempio di codice mostra come utilizzare`list-experiment-templates`.

AWS CLI

Come elencare i modelli di esperimento

L'`list-experiment-templates`esempio seguente elenca i modelli di esperimento presenti nel tuo AWS account.

```
aws fis list-experiment-templates
```

Output:

```
{
    "experimentTemplates": [
        {
            "id": "ABCDE1fgHIJkLmNop",
            "description": "myExperimentTemplate",
            "creationTime": 1616017191.124,
            "lastUpdateTime": 1616017191.124,
            "tags": {
                "key": "value"
            }
        }
    ]
}
```

Per ulteriori informazioni, consulta Modelli di esperimenti nella Guida per l'utente di AWS Fault Injection Simulator.

- Per i dettagli sull'API, consulta ListExperimentTemplates AWS CLICommand Reference.

`list-experiments`

Il seguente esempio di codice mostra come utilizzare`list-experiments`.

AWS CLI

Come elencare gli esperimenti

L'`list-experiments`esempio seguente elenca gli esperimenti nel tuo AWS account.

```
aws fis list-experiments
```

Output:

```
{
    "experiments": [
        {
            "id": "ABCdeF1GHiJkLM23NO",
            "experimentTemplateId": "ABCDE1fgHIJkLmNop",
            "state": {
                "status": "running",
                "reason": "Experiment is running."
            },
            "creationTime": 1616017341.197,
            "tags": {
            "key": "value"
            }
        }
    ]
}
```

Per ulteriori informazioni, consulta Esperimenti nella Guida per l'utente di AWS Fault Injection Simulator.

- Per i dettagli sull'API, consulta ListExperiments AWS CLICommand Reference.

`list-tags-for-resource`

Il seguente esempio di codice mostra come utilizzare`list-tags-for-resource`.

AWS CLI

Come elencare i tag per una risorsa

L'esempio `list-tags-for-resource` seguente elenca i tag per la risorsa specificata.

```
aws fis list-tags-for-resource \
    --resource-arn arn:aws:fis:us-west-2:123456789012:experiment/ABC12DeFGhI3jKLMNOP
```

Output:

```
{
```

```
    "tags": {
        "key1": "value1",
        "key2": "value2"
    }
}
```

Per ulteriori informazioni, consulta Etichettare le risorse FIS nella AWS Guida per l'utente di AWS Fault Injection Simulator.

- Per i dettagli sull'API, vedere ListTagsForResourcein AWS CLI Command Reference.

`start-experiment`

Il seguente esempio di codice mostra come utilizzare`start-experiment`.

AWS CLI

Come avviare un esperimento

L'esempio `start-experiment` seguente avvia l'esperimento specificato.

```
aws fis start-experiment \
    --experiment-template-id ABCDE1fgHIJkLmNop
```

Output:

```
{
    "experiment": {
        "id": "ABC12DeFGhI3jKLMNOP",
        "experimentTemplateId": "ABCDE1fgHIJkLmNop",
        "roleArn": "arn:aws:iam::123456789012:role/myRole",
        "state": {
            "status": "initiating",
            "reason": "Experiment is initiating."
        },
        "targets": {
            "Instances-Target-1": {
                "resourceType": "aws:ec2:instance",
                "resourceArns": [
                    "arn:aws:ec2:us-west-2:123456789012:instance/
i-12a3b4c56d78e9012"
                ],
                "selectionMode": "ALL"
```

```
            }
        },
        "actions": {
            "reboot": {
                "actionId": "aws:ec2:reboot-instances",
                "parameters": {},
                "targets": {
                    "Instances": "Instances-Target-1"
                },
                "state": {
                    "status": "pending",
                    "reason": "Initial state"
                }
            }
        },
        "stopConditions": [
            {
                "source": "none"
            }
        ],
        "creationTime": 1616432464.025,
        "startTime": 1616432464.374,
        "tags": {}
    }
}
```

Per ulteriori informazioni, vedere Experiments for AWS FIS nella Guida per l'utente di AWS Fault Injection Simulator.

- Per i dettagli sull'API, vedere StartExperimentin AWS CLI Command Reference.

`stop-experiment`

Il seguente esempio di codice mostra come utilizzare`stop-experiment`.

AWS CLI

Come interrompere un esperimento

L'esempio `stop-experiment` seguente arresta l'esecuzione dell'esperimento specificato.

```
aws fis stop-experiment \
    --id ABC12DeFGhI3jKLMNOP
```

Output:

```
{
    "experiment": {
        "id": "ABC12DeFGhI3jKLMNOP",
        "experimentTemplateId": "ABCDE1fgHIJkLmNop",
        "roleArn": "arn:aws:iam::123456789012:role/myRole",
        "state": {
            "status": "stopping",
            "reason": "Stopping Experiment."
        },
        "targets": {
            "Instances-Target-1": {
                "resourceType": "aws:ec2:instance",
                "resourceArns": [
                    "arn:aws:ec2:us-west-2:123456789012:instance/
i-12a3b4c56d78e9012"
                ],
                "selectionMode": "ALL"
            }
        },
        "actions": {
            "reboot": {
                "actionId": "aws:ec2:reboot-instances",
                "parameters": {},
                "targets": {
                    "Instances": "Instances-Target-1"
                },
                "startAfter": [
                    "wait"
                ],
                "state": {
                    "status": "pending",
                    "reason": "Initial state."
                }
            },
            "wait": {
                "actionId": "aws:fis:wait",
                "parameters": {
                    "duration": "PT5M"
                },
                "state": {
                    "status": "running",
                    "reason": ""
```

```
                }
            }
        },
        "stopConditions": [
            {
                "source": "none"
            }
        ],
        "creationTime": 1616432680.927,
        "startTime": 1616432681.177,
        "tags": {}
    }
}
```

Per ulteriori informazioni, vedere Experiments for AWS FIS nella Guida per l'utente di AWS Fault Injection Simulator.

- Per i dettagli sull'API, vedere StopExperimentin AWS CLI Command Reference.

`tag-resource`

Il seguente esempio di codice mostra come utilizzare`tag-resource`.

AWS CLI

Come taggare a una risorsa

L'esempio `tag-resource` seguente associa i tag alla risorsa specificata.

```
aws fis tag-resource \
    --resource-arn arn:aws:fis:us-west-2:123456789012:experiment/ABC12DeFGhI3jKLMNOP
 \
    --tags key1=value1,key2=value2
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Etichettare le risorse FIS nella AWS Guida per l'utente di AWS Fault Injection Simulator.

- Per i dettagli sull'API, vedere TagResourcein AWS CLI Command Reference.

`untag-resource`

Il seguente esempio di codice mostra come utilizzare`untag-resource`.

AWS CLI

Come rimuovere un tag da una risorsa

L'esempio `untag-resource` seguente rimuove i tag dall'archivio specificato.

```
aws fis untag-resource \
    --resource-arn arn:aws:fis:us-west-2:123456789012:experiment/ABC12DeFGhI3jKLMNOP
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Etichettare le risorse FIS nella AWS Guida per l'utente di AWS Fault Injection Simulator.

- Per i dettagli sull'API, vedere UntagResourcein AWS CLI Command Reference.

`update-experiment-template`

Il seguente esempio di codice mostra come utilizzare`update-experiment-template`.

AWS CLI

Come aggiornare un modello di esperimento

L'esempio `update-experiment-template` seguente aggiorna la descrizione del modello di esperimento specificato.

```
aws fis update-experiment-template \
    --id ABCDE1fgHIJkLmNop \
    ---description myExperimentTemplate
```

Output:

```
{
    "experimentTemplate": {
        "id": "ABCDE1fgHIJkLmNop",
```

```
        "description": "myExperimentTemplate",
        "targets": {
            "Instances-Target-1": {
                "resourceType": "aws:ec2:instance",
                "resourceArns": [
                    "arn:aws:ec2:us-west-2:123456789012:instance/
i-12a3b4c56d78e9012"
                ],
                "selectionMode": "ALL"
            }
        },
        "actions": {
            "testaction": {
                "actionId": "aws:ec2:stop-instances",
                "parameters": {},
                "targets": {
                    "Instances": "Instances-Target-1"
                }
            }
        },
        "stopConditions": [
            {
                "source": "none"
            }
        ],
        "creationTime": 1616017191.124,
        "lastUpdateTime": 1616017859.607,
        "roleArn": "arn:aws:iam::123456789012:role/FISRole",
        "tags": {
            "key": "value"
        }
    }
}
```

Per ulteriori informazioni, consulta Aggiornare un modello di esperimento nella Guida per l'utente di AWS Fault Injection Simulator.

- Per i dettagli sull'API, consulta UpdateExperimentTemplate AWS CLICommand Reference.

Esempi di Amazon GameLift Servers che utilizzano AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface con Amazon GameLift Servers.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`create-build`

Il seguente esempio di codice mostra come usare`create-build`.

AWS CLI

Esempio 1: come creare una build del gioco da file in un bucket S3

L'esempio `create-build` seguente crea una risorsa build personalizzata del gioco. Utilizza file compressi archiviati in una posizione S3 in un AWS account che controlli tu. Questo esempio presuppone che tu abbia già creato un ruolo IAM che GameLift autorizza Amazon ad accedere alla posizione S3. Poiché la richiesta non specifica un sistema operativo, per impostazione predefinita la nuova risorsa build è WINDOWS_2012.

```
aws gamelift create-build \
    --storage-location file://storage-loc.json \
    --name MegaFrogRaceServer.NA \
    --build-version 12345.678
```

Contenuto di `storage-loc.json`:

```
{
    "Bucket":"MegaFrogRaceServer_NA_build_files"
    "Key":"MegaFrogRaceServer_build_123.zip"
    "RoleArn":"arn:aws:iam::123456789012:role/gamelift"
}
```

Output:

```
{
    "Build": {
        "BuildArn": "arn:aws:gamelift:us-west-2::build/build-a1b2c3d4-5678-90ab-
cdef-EXAMPLE11111",
        "BuildId": "build-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
        "CreationTime": 1496708916.18,
        "Name": "MegaFrogRaceServer.NA",
        "OperatingSystem": "WINDOWS_2012",
        "SizeOnDisk": 479303,
        "Status": "INITIALIZED",
        "Version": "12345.678"
    },
    "StorageLocation": {
        "Bucket": "MegaFrogRaceServer_NA_build_files",
        "Key": "MegaFrogRaceServer_build_123.zip"
    }
}
```

Esempio 2: creare una risorsa per la creazione di un gioco su cui caricare manualmente i file su GameLift

L'esempio `create-build` seguente crea una nuova risorsa build. Inoltre, ottiene una posizione di archiviazione e credenziali temporanee che ti consentono di caricare manualmente la build del gioco nella GameLift posizione in Amazon S3. Dopo aver caricato correttamente la build, il GameLift servizio convalida la build e aggiorna lo stato della nuova build.

```
aws gamelift create-build \
    --name MegaFrogRaceServer.NA \
    --build-version 12345.678 \
    --operating-system AMAZON_LINUX
```

Output:

```
{
    "Build": {
        "BuildArn": "arn:aws:gamelift:us-west-2::build/build-a1b2c3d4-5678-90ab-
cdef-EXAMPLE11111",
        "BuildId": "build-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
        "CreationTime": 1496708916.18,
        "Name": "MegaFrogRaceServer.NA",
        "OperatingSystem": "AMAZON_LINUX",
        "SizeOnDisk": 0,
```

```
        "Status": "INITIALIZED",
        "Version": "12345.678"
    },
    "StorageLocation": {
        "Bucket": "gamelift-builds-us-west-2",
        "Key": "123456789012/build-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
    },
    "UploadCredentials": {
        "AccessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "SecretAccessKey": "wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY",
        "SessionToken": "AgoGb3JpZ2luENz...EXAMPLETOKEN=="
    }
}
```

Per ulteriori informazioni, consulta Caricare una build di server personalizzata GameLift nella Amazon GameLift Developer Guide.

- Per i dettagli sull'API, consulta CreateBuild AWS CLICommand Reference.

`create-fleet`

Il seguente esempio di codice mostra come utilizzare`create-fleet`.

AWS CLI

Esempio 1: come creare un parco istanze Linux di base

L'esempio `create-fleet` seguente crea un parco istanze Linux on demand con configurazione minima per ospitare una build personalizzata del server. Puoi completare la configurazione utilizzando `update-fleet`.

```
aws gamelift create-fleet \
    --name MegaFrogRaceServer.NA.v2 \
    --description 'Hosts for v2 North America' \
    --build-id build-1111aaaa-22bb-33cc-44dd-5555eeee66ff \
    --certificate-configuration 'CertificateType=GENERATED' \
    --ec2-instance-type c4.large \
    --fleet-type ON_DEMAND \
    --runtime-configuration 'ServerProcesses=[{LaunchPath=/local/game/release-na/
MegaFrogRace_Server.exe,ConcurrentExecutions=1}]'
```

Output:

```
{
    "FleetAttributes": {
        "BuildId": "build-1111aaaa-22bb-33cc-44dd-5555eeee66ff",
        "CertificateConfiguration": {
            "CertificateType": "GENERATED"
        },
        "CreationTime": 1496365885.44,
        "Description": "Hosts for v2 North America",
        "FleetArn": "arn:aws:gamelift:us-west-2:444455556666:fleet/
fleet-2222bbbb-33cc-44dd-55ee-6666ffff77aa",
        "FleetId": "fleet-2222bbbb-33cc-44dd-55ee-6666ffff77aa",
        "FleetType": "ON_DEMAND",
        "InstanceType": "c4.large",
        "MetricGroups": ["default"],
        "Name": "MegaFrogRace.NA.v2",
        "NewGameSessionProtectionPolicy": "NoProtection",
        "OperatingSystem": "AMAZON_LINUX",
        "ServerLaunchPath": "/local/game/release-na/MegaFrogRace_Server.exe",
        "Status": "NEW"
    }
}
```

Esempio 2: come creare un parco istanze Windows di base

L'esempio `create-fleet` seguente crea un parco istanze spot Windows con configurazione minima per ospitare una build personalizzata del server. Puoi completare la configurazione utilizzando `update-fleet`.

```
aws gamelift create-fleet \
    --name MegaFrogRace.NA.v2 \
    --description 'Hosts for v2 North America' \
    --build-id build-2222aaaa-33bb-44cc-55dd-6666eeee77ff  \
    --certificate-configuration 'CertificateType=GENERATED' \
    --ec2-instance-type c4.large \
    --fleet-type SPOT \
    --runtime-configuration 'ServerProcesses=[{LaunchPath=C:\game
\Bin64.Release.Dedicated\MegaFrogRace_Server.exe,ConcurrentExecutions=1}]'
```

Output:

```
{
    "FleetAttributes": {
```

```
        "BuildId": "build-2222aaaa-33bb-44cc-55dd-6666eeee77ff",
        "CertificateConfiguration": {
            "CertificateType": "GENERATED"
        },
        "CreationTime": 1496365885.44,
        "Description": "Hosts for v2 North America",
        "FleetArn": "arn:aws:gamelift:us-west-2:444455556666:fleet/
fleet-2222bbbb-33cc-44dd-55ee-6666ffff77aa",
        "FleetId": "fleet-2222bbbb-33cc-44dd-55ee-6666ffff77aa",
        "FleetType": "SPOT",
        "InstanceType": "c4.large",
        "MetricGroups": ["default"],
        "Name": "MegaFrogRace.NA.v2",
        "NewGameSessionProtectionPolicy": "NoProtection",
        "OperatingSystem": "WINDOWS_2012",
        "ServerLaunchPath": "C:\game\Bin64.Release.Dedicated
\MegaFrogRace_Server.exe",
        "Status": "NEW"
    }
}
```

Esempio 3: come creare un parco istanze con configurazione completa

L'esempio `create-fleet` seguente crea un parco istanze spot Windows per una build personalizzata del server, con le impostazioni di configurazione più comunemente utilizzate.

```
aws gamelift create-fleet \
    --name MegaFrogRace.NA.v2 \
    --description 'Hosts for v2 North America' \
    --build-id build-2222aaaa-33bb-44cc-55dd-6666eeee77ff \
    --certificate-configuration 'CertificateType=GENERATED' \
    --ec2-instance-type c4.large \
    --ec2-inbound-permissions
 'FromPort=33435,ToPort=33435,IpRange=10.24.34.0/23,Protocol=UDP' \
    --fleet-type SPOT \
    --new-game-session-protection-policy FullProtection \
    --runtime-configuration file://runtime-config.json \
    --metric-groups default \
    --instance-role-arn 'arn:aws:iam::444455556666:role/GameLiftS3Access'
```

Contenuto di `runtime-config.json`:

```
GameSessionActivationTimeoutSeconds=300,
```

```
 MaxConcurrentGameSessionActivations=2,
 ServerProcesses=[
   {LaunchPath=C:\game\Bin64.Release.Dedicated\MegaFrogRace_Server.exe,Parameters=-
debug,ConcurrentExecutions=1},
   {LaunchPath=C:\game\Bin64.Release.Dedicated
\MegaFrogRace_Server.exe,ConcurrentExecutions=1}]
```

Output:

```
{
    "FleetAttributes": {
        "InstanceRoleArn": "arn:aws:iam::444455556666:role/GameLiftS3Access",
        "Status": "NEW",
        "InstanceType": "c4.large",
        "FleetArn": "arn:aws:gamelift:us-west-2:444455556666:fleet/
fleet-2222bbbb-33cc-44dd-55ee-6666ffff77aa",
        "FleetId": "fleet-2222bbbb-33cc-44dd-55ee-6666ffff77aa",
        "Description": "Hosts for v2 North America",
        "FleetType": "SPOT",
        "OperatingSystem": "WINDOWS_2012",
        "Name": "MegaFrogRace.NA.v2",
        "CreationTime": 1569309011.11,
        "MetricGroups": [
            "default"
        ],
        "BuildId": "build-2222aaaa-33bb-44cc-55dd-6666eeee77ff",
        "ServerLaunchParameters": "abc",
        "ServerLaunchPath": "C:\\game\\Bin64.Release.Dedicated\
\MegaFrogRace_Server.exe",
        "NewGameSessionProtectionPolicy": "FullProtection",
        "CertificateConfiguration": {
            "CertificateType": "GENERATED"
        }
    }
}
```

Esempio 4: come creare un parco istanze server in tempo reale

L'`create-fleet`esempio seguente crea una flotta di istanze Spot con uno script di configurazione in tempo reale che è stato caricato su Amazon. GameLift Tutti i server in tempo reale vengono implementati su computer Linux. Ai fini di questo esempio, presupponiamo che lo script in tempo reale caricato includa più file di script, con la funzione `Init()` inclusa nel file di

script denominato `MainScript.js`. Come mostrato, questo file viene identificato come script di avvio nella configurazione del runtime.

```
aws gamelift create-fleet \
    --name MegaFrogRace.NA.realtime \
    --description 'Mega Frog Race Realtime fleet' \
    --script-id script-1111aaaa-22bb-33cc-44dd-5555eeee66ff \
    --ec2-instance-type c4.large \
    --fleet-type SPOT \
    --certificate-configuration 'CertificateType=GENERATED' --runtime-configuration
 'ServerProcesses=[{LaunchPath=/local/game/MainScript.js,Parameters=+map
 Winter444,ConcurrentExecutions=5}]'
```

Output:

```
{
    "FleetAttributes": {
        "FleetId": "fleet-2222bbbb-33cc-44dd-55ee-6666ffff77aa",
        "Status": "NEW",
        "CreationTime": 1569310745.212,
        "InstanceType": "c4.large",
        "NewGameSessionProtectionPolicy": "NoProtection",
        "CertificateConfiguration": {
            "CertificateType": "GENERATED"
        },
        "Name": "MegaFrogRace.NA.realtime",
        "ScriptId": "script-1111aaaa-22bb-33cc-44dd-5555eeee66ff",
        "FleetArn": "arn:aws:gamelift:us-west-2:444455556666:fleet/
fleet-2222bbbb-33cc-44dd-55ee-6666ffff77aa",
        "FleetType": "SPOT",
        "MetricGroups": [
            "default"
        ],
        "Description": "Mega Frog Race Realtime fleet",
        "OperatingSystem": "AMAZON_LINUX"
    }
}
```

- Per i dettagli sull'API, consulta AWS CLI Command CreateFleetReference.

`create-game-session-queue`

Il seguente esempio di codice mostra come utilizzare`create-game-session-queue`.

AWS CLI

Esempio 1: come configurare una coda ordinata di sessioni di gioco

L'esempio `create-game-session-queue` seguente crea una nuova coda di sessioni di gioco con destinazioni in due Regioni. Configura anche la coda in modo da impostare il timeout delle richieste delle sessioni di gioco dopo un'attesa di 10 minuti per l'assegnazione. Poiché non sono state definite politiche di latenza, GameLift tenta di posizionare tutte le sessioni di gioco con la prima destinazione elencata.

```
aws gamelift create-game-session-queue \
    --name MegaFrogRaceServer-NA \
    --destinations file://destinations.json \
    --timeout-in-seconds 600
```

Contenuto di `destinations.json`:

```
{
    "Destinations": [
        {"DestinationArn": "arn:aws:gamelift:us-west-2::fleet/fleet-
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111" },
        {"DestinationArn": "arn:aws:gamelift:us-west-1::fleet/fleet-
a1b2c3d4-5678-90ab-cdef-EXAMPLE22222" }
    ]
}
```

Output:

```
{
    "GameSessionQueues": [
        {
            "Name": "MegaFrogRaceServer-NA",
            "GameSessionQueueArn": "arn:aws:gamelift:us-
west-2:123456789012:gamesessionqueue/MegaFrogRaceServer-NA",
            "TimeoutInSeconds": 600,
            "Destinations": [
                {"DestinationArn": "arn:aws:gamelift:us-west-2::fleet/fleet-
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"},
```

```
                {"DestinationArn": "arn:aws:gamelift:us-west-1::fleet/fleet-
a1b2c3d4-5678-90ab-cdef-EXAMPLE22222"}
            ]
        }
    ]
}
```

Esempio 2: come configurare una coda di sessioni di gioco con policy di latenza per i giocatori

L'esempio seguente `create-game-session-queue`esempio seguente crea una nuova coda di sessioni di gioco con policy di latenza per due giocatori. La prima policy stabilisce un limite di latenza di 100 ms, che viene applicato durante il primo minuto di un tentativo di assegnazione della sessione di gioco. La seconda policy aumenta il limite di latenza a 200 ms fino al timeout della richiesta di assegnazione, che avviene a 3 minuti.

```
aws gamelift create-game-session-queue \
    --name MegaFrogRaceServer-NA \
    --destinations file://destinations.json \
    --player-latency-policies file://latency-policies.json \
    --timeout-in-seconds 180
```

Contenuto di `destinations.json`:

```
{
    "Destinations": [
        { "DestinationArn": "arn:aws:gamelift:us-west-2::fleet/fleet-
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111" },
        { "DestinationArn": "arn:aws:gamelift:us-east-1::fleet/fleet-
a1b2c3d4-5678-90ab-cdef-EXAMPLE22222" }
    ]
}
```

Contenuto di `latency-policies.json`:

```
{
    "PlayerLatencyPolicies": [
        {"MaximumIndividualPlayerLatencyMilliseconds": 200},
        {"MaximumIndividualPlayerLatencyMilliseconds": 100, "PolicyDurationSeconds":
 60}
    ]
}
```

Output:

```
{
    "GameSessionQueue": {
        "Name": "MegaFrogRaceServer-NA",
        "GameSessionQueueArn": "arn:aws:gamelift:us-
west-2:111122223333:gamesessionqueue/MegaFrogRaceServer-NA",
        "TimeoutInSeconds": 600,
        "PlayerLatencyPolicies": [
            {
                "MaximumIndividualPlayerLatencyMilliseconds": 100,
                "PolicyDurationSeconds": 60
            },
            {
                "MaximumIndividualPlayerLatencyMilliseconds": 200
            }
        ]
        "Destinations": [
            {"DestinationArn": "arn:aws:gamelift:us-west-2::fleet/fleet-
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"},
            {"DestinationArn": "arn:aws:gamelift:us-east-1::fleet/fleet-
a1b2c3d4-5678-90ab-cdef-EXAMPLE22222"}
        ],
    }
}
```

Per ulteriori informazioni, consulta Create a Queue nella Amazon GameLift Developer Guide.

- Per i dettagli sull'API, consulta CreateGameSessionQueue AWS CLICommand Reference.

`delete-build`

Il seguente esempio di codice mostra come utilizzare`delete-build`.

AWS CLI

Come eliminare una build personalizzata del gioco

L'`delete-build`esempio seguente rimuove una build dal tuo GameLift account Amazon. Una volta eliminata, la build non può più essere utilizzata per creare nuovi parchi istanze. Questa operazione non può essere annullata.

```
aws gamelift delete-build \
```

```
    --build-id build-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta DeleteBuild AWS CLICommand Reference.

`delete-fleet`

Il seguente esempio di codice mostra come utilizzare`delete-fleet`.

AWS CLI

Come eliminare un parco istanze non più in uso

L'esempio `delete-fleet` seguente rimuove un parco che è stato ridotto a zero istanze. Se la capacità del parco è maggiore di zero, la richiesta ha esito negativo e viene restituito un errore HTTP 400.

```
aws gamelift delete-fleet \
    --fleet-id fleet-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Manage GameLift Fleets nella Amazon GameLift Developer Guide.

- Per i dettagli sull'API, consulta DeleteFleet AWS CLICommand Reference.

`delete-game-session-queue`

Il seguente esempio di codice mostra come utilizzare`delete-game-session-queue`.

AWS CLI

Come eliminare una coda di sessioni di gioco

L'esempio `delete-game-session-queue` seguente elimina una coda di sessioni di gioco specificata.

```
aws gamelift delete-game-session-queue \
    --name MegaFrogRace-NA
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta DeleteGameSessionQueue AWS CLICommand Reference.

`describe-build`

Il seguente esempio di codice mostra come utilizzare`describe-build`.

AWS CLI

Come ottenere informazioni su una build personalizzata del gioco

L'esempio `describe-build` seguente recupera le proprietà di una risorsa build del server di gioco.

```
aws gamelift describe-build \
    --build-id build-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Output:

```
{
    "Build": {
        "BuildArn": "arn:aws:gamelift:us-west-2::build/build-a1b2c3d4-5678-90ab-
cdef-EXAMPLE11111",
        "BuildId": "build-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
        "CreationTime": 1496708916.18,
        "Name": "My_Game_Server_Build_One",
        "OperatingSystem": "AMAZON_LINUX",
        "SizeOnDisk": 1304924,
        "Status": "READY",
        "Version": "12345.678"
    }
}
```

Per ulteriori informazioni, consulta Caricare una build di server personalizzata GameLift nella Amazon GameLift Developer Guide.

- Per i dettagli sull'API, consulta DescribeBuild AWS CLICommand Reference.

`describe-ec2-instance-limits`

Il seguente esempio di codice mostra come utilizzare`describe-ec2-instance-limits`.

AWS CLI

Per recuperare i limiti di servizio per un tipo di EC2 istanza

L'`describe-ec2-instance-limits`esempio seguente visualizza il numero massimo di istanze consentite e le istanze correnti in uso per il tipo di EC2 istanza specificato nella regione corrente. Il risultato indica che vengono utilizzate solo cinque delle venti istanze consentite.

```
aws gamelift describe-ec2-instance-limits \
    --ec2-instance-type m5.large
```

Output:

```
{
    "EC2InstanceLimits": [
        {
            "EC2InstanceType": ""m5.large",
            "CurrentInstances": 5,
            "InstanceLimit": 20
        }
    ]
}
```

Per ulteriori informazioni, consulta Choose Computing Resources nella Amazon GameLift Developer Guide.

- Per i dettagli sull'API, consulta DescribeEc2 InstanceLimits in AWS CLI Command Reference.

`describe-fleet-attributes`

Il seguente esempio di codice mostra come utilizzare`describe-fleet-attributes`.

AWS CLI

Esempio 1: come visualizzare gli attributi di un elenco di parchi istanze

L'esempio `describe-fleet-attributes` seguente recupera gli attributi di due parchi specificati. Come mostrato, i parchi richiesti vengono implementati con la stessa build, uno per le istanze on demand e uno per le istanze spot, con alcune differenze di configurazione minori.

```
aws gamelift describe-fleet-attributes \
```

```
    --fleet-ids arn:aws:gamelift:us-west-2::fleet/fleet-a1b2c3d4-5678-90ab-cdef-
EXAMPLE11111 fleet-a1b2c3d4-5678-90ab-cdef-EXAMPLE22222
```

Output:

```
{
    "FleetAttributes": [
        {
            "FleetId": "fleet-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
            "FleetArn": "arn:aws:gamelift:us-west-2::fleet/fleet-a1b2c3d4-5678-90ab-
cdef-EXAMPLE11111",
            "FleetType": "ON_DEMAND",
            "InstanceType": "c4.large",
            "Description": "On-demand hosts for v2 North America",
            "Name": "MegaFrogRaceServer.NA.v2-od",
            "CreationTime": 1568836191.995,
            "Status": "ACTIVE",
            "BuildId": "build-a1b2c3d4-5678-90ab-cdef-EXAMPLE33333",
            "BuildArn": "arn:aws:gamelift:us-west-2::build/build-a1b2c3d4-5678-90ab-
cdef-EXAMPLE33333",
            "ServerLaunchPath": "C:\\game\\MegaFrogRace_Server.exe",
            "ServerLaunchParameters": "+gamelift_start_server",
            "NewGameSessionProtectionPolicy": "NoProtection",
            "OperatingSystem": "WINDOWS_2012",
            "MetricGroups": [
                "default"
            ],
            "CertificateConfiguration": {
                "CertificateType": "DISABLED"
            }
        },
        {
            "FleetId": "fleet-a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
            "FleetArn": "arn:aws:gamelift:us-west-2::fleet/fleet-a1b2c3d4-5678-90ab-
cdef-EXAMPLE22222",
            "FleetType": "SPOT",
            "InstanceType": "c4.large",
            "Description": "On-demand hosts for v2 North America",
            "Name": "MegaFrogRaceServer.NA.v2-spot",
            "CreationTime": 1568838275.379,
            "Status": "ACTIVATING",
            "BuildId": "build-a1b2c3d4-5678-90ab-cdef-EXAMPLE33333",
```

```
            "BuildArn": "arn:aws:gamelift:us-west-2::build/build-a1b2c3d4-5678-90ab-
cdef-EXAMPLE33333",
            "ServerLaunchPath": "C:\\game\\MegaFrogRace_Server.exe",
            "NewGameSessionProtectionPolicy": "NoProtection",
            "OperatingSystem": "WINDOWS_2012",
                "MetricGroups": [
                "default"
            ],
            "CertificateConfiguration": {
                "CertificateType": "GENERATED"
            }
        }
    ]
}
```

Esempio 2: come richiedere gli attributi per tutti i parchi istanze

L'esempio `describe-fleet-attributes` seguente restituisce gli attributi di tutti i parchi con qualsiasi stato. Questo esempio mostra l'utilizzo dei parametri di paginazione per restituire un parco per volta.

```
aws gamelift describe-fleet-attributes \
    --limit 1
```

Output:

```
{
    "FleetAttributes": [
        {
            "FleetId": "fleet-a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
            "FleetArn": "arn:aws:gamelift:us-west-2::fleet/fleet-a1b2c3d4-5678-90ab-
cdef-EXAMPLE22222",
            "FleetType": "SPOT",
            "InstanceType": "c4.large",
            "Description": "On-demand hosts for v2 North America",
            "Name": "MegaFrogRaceServer.NA.v2-spot",
            "CreationTime": 1568838275.379,
            "Status": "ACTIVATING",
            "BuildId": "build-a1b2c3d4-5678-90ab-cdef-EXAMPLE33333",
            "BuildArn": "arn:aws:gamelift:us-west-2::build/build-a1b2c3d4-5678-90ab-
cdef-EXAMPLE33333",
            "ServerLaunchPath": "C:\\game\\MegaFrogRace_Server.exe",
```

```
            "NewGameSessionProtectionPolicy": "NoProtection",
            "OperatingSystem": "WINDOWS_2012",
            "MetricGroups": [
                "default"
            ],
            "CertificateConfiguration": {
                "CertificateType": "GENERATED"
            }
        }
    ],
    "NextToken":
 "eyJhd3NBY2NvdW50SWQiOnsicyI6IjMwMjc3NjAxNjM5OCJ9LCJidWlsZElkIjp7InMiOiJidWlsZC01NWYxZTZmMS
}
```

L'output include un valore `NextToken` che puoi utilizzare quando chiami il comando una seconda volta. Passa il valore al parametro `--next-token` per specificare dove recuperare l'output. Il comando seguente restituisce il secondo risultato nell'output.

```
aws gamelift describe-fleet-attributes \
    --limit 1 \
    --next-
token eyJhd3NBY2NvdW50SWQiOnsicyI6IjMwMjc3NjAxNjM5OCJ9LCJidWlsZElkIjp7InMiOiJidWlsZC01NWYxZT
```

Ripeti il comando finché la risposta non include un valore `NextToken`.

Per ulteriori informazioni, consulta Configurazione delle GameLift flotte nella Amazon GameLift Developer Guide.

- Per i dettagli sull'API, consulta DescribeFleetAttributes AWS CLICommand Reference.

`describe-fleet-capacity`

Il seguente esempio di codice mostra come utilizzare`describe-fleet-capacity`.

AWS CLI

Come visualizzare lo stato della capacità per un elenco di parchi istanze

L'esempio `describe-fleet-capacity` seguente recupera la capacità corrente per due parchi specificati.

```
aws gamelift describe-fleet-capacity \
```

```
    --fleet-ids arn:aws:gamelift:us-west-2::fleet/fleet-a1b2c3d4-5678-90ab-cdef-
EXAMPLE11111 fleet-a1b2c3d4-5678-90ab-cdef-EXAMPLE22222
```

Output:

```
{
    "FleetCapacity": [
        {
            "FleetId": "fleet-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
            "InstanceType": "c5.large",
            "InstanceCounts": {
                "DESIRED": 10,
                "MINIMUM": 1,
                "MAXIMUM": 20,
                "PENDING": 0,
                "ACTIVE": 10,
                "IDLE": 3,
                "TERMINATING": 0
            }
        },
        {
            "FleetId": "fleet-a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
            "InstanceType": "c5.large",
            "InstanceCounts": {
                "DESIRED": 13,
                "MINIMUM": 1,
                "MAXIMUM": 20,
                "PENDING": 0,
                "ACTIVE": 15,
                "IDLE": 2,
                "TERMINATING": 2
            }
        }

    ]
}
```

Per ulteriori informazioni, consulta GameLift Metrics for Fleets nella Amazon GameLift Developer Guide.

- Per i dettagli sull'API, consulta AWS CLI Command DescribeFleetCapacityReference.

`describe-fleet-events`

Il seguente esempio di codice mostra come utilizzare`describe-fleet-events`.

AWS CLI

Come richiedere informazioni sugli eventi per un intervallo di tempo specificato

L'esempio `describe-fleet-events` seguente visualizza i dettagli di tutti gli eventi relativi al parco che si sono verificati durante l'intervallo di tempo specificato.

```
aws gamelift describe-fleet-events \
    --fleet-id arn:aws:gamelift:us-west-2::fleet/fleet-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 \
    --start-time 1579647600 \
    --end-time 1579649400 \
    --limit 5
```

Output:

```
{
    "Events": [
        {
            "EventId": "a37b6892-5d07-4d3b-8b47-80244ecf66b9",
            "ResourceId": "fleet-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
            "EventCode": "FLEET_STATE_ACTIVE",
            "Message": "Fleet fleet-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 changed
 state to ACTIVE",
            "EventTime": 1579649342.191
        },
        {
            "EventId": "67da4ec9-92a3-4d95-886a-5d6772c24063",
            "ResourceId": "fleet-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
            "EventCode": "FLEET_STATE_ACTIVATING",
            "Message": "Fleet fleet-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 changed
 state to ACTIVATING",
            "EventTime": 1579649321.427
        },
        {
            "EventId": "23813a46-a9e6-4a53-8847-f12e6a8381ac",
            "ResourceId": "fleet-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
            "EventCode": "FLEET_STATE_BUILDING",
```

```
            "Message": "Fleet fleet-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 changed
 state to BUILDING",
            "EventTime": 1579649321.243
        },
        {
            "EventId": "3bf217d0-1d44-42f9-9202-433ed475d2e8",
            "ResourceId": "fleet-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
            "EventCode": "FLEET_STATE_VALIDATING",
            "Message": "Fleet fleet-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 changed
 state to VALIDATING",
            "EventTime": 1579649197.449
        },
        {
            "EventId": "2ecd0130-5986-44eb-99a7-62df27741084",
            "ResourceId": "fleet-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
            "EventCode": "FLEET_VALIDATION_LAUNCH_PATH_NOT_FOUND",
            "Message": "Failed to find a valid path",
            "EventTime": 1569319075.839,
            "PreSignedLogUrl": "https://gamelift-event-logs-prod-
us-west-2.s3.us-west-2.amazonaws.com/logs/fleet-83422059-8329-42a2-
a4d6-c4444386a6f8/events/2ecd0130-5986-44eb-99a7-62df27741084/
FLEET_VALIDATION_LAUNCH_PATH_NOT_FOUND.txt?X-Amz-Security-
Token=IQoJb3JpZ2luX2VjEB8aCXVzLXdlc3QtMiJHMEUCIHV5K%2FLPx8h310D
%2FAvx0%2FZxsDy5XA3cJOwPdu3T0eBa%2FAiEA1yovokcZYy%2FV4CWW6l26aFyiSHO
%2Bxz%2FBMAhEHYHMQNcqkQMImP%2F%2F%2F%2F%2F%2F%2F%2F%2F%2F
%2FARAAGgw3NDEwNjE1OTIxNzEiDI8rsZtzLzlwEDQhXSrlAtl5Ae
%2Fgo6FCIzqXPbXfBOnSvFYqeDlriZarEpKqKrUt8mXQv9iqHResqCph9AKo49lwgSYTT2QoSxnrD7%2FUgv
%2BZm2pVuczvuKtUA0fcx6s0GxpjIAzdIE%2F5P%2FB7B9M%2BVZ
%2F9KF82hbJi0HTE6Y7BjKsEgFCvk4UXILhfjtan9iQl8%2F21ZTurAcJbm7Y5tuLF9SWSK3%2BEa7VXOcCK4D4O1sMj
%2FIaXoHkNvg0RVTa0hIqdvpaDQlsSBNdqTXbjHTu6fETE9Y9Ky%2BiJK5KiUG
%2F59GjCpDcvS1FqKeLUEmKT7wysGmvjMc2n%2Fr
%2F9VxQfte7w9srXwlLAQuwhiXAAyI5ICMZ5JvzjzQwTqD4CHTVKUUDwL
%2BRZzbuuqkJObZml02CkRGp%2B74RTAzLbWptVqZTIfzctiCTmWxb
%2FmKyELRYsVLrwNJ%2BGJ7%2BCrN0RC%2FjlgfLYIZyeAqjPgAu5HjgX
%2BM7jCo9M7wBTrnAXKOFQuf9dvA84SuwXOJFp17LYGjrHMKv0qC3GfbTMrZ6kzeNV9awKCpXB2Gnx9z2KvIlJdqirWV
%2F9C6%2B4jIZPME3jXmZcEHqqw5uvAVF7aeIavtUZU8pxpDIWT0YE4p3Kriy2AA7ziCRKtVfjV839InyLk8LUjsioWK
%2BYUq8%2FDTlLxqj1S%2FiO4TI0Wo7ilAo%2FKKWWF4guuNDexj8EOOynSp1yImB
%2BZf2Fua3O44W4eEXAMPLE33333&X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-
Date=20170621T231808Z&X-Amz-SignedHeaders=host&X-Amz-Expires=900&X-Amz-
Credential=AKIAIOSFODNN7EXAMPLE%2F20170621%2Fus-west-2%2Fs3%2Faws4_request&X-Amz-
Signature=wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY"
        }
    ],
```

```
    "NextToken":
 "eyJhd3NBY2NvdW50SWQiOnsicyI6IjMwMjc3NjAxNjM5OCJ9LCJidWlsZElkIjp7InMiOiJidWlsZC01NWYxZTZmMS
}
```

Per ulteriori informazioni, consulta Debug GameLift Fleet Issues nella Amazon GameLift Developer Guide.

- Per i dettagli sull'API, consulta DescribeFleetEvents AWS CLICommand Reference.

`describe-fleet-port-settings`

Il seguente esempio di codice mostra come utilizzare`describe-fleet-port-settings`.

AWS CLI

Come visualizzare le autorizzazione di connessione in entrata per un parco istanze

L'esempio `describe-fleet-port-settings` seguente recupera le impostazioni di connessione per un parco specificato.

```
aws gamelift describe-fleet-port-settings \
    --fleet-id arn:aws:gamelift:us-west-2::fleet/fleet-a1b2c3d4-5678-90ab-cdef-
EXAMPLE11111
```

Output:

```
{
    "InboundPermissions": [
        {
            "FromPort": 33400,
            "ToPort": 33500,
            "IpRange": "0.0.0.0/0",
            "Protocol": "UDP"
        },
        {
            "FromPort": 1900,
            "ToPort": 2000,
            "IpRange": "0.0.0.0/0",
            "Protocol": "TCP"
        }
    ]
}
```

Per ulteriori informazioni, consulta Configurazione delle GameLift flotte nella Amazon GameLift Developer Guide.

- Per i dettagli sull'API, consulta DescribeFleetPortSettings AWS CLICommand Reference.

`describe-fleet-utilization`

Il seguente esempio di codice mostra come utilizzare`describe-fleet-utilization`.

AWS CLI

Esempio 1: come visualizzare i dati di utilizzo per un elenco di parchi istanze

L'esempio `describe-fleet-utilization` seguente recupera le informazioni sull'utilizzo corrente per un parco specificato.

```
aws gamelift describe-fleet-utilization \
    --fleet-ids arn:aws:gamelift:us-west-2::fleet/fleet-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Output:

```
{
    "FleetUtilization": [
        {
        "FleetId": "fleet-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
        "ActiveServerProcessCount": 100,
        "ActiveGameSessionCount": 62,
        "CurrentPlayerSessionCount": 329,
        "MaximumPlayerSessionCount": 1000
        }
    ]
}
```

Esempio 2: come richiedere i dati di utilizzo per tutti i parchi istanze

L'esempio `describe-fleet-utilization` seguente restituisce i dati di utilizzo per tutti i parchi con qualsiasi stato. Questo esempio utilizza i parametri di paginazione per restituire i dati per due parchi per volta.

```
aws gamelift describe-fleet-utilization \
    --limit 2
```

Output:

```
{
    "FleetUtilization": [
        {
            "FleetId": "fleet-1111aaaa-22bb-33cc-44dd-5555eeee66ff",
            "ActiveServerProcessCount": 100,
            "ActiveGameSessionCount": 13,
            "CurrentPlayerSessionCount": 98,
            "MaximumPlayerSessionCount": 1000
        },
        {
            "FleetId": "fleet-2222bbbb-33cc-44dd-55ee-6666ffff77aa",
            "ActiveServerProcessCount": 100,
            "ActiveGameSessionCount": 62,
            "CurrentPlayerSessionCount": 329,
            "MaximumPlayerSessionCount": 1000
        }
    ],
    "NextToken":
 "eyJhd3NBY2NvdW50SWQiOnsicyI6IjMwMjc3NjAxNjM5OCJ9LCJidWlsZElkIjp7InMiOiJidWlsZC01NWYxZTZmMS
}
```

Chiama il comando una seconda volta, passando il valore `NextToken` come argomento al parametro `--next-token` per visualizzare i due risultati successivi.

```
aws gamelift describe-fleet-utilization \
    --limit 2 \
    --next-
token eyJhd3NBY2NvdW50SWQiOnsicyI6IjMwMjc3NjAxNjM5OCJ9LCJidWlsZElkIjp7InMiOiJidWlsZC01NWYxZT
```

Ripeti il comando finché la risposta non include più un valore `NextToken` nell'output.

Per ulteriori informazioni, consulta GameLift Metrics for Fleets nella Amazon GameLift Developer Guide.

- Per i dettagli sull'API, consulta AWS CLI Command DescribeFleetUtilizationReference.

`describe-game-session-queues`

Il seguente esempio di codice mostra come utilizzare`describe-game-session-queues`.

AWS CLI

Come visualizzare code di sessioni di gioco

L'esempio `describe-game-session-queues` seguente recupera le proprietà per due code specificate.

```
aws gamelift describe-game-session-queues \
    --names MegaFrogRace-NA MegaFrogRace-EU
```

Output:

```
{
    "GameSessionQueues": [{
            "Destinations": [{
                    "DestinationArn": "arn:aws:gamelift:us-west-2::fleet/fleet-
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
                },
                {
                    "DestinationArn": "arn:aws:gamelift:us-west-2::fleet/fleet-
a1b2c3d4-5678-90ab-cdef-EXAMPLE22222"
                }
            ],
            "Name": "MegaFrogRace-NA",
            "TimeoutInSeconds": 600,
            "GameSessionQueueArn": "arn:aws:gamelift:us-west-2::gamesessionqueue/
MegaFrogRace-NA",
            "PlayerLatencyPolicies": [{
                    "MaximumIndividualPlayerLatencyMilliseconds": 200
                },
                {
                    "MaximumIndividualPlayerLatencyMilliseconds": 100,
                    "PolicyDurationSeconds": 60
                }
            ],
            "FilterConfiguration": {
                "AllowedLocations": ["us-west-2", "ap-south-1", "us-east-1"]
            },
            "PriorityConfiguration": {
                "PriorityOrder": ["LOCATION", "FLEET_TYPE", "DESTINATION"],
                "LocationOrder": ["us-west-2", "ap-south-1", "us-east-1"]
            }
        },
```

```
        {
            "Destinations": [{
                "DestinationArn": "arn:aws:gamelift:eu-west-3::fleet/fleet-
a1b2c3d4-5678-90ab-cdef-EXAMPLE22222"
            }],
            "Name": "MegaFrogRace-EU",
            "TimeoutInSeconds": 600,
            "GameSessionQueueArn": "arn:aws:gamelift:us-west-2::gamesessionqueue/
MegaFrogRace-EU"
        }
    ]
}
```

Per ulteriori informazioni, consulta Using Multi-Region Queues nella Amazon GameLift Developer Guide.

- Per i dettagli sull'API, consulta AWS CLI Command DescribeGameSessionQueuesReference.

`describe-runtime-configuration`

Il seguente esempio di codice mostra come utilizzare`describe-runtime-configuration`.

AWS CLI

Come richiedere la configurazione del runtime per un parco istanze

L'esempio `describe-runtime-configuration` seguente recupera i dettagli sulla configurazione del runtime corrente per un parco specificato.

```
aws gamelift describe-runtime-configuration \
    --fleet-id fleet-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Output:

```
{
    "RuntimeConfiguration": {
        "ServerProcesses": [
            {
                "LaunchPath": "C:\game\Bin64.Release.Dedicated
\MegaFrogRace_Server.exe",
                "Parameters": "+gamelift_start_server",
                "ConcurrentExecutions": 3
```

```
            },
            {
                "LaunchPath": "C:\game\Bin64.Release.Dedicated
\MegaFrogRace_Server.exe",
                "Parameters": "+gamelift_start_server +debug",
                "ConcurrentExecutions": 1
            }
        ],
        "MaxConcurrentGameSessionActivations": 2147483647,
        "GameSessionActivationTimeoutSeconds": 300
    }
}
```

Per ulteriori informazioni, consulta Esegui più processi su una flotta nella Amazon GameLift Developer Guide.

- Per i dettagli sull'API, consulta DescribeRuntimeConfiguration AWS CLICommand Reference.

`list-builds`

Il seguente esempio di codice mostra come utilizzare`list-builds`.

AWS CLI

Esempio 1: come ottenere un elenco di build personalizzate del gioco

L'esempio `list-builds` seguente recupera le proprietà per tutte le build del server di gioco nella Regione corrente. La richiesta di esempio mostra come utilizzare i parametri di paginazione `Limit` e `NextToken` per recuperare i risultati sotto forma di serie sequenziali. Il primo comando recupera le prime due build. Poiché le build sono più di due, la risposta include un valore `NextToken` per indicare che sono disponibili più risultati.

```
aws gamelift list-builds \
    --limit 2
```

Output:

```
{
    "Builds": [
        {
            "BuildArn": "arn:aws:gamelift:us-west-2::build/build-a1b2c3d4-5678-90ab-
cdef-EXAMPLE11111",
```

```
            "BuildId": "build-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
            "CreationTime": 1495664528.723,
            "Name": "My_Game_Server_Build_One",
            "OperatingSystem": "WINDOWS_2012",
            "SizeOnDisk": 8567781,
            "Status": "READY",
            "Version": "12345.678"
        },
        {
            "BuildArn": "arn:aws:gamelift:us-west-2::build/build-a1b2c3d4-5678-90ab-
cdef-EXAMPLE22222",
            "BuildId": "build-a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
            "CreationTime": 1495528748.555,
            "Name": "My_Game_Server_Build_Two",
            "OperatingSystem": "AMAZON_LINUX_2",
            "SizeOnDisk": 8567781,
            "Status": "FAILED",
            "Version": "23456.789"
        }
    ],
    "NextToken":
 "eyJhd3NBY2NvdW50SWQiOnsicyI6IjMwMjc3NjAxNjM5OCJ9LCJidWlsZElkIjp7InMiOiJidWlsZC01NWYxZTZmMS
}
```


Puoi quindi chiamare di nuovo il comando con il parametro `--next-token` nel modo seguente per visualizzare le due build successive.

```
aws gamelift list-builds \
    --limit 2
    --next-
token eyJhd3NBY2NvdW50SWQiOnsicyI6IjMwMjc3NjAxNjM5OCJ9LCJidWlsZElkIjp7InMiOiJidWlsZC01NWYxZT
```

Ripeti il comando finché la risposta non include un valore `NextToken`.

Esempio 2: come ottenere un elenco di build personalizzate del gioco con stato di errore

L'esempio `list-builds` seguente recupera le proprietà per tutte le build del server di gioco con stato FAILED nella Regione corrente.

```
aws gamelift list-builds \
    --status FAILED
```

Output:

```
{
    "Builds": [
        {
            "BuildArn": "arn:aws:gamelift:us-west-2::build/build-a1b2c3d4-5678-90ab-
cdef-EXAMPLE22222",
            "BuildId": "build-a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
            "CreationTime": 1495528748.555,
            "Name": "My_Game_Server_Build_Two",
            "OperatingSystem": "AMAZON_LINUX_2",
            "SizeOnDisk": 8567781,
            "Status": "FAILED",
            "Version": "23456.789"
        }
    ]
}
```

- Per i dettagli sull'API, consulta ListBuilds AWS CLICommand Reference.

`list-fleets`

Il seguente esempio di codice mostra come utilizzare`list-fleets`.

AWS CLI

Esempio 1: come ottenere un elenco di tutti i parchi istanze in una Regione

L'`list-fleets`esempio seguente mostra la flotta IDs di tutte le flotte della regione corrente. Questo esempio utilizza i parametri di impaginazione per recuperare due flotte IDs alla volta. La risposta include un attributo `next-token`, che indica la presenza di altri risultati da recuperare.

```
aws gamelift list-fleets \
    --limit 2
```

Output:

```
{
    "FleetIds": [
        "fleet-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
        "fleet-a1b2c3d4-5678-90ab-cdef-EXAMPLE22222"
```

```
    ],
    "NextToken":
 "eyJhd3NBY2NvdW50SWQiOnsicyI6IjMwMjc3NjAxNjM5OCJ9LCJidWlsZElkIjp7InMiOiJidWlsZC01NWYxZTZmMS
}
```

Puoi passare il valore `NextToken` della risposta precedente nel comando successivo, come mostrato qui, per ottenere i due risultati successivi.

```
aws gamelift list-fleets \
    --limit 2 \
    --next-
token eyJhd3NBY2NvdW50SWQiOnsicyI6IjMwMjc3NjAxNjM5OCJ9LCJidWlsZElkIjp7InMiOiJidWlsZC00NDRlZj
```

Esempio 2: come ottenere un elenco di tutti i parchi istanze in una Regione con una build o uno script specifico

L'`list-builds`esempio seguente recupera le IDs flotte dispiegate con la build di gioco specificata. Se utilizzi server in tempo reale, puoi fornire un ID script al posto di un ID build. Poiché questo esempio non specifica il parametro limit, i risultati possono includere fino a 16 flotte. IDs

```
aws gamelift list-fleets \
    --build-id build-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Output:

```
{
    "FleetIds": [
        "fleet-a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
        "fleet-a1b2c3d4-5678-90ab-cdef-EXAMPLE33333",
        "fleet-a1b2c3d4-5678-90ab-cdef-EXAMPLE44444"
    ]
}
```

- Per i dettagli sull'API, consulta ListFleets AWS CLICommand Reference.

`request-upload-credentials`

Il seguente esempio di codice mostra come utilizzare`request-upload-credentials`.

AWS CLI

Come aggiornare le credenziali di accesso per caricare una build

L'`create-build`esempio seguente ottiene nuove credenziali di accesso valide per caricare un file di GameLift build in una posizione Amazon S3. Le credenziali hanno una durata limitata. Puoi ottenere l'ID build dalla risposta alla richiesta `CreateBuild` originale.

```
aws gamelift request-upload-credentials \
    --build-id build-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Output:

```
{
    "StorageLocation": {
        "Bucket": "gamelift-builds-us-west-2",
        "Key": "123456789012/build-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
    },
    "UploadCredentials": {
        "AccessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "SecretAccessKey": "wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY",
        "SessionToken": "AgoGb3JpZ2luENz...EXAMPLETOKEN=="
    }
}
```

Per ulteriori informazioni, consulta Caricare una build di server personalizzata GameLift nella Amazon GameLift Developer Guide.

- Per i dettagli sull'API, consulta RequestUploadCredentials AWS CLICommand Reference.

`start-fleet-actions`

Il seguente esempio di codice mostra come utilizzare`start-fleet-actions`.

AWS CLI

Come riavviare l'attività di dimensionamento automatico di un parco istanze

L'`start-fleet-actions`esempio seguente riprende l'uso di tutte le politiche di scalabilità definite per il parco veicoli specificato ma che sono state interrotte chiamando`` ``. stop-fleet-

actions Dopo l'avvio, le policy di dimensionamento iniziano immediatamente a monitorare le rispettive metriche.

```
aws gamelift start-fleet-actions \
    --fleet-id fleet-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 \
    --actions AUTO_SCALING
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta Command Reference. StartFleetActionsAWS CLI

`stop-fleet-actions`

Il seguente esempio di codice mostra come utilizzare`stop-fleet-actions`.

AWS CLI

Come arrestare l'attività di dimensionamento automatico di un parco istanze

L'esempio `stop-fleet-actions` seguente arresta l'utilizzo di tutte le policy di dimensionamento definite per il parco specificato. Una volta sospese le policy, la capacità del parco mantiene lo stesso numero di istanze attive, se questo non viene modificato manualmente.

```
aws gamelift start-fleet-actions \
    --fleet-id fleet-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 \
    --actions AUTO_SCALING
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta StopFleetActions AWS CLICommand Reference.

`update-build`

Il seguente esempio di codice mostra come utilizzare`update-build`.

AWS CLI

Come aggiornare una build personalizzata del gioco

L'esempio `update-build` seguente modifica le informazioni sul nome e sulla versione associate a una risorsa build specificata. L'oggetto build restituito verifica che le modifiche siano state apportate correttamente.

```
aws gamelift update-build \
    --build-id build-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 \
    --name MegaFrogRaceServer.NA.east \
    --build-version 12345.east
```

Output:

```
{
    "Build": {
        "BuildArn": "arn:aws:gamelift:us-west-2::build/build-a1b2c3d4-5678-90ab-
cdef-EXAMPLE11111",
        "BuildId": "build-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
        "CreationTime": 1496708916.18,
        "Name": "MegaFrogRaceServer.NA.east",
        "OperatingSystem": "AMAZON_LINUX_2",
        "SizeOnDisk": 1304924,
        "Status": "READY",
        "Version": "12345.east"
    }
}
```

Per ulteriori informazioni, consulta Update Your Build Files nella Amazon GameLift Developer Guide.

- Per i dettagli sull'API, consulta UpdateBuild AWS CLICommand Reference.

`update-game-session-queue`

Il seguente esempio di codice mostra come utilizzare`update-game-session-queue`.

AWS CLI

Come aggiornare la configurazione di una coda di sessioni di gioco

L'esempio `update-game-session-queue` seguente aggiunge una nuova destinazione e aggiorna le policy di latenza dei giocatori per una coda di sessioni di gioco esistente.

```
aws gamelift update-game-session-queue \
    --name MegaFrogRace-NA \
    --destinations file://destinations.json \
    --player-latency-policies file://latency-policies.json
```

Contenuto di `destinations.json`:

```
{
    "Destinations": [
        {"DestinationArn": "arn:aws:gamelift:us-west-2::fleet/
fleet-1a2b3c4d-5e6f-7a8b-9c0d-1e2f3a4b5c6d"},
        {"DestinationArn": "arn:aws:gamelift:us-east-1::fleet/
fleet-5c6d3c4d-5e6f-7a8b-9c0d-1e2f3a4b5a2b"},
        {"DestinationArn": "arn:aws:gamelift:us-east-1::alias/
alias-11aa22bb-3c4d-5e6f-000a-1111aaaa22bb"}
    ]
}
```

Contenuto di `latency-policies.json`:

```
{
    "PlayerLatencyPolicies": [
        {"MaximumIndividualPlayerLatencyMilliseconds": 200},
        {"MaximumIndividualPlayerLatencyMilliseconds": 150, "PolicyDurationSeconds":
 120},
        {"MaximumIndividualPlayerLatencyMilliseconds": 100, "PolicyDurationSeconds":
 120}
    ]
}
```

Output:

```
{
    "GameSessionQueue": {
        "Destinations": [
            {"DestinationArn": "arn:aws:gamelift:us-west-2::fleet/
fleet-1a2b3c4d-5e6f-7a8b-9c0d-1e2f3a4b5c6d"},
            {"DestinationArn": "arn:aws:gamelift:us-east-1::fleet/
fleet-5c6d3c4d-5e6f-7a8b-9c0d-1e2f3a4b5a2b"},
            {"DestinationArn": "arn:aws:gamelift:us-east-1::alias/
alias-11aa22bb-3c4d-5e6f-000a-1111aaaa22bb"}
        ],
        "GameSessionQueueArn": "arn:aws:gamelift:us-
west-2:111122223333:gamesessionqueue/MegaFrogRace-NA",
        "Name": "MegaFrogRace-NA",
        "TimeoutInSeconds": 600,
        "PlayerLatencyPolicies": [
```

```
            {"MaximumIndividualPlayerLatencyMilliseconds": 200},
            {"MaximumIndividualPlayerLatencyMilliseconds": 150,
 "PolicyDurationSeconds": 120},
            {"MaximumIndividualPlayerLatencyMilliseconds": 100,
 "PolicyDurationSeconds": 120}
        ]
    }
}
```

Per ulteriori informazioni, consulta Using Multi-Region Queues nella Amazon GameLift Developer Guide.

- Per i dettagli sull'API, consulta AWS CLI Command UpdateGameSessionQueueReference.

`upload-build`

Il seguente esempio di codice mostra come utilizzare`upload-build`.

AWS CLI

Esempio 1: come caricare una build del server di gioco Linux

L'`upload-build`esempio seguente carica i file di build del server di gioco Linux da una directory di file al GameLift servizio e crea una risorsa di compilazione.

```
aws gamelift upload-build \
    --name MegaFrogRaceServer.NA \
    --build-version 2.0.1 \
    --build-root ~/MegaFrogRace_Server/release-na \
    --operating-system AMAZON_LINUX_2
    --server-sdk-version 4.0.2
```

Output:

```
Uploading ~/MegaFrogRace_Server/release-na:  16.0 KiB / 74.6 KiB (21.45%)
Uploading ~/MegaFrogRace_Server/release-na:  32.0 KiB / 74.6 KiB (42.89%)
Uploading ~/MegaFrogRace_Server/release-na:  48.0 KiB / 74.6 KiB (64.34%)
Uploading ~/MegaFrogRace_Server/release-na:  64.0 KiB / 74.6 KiB (85.79%)
Uploading ~/MegaFrogRace_Server/release-na:  74.6 KiB / 74.6 KiB (100.00%)
Successfully uploaded ~/MegaFrogRace_Server/release-na to AWS GameLift
Build ID: build-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Esempio 2: come caricare una build del server di gioco Windows

L'`upload-build`esempio seguente carica i file di build del server di gioco Windows da una directory al GameLift servizio e crea un record di build.

```
aws gamelift upload-build \
    --name MegaFrogRaceServer.NA \
    --build-version 2.0.1 \
    --build-root C:\MegaFrogRace_Server\release-na \
    --operating-system WINDOWS_2012
    --server-sdk-version 4.0.2
```

Output:

```
Uploading C:\MegaFrogRace_Server\release-na:  16.0 KiB / 74.6 KiB (21.45%)
Uploading C:\MegaFrogRace_Server\release-na:  32.0 KiB / 74.6 KiB (42.89%)
Uploading C:\MegaFrogRace_Server\release-na:  48.0 KiB / 74.6 KiB (64.34%)
Uploading C:\MegaFrogRace_Server\release-na:  64.0 KiB / 74.6 KiB (85.79%)
Uploading C:\MegaFrogRace_Server\release-na:  74.6 KiB / 74.6 KiB (100.00%)
Successfully uploaded C:\MegaFrogRace_Server\release-na to AWS GameLift
Build ID: build-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Per ulteriori informazioni, consulta Caricare una build di server personalizzata GameLift nella Amazon GameLift Developer Guide.

- Per i dettagli sull'API, consulta UploadBuild AWS CLICommand Reference.

Esempi di utilizzo di Amazon Glacier AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando Amazon Glacier. AWS Command Line Interface

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`abort-multipart-upload`

Il seguente esempio di codice mostra come usare. `abort-multipart-upload`

AWS CLI

Il comando seguente elimina un caricamento in più parti in corso in una vault denominata `my-vault`:

```
aws glacier abort-multipart-upload --account-id - --vault-name my-vault
 --upload-id 19gaRezEXAMPLES6Ry5YYdqthHOC_kGRCT03L9yetr220UmPtBYKk-
OssZtLqyFu7sY1_lR7vgFuJV6NtcV5zpsJ
```

Il comando non produce output. Amazon Glacier richiede un argomento ID account durante l'esecuzione delle operazioni, ma puoi utilizzare un trattino per specificare l'account in uso. L'ID di caricamento viene restituito dal comando `aws glacier initiate-multipart-upload` e può essere ottenuto anche utilizzando `aws glacier list-multipart-uploads`.

Per ulteriori informazioni sui caricamenti multiparte su Amazon Glacier tramite l'interfaccia a riga di comando, consulta Using Amazon Glacier AWS nella CLI User Guide.AWS

- Per i dettagli sull'API, consulta Command Reference. AbortMultipartUploadAWS CLI

`abort-vault-lock`

Il seguente esempio di codice mostra come utilizzare`abort-vault-lock`.

AWS CLI

Come interrompere un processo di blocco vault in corso

L'esempio `abort-vault-lock` seguente elimina una policy di Vault Lock dalla vault specificata e reimposta lo stato del blocco della vault su sbloccato.

```
aws glacier abort-vault-lock \
    --account-id - \
```

```
    --vault-name MyVaultName
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Interrompere Vault Lock (policy di blocco DELETE) nella Guida per gli sviluppatori dell'API Amazon Glacier.

- Per i dettagli sull'API, consulta AbortVaultLock AWS CLICommand Reference.

`add-tags-to-vault`

Il seguente esempio di codice mostra come utilizzare`add-tags-to-vault`.

AWS CLI

Il comando seguente aggiunge due tag a un vault denominato `my-vault`:

```
aws glacier add-tags-to-vault --account-id - --vault-name my-vault --
tags id=1234,date=july2015
```

Amazon Glacier richiede un argomento ID account durante l'esecuzione delle operazioni, ma puoi utilizzare un trattino per specificare l'account in uso.

- Per i dettagli sull'API, consulta AddTagsToVault AWS CLICommand Reference.

`complete-multipart-upload`

Il seguente esempio di codice mostra come utilizzare`complete-multipart-upload`.

AWS CLI

Il comando seguente completa il caricamento in più parti per un archivio da 3 MiB:

```
aws glacier complete-multipart-upload --archive-size 3145728 --
checksum 9628195fcdbcbbe76cdde456d4646fa7de5f219fb39823836d81f0cc0e18aa67
 --upload-id 19gaRezEXAMPLES6Ry5YYdqthHOC_kGRCT03L9yetr220UmPtBYKk-
OssZtLqyFu7sY1_lR7vgFuJV6NtcV5zpsJ --account-id - --vault-name my-vault
```

Amazon Glacier richiede un argomento ID account durante l'esecuzione delle operazioni, ma puoi utilizzare un trattino per specificare l'account in uso.

L'ID di caricamento viene restituito dal comando `aws glacier initiate-multipart-upload` e può essere ottenuto anche utilizzando `aws glacier list-multipart-uploads`. Il parametro checksum considera una struttura hash SHA-256 dell'archivio in formato esadecimale.

Per ulteriori informazioni sui caricamenti multiparte su Amazon Glacier tramite AWS l'interfaccia a riga di comando, incluse le istruzioni sul calcolo di un tree hash, consulta Using Amazon Glacier nella CLI User Guide.AWS

- Per i CompleteMultipartUploaddettagli AWS CLI sull'API, consulta Command Reference.

`complete-vault-lock`

Il seguente esempio di codice mostra come utilizzare`complete-vault-lock`.

AWS CLI

Come completare un processo di blocco vault in corso

L'esempio `complete-vault-lock` seguente completa l'esecuzione del blocco in corso per la vault specificata e imposta lo stato del blocco della vault su `Locked`. Il valore per il parametro `lock-id` si ottiene quando si esegue `initiate-lock-process`.

```
aws glacier complete-vault-lock \
    --account-id - \
    --vault-name MyVaultName \
    --lock-id 9QZgEXAMPLEPhvL6xEXAMPLE
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Completare Vault Lock (ID blocco POST) nella Guida per gli sviluppatori dell'API Amazon Glacier.

- Per i dettagli sull'API, consulta CompleteVaultLock AWS CLICommand Reference.

`create-vault`

Il seguente esempio di codice mostra come utilizzare`create-vault`.

AWS CLI

Il comando seguente crea un nuovo vault denominato `my-vault`:

```
aws glacier create-vault --vault-name my-vault --account-id -
```

Amazon Glacier richiede un argomento ID account durante l'esecuzione delle operazioni, ma puoi utilizzare un trattino per specificare l'account in uso.

- Per i dettagli sull'API, consulta CreateVault AWS CLICommand Reference.

`delete-archive`

Il seguente esempio di codice mostra come utilizzare`delete-archive`.

AWS CLI

Eliminazione di un archivio da un vault

L'esempio `delete-archive` seguente rimuove l'archivio specificato da `example_vault`.

```
aws glacier delete-archive \
    --account-id 111122223333  \
    --vault-name example_vault \
    --archive-id Sc0u9ZP8yaWkmh-XGlIvAVprtLhaLCGnNwNl5I5x9HqPIkX5mjc0DrId3Ln-
Gi_k2HzmlIDZUz117KSdVMdMXLuFWi9PJUitxWO73edQ43eTlMWkH0pd9zVSAuV_XXZBVhKhyGhJ7w
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta DeleteArchive AWS CLICommand Reference.

`delete-vault-access-policy`

Il seguente esempio di codice mostra come utilizzare`delete-vault-access-policy`.

AWS CLI

Come rimuovere la policy di accesso di una vault

L'esempio `delete-vault-access-policy` seguente rimuove la policy di accesso per la vault specificata.

```
aws glacier delete-vault-access-policy \
    --account-id 111122223333 \
    --vault-name example_vault
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta DeleteVaultAccessPolicy AWS CLICommand Reference.

`delete-vault-notifications`

Il seguente esempio di codice mostra come utilizzare`delete-vault-notifications`.

AWS CLI

Rimuovere le notifiche SNS per un vault

L'esempio `delete-vault-notifications` seguente rimuove le notifiche inviate da Amazon Simple Notification Service (Amazon SNS) per la vault specificata.

```
aws glacier delete-vault-notifications \
    --account-id 111122223333 \
    --vault-name example_vault
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta DeleteVaultNotifications AWS CLICommand Reference.

`delete-vault`

Il seguente esempio di codice mostra come utilizzare`delete-vault`.

AWS CLI

Il comando seguente elimina un vault denominato `my-vault`:

```
aws glacier delete-vault --vault-name my-vault --account-id -
```

Il comando non produce output. Amazon Glacier richiede un argomento ID account durante l'esecuzione delle operazioni, ma puoi utilizzare un trattino per specificare l'account in uso.

- Per i dettagli sull'API, consulta DeleteVault AWS CLICommand Reference.

`describe-job`

Il seguente esempio di codice mostra come utilizzare`describe-job`.

AWS CLI

Il comando seguente recupera le informazioni su un processo di recupero dell'inventario su un vault denominato `my-vault`:

```
aws glacier describe-job --account-id - --vault-name my-
vault --job-id zbxcm3Z_3z5UkoroF7SuZKrxgGoDc3RloGduS7Eg-
RO47Yc6FxsdGBgf_Q2DK5Ejh18CnTS5XW4_XqlNHS61dsO4CnMW
```

Output:

```
{
    "InventoryRetrievalParameters": {
        "Format": "JSON"
    },
    "VaultARN": "arn:aws:glacier:us-west-2:0123456789012:vaults/my-vault",
    "Completed": false,
    "JobId": "zbxcm3Z_3z5UkoroF7SuZKrxgGoDc3RloGduS7Eg-
RO47Yc6FxsdGBgf_Q2DK5Ejh18CnTS5XW4_XqlNHS61dsO4CnMW",
    "Action": "InventoryRetrieval",
    "CreationDate": "2015-07-17T20:23:41.616Z",
    "StatusCode": "InProgress"
}
```

L'ID del processo si trova nell'output di `aws glacier initiate-job` e `aws glacier list-jobs`. Amazon Glacier richiede un argomento ID account durante l'esecuzione delle operazioni, ma puoi utilizzare un trattino per specificare l'account in uso.

- Per i dettagli sull'API, consulta DescribeJob AWS CLICommand Reference.

`describe-vault`

Il seguente esempio di codice mostra come utilizzare`describe-vault`.

AWS CLI

Il comando seguente recupera i dati relativi a un vault denominato `my-vault`:

```
aws glacier describe-vault --vault-name my-vault --account-id -
```

Amazon Glacier richiede un argomento ID account durante l'esecuzione delle operazioni, ma puoi utilizzare un trattino per specificare l'account in uso.

- Per i dettagli sull'API, consulta DescribeVault AWS CLICommand Reference.

`get-data-retrieval-policy`

Il seguente esempio di codice mostra come utilizzare`get-data-retrieval-policy`.

AWS CLI

Il comando seguente ottiene la policy di recupero dei dati per l'account in uso:

```
aws glacier get-data-retrieval-policy --account-id -
```

Output:

```
{
    "Policy": {
        "Rules": [
            {
                "BytesPerHour": 10737418240,
                "Strategy": "BytesPerHour"
            }
        ]
    }
}
```

Amazon Glacier richiede un argomento ID account durante l'esecuzione delle operazioni, ma puoi utilizzare un trattino per specificare l'account in uso.

- Per i dettagli sull'API, consulta GetDataRetrievalPolicy AWS CLICommand Reference.

`get-job-output`

Il seguente esempio di codice mostra come utilizzare`get-job-output`.

AWS CLI

Il comando seguente salva l'output di un processo di inventario del vault in un file nella directory corrente denominata `output.json`:

```
aws glacier get-job-output --account-id - --vault-name my-
vault --job-id zbxcm3Z_3z5UkoroF7SuZKrxgGoDc3RloGduS7Eg-
RO47Yc6FxsdGBgf_Q2DK5Ejh18CnTS5XW4_XqlNHS61dsO4CnMW output.json
```

`job-id` è disponibile nell'output di `aws glacier list-jobs`. Nota che il nome del file di output è un argomento posizionale non preceduto dal nome di un'opzione. Amazon Glacier richiede un argomento ID account durante l'esecuzione delle operazioni, ma puoi utilizzare un trattino per specificare l'account in uso.

Output:

```
{
    "status": 200,
    "acceptRanges": "bytes",
    "contentType": "application/json"
}
```

`output.json`:

```
{"VaultARN":"arn:aws:glacier:us-west-2:0123456789012:vaults/
my-vault","InventoryDate":"2015-04-07T00:26:18Z","ArchiveList":
[{"ArchiveId":"kKB7ymWJVpPSwhGP6ycSOAekp9ZYe_--zM_mw6k76ZFGEIWQX-
ybtRDvc2VkPSDtfKmQrj0IRQLSGsNuDp-
AJVlu2ccmDSyDUmZwKbwbpAdGATGDiB3hHO0bjbGehXTcApVud_wyDw","ArchiveDescription":"multipart
 upload
 test","CreationDate":"2015-04-06T22:24:34Z","Size":3145728,"SHA256TreeHash":"9628195fcdbcbb
```

- Per i dettagli sull'API, consulta GetJobOutput AWS CLICommand Reference.

`get-vault-access-policy`

Il seguente esempio di codice mostra come utilizzare`get-vault-access-policy`.

AWS CLI

Come recuperare la policy di accesso di una vault

L'esempio `get-vault-access-policy` seguente recupera la policy di accesso per la vault specificata.

```
aws glacier get-vault-access-policy \
```

```
    --account-id 111122223333 \
    --vault-name example_vault
```

Output:

```
{
    "policy": {
        "Policy": "{\"Version\":\"2012-10-17\",\"Statement\":[{\"Effect
\":\"Allow\",\"Principal\":{\"AWS\":\"arn:aws:iam::444455556666:root
\"},\"Action\":\"glacier:ListJobs\",\"Resource\":\"arn:aws:glacier:us-
east-1:111122223333:vaults/example_vault\"},{\"Effect\":\"Allow\",\"Principal\":
{\"AWS\":\"arn:aws:iam::444455556666:root\"},\"Action\":\"glacier:UploadArchive\",
\"Resource\":\"arn:aws:glacier:us-east-1:111122223333:vaults/example_vault\"}]}"
    }
}
```

- Per i dettagli sull'API, consulta GetVaultAccessPolicy AWS CLICommand Reference.

`get-vault-lock`

Il seguente esempio di codice mostra come utilizzare`get-vault-lock`.

AWS CLI

Come ottenere i dettagli di un blocco vault

L'esempio `get-vault-lock` seguente recupera i dettagli sul blocco della vault specificata.

```
aws glacier get-vault-lock \
    --account-id - \
    --vault-name MyVaultName
```

Output:

```
{
    "Policy": "{\"Version\":\"2012-10-17\",\"Statement\":[{\"Sid\":\"Define-vault-
lock\",\"Effect\":\"Deny\",\"Principal\":{\"AWS\":\"arn:aws:iam::999999999999:root
\"},\"Action\":\"glacier:DeleteArchive\",\"Resource\":\"arn:aws:glacier:us-
west-2:99999999999:vaults/MyVaultName\",\"Condition\":{\"NumericLessThanEquals\":
{\"glacier:ArchiveAgeinDays\":\"365\"}}}]}",
    "State": "Locked",
```

```
    "CreationDate": "2019-07-29T22:25:28.640Z"
}
```

Per ulteriori informazioni, consulta Ottenere Vault Lock (policy di blocco GET) nella Guida per gli sviluppatori dell'API Amazon Glacier.

- Per i dettagli sull'API, consulta GetVaultLock AWS CLICommand Reference.

`get-vault-notifications`

Il seguente esempio di codice mostra come utilizzare`get-vault-notifications`.

AWS CLI

Il comando seguente ottiene una descrizione della configurazione della notifica per un vault denominato `my-vault`:

```
aws glacier get-vault-notifications --account-id - --vault-name my-vault
```

Output:

```
{
    "vaultNotificationConfig": {
        "Events": [
            "InventoryRetrievalCompleted",
            "ArchiveRetrievalCompleted"
        ],
        "SNSTopic": "arn:aws:sns:us-west-2:0123456789012:my-vault"
    }
}
```

Se non è stata configurata alcuna notifica per il vault, viene restituito un errore. Amazon Glacier richiede un argomento ID account durante l'esecuzione delle operazioni, ma puoi utilizzare un trattino per specificare l'account in uso.

- Per i dettagli sull'API, consulta GetVaultNotifications AWS CLICommand Reference.

`initiate-job`

Il seguente esempio di codice mostra come utilizzare`initiate-job`.

AWS CLI

Il comando seguente avvia un processo per ottenere un inventario del vault `my-vault`.

```
aws glacier initiate-job --account-id - --vault-name my-vault --job-parameters
 '{"Type": "inventory-retrieval"}'
```

Output:

```
{
    "location": "/0123456789012/vaults/my-vault/jobs/
zbxcm3Z_3z5UkoroF7SuZKrxgGoDc3RloGduS7Eg-
RO47Yc6FxsdGBgf_Q2DK5Ejh18CnTS5XW4_XqlNHS61dsO4CnMW",
    "jobId": "zbxcm3Z_3z5UkoroF7SuZKrxgGoDc3RloGduS7Eg-
RO47Yc6FxsdGBgf_Q2DK5Ejh18CnTS5XW4_XqlNHS61dsO4CnMW"
}
```

Amazon Glacier richiede un argomento ID account durante l'esecuzione delle operazioni, ma puoi utilizzare un trattino per specificare l'account in uso.

Il comando seguente avvia un processo per recuperare un archivio dal vault `my-vault`.

```
aws glacier initiate-job --account-id - --vault-name my-vault --job-
parameters file://job-archive-retrieval.json
```

`job-archive-retrieval.json` è un file JSON nella cartella locale che specifica il tipo di processo, l'ID di archivio e alcuni parametri facoltativi:

```
{
  "Type": "archive-retrieval",
  "ArchiveId": "kKB7ymWJVpPSwhGP6ycSOAekp9ZYe_--zM_mw6k76ZFGEIWQX-
ybtRDvc2VkPSDtfKmQrj0IRQLSGsNuDp-
AJVlu2ccmDSyDUmZwKbwbpAdGATGDiB3hHO0bjbGehXTcApVud_wyDw",
  "Description": "Retrieve archive on 2015-07-17",
  "SNSTopic": "arn:aws:sns:us-west-2:0123456789012:my-topic"
}
```

IDs Gli archivi sono disponibili nell'output di `aws glacier upload-archive` e`aws glacier get-job-output`.

Output:

```
{
    "location": "/011685312445/vaults/mwunderl/jobs/l7IL5-
EkXyEY9Ws95fClzIbk2O5uLYaFdAYOi-
azsX_Z8V6NH4yERHzars8wTKYQMX6nBDI9cMNHzyZJO59-8N9aHWav",
    "jobId": "l7IL5-EkXy2O5uLYaFdAYOiEY9Ws95fClzIbk-
azsX_Z8V6NH4yERHzars8wTKYQMX6nBDI9cMNHzyZJO59-8N9aHWav"
}
```

Per informazioni sul formato dei parametri del processo, consulta la sezione relativa all'avvio del processo nella documentazione di riferimento dell'API Amazon Glacier.

- Per i dettagli sull'API, consulta InitiateJob AWS CLICommand Reference.

`initiate-multipart-upload`

Il seguente esempio di codice mostra come utilizzare`initiate-multipart-upload`.

AWS CLI

Il comando seguente avvia un caricamento in più parti in una vault denominata `my-vault` con una dimensione della parte di 1 MiB (1024 x 1024 byte) per file:

```
aws glacier initiate-multipart-upload --account-id - --part-size 1048576 --vault-
name my-vault --archive-description "multipart upload test"
```

Il parametro di descrizione dell'archivio è facoltativo. Amazon Glacier richiede un argomento ID account durante l'esecuzione delle operazioni, ma puoi utilizzare un trattino per specificare l'account in uso.

Questo comando restituisce un ID di caricamento in caso di esito positivo. Utilizza l'ID di caricamento quando carichi le singole parti del tuo archivio con `aws glacier upload-multipart-part`. Per ulteriori informazioni sui caricamenti multiparte su Amazon Glacier tramite l'interfaccia a riga di comando, consulta Using Amazon Glacier AWS nella CLI User Guide.AWS

- Per i dettagli sull'API, consulta Command Reference. InitiateMultipartUploadAWS CLI

`initiate-vault-lock`

Il seguente esempio di codice mostra come utilizzare`initiate-vault-lock`.

AWS CLI

Come avviare il processo di blocco della vault

L'esempio `initiate-vault-lock` seguente installa una policy di Vault Lock nella vault specificata e imposta lo stato del blocco della vault su `InProgress`. Devi completare il processo chiamando `complete-vault-lock` entro 24 ore per impostare lo stato del blocco della vault su `Locked`.

```
aws glacier initiate-vault-lock \
    --account-id - \
    --vault-name MyVaultName \
    --policy file://vault_lock_policy.json
```

Contenuto di `vault_lock_policy.json`:

```
{"Policy":"{\"Version\":\"2012-10-17\",\"Statement\":[{\"Sid\":\"Define-vault-
lock\",\"Effect\":\"Deny\",\"Principal\":{\"AWS\":\"arn:aws:iam::999999999999:root
\"},\"Action\":\"glacier:DeleteArchive\",\"Resource\":\"arn:aws:glacier:us-
west-2:999999999999:vaults/examplevault\",\"Condition\":{\"NumericLessThanEquals\":
{\"glacier:ArchiveAgeinDays\":\"365\"}}}]}"}
```

L'output è l'ID di Vault Lock che puoi utilizzare per completare il processo di blocco della vault.

```
{
    "lockId": "9QZgEXAMPLEPhvL6xEXAMPLE"
}
```

Per ulteriori informazioni, consulta Avviare Vault Lock (policy di blocco POST) nella Guida per gli sviluppatori dell'API Amazon Glacier.

- Per i dettagli sull'API, consulta InitiateVaultLock AWS CLICommand Reference.

`list-jobs`

Il seguente esempio di codice mostra come utilizzare`list-jobs`.

AWS CLI

Il comando seguente elenca i lavori in corso e quelli completati di recente per un vault denominato `my-vault`:

```
aws glacier list-jobs --account-id - --vault-name my-vault
```

Output:

```
{
    "JobList": [
        {
            "VaultARN": "arn:aws:glacier:us-west-2:0123456789012:vaults/my-vault",
            "RetrievalByteRange": "0-3145727",
            "SNSTopic": "arn:aws:sns:us-west-2:0123456789012:my-vault",
            "Completed": false,
            "SHA256TreeHash":
 "9628195fcdbcbbe76cdde932d4646fa7de5f219fb39823836d81f0cc0e18aa67",
            "JobId": "l7IL5-EkXyEY9Ws95fClzIbk2O5uLYaFdAYOi-
azsX_Z8V6NH4yERHzars8wTKYQMX6nBDI9cMNHzyZJO59-8N9aHWav",
            "ArchiveId": "kKB7ymWJVpPSwhGP6ycSOAekp9ZYe_--zM_mw6k76ZFGEIWQX-
ybtRDvc2VkPSDtfKmQrj0IRQLSGsNuDp-
AJVlu2ccmDSyDUmZwKbwbpAdGATGDiB3hHO0bjbGehXTcApVud_wyDw",
            "JobDescription": "Retrieve archive on 2015-07-17",
            "ArchiveSizeInBytes": 3145728,
            "Action": "ArchiveRetrieval",
            "ArchiveSHA256TreeHash":
 "9628195fcdbcbbe76cdde932d4646fa7de5f219fb39823836d81f0cc0e18aa67",
            "CreationDate": "2015-07-17T21:16:13.840Z",
            "StatusCode": "InProgress"
        },
        {
            "InventoryRetrievalParameters": {
                "Format": "JSON"
            },
            "VaultARN": "arn:aws:glacier:us-west-2:0123456789012:vaults/my-vault",
            "Completed": false,
            "JobId": "zbxcm3Z_3z5UkoroF7SuZKrxgGoDc3RloGduS7Eg-
RO47Yc6FxsdGBgf_Q2DK5Ejh18CnTS5XW4_XqlNHS61dsO4CnMW",
            "Action": "InventoryRetrieval",
            "CreationDate": "2015-07-17T20:23:41.616Z",
            "StatusCode": ""InProgress""
        }
    ]
}
```

Amazon Glacier richiede un argomento ID account durante l'esecuzione delle operazioni, ma puoi utilizzare un trattino per specificare l'account in uso.

- Per i dettagli sull'API, consulta ListJobs AWS CLICommand Reference.

`list-multipart-uploads`

Il seguente esempio di codice mostra come utilizzare`list-multipart-uploads`.

AWS CLI

Il comando seguente mostra tutti i caricamenti in più parti in corso per una vault denominata `my-vault`:

```
aws glacier list-multipart-uploads --account-id - --vault-name my-vault
```

Amazon Glacier richiede un argomento ID account durante l'esecuzione delle operazioni, ma puoi utilizzare un trattino per specificare l'account in uso.

Per ulteriori informazioni sui caricamenti multiparte su Amazon Glacier tramite l'interfaccia a riga di comando, consulta Using Amazon Glacier AWS nella CLI User Guide.AWS

- Per i dettagli sull'API, consulta Command Reference. ListMultipartUploadsAWS CLI

`list-parts`

Il seguente esempio di codice mostra come utilizzare`list-parts`.

AWS CLI

Il comando seguente elenca le parti caricate per un caricamento in più parti in una vault denominata `my-vault`:

```
aws glacier list-parts --account-id - --vault-name my-vault --upload-id "SYZi7qnL-YGqGwAm8Kn3BLP2ElNCvnB-5961R09CSaPmPwkYGHOqeN_nX3-Vhnd2yF0KfB5FkmbnBU9GubbdrCs8ut-D"
```

Output:

```
{
    "MultipartUploadId": "SYZi7qnL-
YGqGwAm8Kn3BLP2ElNCvnB-5961R09CSaPmPwkYGHOqeN_nX3-Vhnd2yF0KfB5FkmbnBU9GubbdrCs8ut-
D",
```

```
    "Parts": [
        {
            "RangeInBytes": "0-1048575",
            "SHA256TreeHash":
 "e1f2a7cd6e047350f69b9f8cfa60fa606fe2f02802097a9a026360a7edc1f553"
        },
        {
            "RangeInBytes": "1048576-2097151",
            "SHA256TreeHash":
 "43cf3061fb95796aed99a11a6aa3cd8f839eed15e655ab0a597126210636aee6"
        }
    ],
    "VaultARN": "arn:aws:glacier:us-west-2:0123456789012:vaults/my-vault",
    "CreationDate": "2015-07-18T00:05:23.830Z",
    "PartSizeInBytes": 1048576
}
```

Amazon Glacier richiede un argomento ID account durante l'esecuzione delle operazioni, ma puoi utilizzare un trattino per specificare l'account in uso.

Per ulteriori informazioni sui caricamenti multiparte su Amazon Glacier tramite l'interfaccia a riga di comando, consulta Using Amazon Glacier AWS nella CLI User Guide.AWS

- Per i dettagli sull'API, consulta Command Reference. ListPartsAWS CLI

`list-provisioned-capacity`

Il seguente esempio di codice mostra come utilizzare`list-provisioned-capacity`.

AWS CLI

Come recuperare le unità di capacità assegnate

L'esempio `list-provisioned-capacity` seguente recupera i dettagli per tutte le unità di capacità assegnate per l'account specificato.

```
aws glacier list-provisioned-capacity \
    --account-id 111122223333
```

Output:

```
{
```

```
    "ProvisionedCapacityList": [
        {
            "CapacityId": "HpASAuvfRFiVDbOjMfEIcr8K",
            "ExpirationDate": "2020-03-18T19:59:24.000Z",
            "StartDate": "2020-02-18T19:59:24.912Z"
        }
    ]
}
```

- Per i dettagli sull'API, consulta ListProvisionedCapacity AWS CLICommand Reference.

`list-tags-for-vault`

Il seguente esempio di codice mostra come utilizzare`list-tags-for-vault`.

AWS CLI

Il comando seguente consente di elencare i tag applicati a un vault denominato `my-vault`:

```
aws glacier list-tags-for-vault --account-id - --vault-name my-vault
```

Output:

```
{
    "Tags": {
        "date": "july2015",
        "id": "1234"
    }
}
```

Amazon Glacier richiede un argomento ID account durante l'esecuzione delle operazioni, ma puoi utilizzare un trattino per specificare l'account in uso.

- Per i dettagli sull'API, consulta ListTagsForVault AWS CLICommand Reference.

`list-vaults`

Il seguente esempio di codice mostra come utilizzare`list-vaults`.

AWS CLI

Il comando seguente consente di elencare le vault nell'account e nella Regione predefiniti:

```
aws glacier list-vaults --account-id -
```

Output:

```
{
    "VaultList": [
        {
            "SizeInBytes": 3178496,
            "VaultARN": "arn:aws:glacier:us-west-2:0123456789012:vaults/my-vault",
            "LastInventoryDate": "2015-04-07T00:26:19.028Z",
            "VaultName": "my-vault",
            "NumberOfArchives": 1,
            "CreationDate": "2015-04-06T21:23:45.708Z"
        }
    ]
}
```

Amazon Glacier richiede un argomento ID account durante l'esecuzione delle operazioni, ma puoi utilizzare un trattino per specificare l'account in uso.

- Per i dettagli sull'API, consulta ListVaults AWS CLICommand Reference.

`purchase-provisioned-capacity`

Il seguente esempio di codice mostra come utilizzare`purchase-provisioned-capacity`.

AWS CLI

Come acquistare un'unità di capacità assegnata

L'esempio `purchase-provisioned-capacity` seguente determina l'acquisto di un'unità di capacità assegnata.

```
aws glacier purchase-provisioned-capacity \
    --account-id 111122223333
```

Output:

```
{
    "capacityId": "HpASAuvfRFiVDbOjMfEIcr8K"
}
```

- Per i dettagli sull'API, consulta PurchaseProvisionedCapacity AWS CLICommand Reference.

`remove-tags-from-vault`

Il seguente esempio di codice mostra come utilizzare`remove-tags-from-vault`.

AWS CLI

Il comando seguente rimuove un tag con la chiave `date` da un archivio denominato `my-vault`:

```
aws glacier remove-tags-from-vault --account-id - --vault-name my-vault --tag-
keys date
```

Amazon Glacier richiede un argomento ID account durante l'esecuzione delle operazioni, ma puoi utilizzare un trattino per specificare l'account in uso.

- Per i dettagli sull'API, consulta RemoveTagsFromVault AWS CLICommand Reference.

`set-data-retrieval-policy`

Il seguente esempio di codice mostra come utilizzare`set-data-retrieval-policy`.

AWS CLI

Il comando seguente configura un policy di recupero dei dati per l'account in uso:

```
aws glacier set-data-retrieval-policy --account-id - --policy file://data-retrieval-
policy.json
```

`data-retrieval-policy.json` è un file JSON nella cartella corrente che specifica una policy di recupero dei dati:

```
{
  "Rules":[
     {
         "Strategy":"BytesPerHour",
         "BytesPerHour":10737418240
      }
   ]
}
```

Amazon Glacier richiede un argomento ID account durante l'esecuzione delle operazioni, ma puoi utilizzare un trattino per specificare l'account in uso.

Il comando seguente imposta la policy di recupero dei dati su `FreeTier` utilizzando JSON in linea:

```
aws glacier set-data-retrieval-policy --account-id - --policy '{"Rules":
[{"Strategy":"FreeTier"}]}'
```

Per dettagli sul formato della policy, consulta la sezione relativa all'impostazione della policy di recupero dei dati nella documentazione di riferimento dell'API Amazon Glacier.

- Per i dettagli sull'API, consulta SetDataRetrievalPolicy AWS CLICommand Reference.

`set-vault-access-policy`

Il seguente esempio di codice mostra come utilizzare`set-vault-access-policy`.

AWS CLI

Come impostare la policy di accesso di una vault

L'esempio `set-vault-access-policy` seguente collega una policy di autorizzazione alla vault specificata.

```
aws glacier set-vault-access-policy \
    --account-id 111122223333 \
    --vault-name example_vault
    --policy '{"Policy": "{\"Version\":\"2012-10-17\",\"Statement\":
[{\"Effect\":\"Allow\",\"Principal\":{\"AWS\":\"arn:aws:iam::444455556666:root
\"},\"Action\":\"glacier:ListJobs\",\"Resource\":\"arn:aws:glacier:us-
east-1:111122223333:vaults/example_vault\"},{\"Effect\":\"Allow\",\"Principal\":
{\"AWS\":\"arn:aws:iam::444455556666:root\"},\"Action\":\"glacier:UploadArchive\",
\"Resource\":\"arn:aws:glacier:us-east-1:111122223333:vaults/example_vault\"}]}"}'
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta SetVaultAccessPolicy AWS CLICommand Reference.

`set-vault-notifications`

Il seguente esempio di codice mostra come utilizzare`set-vault-notifications`.

AWS CLI

Il comando seguente configura le notifiche SNS per un vault denominato `my-vault`:

```
aws glacier set-vault-notifications --account-id - --vault-name my-vault --vault-notification-config file://notificationconfig.json
```

`notificationconfig.json` è un file JSON nella cartella corrente che specifica un argomento SNS e gli eventi da pubblicare:

```
{
  "SNSTopic": "arn:aws:sns:us-west-2:0123456789012:my-vault",
  "Events": ["ArchiveRetrievalCompleted", "InventoryRetrievalCompleted"]
}
```

Amazon Glacier richiede un argomento ID account durante l'esecuzione delle operazioni, ma puoi utilizzare un trattino per specificare l'account in uso.

- Per i dettagli sull'API, consulta SetVaultNotifications AWS CLICommand Reference.

`upload-archive`

Il seguente esempio di codice mostra come utilizzare`upload-archive`.

AWS CLI

Il comando seguente carica un archivio nella cartella corrente denominata `archive.zip` in un archivio denominato `my-vault`:

```
aws glacier upload-archive --account-id - --vault-name my-vault --body archive.zip
```

Output:

```
{
    "archiveId": "kKB7ymWJVpPSwhGP6ycSOAekp9ZYe_--zM_mw6k76ZFGEIWQX-
ybtRDvc2VkPSDtfKmQrj0IRQLSGsNuDp-
AJVlu2ccmDSyDUmZwKbwbpAdGATGDiB3hHO0bjbGehXTcApVud_wyDw",
    "checksum": "969fb39823836d81f0cc028195fcdbcbbe76cdde932d4646fa7de5f21e18aa67",
    "location": "/0123456789012/vaults/my-vault/archives/
kKB7ymWJVpPSwhGP6ycSOAekp9ZYe_--zM_mw6k76ZFGEIWQX-ybtRDvc2VkPSDtfKmQrj0IRQLSGsNuDp-
AJVlu2ccmDSyDUmZwKbwbpAdGATGDiB3hHO0bjbGehXTcApVud_wyDw"
```

```
}
```

Amazon Glacier richiede un argomento ID account durante l'esecuzione delle operazioni, ma puoi utilizzare un trattino per specificare l'account in uso.

Per recuperare un archivio caricato, avvia un processo di recupero con il comando initiate-job di aws glacier.

- Per i dettagli sull'API, consulta UploadArchive AWS CLICommand Reference.

`upload-multipart-part`

Il seguente esempio di codice mostra come utilizzare`upload-multipart-part`.

AWS CLI

Il comando seguente carica la prima parte da 1 MiB (1024 x 1024 byte) di un archivio:

```
aws glacier upload-multipart-part --body part1 --range 'bytes
 0-1048575/*' --account-id - --vault-name my-vault --upload-
id 19gaRezEXAMPLES6Ry5YYdqthHOC_kGRCT03L9yetr220UmPtBYKk-
OssZtLqyFu7sY1_lR7vgFuJV6NtcV5zpsJ
```

Amazon Glacier richiede un argomento ID account durante l'esecuzione delle operazioni, ma puoi utilizzare un trattino per specificare l'account in uso.

Il parametro body indica il percorso di un file delle parti sul filesystem locale. Il parametro range utilizza un intervallo di contenuti HTTP che indica i byte occupati dalla parte nell'archivio completato. L'ID di caricamento viene restituito dal comando `aws glacier initiate-multipart-upload` e può essere ottenuto anche utilizzando `aws glacier list-multipart-uploads`.

Per ulteriori informazioni sui caricamenti multiparte su Amazon Glacier tramite l'interfaccia a riga di comando, consulta Using Amazon Glacier AWS nella CLI User Guide.AWS

- Per i dettagli sull'API, consulta Command Reference. UploadMultipartPartAWS CLI

Esempi di Global Accelerator utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando Global Accelerator. AWS Command Line Interface

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`add-custom-routing-endpoints`

Il seguente esempio di codice mostra come utilizzare`add-custom-routing-endpoints`.

AWS CLI

Come aggiungere un endpoint di sottorete VPC a un gruppo di endpoint per un acceleratore di instradamento personalizzato

L'esempio `add-custom-routing-endpoints` seguente aggiunge un endpoint di sottorete VPC a un gruppo di endpoint per un acceleratore di instradamento personalizzato.

```
aws globalaccelerator add-custom-routing-endpoints \
    --endpoint-group-
arn arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-abcd-1234-
abcd-1234abcdefgh/listener/0123vxyz/endpoint-group/4321abcd \
    --endpoint-configurations "EndpointId=subnet-1234567890abcdef0"
```

Output:

```
{
    "EndpointDescriptions": [
        {
            "EndpointId": "subnet-1234567890abcdef0"
        }
    ],

 "EndpointGroupArn":"arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-
abcd-1234-abcd-1234abcdefgh/listener/0123vxyz/endpoint-group/4321abcd"
```

```
}
```

Per ulteriori informazioni, consulta gli endpoint di sottorete VPC per acceleratori di routing personalizzati in Global Accelerator nella AWS Global Accelerator Developer Guide.AWS

- Per i dettagli sull'API, consulta Command Reference. AddCustomRoutingEndpointsAWS CLI

`advertise-byoip-cidr`

Il seguente esempio di codice mostra come utilizzare`advertise-byoip-cidr`.

AWS CLI

Come pubblicizzare un intervallo di indirizzi

L'`advertise-byoip-cidr`esempio seguente richiede AWS di pubblicizzare un intervallo di indirizzi che hai predisposto per essere utilizzato con le tue AWS risorse.

```
aws globalaccelerator advertise-byoip-cidr \
    --cidr 198.51.100.0/24
```

Output:

```
{
    "ByoipCidr": {
        "Cidr": "198.51.100.0/24",
        "State": "PENDING_ADVERTISING"
    }
}
```

Per ulteriori informazioni, consulta Bring Your Own IP Address in AWS Global Accelerator nella Global Accelerator Developer AWS Guide.

- Per i dettagli sull'API, consulta AWS CLI Command AdvertiseByoipCidrReference.

`allow-custom-routing-traffic`

Il seguente esempio di codice mostra come utilizzare`allow-custom-routing-traffic`.

AWS CLI

Per consentire il traffico verso destinazioni specifiche di EC2 istanze Amazon in una sottorete VPC per un acceleratore di routing personalizzato

L'`allow-custom-routing-traffic`esempio seguente specifica che il traffico è consentito verso determinati indirizzi IP (destinazione) di EC2 istanze Amazon e che le porte di un endpoint di sottorete VPC in un acceleratore di routing personalizzato possono ricevere traffico.

```
aws globalaccelerator allow-custom-routing-traffic \
    --endpoint-group-
arn arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-abcd-1234-
abcd-1234abcdefgh/listener/0123vxyz/endpoint-group/ab88888example \
    --endpoint-id subnet-abcd123example \
    --destination-addresses "172.31.200.6" "172.31.200.7" \
    --destination-ports 80 81
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta gli endpoint di sottorete VPC per acceleratori di routing personalizzati in Global Accelerator nella AWS Global Accelerator Developer Guide.AWS

- Per i dettagli sull'API, consulta Command Reference. AllowCustomRoutingTrafficAWS CLI

`create-accelerator`

Il seguente esempio di codice mostra come utilizzare`create-accelerator`.

AWS CLI

Come creare un acceleratore

L'esempio `create-accelerator` seguente crea un acceleratore con due tag con due indirizzi IP statici BYOIP. È necessario specificare la Regione `US-West-2 (Oregon)` per creare o aggiornare un acceleratore.

```
aws globalaccelerator create-accelerator \
    --name ExampleAccelerator \
    --tags Key="Name",Value="Example Name" Key="Project",Value="Example Project" \
    --ip-addresses 192.0.2.250 198.51.100.52
```

Output:

```
{
    "Accelerator": {
        "AcceleratorArn":
 "arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-abcd-1234-
abcd-1234abcdefgh",
        "IpAddressType": "IPV4",
        "Name": "ExampleAccelerator",
        "Enabled": true,
        "Status": "IN_PROGRESS",
        "IpSets": [
            {
                "IpAddresses": [
                    "192.0.2.250",
                    "198.51.100.52"
                ],
                "IpFamily": "IPv4"
            }
        ],
        "DnsName":"a1234567890abcdef.awsglobalaccelerator.com",
        "CreatedTime": 1542394847.0,
        "LastModifiedTime": 1542394847.0
    }
}
```

Per ulteriori informazioni, consulta Accelerators in AWS Global Accelerator nella Global Accelerator AWS Developer Guide.

- Per i dettagli sull'API, consulta Command CreateAcceleratorReference AWS CLI .

`create-custom-routing-accelerator`

Il seguente esempio di codice mostra come utilizzare`create-custom-routing-accelerator`.

AWS CLI

Come creare un acceleratore di instradamento personalizzato

L'esempio `create-custom-routing-accelerator` seguente crea un acceleratore di instradamento personalizzato con i tag `Name` e `Project`.

```
aws globalaccelerator create-custom-routing-accelerator \
    --name ExampleCustomRoutingAccelerator \
    --tags Key="Name",Value="Example Name" Key="Project",Value="Example Project" \
```

```
    --ip-addresses 192.0.2.250 198.51.100.52
```

Output:

```
{
    "Accelerator": {
        "AcceleratorArn":
 "arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-abcd-1234-
abcd-1234abcdefgh",
        "IpAddressType": "IPV4",
        "Name": "ExampleCustomRoutingAccelerator",
        "Enabled": true,
        "Status": "IN_PROGRESS",
        "IpSets": [
            {
                "IpAddresses": [
                    "192.0.2.250",
                    "198.51.100.52"
                ],
                "IpFamily": "IPv4"
            }
        ],
        "DnsName":"a1234567890abcdef.awsglobalaccelerator.com",
        "CreatedTime": 1542394847.0,
        "LastModifiedTime": 1542394847.0
    }
}
```

Per ulteriori informazioni, consulta la sezione Acceleratori di routing personalizzati in Global Accelerator nella AWS Global Accelerator Developer AWS Guide.

- Per i dettagli sull'API, consulta CreateCustomRoutingAcceleratorCommand Reference.AWS CLI

`create-custom-routing-endpoint-group`

Il seguente esempio di codice mostra come utilizzare`create-custom-routing-endpoint-group`.

AWS CLI

Come creare un gruppo di endpoint per un acceleratore di instradamento personalizzato

L'esempio `create-custom-routing-endpoint-group` seguente crea un gruppo di endpoint per un acceleratore di instradamento personalizzato.

```
aws globalaccelerator create-custom-routing-endpoint-group \
    --listener-arn arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-abcd-1234-abcd-1234abcdefgh/listener/0123vxyz \
    --endpoint-group-region us-east-2 \
    --destination-configurations "FromPort=80,ToPort=81,Protocols=TCP,UDP"
```

Output:

```
{
    "EndpointGroup": {
        "EndpointGroupArn":
 "arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-abcd-1234-abcd-1234abcdefgh/listener/0123vxyz/endpoint-group/4321abcd",
        "EndpointGroupRegion": "us-east-2",
        "DestinationDescriptions": [
            {
                "FromPort": 80,
                "ToPort": 81,
                "Protocols": [
                    "TCP",
                    "UDP"
                ]
            }
        ],
        "EndpointDescriptions": []
    }
}
```

Per ulteriori informazioni, consulta Endpoint groups for custom routing accelerators in AWS Global Accelerator nella Global Accelerator Developer Guide AWS .

- Per i dettagli sull'API, consulta Command Reference. CreateCustomRoutingEndpointGroupAWS CLI

`create-custom-routing-listener`

Il seguente esempio di codice mostra come utilizzare`create-custom-routing-listener`.

AWS CLI

Come creare un listener per un acceleratore di instradamento personalizzato

L'esempio `create-custom-routing-listener` seguente crea un listener con un intervallo di porte compreso tra 5000 e 10000 per un acceleratore di instradamento personalizzato.

```
aws globalaccelerator create-custom-routing-listener \
    --accelerator-arn arn:aws:globalaccelerator::123456789012:accelerator/1234abcd-abcd-1234-abcd-1234abcdefgh \
    --port-ranges FromPort=5000,ToPort=10000
```

Output:

```
{
    "Listener": {
        "PortRange": [
            "FromPort": 5000,
            "ToPort": 10000
        ],
        "ListenerArn":
 "arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-abcd-1234-abcd-1234abcdefgh/listener/0123vxyz"
    }
}
```

Per ulteriori informazioni, consulta Listeners for custom routing accelerators in AWS Global Accelerator nella Global Accelerator Developer Guide AWS .

- Per i dettagli sull'API, consulta Command Reference. CreateCustomRoutingListenerAWS CLI

`create-endpoint-group`

Il seguente esempio di codice mostra come utilizzare`create-endpoint-group`.

AWS CLI

Come creare un gruppo di endpoint

L'esempio `create-endpoint-group` seguente crea un gruppo di endpoint con un endpoint.

```
aws globalaccelerator create-endpoint-group \
```

```
    --listener-arn arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-
abcd-1234-abcd-1234abcdefgh/listener/0123vxyz \
    --endpoint-group-region us-east-1 \
    --endpoint-configurations EndpointId=i-1234567890abcdef0,Weight=128
```

Output:

```
{
    "EndpointGroup": {
        "TrafficDialPercentage": 100.0,
        "EndpointDescriptions": [
            {
                "Weight": 128,
                "EndpointId": "i-1234567890abcdef0"
            }
        ],
        "EndpointGroupArn":
 "arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-abcd-1234-
abcd-1234abcdefgh/listener/0123vxyz/endpoint-group/098765zyxwvu",
        "EndpointGroupRegion": "us-east-1"
    }
}
```

Per ulteriori informazioni, consulta Endpoint groups in AWS Global Accelerator nella Global Accelerator AWS Developer Guide.

- Per i dettagli sull'API, consulta Command CreateEndpointGroupReference AWS CLI .

`create-listener`

Il seguente esempio di codice mostra come utilizzare`create-listener`.

AWS CLI

Come creare un listener

L'esempio `create-listener` seguente crea un listener con due porte.

```
aws globalaccelerator create-listener \
    --accelerator-arn arn:aws:globalaccelerator::123456789012:accelerator/1234abcd-
abcd-1234-abcd-1234abcdefgh \
    --port-ranges FromPort=80,ToPort=80 FromPort=81,ToPort=81 \
```

```
    --protocol TCP
```

Output:

```
{
    "Listener": {
        "PortRanges": [
            {
                "ToPort": 80,
                "FromPort": 80
            },
            {
                "ToPort": 81,
                "FromPort": 81
            }
        ],
        "ClientAffinity": "NONE",
        "Protocol": "TCP",
        "ListenerArn":
 "arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-abcd-1234-
abcd-1234abcdefgh/listener/0123vxyz"
    }
}
```

Per ulteriori informazioni, consulta Listeners in AWS Global Accelerator nella Global Accelerator AWS Developer Guide.

- Per i dettagli sull'API, consulta Command CreateListenerReference AWS CLI .

`deny-custom-routing-traffic`

Il seguente esempio di codice mostra come utilizzare`deny-custom-routing-traffic`.

AWS CLI

Come specificare un indirizzo di destinazione che non può ricevere traffico in un acceleratore di instradamento personalizzato

L'esempio `deny-custom-routing-traffic` seguente specifica l'indirizzo o gli indirizzi di destinazione in un endpoint di sottorete che non possono ricevere traffico per un acceleratore di instradamento personalizzato. Per specificare più indirizzi di destinazione, separali con uno spazio. Nessuna risposta per una deny-custom-routing-traffic chiamata riuscita.

```
aws globalaccelerator deny-custom-routing-traffic \
    --endpoint-group-
arn "arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-abcd-1234-
abcd-1234abcdefgh/listener/0123vxyz/endpoint-group/ab88888example" \
    --endpoint-id "subnet-abcd123example" \
    --destination-addresses "198.51.100.52"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta gli endpoint di sottorete VPC per acceleratori di routing personalizzati in Global Accelerator nella AWS Global Accelerator Developer Guide.AWS

- Per i dettagli sull'API, consulta Command Reference. DenyCustomRoutingTrafficAWS CLI

`deprovision-byoip-cidr`

Il seguente esempio di codice mostra come utilizzare`deprovision-byoip-cidr`.

AWS CLI

Come annullare il provisioning di un intervallo di indirizzi

L'`deprovision-byoip-cidr`esempio seguente rilascia l'intervallo di indirizzi specificato che avete predisposto per l'utilizzo con AWS le vostre risorse.

```
aws globalaccelerator deprovision-byoip-cidr \
    --cidr "198.51.100.0/24"
```

Output:

```
{
    "ByoipCidr": {
        "Cidr": "198.51.100.0/24",
        "State": "PENDING_DEPROVISIONING"
    }
}
```

Per ulteriori informazioni, consulta Bring your own IP address in AWS Global Accelerator nella Global Accelerator AWS Developer Guide.

- Per i dettagli sull'API, consulta AWS CLI Command DeprovisionByoipCidrReference.

`describe-accelerator-attributes`

Il seguente esempio di codice mostra come utilizzare`describe-accelerator-attributes`.

AWS CLI

Come descrivere gli attributi di un acceleratore

L'esempio `describe-accelerator-attributes` seguente recupera i dettagli degli attributi per un acceleratore.

```
aws globalaccelerator describe-accelerator-attributes \
    --accelerator-arn arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-abcd-1234-abcd-1234abcdefgh
```

Output:

```
{
    "AcceleratorAttributes": {
        "FlowLogsEnabled": true
        "FlowLogsS3Bucket": flowlogs-abc
        "FlowLogsS3Prefix": bucketprefix-abc
    }
}
```

Per ulteriori informazioni, consulta Accelerators in AWS Global Accelerator nella Global Accelerator AWS Developer Guide.

- Per i dettagli sull'API, consulta Command DescribeAcceleratorAttributesReference AWS CLI .

`describe-accelerator`

Il seguente esempio di codice mostra come utilizzare`describe-accelerator`.

AWS CLI

Come descrivere un acceleratore

L'esempio `describe-accelerator` seguente recupera i dettagli sull'acceleratore specificato.

```
aws globalaccelerator describe-accelerator \
```

```
    --accelerator-arn arn:aws:globalaccelerator::123456789012:accelerator/1234abcd-abcd-1234-abcd-1234abcdefgh
```

Output:

```
{
    "Accelerator": {
        "AcceleratorArn":
 "arn:aws:globalaccelerator::123456789012:accelerator/1234abcd-abcd-1234-
abcd-1234abcdefgh",
        "IpAddressType": "IPV4",
        "Name": "ExampleAccelerator",
        "Enabled": true,
        "Status": "IN_PROGRESS",
        "IpSets": [
            {
                "IpAddresses": [
                    "192.0.2.250",
                    "198.51.100.52"
                ],
                "IpFamily": "IPv4"
            }
        ],
        "DnsName":"a1234567890abcdef.awsglobalaccelerator.com",
        "CreatedTime": 1542394847,
        "LastModifiedTime": 1542395013
    }
}
```

Per ulteriori informazioni, consulta Accelerators in AWS Global Accelerator nella Global Accelerator AWS Developer Guide.

- Per i dettagli sull'API, consulta Command DescribeAcceleratorReference AWS CLI .

`describe-custom-routing-accelerator-attributes`

Il seguente esempio di codice mostra come utilizzare`describe-custom-routing-accelerator-attributes`.

AWS CLI

Come descrivere gli attributi di un acceleratore di instradamento personalizzato

L'esempio `describe-custom-routing-accelerator-attributes` seguente descrive gli attributi per un acceleratore di instradamento personalizzato.

```
aws globalaccelerator describe-custom-routing-accelerator-attributes \
   --accelerator-arn arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-abcd-1234-abcd-1234abcdefgh
```

Output:

```
{
    "AcceleratorAttributes": {
    "FlowLogsEnabled": false
    }
}
```

Per ulteriori informazioni, consulta la sezione Acceleratori di routing personalizzati in Global Accelerator nella AWS Global Accelerator Developer AWS Guide.

- Per i dettagli sull'API, consulta DescribeCustomRoutingAcceleratorAttributesCommand Reference.AWS CLI

`describe-custom-routing-accelerator`

Il seguente esempio di codice mostra come utilizzare`describe-custom-routing-accelerator`.

AWS CLI

Come descrivere un acceleratore di instradamento personalizzato

L'esempio `describe-custom-routing-accelerator` seguente recupera i dettagli sull'acceleratore di instradamento personalizzato specificato.

```
aws globalaccelerator describe-custom-routing-accelerator \
    --accelerator-arn arn:aws:globalaccelerator::123456789012:accelerator/1234abcd-abcd-1234-abcd-1234abcdefgh
```

Output:

```
{
```

```
    "Accelerator": {
        "AcceleratorArn":
 "arn:aws:globalaccelerator::123456789012:accelerator/1234abcd-abcd-1234-
abcd-1234abcdefgh",
        "IpAddressType": "IPV4",
        "Name": "ExampleCustomRoutingAccelerator",
        "Enabled": true,
        "Status": "IN_PROGRESS",
        "IpSets": [
            {
                "IpAddresses": [
                    "192.0.2.250",
                    "198.51.100.52"
                ],
                "IpFamily": "IPv4"
            }
        ],
        "DnsName":"a1234567890abcdef.awsglobalaccelerator.com",
        "CreatedTime": 1542394847,
        "LastModifiedTime": 1542395013
    }
}
```

Per ulteriori informazioni, consulta la sezione Acceleratori di routing personalizzati in Global Accelerator nella AWS Global Accelerator Developer AWS Guide.

- Per i dettagli sull'API, consulta DescribeCustomRoutingAcceleratorCommand Reference.AWS CLI

`describe-custom-routing-endpoint-group`

Il seguente esempio di codice mostra come utilizzare`describe-custom-routing-endpoint-group`.

AWS CLI

Come descrivere un gruppo di endpoint per un acceleratore di instradamento personalizzato

L'esempio `describe-custom-routing-endpoint-group` seguente descrive un gruppo di endpoint per un acceleratore di instradamento personalizzato.

```
aws globalaccelerator describe-custom-routing-endpoint-group \
```

```
    --endpoint-group-
arn arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-abcd-1234-
abcd-1234abcdefgh/listener/6789vxyz/endpoint-group/ab88888example
```

Output:

```
{
    "EndpointGroup": {
        "EndpointGroupArn":
 "arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-abcd-1234-
abcd-1234abcdefgh/listener/6789vxyz/endpoint-group/ab88888example",
        "EndpointGroupRegion": "us-east-2",
        "DestinationDescriptions": [
            {
                "FromPort": 5000,
                "ToPort": 10000,
                "Protocols": [
                    "UDP"
                ]
            }
        ],
        "EndpointDescriptions": [
            {
                "EndpointId": "subnet-1234567890abcdef0"
            }
        ]
    }
}
```

Per ulteriori informazioni, consulta Endpoint groups for custom routing accelerators in AWS Global Accelerator nella Global Accelerator Developer Guide AWS .

- Per i dettagli sull'API, consulta Command Reference. DescribeCustomRoutingEndpointGroupAWS CLI

`describe-custom-routing-listener`

Il seguente esempio di codice mostra come utilizzare`describe-custom-routing-listener`.

AWS CLI

Come descrivere un listener per un acceleratore di instradamento personalizzato

L'esempio `describe-custom-routing-listener` seguente descrive un listener per un acceleratore di instradamento personalizzato.

```
aws globalaccelerator describe-custom-routing-listener \
    --listener-arn arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-abcd-1234-abcd-1234abcdefgh/listener/abcdef1234
```

Output:

```
{
    "Listener": {
        "PortRanges": [
            "FromPort": 5000,
            "ToPort": 10000
        ],
        "ListenerArn":
 "arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-abcd-1234-abcd-1234abcdefgh/listener/abcdef1234"
    }
}
```

Per ulteriori informazioni, consulta Listeners for custom routing accelerators in AWS Global Accelerator nella Global Accelerator Developer Guide AWS .

- Per i dettagli sull'API, consulta Command Reference. DescribeCustomRoutingListenerAWS CLI

`describe-endpoint-group`

Il seguente esempio di codice mostra come utilizzare`describe-endpoint-group`.

AWS CLI

Come descrivere un gruppo di endpoint

L'`describe-endpoint-group`esempio seguente recupera i dettagli su un gruppo di endpoint con i seguenti endpoint: un' EC2 istanza Amazon, un ALB e un NLB.

```
aws globalaccelerator describe-endpoint-group \
    --endpoint-group-
arn arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-abcd-1234-abcd-1234abcdefgh/listener/6789vxyz-vxyz-6789-vxyz-6789lmnopqrs/endpoint-group/ab88888example
```

Output:

```
{
    "EndpointGroup": {
        "TrafficDialPercentage": 100.0,
        "EndpointDescriptions": [
        {
            "Weight": 128,
            "EndpointId": "i-1234567890abcdef0"
        },
        {
            "Weight": 128,
            "EndpointId": "arn:aws:elasticloadbalancing:us-
east-1:000123456789:loadbalancer/app/ALBTesting/alb01234567890xyz"
        },
        {
            "Weight": 128,
            "EndpointId": "arn:aws:elasticloadbalancing:us-
east-1:000123456789:loadbalancer/net/NLBTesting/alb01234567890qrs"
        }
        ],
        "EndpointGroupArn":
 "arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-abcd-1234-
abcd-1234abcdefgh/listener/6789vxyz-vxyz-6789-vxyz-6789lmnopqrs/endpoint-
group/4321abcd-abcd-4321-abcd-4321abcdefg",
        "EndpointGroupRegion": "us-east-1"
    }
}
```

Per ulteriori informazioni, consulta i gruppi di endpoint in Global Accelerator nella AWS Global Accelerator Developer Guide.AWS

- Per i dettagli sull'API, consulta Command DescribeEndpointGroupReference AWS CLI .

`describe-listener`

Il seguente esempio di codice mostra come utilizzare`describe-listener`.

AWS CLI

Come descrivere un listener

L'esempio `describe-listener` seguente descrive un listener.

```
aws globalaccelerator describe-listener \
    --listener-arn arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-abcd-1234-abcd-1234abcdefgh/listener/abcdef1234
```

Output:

```
{
    "Listener": {
        "ListenerArn":
 "arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-abcd-1234-abcd-1234abcdefgh/listener/abcdef1234",
        "PortRanges": [
            {
                "FromPort": 80,
                "ToPort": 80
            }
        ],
        "Protocol": "TCP",
        "ClientAffinity": "NONE"
    }
}
```

Per ulteriori informazioni, consulta Listeners in AWS Global Accelerator nella Global Accelerator AWS Developer Guide.

- Per i dettagli sull'API, consulta Command DescribeListenerReference AWS CLI .

`list-accelerators`

Il seguente esempio di codice mostra come utilizzare`list-accelerators`.

AWS CLI

Come elencare gli acceleratori

L'`list-accelerators`esempio seguente elenca gli acceleratori presenti nel tuo AWS account. In questo account sono presenti due acceleratori.

```
aws globalaccelerator list-accelerators
```

Output:

```
{
    "Accelerators": [
        {
            "AcceleratorArn":
 "arn:aws:globalaccelerator::012345678901:accelerator/5555abcd-abcd-5555-
abcd-5555EXAMPLE1",
            "Name": "TestAccelerator",
            "IpAddressType": "IPV4",
            "Enabled": true,
            "IpSets": [
                {
                    "IpFamily": "IPv4",
                    "IpAddresses": [
                        "192.0.2.250",
                        "198.51.100.52"
                    ]
                }
            ],
            "DnsName": "5a5a5a5a5a5a5a5a.awsglobalaccelerator.com",
            "Status": "DEPLOYED",
            "CreatedTime": 1552424416.0,
            "LastModifiedTime": 1569375641.0
        },
        {
            "AcceleratorArn":
 "arn:aws:globalaccelerator::888888888888:accelerator/8888abcd-abcd-8888-
abcd-8888EXAMPLE2",
            "Name": "ExampleAccelerator",
            "IpAddressType": "IPV4",
            "Enabled": true,
            "IpSets": [
                {
                    "IpFamily": "IPv4",
                    "IpAddresses": [
                        "192.0.2.100",
                        "198.51.100.10"
                    ]
                }
            ],
            "DnsName": "6a6a6a6a6a6a6a6a.awsglobalaccelerator.com",
            "Status": "DEPLOYED",
            "CreatedTime": 1575585564.0,
            "LastModifiedTime": 1579809243.0
```

```
        },
    ]
}
```

Per ulteriori informazioni, consulta Accelerators in AWS Global Accelerator nella Global Accelerator Developer AWS Guide.

- Per i dettagli sull'API, consulta Command ListAcceleratorsReference AWS CLI .

`list-byoip-cidr`

Il seguente esempio di codice mostra come utilizzare`list-byoip-cidr`.

AWS CLI

Come elencare gli intervalli di indirizzi

L'esempio `list-byoip-cidr` seguente elenca gli intervalli di indirizzi BYOIP (Bring Your Own IP) di cui hai effettuato il provisioning per l'uso con Global Accelerator.

```
aws globalaccelerator list-byoip-cidrs
```

Output:

```
{
    "ByoipCidrs": [
        {
            "Cidr": "198.51.100.0/24",
            "State": "READY"
        }
        {
            "Cidr": "203.0.113.25/24",
            "State": "READY"
        }
    ]
}
```

Per ulteriori informazioni, consulta Bring your own IP address in AWS Global Accelerator nella Global Accelerator Developer Guide.AWS

- Per i dettagli sull'API, consulta AWS CLI Command ListByoipCidrReference.

`list-custom-routing-accelerators`

Il seguente esempio di codice mostra come utilizzare`list-custom-routing-accelerators`.

AWS CLI

Come elencare i tuoi acceleratori di instradamento personalizzati

L'`list-custom-routing-accelerators`esempio seguente elenca gli acceleratori di routing personalizzati in un AWS account.

```
aws globalaccelerator list-custom-routing-accelerators
```

Output:

```
{
    "Accelerators": [
        {
            "AcceleratorArn":
 "arn:aws:globalaccelerator::012345678901:accelerator/5555abcd-abcd-5555-
abcd-5555EXAMPLE1",
            "Name": "TestCustomRoutingAccelerator",
            "IpAddressType": "IPV4",
            "Enabled": true,
            "IpSets": [
                {
                    "IpFamily": "IPv4",
                    "IpAddresses": [
                        "192.0.2.250",
                        "198.51.100.52"
                    ]
                }
            ],
            "DnsName": "5a5a5a5a5a5a5a5a.awsglobalaccelerator.com",
            "Status": "DEPLOYED",
            "CreatedTime": 1552424416.0,
            "LastModifiedTime": 1569375641.0
        },
        {
            "AcceleratorArn":
 "arn:aws:globalaccelerator::888888888888:accelerator/8888abcd-abcd-8888-
abcd-8888EXAMPLE2",
            "Name": "ExampleCustomRoutingAccelerator",
```

```
            "IpAddressType": "IPV4",
            "Enabled": true,
            "IpSets": [
                {
                    "IpFamily": "IPv4",
                    "IpAddresses": [
                        "192.0.2.100",
                        "198.51.100.10"
                    ]
                }
            ],
            "DnsName": "6a6a6a6a6a6a6a.awsglobalaccelerator.com",
            "Status": "DEPLOYED",
            "CreatedTime": 1575585564.0,
            "LastModifiedTime": 1579809243.0
        },
    ]
}
```

Per ulteriori informazioni, consulta la sezione Acceleratori di routing personalizzati in Global Accelerator nella AWS Global Accelerator Developer Guide AWS .

- Per i dettagli sull'API, consulta ListCustomRoutingAcceleratorsCommand Reference.AWS CLI

`list-custom-routing-endpoint-groups`

Il seguente esempio di codice mostra come utilizzare`list-custom-routing-endpoint-groups`.

AWS CLI

Come elencare i gruppi di endpoint per un listener in un acceleratore di instradamento personalizzato

L'esempio `list-custom-routing-endpoint-groups` seguente elenca i gruppi di endpoint per un listener in un acceleratore di instradamento personalizzato.

```
aws globalaccelerator list-custom-routing-endpoint-groups \
    --listener-arn arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-abcd-1234-abcd-1234abcdefgh/listener/abcdef1234
```

Output:

```
{
```

```
    "EndpointGroups": [
        {
            "EndpointGroupArn":
 "arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-abcd-1234-
abcd-1234abcdefgh/listener/abcdef1234/endpoint-group/ab88888example",
            "EndpointGroupRegion": "eu-central-1",
            "DestinationDescriptions": [
                {
                    "FromPort": 80,
                    "ToPort": 80,
                    "Protocols": [
                        "TCP",
                        "UDP"
                    ]
                }
            ]
            "EndpointDescriptions": [
                {
                    "EndpointId": "subnet-abcd123example"
                }
            ]
        }
    ]
}
```

Per ulteriori informazioni, consulta Endpoint groups for custom routing accelerators in AWS Global Accelerator nella Global Accelerator Developer Guide AWS .

- Per i dettagli sull'API, consulta Command Reference. ListCustomRoutingEndpointGroupsAWS CLI

`list-custom-routing-listeners`

Il seguente esempio di codice mostra come utilizzare`list-custom-routing-listeners`.

AWS CLI

Come elencare i listener per gli acceleratori di instradamento personalizzati

L'esempio `list-custom-routing-listeners` seguente elenca i listener per un acceleratore di instradamento personalizzato.

```
aws globalaccelerator list-custom-routing-listeners \
```

```
    --accelerator-arn arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-abcd-1234-abcd-1234abcdefgh
```

Output:

```
{
    "Listeners": [
        {
            "ListenerArn":
 "arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-abcd-1234-abcd-1234abcdefgh/listener/abcdef1234",
            "PortRanges": [
                {
                    "FromPort": 5000,
                    "ToPort": 10000
                }
            ],
            "Protocol": "TCP"
        }
    ]
}
```

Per ulteriori informazioni, consulta Listeners for custom routing accelerators in AWS Global Accelerator nella Global Accelerator Developer Guide AWS .

- Per i dettagli sull'API, consulta Command Reference. ListCustomRoutingListenersAWS CLI

`list-custom-routing-port-mappings-by-destination`

Il seguente esempio di codice mostra come utilizzare`list-custom-routing-port-mappings-by-destination`.

AWS CLI

Come elencare le mappature delle porte per una determinata destinazione dell'acceleratore di instradamento personalizzato

L'`list-custom-routing-port-mappings-by-destination`esempio seguente fornisce le mappature delle porte per un EC2 server di destinazione specifico (all'indirizzo di destinazione) per un acceleratore di routing personalizzato.

```
aws globalaccelerator list-custom-routing-port-mappings-by-destination \
```

```
    --endpoint-id subnet-abcd123example \
    --destination-address 198.51.100.52
```

Output:

```
{
    "DestinationPortMappings": [
        {
            "AcceleratorArn":
 "arn:aws:globalaccelerator::402092451327:accelerator/24ea29b8-
d750-4489-8919-3095f3c4b0a7",
                "AcceleratorSocketAddresses": [
                    {
                        "IpAddress": "192.0.2.250",
                        "Port": 65514
                    },
                    {
                        "IpAddress": "192.10.100.99",
                        "Port": 65514
                    }
                ],
                "EndpointGroupArn":
 "arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-abcd-1234-
abcd-1234abcdefgh/listener/0123vxyz/endpoint-group/ab88888example",
                "EndpointId": "subnet-abcd123example",
                "EndpointGroupRegion": "us-west-2",
                "DestinationSocketAddress": {
                    "IpAddress": "198.51.100.52",
                    "Port": 80
                },
                "IpAddressType": "IPv4",
                "DestinationTrafficState": "ALLOW"
        }
    ]
}
```

Per ulteriori informazioni, consulta Come funzionano gli acceleratori di routing personalizzati in Global Accelerator nella AWS Global Accelerator Developer Guide.AWS

- Per i dettagli sull'API, consulta ListCustomRoutingPortMappingsByDestinationCommand Reference.AWS CLI

`list-custom-routing-port-mappings`

Il seguente esempio di codice mostra come utilizzare`list-custom-routing-port-mappings`.

AWS CLI

Come elencare le mappature delle porte in un acceleratore di instradamento personalizzato

L'esempio `list-custom-routing-port-mappings` seguente fornisce un elenco parziale delle mappature delle porte in un acceleratore di instradamento personalizzato.

```
aws globalaccelerator list-custom-routing-port-mappings \
    --accelerator-arn arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-abcd-1234-abcd-1234abcdefgh
```

Output:

```
{
    "PortMappings": [
        {
            "AcceleratorPort": 40480,
            "EndpointGroupArn":
 "arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-abcd-1234-
abcd-1234abcdefgh/listener/0123vxyz/endpoint-group/098765zyxwvu",
            "EndpointId": "subnet-1234567890abcdef0",
            "DestinationSocketAddress": {
                "IpAddress": "192.0.2.250",
                "Port": 80
            },
            "Protocols": [
                "TCP",
                "UDP"
            ],
            "DestinationTrafficState": "ALLOW"
        }
        {
            "AcceleratorPort": 40481,
            "EndpointGroupArn":
 "arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-abcd-1234-
abcd-1234abcdefgh/listener/0123vxyz/endpoint-group/098765zyxwvu",
            "EndpointId": "subnet-1234567890abcdef0",
            "DestinationSocketAddress": {
               "IpAddress": "192.0.2.251",
```

```
                "Port": 80
            },
            "Protocols": [
                "TCP",
                "UDP"
            ],
            "DestinationTrafficState": "ALLOW"
        }
    ]
}
```

Per ulteriori informazioni, consulta Come funzionano gli acceleratori di routing personalizzati in Global Accelerator nella AWS Global Accelerator Developer AWS Guide.

- Per i dettagli sull'API, consulta ListCustomRoutingPortMappingsCommand Reference.AWS CLI

`list-endpoint-groups`

Il seguente esempio di codice mostra come utilizzare`list-endpoint-groups`.

AWS CLI

Come elencare i gruppi di endpoint

L'esempio `list-endpoint-groups` seguente elenca i gruppi di endpoint per un listener. In questo listener sono presenti due gruppi di endpoint.

```
aws globalaccelerator --region us-west-2 list-endpoint-groups \
    --listener-arn arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-abcd-1234-abcd-1234abcdefgh/listener/abcdef1234
```

Output:

```
{
    "EndpointGroups": [
        {
            "EndpointGroupArn":
 "arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-abcd-1234-abcd-1234abcdefgh/listener/abcdef1234/endpoint-group/ab88888example",
            "EndpointGroupRegion": "eu-central-1",
            "EndpointDescriptions": [],
            "TrafficDialPercentage": 100.0,
```

```
            "HealthCheckPort": 80,
            "HealthCheckProtocol": "TCP",
            "HealthCheckIntervalSeconds": 30,
            "ThresholdCount": 3
        }
        {
            "EndpointGroupArn":
 "arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-abcd-1234-
abcd-1234abcdefgh/listener/abcdef1234/endpoint-group/ab99999example",
            "EndpointGroupRegion": "us-east-1",
            "EndpointDescriptions": [],
            "TrafficDialPercentage": 50.0,
            "HealthCheckPort": 80,
            "HealthCheckProtocol": "TCP",
            "HealthCheckIntervalSeconds": 30,
            "ThresholdCount": 3
        }
    ]
}
```

Per ulteriori informazioni, consulta Endpoint Groups in AWS Global Accelerator nella Global Accelerator AWS Developer Guide.

- Per i dettagli sull'API, consulta Command ListEndpointGroupsReference AWS CLI .

`list-listeners`

Il seguente esempio di codice mostra come utilizzare`list-listeners`.

AWS CLI

Come elencare i listener

L'esempio `list-listeners` seguente elenca i listener per un acceleratore.

```
aws globalaccelerator list-listeners \
    --accelerator-arn arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-
abcd-1234-abcd-1234abcdefgh
```

Output:

```
{
    "Listeners": [
```

```
        {
            "ListenerArn":
 "arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-abcd-1234-
abcd-1234abcdefgh/listener/abcdef1234",
            "PortRanges": [
                {
                    "FromPort": 80,
                    "ToPort": 80
                }
            ],
            "Protocol": "TCP",
            "ClientAffinity": "NONE"
        }
    ]
}
```

Per ulteriori informazioni, consulta Listeners in AWS Global Accelerator nella Global Accelerator AWS Developer Guide.

- Per i dettagli sull'API, consulta Command ListListenersReference AWS CLI .

`list-tags-for-resource`

Il seguente esempio di codice mostra come utilizzare`list-tags-for-resource`.

AWS CLI

Come elencare i tag per un acceleratore

L'esempio `list-tags-for-resource` seguente elenca i tag per un determinato acceleratore.

```
aws globalaccelerator list-tags-for-resource \
    --accelerator-arn arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-
abcd-1234-abcd-1234abcdefgh
```

Output:

```
{
    "Tags": [
        {
            "Key": "Project",
            "Value": "A123456"
```

```
        }
    ]
}
```

Per ulteriori informazioni, consulta Tagging in AWS Global Accelerator nella Global Accelerator AWS Developer Guide.

- Per i dettagli sull'API, consulta Command ListTagsForResourceReference AWS CLI .

`provision-byoip-cidr`

Il seguente esempio di codice mostra come utilizzare`provision-byoip-cidr`.

AWS CLI

Come effettuare il provisioning di un intervallo di indirizzi

L'`provision-byoip-cidr`esempio seguente fornisce l'intervallo di indirizzi specificato da utilizzare con le AWS risorse.

```
aws globalaccelerator provision-byoip-cidr \
    --cidr 192.0.2.250/24 \
    --cidr-authorization-context Message="$text_message",Signature="$signed_message"
```

Output:

```
{
    "ByoipCidr": {
        "Cidr": "192.0.2.250/24",
        "State": "PENDING_PROVISIONING"
    }
}
```

Per ulteriori informazioni, consulta Bring your own IP address in AWS Global Accelerator nella Global Accelerator Developer Guide.AWS

- Per i dettagli sull'API, consulta AWS CLI Command ProvisionByoipCidrReference.

`tag-resource`

Il seguente esempio di codice mostra come utilizzare`tag-resource`.

AWS CLI

Come taggare un acceleratore

L'esempio `tag-resource` seguente aggiunge i tag Name e Project a un acceleratore, nonché i valori corrispondenti per ciascuno di essi.

```
aws globalaccelerator tag-resource \
    --resource-arn arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-abcd-1234-abcd-1234abcdefgh \
    --tags Key="Name",Value="Example Name" Key="Project",Value="Example Project"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Tagging in AWS Global Accelerator nella Global Accelerator AWS Developer Guide.

- Per i dettagli sull'API, consulta Command TagResourceReference AWS CLI .

`untag-resource`

Il seguente esempio di codice mostra come utilizzare`untag-resource`.

AWS CLI

Come rimuovere un tag da un acceleratore

L'esempio `untag-resource` seguente rimuove i tag Name e Project da un acceleratore.

```
aws globalaccelerator untag-resource \
    --resource-arn arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-abcd-1234-abcd-1234abcdefgh \
    --tag-keys Key="Name" Key="Project"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Tagging in AWS Global Accelerator nella Global Accelerator AWS Developer Guide.

- Per i dettagli sull'API, consulta Command UntagResourceReference AWS CLI .

`update-accelerator-attributes`

Il seguente esempio di codice mostra come utilizzare`update-accelerator-attributes`.

AWS CLI

Come aggiornare gli attributi di un acceleratore

L'esempio `update-accelerator-attributes` seguente aggiorna un acceleratore per abilitare i log di flusso. È necessario specificare la Regione `US-West-2 (Oregon)` per creare o aggiornare gli attributi di un acceleratore.

```
aws globalaccelerator update-accelerator-attributes \
    --accelerator-arn arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-abcd-1234-abcd-1234abcdefgh \
    --flow-logs-enabled \
    --flow-logs-s3-bucket flowlogs-abc \
    --flow-logs-s3-prefix bucketprefix-abc
```

Output:

```
{
    "AcceleratorAttributes": {
        "FlowLogsEnabled": true
        "FlowLogsS3Bucket": flowlogs-abc
        "FlowLogsS3Prefix": bucketprefix-abc
    }
}
```

Per ulteriori informazioni, consulta Accelerators in AWS Global Accelerator nella Global Accelerator AWS Developer Guide.

- Per i dettagli sull'API, consulta Command UpdateAcceleratorAttributesReference AWS CLI .

`update-accelerator`

Il seguente esempio di codice mostra come utilizzare`update-accelerator`.

AWS CLI

Come aggiornare un acceleratore

L'esempio `update-accelerator` seguente modifica un acceleratore rinominandolo in `ExampleAcceleratorNew`. È necessario specificare la Regione `US-West-2 (Oregon)` per creare o aggiornare gli acceleratori.

```
aws globalaccelerator update-accelerator \
    --accelerator-arn arn:aws:globalaccelerator::123456789012:accelerator/1234abcd-abcd-1234-abcd-1234abcdefgh \
    --name ExampleAcceleratorNew
```

Output:

```
{
    "Accelerator": {
        "AcceleratorArn":
 "arn:aws:globalaccelerator::123456789012:accelerator/1234abcd-abcd-1234-
abcd-1234abcdefgh",
        "IpAddressType": "IPV4",
        "Name": "ExampleAcceleratorNew",
        "Enabled": true,
        "Status": "IN_PROGRESS",
        "IpSets": [
            {
                "IpAddresses": [
                    "192.0.2.250",
                    "198.51.100.52"
                ],
                "IpFamily": "IPv4"
            }
        ],
        "DnsName":"a1234567890abcdef.awsglobalaccelerator.com",
        "CreatedTime": 1232394847,
        "LastModifiedTime": 1232395654
    }
}
```

Per ulteriori informazioni, consulta Accelerators in AWS Global Accelerator nella Global Accelerator AWS Developer Guide.

- Per i dettagli sull'API, consulta Command UpdateAcceleratorReference AWS CLI .

`update-custom-routing-accelerator-attributes`

Il seguente esempio di codice mostra come utilizzare`update-custom-routing-accelerator-attributes`.

AWS CLI

Come aggiornare gli attributi di un acceleratore di instradamento personalizzato

L'esempio `update-custom-routing-accelerator-attributes` seguente aggiorna un acceleratore di instradamento personalizzato per abilitare i log di flusso.

```
aws globalaccelerator update-custom-routing-accelerator-attributes \
    --accelerator-arn arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-abcd-1234-abcd-1234abcdefgh \
    --flow-logs-enabled \
    --flow-logs-s3-bucket flowlogs-abc \
    --flow-logs-s3-prefix bucketprefix-abc
```

Output:

```
{
    "AcceleratorAttributes": {
        "FlowLogsEnabled": true
        "FlowLogsS3Bucket": flowlogs-abc
        "FlowLogsS3Prefix": bucketprefix-abc
    }
}
```

Per ulteriori informazioni, consulta la sezione Acceleratori di routing personalizzati in Global Accelerator nella AWS Global Accelerator Developer AWS Guide.

- Per i dettagli sull'API, consulta UpdateCustomRoutingAcceleratorAttributesCommand Reference.AWS CLI

`update-custom-routing-accelerator`

Il seguente esempio di codice mostra come utilizzare`update-custom-routing-accelerator`.

AWS CLI

Come aggiornare un acceleratore di instradamento personalizzato

L'esempio `update-custom-routing-accelerator` seguente modifica un acceleratore di instradamento personalizzato cambiandone il nome.

```
aws globalaccelerator --region us-west-2 update-custom-routing-accelerator \
    --accelerator-arn arn:aws:globalaccelerator::123456789012:accelerator/1234abcd-abcd-1234-abcd-1234abcdefgh \
    --name ExampleCustomRoutingAcceleratorNew
```

Output:

```
{
    "Accelerator": {
        "AcceleratorArn":
 "arn:aws:globalaccelerator::123456789012:accelerator/1234abcd-abcd-1234-abcd-1234abcdefgh",
        "IpAddressType": "IPV4",
        "Name": "ExampleCustomRoutingAcceleratorNew",
        "Enabled": true,
        "Status": "IN_PROGRESS",
        "IpSets": [
            {
                "IpAddresses": [
                    "192.0.2.250",
                    "198.51.100.52"
                ],
                "IpFamily": "IPv4"
            }
        ],
        "DnsName":"a1234567890abcdef.awsglobalaccelerator.com",
        "CreatedTime": 1232394847,
        "LastModifiedTime": 1232395654
    }
}
```

Per ulteriori informazioni, consulta la sezione Acceleratori di routing personalizzati in Global Accelerator nella AWS Global Accelerator Developer AWS Guide.

- Per i dettagli sull'API, consulta UpdateCustomRoutingAcceleratorCommand Reference.AWS CLI

update-custom-routing-listener

Il seguente esempio di codice mostra come utilizzareupdate-custom-routing-listener.

AWS CLI

Come aggiornare un listener per un acceleratore di instradamento personalizzato

L'esempio update-custom-routing-listener seguente aggiorna un listener per modificare l'intervallo di porte.

```
aws globalaccelerator update-custom-routing-listener \
    --listener-arn arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-abcd-1234-abcd-1234abcdefgh/listener/0123vxyz \
    --port-ranges FromPort=10000,ToPort=20000
```

Output:

```
{
    "Listener": {
        "ListenerArn":
 "arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-abcd-1234-
abcd-1234abcdefgh/listener/0123vxyz
        "PortRanges": [
            {
                "FromPort": 10000,
                "ToPort": 20000
            }
        ],
        "Protocol": "TCP"
    }
}
```

Per ulteriori informazioni, consulta Listeners for custom routing accelerators in AWS Global Accelerator nella Global Accelerator Developer Guide AWS .

- Per i dettagli sull'API, consulta Command Reference. UpdateCustomRoutingListenerAWS CLI

update-endpoint-group

Il seguente esempio di codice mostra come utilizzareupdate-endpoint-group.

AWS CLI

Come aggiornare un gruppo di endpoint

L'esempio `update-endpoint-group` seguente aggiunge tre endpoint a un gruppo di endpoint: un indirizzo IP elastico, un ALB e un NLB.

```
aws globalaccelerator update-endpoint-group \
    --endpoint-group-
arn arn:aws:globalaccelerator::123456789012:accelerator/1234abcd-abcd-1234-
abcd-1234abcdefgh/listener/6789vxyz-vxyz-6789-vxyz-6789lmnopqrs/endpoint-group/
ab88888example \
    --endpoint-configurations \
        EndpointId=eipalloc-eip01234567890abc,Weight=128 \
        EndpointId=arn:aws:elasticloadbalancing:us-east-1:000123456789:loadbalancer/
app/ALBTesting/alb01234567890xyz,Weight=128 \
        EndpointId=arn:aws:elasticloadbalancing:us-east-1:000123456789:loadbalancer/
net/NLBTesting/alb01234567890qrs,Weight=128
```

Output:

```
{
    "EndpointGroup": {
        "TrafficDialPercentage": 100,
        "EndpointDescriptions": [
            {
                "Weight": 128,
                "EndpointId": "eip01234567890abc"
            },
            {
                "Weight": 128,
                "EndpointId": "arn:aws:elasticloadbalancing:us-
east-1:000123456789:loadbalancer/app/ALBTesting/alb01234567890xyz"
            },
            {
                "Weight": 128,
                "EndpointId": "arn:aws:elasticloadbalancing:us-
east-1:000123456789:loadbalancer/net/NLBTesting/alb01234567890qrs"
            }
        ],
        "EndpointGroupArn":
 "arn:aws:globalaccelerator::123456789012:accelerator/1234abcd-abcd-1234-
```

```
abcd-1234abcdefgh/listener/6789vxyz-vxyz-6789-vxyz-6789lmnopqrs/endpoint-
group/4321abcd-abcd-4321-abcd-4321abcdefg",
        "EndpointGroupRegion": "us-east-1"
    }
}
```

Per ulteriori informazioni, consulta Endpoint groups in AWS Global Accelerator nella Global Accelerator AWS Developer Guide.

- Per i dettagli sull'API, consulta Command UpdateEndpointGroupReference AWS CLI .

`update-listener`

Il seguente esempio di codice mostra come utilizzareup`date-listener`.

AWS CLI

Come aggiornare un listener

L'esempio `update-listener` seguente aggiorna un listener per cambiare la porta in 100.

```
aws globalaccelerator update-listener \
    --listener-arn arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-
abcd-1234-abcd-1234abcdefgh/listener/0123vxyz \
    --port-ranges FromPort=100,ToPort=100
```

Output:

```
{
    "Listener": {
        "ListenerArn":
 "arn:aws:globalaccelerator::012345678901:accelerator/1234abcd-abcd-1234-
abcd-1234abcdefgh/listener/0123vxyz
        "PortRanges": [
            {
                "FromPort": 100,
                "ToPort": 100
            }
        ],
        "Protocol": "TCP",
        "ClientAffinity": "NONE"
    }
```

```
}
```

Per ulteriori informazioni, consulta Listeners in AWS Global Accelerator nella Global Accelerator AWS Developer Guide.

- Per i dettagli sull'API, consulta Command UpdateListenerReference AWS CLI .

`withdraw-byoip-cidr`

Il seguente esempio di codice mostra come utilizzare`withdraw-byoip-cidr`.

AWS CLI

Come ritirare un intervallo di indirizzi

L'`withdraw-byoip-cidr`esempio seguente rimuove da AWS Global Accelerator un intervallo di indirizzi che avevate precedentemente pubblicizzato per l'utilizzo con le vostre risorse. AWS

```
aws globalaccelerator withdraw-byoip-cidr \
    --cidr 192.0.2.250/24
```

Output:

```
{
    "ByoipCidr": {
        "Cidr": "192.0.2.250/24",
        "State": "PENDING_WITHDRAWING"
    }
}
```

Per ulteriori informazioni, consulta Bring your own IP address in AWS Global Accelerator nella Global Accelerator Developer Guide AWS .

- Per i dettagli sull'API, consulta AWS CLI Command WithdrawByoipCidrReference.

AWS Glue esempi utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with AWS Glue.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`batch-stop-job-run`

Il seguente esempio di codice mostra come utilizzare`batch-stop-job-run`.

AWS CLI

Come arrestare le esecuzioni dei processi

L'esempio `batch-stop-job-run` seguente avvia un processo.

```
aws glue batch-stop-job-run \
    --job-name "my-testing-job" \
    --job-run-id jr_852f1de1f29fb62e0ba4166c33970803935d87f14f96cfdee5089d5274a61d3f
```

Output:

```
{
    "SuccessfulSubmissions": [
        {
            "JobName": "my-testing-job",
            "JobRunId":
 "jr_852f1de1f29fb62e0ba4166c33970803935d87f14f96cfdee5089d5274a61d3f"
        }
    ],
    "Errors": [],
    "ResponseMetadata": {
        "RequestId": "66bd6b90-01db-44ab-95b9-6aeff0e73d88",
        "HTTPStatusCode": 200,
        "HTTPHeaders": {
            "date": "Fri, 16 Oct 2020 20:54:51 GMT",
```

```
            "content-type": "application/x-amz-json-1.1",
            "content-length": "148",
            "connection": "keep-alive",
            "x-amzn-requestid": "66bd6b90-01db-44ab-95b9-6aeff0e73d88"
        },
        "RetryAttempts": 0
    }
}
```

Per ulteriori informazioni, consulta Esecuzioni di processi nella Guida per gli sviluppatori di AWS Glue.

- Per i dettagli sull'API, consulta BatchStopJobRun AWS CLICommand Reference.

`create-connection`

Il seguente esempio di codice mostra come utilizzare`create-connection`.

AWS CLI

Per creare una connessione per gli archivi dati di AWS Glue

L'`create-connection`esempio seguente crea una connessione nel AWS Glue Data Catalog che fornisce informazioni di connessione per un data store Kafka.

```
aws glue create-connection \
    --connection-input '{ \
        "Name":"conn-kafka-custom", \
        "Description":"kafka connection with ssl to custom kafka", \
        "ConnectionType":"KAFKA",  \
        "ConnectionProperties":{  \
            "KAFKA_BOOTSTRAP_SERVERS":"<Kafka-broker-server-url>:<SSL-Port>", \
            "KAFKA_SSL_ENABLED":"true", \
            "KAFKA_CUSTOM_CERT": "s3://bucket/prefix/cert-file.pem" \
        }, \
        "PhysicalConnectionRequirements":{ \
            "SubnetId":"subnet-1234", \
            "SecurityGroupIdList":["sg-1234"], \
            "AvailabilityZone":"us-east-1a"} \
    }' \
    --region us-east-1
    --endpoint https://glue.us-east-1.amazonaws.com
```

Questo comando non produce alcun output.

Per ulteriori informazioni, vedere Defining Connections in the AWS Glue Data Catalog nella AWS Glue Developer Guide.

- Per i dettagli sulle API, consulta CreateConnection AWS CLICommand Reference.

`create-database`

Il seguente esempio di codice mostra come utilizzare`create-database`.

AWS CLI

Come creare un database

L'`create-database`esempio seguente crea un database nel AWS Glue Data Catalog.

```
aws glue create-database \
    --database-input "{\"Name\":\"tempdb\"}" \
    --profile my_profile \
    --endpoint https://glue.us-east-1.amazonaws.com
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Definizione di un database nel catalogo dati nella Guida per gli sviluppatori di AWS Glue.

- Per i dettagli sull'API, vedere CreateDatabasein AWS CLI Command Reference.

`create-job`

Il seguente esempio di codice mostra come utilizzare`create-job`.

AWS CLI

Come creare un processo di trasformazione dei dati

L'esempio `create-job` seguente crea un processo di streaming che esegue uno script archiviato in S3.

```
aws glue create-job \
    --name my-testing-job \
```

```
    --role AWSGlueServiceRoleDefault \
    --command '{ \
        "Name": "gluestreaming", \
        "ScriptLocation": "s3://amzn-s3-demo-bucket/folder/" \
    }' \
    --region us-east-1 \
    --output json \
    --default-arguments '{ \
        "--job-language":"scala", \
        "--class":"GlueApp" \
    }' \
    --profile my-profile \
    --endpoint https://glue.us-east-1.amazonaws.com
```

Contenuto di `test_script.scala`:

```
import com.amazonaws.services.glue.ChoiceOption
import com.amazonaws.services.glue.GlueContext
import com.amazonaws.services.glue.MappingSpec
import com.amazonaws.services.glue.ResolveSpec
import com.amazonaws.services.glue.errors.CallSite
import com.amazonaws.services.glue.util.GlueArgParser
import com.amazonaws.services.glue.util.Job
import com.amazonaws.services.glue.util.JsonOptions
import org.apache.spark.SparkContext
import scala.collection.JavaConverters._

object GlueApp {
    def main(sysArgs: Array[String]) {
        val spark: SparkContext = new SparkContext()
        val glueContext: GlueContext = new GlueContext(spark)
        // @params: [JOB_NAME]
        val args = GlueArgParser.getResolvedOptions(sysArgs,
 Seq("JOB_NAME").toArray)
        Job.init(args("JOB_NAME"), glueContext, args.asJava)
        // @type: DataSource
        // @args: [database = "tempdb", table_name = "s3-source", transformation_ctx
 = "datasource0"]
        // @return: datasource0
        // @inputs: []
        val datasource0 = glueContext.getCatalogSource(database = "tempdb",
 tableName = "s3-source", redshiftTmpDir = "", transformationContext =
 "datasource0").getDynamicFrame()
```

```
        // @type: ApplyMapping
        // @args: [mapping = [("sensorid", "int", "sensorid", "int"),
 ("currenttemperature", "int", "currenttemperature", "int"), ("status", "string",
 "status", "string")], transformation_ctx = "applymapping1"]
        // @return: applymapping1
        // @inputs: [frame = datasource0]
        val applymapping1 = datasource0.applyMapping(mappings = Seq(("sensorid",
 "int", "sensorid", "int"), ("currenttemperature", "int", "currenttemperature",
 "int"), ("status", "string", "status", "string")), caseSensitive = false,
 transformationContext = "applymapping1")
        // @type: SelectFields
        // @args: [paths = ["sensorid", "currenttemperature", "status"],
 transformation_ctx = "selectfields2"]
        // @return: selectfields2
        // @inputs: [frame = applymapping1]
        val selectfields2 = applymapping1.selectFields(paths = Seq("sensorid",
 "currenttemperature", "status"), transformationContext = "selectfields2")
        // @type: ResolveChoice
        // @args: [choice = "MATCH_CATALOG", database = "tempdb", table_name = "my-
s3-sink", transformation_ctx = "resolvechoice3"]
        // @return: resolvechoice3
        // @inputs: [frame = selectfields2]
        val resolvechoice3 = selectfields2.resolveChoice(choiceOption =
 Some(ChoiceOption("MATCH_CATALOG")), database = Some("tempdb"), tableName =
 Some("my-s3-sink"), transformationContext = "resolvechoice3")
        // @type: DataSink
        // @args: [database = "tempdb", table_name = "my-s3-sink",
 transformation_ctx = "datasink4"]
        // @return: datasink4
        // @inputs: [frame = resolvechoice3]
        val datasink4 = glueContext.getCatalogSink(database = "tempdb",
 tableName = "my-s3-sink", redshiftTmpDir = "", transformationContext =
 "datasink4").writeDynamicFrame(resolvechoice3)
        Job.commit()
    }
}
```

Output:

```
{
    "Name": "my-testing-job"
}
```

Per ulteriori informazioni, consulta Authoring Jobs in AWS Glue nella Glue Developer Guide.AWS

- Per i dettagli sull'API, consulta CreateJob AWS CLICommand Reference.

`create-table`

Il seguente esempio di codice mostra come utilizzare`create-table`.

AWS CLI

Esempio 1: come creare una tabella per un flusso di dati Kinesis

L'`create-table`esempio seguente crea una tabella nel AWS Glue Data Catalog che descrive un flusso di dati Kinesis.

```
aws glue create-table \
    --database-name tempdb \
    --table-input  '{"Name":"test-kinesis-input", "StorageDescriptor":{ \
            "Columns":[ \
                {"Name":"sensorid", "Type":"int"}, \
                {"Name":"currenttemperature", "Type":"int"}, \
                {"Name":"status", "Type":"string"} \
            ], \
            "Location":"my-testing-stream", \
            "Parameters":{ \
                "typeOfData":"kinesis","streamName":"my-testing-stream", \
                "kinesisUrl":"https://kinesis.us-east-1.amazonaws.com" \
            }, \
            "SerdeInfo":{ \
                "SerializationLibrary":"org.openx.data.jsonserde.JsonSerDe"} \
        }, \
        "Parameters":{ \
            "classification":"json"} \
        }' \
    --profile my-profile \
    --endpoint https://glue.us-east-1.amazonaws.com
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Definizione delle tabelle nel AWS Glue Data Catalog nella AWS Glue Developer Guide.

Esempio 2: come creare una tabella per un datastore Kafka

L'`create-table`esempio seguente crea una tabella nel AWS Glue Data Catalog che descrive un data store Kafka.

```
aws glue create-table \
    --database-name tempdb \
    --table-input  '{"Name":"test-kafka-input", "StorageDescriptor":{ \
            "Columns":[ \
                {"Name":"sensorid", "Type":"int"}, \
                {"Name":"currenttemperature", "Type":"int"}, \
                {"Name":"status", "Type":"string"} \
            ], \
            "Location":"glue-topic", \
            "Parameters":{ \
                "typeOfData":"kafka","topicName":"glue-topic", \
                "connectionName":"my-kafka-connection" \
            }, \
            "SerdeInfo":{ \
                "SerializationLibrary":"org.apache.hadoop.hive.serde2.OpenCSVSerde"}
 \
        }, \
        "Parameters":{ \
            "separatorChar":","} \
        }' \
    --profile my-profile \
    --endpoint https://glue.us-east-1.amazonaws.com
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Definizione delle tabelle nel AWS Glue Data Catalog nella AWS Glue Developer Guide.

Esempio 3: Per creare una tabella per un data AWS store S3

L'`create-table`esempio seguente crea una tabella nel AWS Glue Data Catalog che descrive un data store AWS Simple Storage Service (AWS S3).

```
aws glue create-table \
    --database-name tempdb \
    --table-input  '{"Name":"s3-output", "StorageDescriptor":{ \
            "Columns":[ \
                {"Name":"s1", "Type":"string"}, \
                {"Name":"s2", "Type":"int"}, \
```

```
                {"Name":"s3", "Type":"string"}
            ], \
            "Location":"s3://bucket-path/", \
            "SerdeInfo":{ \
                "SerializationLibrary":"org.openx.data.jsonserde.JsonSerDe"} \
        }, \
        "Parameters":{ \
            "classification":"json"} \
        }' \
    --profile my-profile \
    --endpoint https://glue.us-east-1.amazonaws.com
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Definizione delle tabelle nel AWS Glue Data Catalog nella AWS Glue Developer Guide.

- Per i dettagli sulle API, consulta CreateTable AWS CLICommand Reference.

`delete-job`

Il seguente esempio di codice mostra come utilizzare`delete-job`.

AWS CLI

Come eliminare un processo

L'esempio `delete-job` seguente elimina un processo non più necessario.

```
aws glue delete-job \
    --job-name my-testing-job
```

Output:

```
{
    "JobName": "my-testing-job"
}
```

Per ulteriori informazioni, consulta Working with Jobs on the AWS Glue Console nella AWS Glue Developer Guide.

- Per i dettagli sull'API, consulta DeleteJob AWS CLICommand Reference.

`get-databases`

Il seguente esempio di codice mostra come utilizzare`get-databases`.

AWS CLI

Per elencare le definizioni di alcuni o tutti i database del AWS Glue Data Catalog

L'esempio `get-databases` seguente restituisce informazioni sui database del Catalogo dati.

```
aws glue get-databases
```

Output:

```
{
    "DatabaseList": [
        {
            "Name": "default",
            "Description": "Default Hive database",
            "LocationUri": "file:/spark-warehouse",
            "CreateTime": 1602084052.0,
            "CreateTableDefaultPermissions": [
                {
                    "Principal": {
                        "DataLakePrincipalIdentifier": "IAM_ALLOWED_PRINCIPALS"
                    },
                    "Permissions": [
                        "ALL"
                    ]
                }
            ],
            "CatalogId": "111122223333"
        },
        {
            "Name": "flights-db",
            "CreateTime": 1587072847.0,
            "CreateTableDefaultPermissions": [
                {
                    "Principal": {
                        "DataLakePrincipalIdentifier": "IAM_ALLOWED_PRINCIPALS"
                    },
                    "Permissions": [
                        "ALL"
```

```
                    ]
                }
            ],
            "CatalogId": "111122223333"
        },
        {
            "Name": "legislators",
            "CreateTime": 1601415625.0,
            "CreateTableDefaultPermissions": [
                {
                    "Principal": {
                        "DataLakePrincipalIdentifier": "IAM_ALLOWED_PRINCIPALS"
                    },
                    "Permissions": [
                        "ALL"
                    ]
                }
            ],
            "CatalogId": "111122223333"
        },
        {
            "Name": "tempdb",
            "CreateTime": 1601498566.0,
            "CreateTableDefaultPermissions": [
                {
                    "Principal": {
                        "DataLakePrincipalIdentifier": "IAM_ALLOWED_PRINCIPALS"
                    },
                    "Permissions": [
                        "ALL"
                    ]
                }
            ],
            "CatalogId": "111122223333"
        }
    ]
}
```

Per ulteriori informazioni, consulta Definizione di un database nel catalogo dati nella Guida per gli sviluppatori di AWS Glue.

- Per i dettagli sulle API, consultate GetDatabases AWS CLICommand Reference.

`get-job-run`

Il seguente esempio di codice mostra come utilizzare`get-job-run`.

AWS CLI

Come ottenere informazioni relative all'esecuzione di un processo

L'esempio `get-job-run` seguente recupera le informazioni relative all'esecuzione di un processo.

```
aws glue get-job-run \
    --job-name "Combine legistators data" \
    --run-id jr_012e176506505074d94d761755e5c62538ee1aad6f17d39f527e9140cf0c9a5e
```

Output:

```
{
    "JobRun": {
        "Id": "jr_012e176506505074d94d761755e5c62538ee1aad6f17d39f527e9140cf0c9a5e",
        "Attempt": 0,
        "JobName": "Combine legistators data",
        "StartedOn": 1602873931.255,
        "LastModifiedOn": 1602874075.985,
        "CompletedOn": 1602874075.985,
        "JobRunState": "SUCCEEDED",
        "Arguments": {
            "--enable-continuous-cloudwatch-log": "true",
            "--enable-metrics": "",
            "--enable-spark-ui": "true",
            "--job-bookmark-option": "job-bookmark-enable",
            "--spark-event-logs-path": "s3://aws-glue-assets-111122223333-us-east-1/
sparkHistoryLogs/"
        },
        "PredecessorRuns": [],
        "AllocatedCapacity": 10,
        "ExecutionTime": 117,
        "Timeout": 2880,
        "MaxCapacity": 10.0,
        "WorkerType": "G.1X",
        "NumberOfWorkers": 10,
        "LogGroupName": "/aws-glue/jobs",
        "GlueVersion": "2.0"
```

```
    }
}
```

Per ulteriori informazioni, consulta Esecuzioni di processi nella Guida per gli sviluppatori di AWS Glue.

- Per i dettagli sull'API, consulta GetJobRun AWS CLICommand Reference.

`get-job-runs`

Il seguente esempio di codice mostra come utilizzare`get-job-runs`.

AWS CLI

Come ottenere informazioni su tutte le esecuzioni di processo per un determinato processo

L'esempio `get-job-runs` seguente recupera informazioni sulle esecuzioni di processo per un determinato processo.

```
aws glue get-job-runs \
    --job-name "my-testing-job"
```

Output:

```
{
    "JobRuns": [
        {
            "Id":
 "jr_012e176506505074d94d761755e5c62538ee1aad6f17d39f527e9140cf0c9a5e",
            "Attempt": 0,
            "JobName": "my-testing-job",
            "StartedOn": 1602873931.255,
            "LastModifiedOn": 1602874075.985,
            "CompletedOn": 1602874075.985,
            "JobRunState": "SUCCEEDED",
            "Arguments": {
                "--enable-continuous-cloudwatch-log": "true",
                "--enable-metrics": "",
                "--enable-spark-ui": "true",
                "--job-bookmark-option": "job-bookmark-enable",
                "--spark-event-logs-path": "s3://aws-glue-assets-111122223333-us-
east-1/sparkHistoryLogs/"
            },
```

```
            "PredecessorRuns": [],
            "AllocatedCapacity": 10,
            "ExecutionTime": 117,
            "Timeout": 2880,
            "MaxCapacity": 10.0,
            "WorkerType": "G.1X",
            "NumberOfWorkers": 10,
            "LogGroupName": "/aws-glue/jobs",
            "GlueVersion": "2.0"
        },
        {
            "Id":
 "jr_03cc19ddab11c4e244d3f735567de74ff93b0b3ef468a713ffe73e53d1aec08f_attempt_2",
            "Attempt": 2,
            "PreviousRunId":
 "jr_03cc19ddab11c4e244d3f735567de74ff93b0b3ef468a713ffe73e53d1aec08f_attempt_1",
            "JobName": "my-testing-job",
            "StartedOn": 1602811168.496,
            "LastModifiedOn": 1602811282.39,
            "CompletedOn": 1602811282.39,
            "JobRunState": "FAILED",
            "ErrorMessage": "An error occurred while calling
 o122.pyWriteDynamicFrame.
                Access Denied (Service: Amazon S3; Status Code: 403; Error Code:
 AccessDenied;
                Request ID: 021AAB703DB20A2D;
                S3 Extended Request ID: teZk24Y09TkXzBvMPG502L5VJBhe9DJuWA9/
TXtuGOqfByajkfL/Tlqt5JBGdEGpigAqzdMDM/U=)",
            "PredecessorRuns": [],
            "AllocatedCapacity": 10,
            "ExecutionTime": 110,
            "Timeout": 2880,
            "MaxCapacity": 10.0,
            "WorkerType": "G.1X",
            "NumberOfWorkers": 10,
            "LogGroupName": "/aws-glue/jobs",
            "GlueVersion": "2.0"
        },
        {
            "Id":
 "jr_03cc19ddab11c4e244d3f735567de74ff93b0b3ef468a713ffe73e53d1aec08f_attempt_1",
            "Attempt": 1,
            "PreviousRunId":
 "jr_03cc19ddab11c4e244d3f735567de74ff93b0b3ef468a713ffe73e53d1aec08f",
```

```
            "JobName": "my-testing-job",
            "StartedOn": 1602811020.518,
            "LastModifiedOn": 1602811138.364,
            "CompletedOn": 1602811138.364,
            "JobRunState": "FAILED",
            "ErrorMessage": "An error occurred while calling
 o122.pyWriteDynamicFrame.
                Access Denied (Service: Amazon S3; Status Code: 403; Error Code:
 AccessDenied;
                Request ID: 2671D37856AE7ABB;
                S3 Extended Request ID: RLJCJw20brV
+PpC6GpORahyF2fp9flB5SSb2bTGPnUSPVizLXRl1PN3QZldb+v1o9qRVktNYbW8=)",
            "PredecessorRuns": [],
            "AllocatedCapacity": 10,
            "ExecutionTime": 113,
            "Timeout": 2880,
            "MaxCapacity": 10.0,
            "WorkerType": "G.1X",
            "NumberOfWorkers": 10,
            "LogGroupName": "/aws-glue/jobs",
            "GlueVersion": "2.0"
        }
    ]
}
```

Per ulteriori informazioni, consulta Esecuzioni di processi nella Guida per gli sviluppatori di AWS Glue.

- Per i dettagli sull'API, consulta GetJobRuns AWS CLICommand Reference.

`get-job`

Il seguente esempio di codice mostra come utilizzare`get-job`.

AWS CLI

Come recuperare le informazioni relative a un processo

L'esempio `get-job` seguente recupera le informazioni relative a un processo.

```
aws glue get-job \
    --job-name my-testing-job
```

Output:

```
{
    "Job": {
        "Name": "my-testing-job",
        "Role": "Glue_DefaultRole",
        "CreatedOn": 1602805698.167,
        "LastModifiedOn": 1602805698.167,
        "ExecutionProperty": {
            "MaxConcurrentRuns": 1
        },
        "Command": {
            "Name": "gluestreaming",
            "ScriptLocation": "s3://janetst-bucket-01/Scripts/test_script.scala",
            "PythonVersion": "2"
        },
        "DefaultArguments": {
            "--class": "GlueApp",
            "--job-language": "scala"
        },
        "MaxRetries": 0,
        "AllocatedCapacity": 10,
        "MaxCapacity": 10.0,
        "GlueVersion": "1.0"
    }
}
```

Per ulteriori informazioni, consulta Processi nella Guida per gli sviluppatori di AWS Glue.

- Per i dettagli sull'API, consulta GetJob AWS CLICommand Reference.

`get-plan`

Il seguente esempio di codice mostra come utilizzare`get-plan`.

AWS CLI

Come ottenere il codice generato per mappare i dati dalle tabelle di origine alle tabelle di destinazione

Il comando `get-plan` seguente recupera il codice generato per mappare le colonne dall'origine dati alla destinazione dei dati.

```
aws glue get-plan --mapping '[ \
    { \
        "SourcePath":"sensorid", \
        "SourceTable":"anything", \
        "SourceType":"int", \
        "TargetPath":"sensorid", \
        "TargetTable":"anything", \
        "TargetType":"int" \
    }, \
    { \
        "SourcePath":"currenttemperature", \
        "SourceTable":"anything", \
        "SourceType":"int", \
        "TargetPath":"currenttemperature", \
        "TargetTable":"anything", \
        "TargetType":"int" \
    }, \
    { \
        "SourcePath":"status", \
        "SourceTable":"anything", \
        "SourceType":"string", \
        "TargetPath":"status", \
        "TargetTable":"anything", \
        "TargetType":"string" \
    }]' \
    --source '{ \
        "DatabaseName":"tempdb", \
        "TableName":"s3-source" \
    }' \
    --sinks '[ \
        { \
            "DatabaseName":"tempdb", \
            "TableName":"my-s3-sink" \
        }]' \
    --language "scala" \
    --endpoint https://glue.us-east-1.amazonaws.com \
    --output "text"
```

Output:

```
import com.amazonaws.services.glue.ChoiceOption
import com.amazonaws.services.glue.GlueContext
import com.amazonaws.services.glue.MappingSpec
```

```
import com.amazonaws.services.glue.ResolveSpec
import com.amazonaws.services.glue.errors.CallSite
import com.amazonaws.services.glue.util.GlueArgParser
import com.amazonaws.services.glue.util.Job
import com.amazonaws.services.glue.util.JsonOptions
import org.apache.spark.SparkContext
import scala.collection.JavaConverters._

object GlueApp {
  def main(sysArgs: Array[String]) {
    val spark: SparkContext = new SparkContext()
    val glueContext: GlueContext = new GlueContext(spark)
    // @params: [JOB_NAME]
    val args = GlueArgParser.getResolvedOptions(sysArgs, Seq("JOB_NAME").toArray)
    Job.init(args("JOB_NAME"), glueContext, args.asJava)
    // @type: DataSource
    // @args: [database = "tempdb", table_name = "s3-source", transformation_ctx =
 "datasource0"]
    // @return: datasource0
    // @inputs: []
    val datasource0 = glueContext.getCatalogSource(database = "tempdb",
 tableName = "s3-source", redshiftTmpDir = "", transformationContext =
 "datasource0").getDynamicFrame()
    // @type: ApplyMapping
    // @args: [mapping = [("sensorid", "int", "sensorid", "int"),
 ("currenttemperature", "int", "currenttemperature", "int"), ("status", "string",
 "status", "string")], transformation_ctx = "applymapping1"]
    // @return: applymapping1
    // @inputs: [frame = datasource0]
    val applymapping1 = datasource0.applyMapping(mappings = Seq(("sensorid",
 "int", "sensorid", "int"), ("currenttemperature", "int", "currenttemperature",
 "int"), ("status", "string", "status", "string")), caseSensitive = false,
 transformationContext = "applymapping1")
    // @type: SelectFields
    // @args: [paths = ["sensorid", "currenttemperature", "status"],
 transformation_ctx = "selectfields2"]
    // @return: selectfields2
    // @inputs: [frame = applymapping1]
    val selectfields2 = applymapping1.selectFields(paths = Seq("sensorid",
 "currenttemperature", "status"), transformationContext = "selectfields2")
    // @type: ResolveChoice
    // @args: [choice = "MATCH_CATALOG", database = "tempdb", table_name = "my-s3-
sink", transformation_ctx = "resolvechoice3"]
    // @return: resolvechoice3
```

```
    // @inputs: [frame = selectfields2]
    val resolvechoice3 = selectfields2.resolveChoice(choiceOption =
 Some(ChoiceOption("MATCH_CATALOG")), database = Some("tempdb"), tableName =
 Some("my-s3-sink"), transformationContext = "resolvechoice3")
    // @type: DataSink
    // @args: [database = "tempdb", table_name = "my-s3-sink", transformation_ctx =
 "datasink4"]
    // @return: datasink4
    // @inputs: [frame = resolvechoice3]
    val datasink4 = glueContext.getCatalogSink(database = "tempdb",
 tableName = "my-s3-sink", redshiftTmpDir = "", transformationContext =
 "datasink4").writeDynamicFrame(resolvechoice3)
    Job.commit()
  }
}
```

Per ulteriori informazioni, consulta Modifica degli script in AWS Glue nella Glue Developer Guide.AWS

- Per i dettagli sulle API, consulta GetPlan AWS CLICommand Reference.

`get-tables`

Il seguente esempio di codice mostra come utilizzare`get-tables`.

AWS CLI

Come elencare le definizioni di alcune o tutte le tabelle del database specificato

L'esempio `get-tables` seguente restituisce le informazioni relative alle tabelle del database specificato.

```
aws glue get-tables --database-name 'tempdb'
```

Output:

```
{
    "TableList": [
        {
            "Name": "my-s3-sink",
            "DatabaseName": "tempdb",
            "CreateTime": 1602730539.0,
            "UpdateTime": 1602730539.0,
```

```
            "Retention": 0,
            "StorageDescriptor": {
                "Columns": [
                    {
                        "Name": "sensorid",
                        "Type": "int"
                    },
                    {
                        "Name": "currenttemperature",
                        "Type": "int"
                    },
                    {
                        "Name": "status",
                        "Type": "string"
                    }
                ],
                "Location": "s3://janetst-bucket-01/test-s3-output/",
                "Compressed": false,
                "NumberOfBuckets": 0,
                "SerdeInfo": {
                    "SerializationLibrary": "org.openx.data.jsonserde.JsonSerDe"
                },
                "SortColumns": [],
                "StoredAsSubDirectories": false
            },
            "Parameters": {
                "classification": "json"
            },
            "CreatedBy": "arn:aws:iam::007436865787:user/JRSTERN",
            "IsRegisteredWithLakeFormation": false,
            "CatalogId": "007436865787"
        },
        {
            "Name": "s3-source",
            "DatabaseName": "tempdb",
            "CreateTime": 1602730658.0,
            "UpdateTime": 1602730658.0,
            "Retention": 0,
            "StorageDescriptor": {
                "Columns": [
                    {
                        "Name": "sensorid",
                        "Type": "int"
                    },
```

```
                {
                    "Name": "currenttemperature",
                    "Type": "int"
                },
                {
                    "Name": "status",
                    "Type": "string"
                }
            ],
            "Location": "s3://janetst-bucket-01/",
            "Compressed": false,
            "NumberOfBuckets": 0,
            "SortColumns": [],
            "StoredAsSubDirectories": false
        },
        "Parameters": {
            "classification": "json"
        },
        "CreatedBy": "arn:aws:iam::007436865787:user/JRSTERN",
        "IsRegisteredWithLakeFormation": false,
        "CatalogId": "007436865787"
    },
    {
        "Name": "test-kinesis-input",
        "DatabaseName": "tempdb",
        "CreateTime": 1601507001.0,
        "UpdateTime": 1601507001.0,
        "Retention": 0,
        "StorageDescriptor": {
            "Columns": [
                {
                    "Name": "sensorid",
                    "Type": "int"
                },
                {
                    "Name": "currenttemperature",
                    "Type": "int"
                },
                {
                    "Name": "status",
                    "Type": "string"
                }
            ],
            "Location": "my-testing-stream",
```

```
                "Compressed": false,
                "NumberOfBuckets": 0,
                "SerdeInfo": {
                    "SerializationLibrary": "org.openx.data.jsonserde.JsonSerDe"
                },
                "SortColumns": [],
                "Parameters": {
                    "kinesisUrl": "https://kinesis.us-east-1.amazonaws.com",
                    "streamName": "my-testing-stream",
                    "typeOfData": "kinesis"
                },
                "StoredAsSubDirectories": false
            },
            "Parameters": {
                "classification": "json"
            },
            "CreatedBy": "arn:aws:iam::007436865787:user/JRSTERN",
            "IsRegisteredWithLakeFormation": false,
            "CatalogId": "007436865787"
        }
    ]
}
```

Per ulteriori informazioni, consulta Definizione delle tabelle nel AWS Glue Data Catalog nella AWS Glue Developer Guide.

- Per i dettagli sulle API, consulta GetTables AWS CLICommand Reference.

`start-crawler`

Il seguente esempio di codice mostra come utilizzare`start-crawler`.

AWS CLI

Come avviare un crawler

L'esempio `start-crawler` seguente avvia un crawler.

```
aws glue start-crawler --name my-crawler
```

Output:

```
None
```

Per ulteriori informazioni, consulta Definizione di crawler nella Guida per gli sviluppatori di AWS Glue.

- Per i dettagli sull'API, consulta StartCrawler AWS CLICommand Reference.

`start-job-run`

Il seguente esempio di codice mostra come utilizzare`start-job-run`.

AWS CLI

Come avviare l'esecuzione di un processo

L'esempio `start-job-run` seguente avvia un processo.

```
aws glue start-job-run \
    --job-name my-job
```

Output:

```
{
    "JobRunId":
 "jr_22208b1f44eb5376a60569d4b21dd20fcb8621e1a366b4e7b2494af764b82ded"
}
```

Per ulteriori informazioni, consulta Creazione di processi nella Guida per gli sviluppatori di AWS Glue.

- Per i dettagli sull'API, consulta StartJobRun AWS CLICommand Reference.

GuardDuty esempi utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with GuardDuty.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`accept-invitation`

Il seguente esempio di codice mostra come utilizzare`accept-invitation`.

AWS CLI

Accettare un invito a diventare un account GuardDuty membro nella regione corrente

L'`accept-invitation`esempio seguente mostra come accettare un invito a diventare un account GuardDuty membro nella regione corrente.

```
aws guardduty accept-invitation  \
    --detector-id 12abc34d567e8fa901bc2d34eexample \
    --master-id 123456789111 \
    --invitation-id d6b94fb03a66ff665f7db8764example
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Gestire GuardDuty gli account su invito nella Guida GuardDuty per l'utente.

- Per i dettagli sull'API, consulta AcceptInvitation AWS CLICommand Reference.

`archive-findings`

Il seguente esempio di codice mostra come utilizzare`archive-findings`.

AWS CLI

Come archiviare gli esiti nella Regione corrente

In questo esempio `archive-findings` viene mostrato come archiviare gli esiti nella Regione corrente.

```
aws guardduty archive-findings \
    --detector-id 12abc34d567e8fa901bc2d34eexample \
    --finding-ids d6b94fb03a66ff665f7db8764example 3eb970e0de00c16ec14e6910fexample
```

Questo comando non produce alcun output.

Per ulteriori informazioni, vedere Creazione di regole di soppressione nella Guida per l'GuardDuty utente.

- Per i dettagli sull'API, consulta ArchiveFindings AWS CLICommand Reference.

`create-detector`

Il seguente esempio di codice mostra come utilizzare`create-detector`.

AWS CLI

Da abilitare GuardDuty nella regione corrente

Questo esempio mostra come creare un nuovo rilevatore, che abiliti GuardDuty, nella regione corrente. :

```
aws guardduty create-detector \
    --enable
```

Output:

```
{
    "DetectorId": "b6b992d6d2f48e64bc59180bfexample"
}
```

Per ulteriori informazioni, consulta Enable Amazon GuardDuty nella GuardDuty User Guide.

- Per i dettagli sull'API, consulta CreateDetector AWS CLICommand Reference.

`create-filter`

Il seguente esempio di codice mostra come utilizzare`create-filter`.

AWS CLI

Esempio 1: come creare un nuovo filtro nella Regione corrente

L'esempio `create-filter` seguente crea un filtro che corrisponde a tutti gli esiti Portscan per l'istanza creata da un'immagine specifica. Tali esiti non vengono soppressi.

```
aws guardduty create-filter \
    --detector-id b6b992d6d2f48e64bc59180bfexample \
    --name myFilterExample \
    --finding-criteria '{"Criterion": {"type": {"Eq": ["Recon:EC2/
Portscan"]},"resource.instanceDetails.imageId": {"Eq": ["ami-0a7a207083example"]}}}'
```

Output:

```
{
    "Name": "myFilterExample"
}
```

Per ulteriori informazioni, vedere Filtraggio dei GuardDuty risultati nella Guida per l'GuardDuty utente.

Esempio 2: come creare un nuovo filtro ed eliminare gli esiti nella Regione corrente

L'esempio `create-filter` seguente crea un filtro che corrisponde a tutti gli esiti Portscan per l'istanza creata da un'immagine specifica. Questo filtro archivia tali esiti in modo che non compaiano negli esiti correnti.

```
aws guardduty create-filter \
    --detector-id b6b992d6d2f48e64bc59180bfexample \
    --action ARCHIVE \
    --name myFilterSecondExample \
    --finding-criteria '{"Criterion": {"type": {"Eq": ["Recon:EC2/
Portscan"]},"resource.instanceDetails.imageId": {"Eq": ["ami-0a7a207083example"]}}}'
```

Output:

```
{
    "Name": "myFilterSecondExample"
}
```

Per ulteriori informazioni, consulta Filtraggio dei GuardDuty risultati nella Guida per l'GuardDuty utente.

- Per i dettagli sull'API, consulta CreateFilter AWS CLICommand Reference.

`create-ip-set`

Il seguente esempio di codice mostra come utilizzare`create-ip-set`.

AWS CLI

Come creare e attivare un set di IP attendibili

L'esempio `create-ip-set` seguente crea e attiva un set di IP attendibili nella Regione corrente.

```
aws guardduty create-ip-set \
    --detector-id 12abc34d567e8fa901bc2d34eexample \
    --name new-ip-set-example \
    --format TXT \
    --location s3://amzn-s3-demo-bucket/customtrustlist.csv \
    --activate
```

Output:

```
{
    "IpSetId": "d4b94fc952d6912b8f3060768example"
}
```

Per ulteriori informazioni, vedere Utilizzo degli elenchi di IP affidabili e degli elenchi di minacce nella Guida GuardDuty per l'utente.

- Per i dettagli sulle API, consulta CreateIpSet AWS CLICommand Reference.

`create-members`

Il seguente esempio di codice mostra come utilizzare`create-members`.

AWS CLI

Per associare un nuovo membro al tuo account GuardDuty principale nella regione corrente.

Questo esempio mostra come associare gli account dei membri che devono essere gestiti dall'account corrente come GuardDuty master.

```
aws guardduty create-members
    --detector-id b6b992d6d2f48e64bc59180bfexample \
```

```
    --account-details AccountId=111122223333,Email=first
+member@example.com AccountId=111111111111 ,Email=another+member@example.com
```

Output:

```
{
   "UnprocessedAccounts": []
}
```

Per ulteriori informazioni, consulta Gestione di più account nella Guida GuardDuty per l'utente.

- Per i dettagli sull'API, consulta CreateMembers AWS CLICommand Reference.

`create-publishing-destination`

Il seguente esempio di codice mostra come utilizzare`create-publishing-destination`.

AWS CLI

Per creare una destinazione di pubblicazione in cui esportare i GuardDuty risultati nella regione corrente.

L'`create-publishing-destination`esempio seguente mostra come impostare una destinazione di pubblicazione per esportare i GuardDuty risultati correnti (non archiviati) per tenere traccia dei dati storici dei risultati.

```
aws guardduty create-publishing-destination \
    --detector-id b6b992d6d2f48e64bc59180bfexample \
    --destination-type S3 \
    --destination-properties 'DestinationArn=arn:aws:s3:::amzn-s3-demo-
bucket,KmsKeyArn=arn:aws:kms:us-west-1:111122223333:key/84cee9c5-dea1-401a-ab6d-
e1de7example'
```

Output:

```
{
    "DestinationId": "46b99823849e1bbc242dfbe3cexample"
}
```

Per ulteriori informazioni, consulta Esportazione dei GuardDuty risultati generati nei bucket Amazon S3 nella GuardDuty Guida per l'utente.

- Per i dettagli sull'API, consulta Command CreatePublishingDestinationReference AWS CLI .

`create-sample-findings`

Il seguente esempio di codice mostra come utilizzare`create-sample-findings`.

AWS CLI

Per creare GuardDuty risultati di esempio nella regione corrente.

Questo esempio mostra come creare un esito di esempio dei tipi forniti.

```
aws guardduty create-sample-findings \
    --detector-id b6b992d6d2f48e64bc59180bfexample \
    --finding-types UnauthorizedAccess:EC2/TorClient UnauthorizedAccess:EC2/TorRelay
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Esempi di risultati nella Guida GuardDuty per l'utente.

- Per i dettagli sull'API, consulta CreateSampleFindings AWS CLICommand Reference.

`create-threat-intel-set`

Il seguente esempio di codice mostra come utilizzare`create-threat-intel-set`.

AWS CLI

Come creare e attivare un nuovo set di informazioni sulle minacce

L'esempio `create-threat-intel-set` seguente crea e attiva un set di intelligence delle minacce nella Regione corrente.

```
aws guardduty create-threat-intel-set \
    --detector-id b6b992d6d2f48e64bc59180bfexample \
    --name myThreatSet-example \
    --format TXT \
    --location s3://amzn-s3-demo-bucket/threatlist.csv \
    --activate
```

Output:

```
{
    "ThreatIntelSetId": "20b9a4691aeb33506b808878cexample"
}
```

Per ulteriori informazioni, vedere Utilizzo degli elenchi di IP affidabili e degli elenchi di minacce nella Guida GuardDuty per l'utente.

- Per i dettagli sulle API, consulta CreateThreatIntelSet AWS CLICommand Reference.

`decline-invitations`

Il seguente esempio di codice mostra come utilizzare`decline-invitations`.

AWS CLI

Come rifiutare un invito a far gestire GuardDuty da un altro account nella Regione corrente.

Questo esempio mostra come rifiutare un invito all'iscrizione.

```
aws guardduty decline-invitations \
    --account-ids 111122223333
```

Output:

```
{
    "UnprocessedAccounts": []
}
```

Per ulteriori informazioni, vedere Gestione GuardDuty degli account su invito nella Guida GuardDuty per l'utente.

- Per i dettagli sull'API, consulta DeclineInvitations AWS CLICommand Reference.

`delete-detector`

Il seguente esempio di codice mostra come utilizzare`delete-detector`.

AWS CLI

Per eliminare e disattivare GuardDuty un rilevatore nella regione corrente.

Questo esempio mostra come eliminare un rilevatore. In caso di successo, questa operazione verrà disattivata GuardDuty nella regione associata a quel rilevatore.

```
aws guardduty delete-detector \
    --detector-id b6b992d6d2f48e64bc59180bfexample
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consultate Sospensione o disabilitazione GuardDuty nella Guida per l'utente. GuardDuty

- Per i dettagli sull'API, consulta AWS CLI Command DeleteDetectorReference.

`delete-filter`

Il seguente esempio di codice mostra come utilizzare`delete-filter`.

AWS CLI

Come eliminare un filtro esistente nella Regione corrente

In questo esempio viene mostrato come creare ed eliminare un filtro.

```
aws guardduty delete-filter \
    --detector-id b6b992d6d2f48e64bc59180bfexample \
    --filter-name byebyeFilter
```

Questo comando non produce alcun output.

Per ulteriori informazioni, vedere Filtraggio dei risultati nella Guida per l' GuardDuty utente.

- Per i dettagli sull'API, consulta DeleteFilter AWS CLICommand Reference.

`disable-organization-admin-account`

Il seguente esempio di codice mostra come utilizzare`disable-organization-admin-account`.

AWS CLI

Per rimuovere un account come amministratore delegato all' GuardDuty interno dell'organizzazione

Questo esempio mostra come rimuovere un account come amministratore delegato di. GuardDuty

```
aws guardduty disable-organization-admin-account \
    --admin-account-id 111122223333
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Gestire gli account con AWS le organizzazioni nella Guida per l'GuardDuty utente.

- Per i dettagli sull'API, consulta DisableOrganizationAdminAccount AWS CLICommand Reference.

`disassociate-from-master-account`

Il seguente esempio di codice mostra come utilizzare`disassociate-from-master-account`.

AWS CLI

Come annullare l'associazione dal tuo attuale account amministratore nella Regione corrente

L'`disassociate-from-master-account`esempio seguente dissocia l'account dell'utente dall'account GuardDuty amministratore corrente nell'area corrente. AWS

```
aws guardduty disassociate-from-master-account \
    --detector-id d4b040365221be2b54a6264dcexample
```

Questo comando non produce alcun output.

Per ulteriori informazioni, vedere Comprensione della relazione tra account GuardDuty amministratore e account membro nella Guida per l'GuardDuty utente.

- Per i dettagli sull'API, consulta DisassociateFromMasterAccount AWS CLICommand Reference.

`get-detector`

Il seguente esempio di codice mostra come utilizzare`get-detector`.

AWS CLI

Come recuperare i dettagli di un rilevatore specifico

L'esempio `get-detector` seguente visualizza i dettagli delle configurazioni del rilevatore specificato.

```
aws guardduty get-detector \
    --detector-id 12abc34d567e8fa901bc2d34eexample
```

Output:

```
{
    "Status": "ENABLED",
    "ServiceRole": "arn:aws:iam::111122223333:role/aws-service-role/
guardduty.amazonaws.com/AWSServiceRoleForAmazonGuardDuty",
    "Tags": {},
    "FindingPublishingFrequency": "SIX_HOURS",
    "UpdatedAt": "2018-11-07T03:24:22.938Z",
    "CreatedAt": "2017-12-22T22:51:31.940Z"
}
```

Per ulteriori informazioni, vedere Concetti e terminologia nella Guida per l' GuardDuty utente.

- Per i dettagli sull'API, consulta GetDetector AWS CLICommand Reference.

`get-findings`

Il seguente esempio di codice mostra come utilizzare`get-findings`.

AWS CLI

Esempio 1: come recuperare i dettagli di un esito specifico

L'esempio `get-findings` seguente recupera i dettagli completi degli esiti JSON dell'esito specificato.

```
aws guardduty get-findings \
    --detector-id 12abc34d567e8fa901bc2d34eexample \
    --finding-id 1ab92989eaf0e742df4a014d5example
```

Output:

```
{
    "Findings": [
        {
```

```
            "Resource": {
                "ResourceType": "AccessKey",
                "AccessKeyDetails": {
                    "UserName": "testuser",
                    "UserType": "IAMUser",
                    "PrincipalId": "AIDACKCEVSQ6C2EXAMPLE",
                    "AccessKeyId": "ASIASZ4SI7REEEXAMPLE"
                }
            },
            "Description": "APIs commonly used to discover the users, groups,
 policies and permissions in an account, was invoked by IAM principal testuser under
 unusual circumstances. Such activity is not typically seen from this principal.",
            "Service": {
                "Count": 5,
                "Archived": false,
                "ServiceName": "guardduty",
                "EventFirstSeen": "2020-05-26T22:02:24Z",
                "ResourceRole": "TARGET",
                "EventLastSeen": "2020-05-26T22:33:55Z",
                "DetectorId": "d4b040365221be2b54a6264dcexample",
                "Action": {
                    "ActionType": "AWS_API_CALL",
                    "AwsApiCallAction": {
                        "RemoteIpDetails": {
                            "GeoLocation": {
                                "Lat": 51.5164,
                                "Lon": -0.093
                            },
                            "City": {
                                "CityName": "London"
                            },
                            "IpAddressV4": "52.94.36.7",
                            "Organization": {
                                "Org": "Amazon.com",
                                "Isp": "Amazon.com",
                                "Asn": "16509",
                                "AsnOrg": "AMAZON-02"
                            },
                            "Country": {
                                "CountryName": "United Kingdom"
                            }
                        },
                        "Api": "ListPolicyVersions",
                        "ServiceName": "iam.amazonaws.com",
```

```
                        "CallerType": "Remote IP"
                    }
                }
            },
            "Title": "Unusual user permission reconnaissance activity by testuser.",
            "Type": "Recon:IAMUser/UserPermissions",
            "Region": "us-east-1",
            "Partition": "aws",
            "Arn": "arn:aws:guardduty:us-east-1:111122223333:detector/
d4b040365221be2b54a6264dcexample/finding/1ab92989eaf0e742df4a014d5example",
            "UpdatedAt": "2020-05-26T22:55:21.703Z",
            "SchemaVersion": "2.0",
            "Severity": 5,
            "Id": "1ab92989eaf0e742df4a014d5example",
            "CreatedAt": "2020-05-26T22:21:48.385Z",
            "AccountId": "111122223333"
        }
    ]
}
```

Per ulteriori informazioni, vedere Findings in the GuardDuty User Guide.

- Per i dettagli sull'API, consulta GetFindings AWS CLICommand Reference.

`get-ip-set`

Il seguente esempio di codice mostra come utilizzare`get-ip-set`.

AWS CLI

Come elencare i dettagli ottenuti su un set di IP attendibili specificato

L'esempio `get-ip-set` seguente mostra lo stato e i dettagli del set di IP attendibili specificato.

```
aws guardduty get-ip-set \
    --detector-id 12abc34d567e8fa901bc2d34eexample \
    --ip-set-id d4b94fc952d6912b8f3060768example
```

Output:

```
{
    "Status": "ACTIVE",
```

```
    "Location": "s3://amzn-s3-demo-bucket.s3-us-west-2.amazonaws.com/
customlist.csv",
    "Tags": {},
    "Format": "TXT",
    "Name": "test-ip-set-example"
}
```

Per ulteriori informazioni, vedere Utilizzo degli elenchi di IP affidabili e degli elenchi di minacce nella Guida GuardDuty per l'utente.

- Per i dettagli sulle API, consulta GetIpSet AWS CLICommand Reference.

`get-master-account`

Il seguente esempio di codice mostra come utilizzare`get-master-account`.

AWS CLI

Come recuperare i dettagli del tuo account master nella Regione corrente

L'esempio `get-master-account` seguente visualizza lo stato e i dettagli dell'account master associato al rilevatore nella Regione corrente.

```
aws guardduty get-master-account \
    --detector-id 12abc34d567e8fa901bc2d34eexample
```

Output:

```
{
    "Master": {
        "InvitationId": "04b94d9704854a73f94e061e8example",
        "InvitedAt": "2020-06-09T22:23:04.970Z",
        "RelationshipStatus": "Enabled",
        "AccountId": "111122223333"
    }
}
```

Per ulteriori informazioni, vedere Comprensione della relazione tra account GuardDuty amministratore e account membro nella Guida per l'GuardDuty utente.

- Per i dettagli sull'API, consulta GetMasterAccount AWS CLICommand Reference.

`list-detectors`

Il seguente esempio di codice mostra come utilizzare`list-detectors`.

AWS CLI

Come elencare i rilevatori disponibili nella Regione corrente

L'`list-detectors`esempio seguente elenca i rilevatori disponibili nella AWS regione corrente.

```
aws guardduty list-detectors
```

Output:

```
{
    "DetectorIds": [
        "12abc34d567e8fa901bc2d34eexample"
    ]
}
```

Per ulteriori informazioni, vedere Concetti e terminologia nella Guida per l' GuardDuty utente.

- Per i dettagli sull'API, consulta ListDetectors AWS CLICommand Reference.

`list-findings`

Il seguente esempio di codice mostra come utilizzare`list-findings`.

AWS CLI

Esempio 1: come elencare tutti gli esiti per la Regione corrente

L'esempio `list-findings` seguente visualizza un elenco di tutti i findingId per la Regione corrente ordinati per gravità dalla più alta alla più bassa.

```
aws guardduty list-findings \
    --detector-id 12abc34d567e8fa901bc2d34eexample \
    --sort-criteria '{"AttributeName": "severity","OrderBy":"DESC"}'
```

Output:

```
{
```

```
    "FindingIds": [
        "04b8ab50fd29c64fc771b232dexample",
        "5ab8ab50fd21373735c826d3aexample",
        "90b93de7aba69107f05bbe60bexample",
        ...
    ]
}
```

Per ulteriori informazioni, vedere Findings in the GuardDuty User Guide.

Esempio 2: come elencare gli esiti per la Regione corrente corrispondenti a un criterio di ricerca specifico

L'esempio `list-findings` seguente visualizza un elenco di tutti i findingId che corrispondono a un tipo di esito specificato.

```
aws guardduty list-findings \
    --detector-id 12abc34d567e8fa901bc2d34eexample \
    --finding-criteria  '{"Criterion":{"type": {"Eq":["UnauthorizedAccess:EC2/
SSHBruteForce"]}}}'
```

Output:

```
{
    "FindingIds": [
        "90b93de7aba69107f05bbe60bexample",
        "6eb9430d7023d30774d6f05e3example",
        "2eb91a2d060ac9a21963a5848example",
        "44b8ab50fd2b0039a9e48f570example",
        "9eb8ab4cd2b7e5b66ba4f5e96example",
        "e0b8ab3a38e9b0312cc390ceeexample"
    ]
}
```

Per ulteriori informazioni, vedere Findings in the GuardDuty User Guide.

Esempio 3: come elencare gli esiti per la Regione corrente corrispondenti a un set di criteri di ricerca specifico definito in un file JSON

L'esempio `list-findings` seguente visualizza un elenco di tutti i findingId non archiviati e che non coinvolgono l'utente IAM denominato "testuser", come specificato in un file JSON.

```
aws guardduty list-findings \
    --detector-id 12abc34d567e8fa901bc2d34eexample \
    --finding-criteria  file://myfile.json
```

Contenuto di `myfile.json`:

```
{"Criterion": {
    "resource.accessKeyDetails.userName":{
                "Eq":[
                    "testuser"
                    ]
                },
    "service.archived": {
                "Eq": [
                    "false"
                ]
            }
        }
}
```

Output:

```
{
    "FindingIds": [
        "1ab92989eaf0e742df4a014d5example"
    ]
}
```

Per ulteriori informazioni, vedere Findings in the GuardDuty User Guide.

- Per i dettagli sull'API, consulta ListFindings AWS CLICommand Reference.

`list-invitations`

Il seguente esempio di codice mostra come utilizzare`list-invitations`.

AWS CLI

Come elencare i dettagli dei tuoi inviti a diventare account membro nella Regione corrente

L'`list-invitations`esempio seguente elenca i dettagli e lo stato degli inviti a diventare un account GuardDuty membro nella regione corrente.

```
aws guardduty list-invitations
```

Output:

```
{
    "Invitations": [
        {
            "InvitationId": "d6b94fb03a66ff665f7db8764example",
            "InvitedAt": "2020-06-10T17:56:38.221Z",
            "RelationshipStatus": "Invited",
            "AccountId": "123456789111"
        }
    ]
}
```

Per ulteriori informazioni, consulta Gestire GuardDuty gli account tramite invito nella Guida per l' GuardDuty utente.

- Per i dettagli sull'API, consulta ListInvitations AWS CLICommand Reference.

`list-ip-sets`

Il seguente esempio di codice mostra come utilizzare`list-ip-sets`.

AWS CLI

Come elencare i set di IP attendibili nella Regione corrente

L'`list-ip-sets`esempio seguente elenca i set IP affidabili nella AWS regione corrente.

```
aws guardduty list-ip-sets \
    --detector-id 12abc34d567e8fa901bc2d34eexample
```

Output:

```
{
    "IpSetIds": [
        "d4b94fc952d6912b8f3060768example"
    ]
}
```

Per ulteriori informazioni, vedere Utilizzo degli elenchi di IP affidabili e degli elenchi di minacce nella Guida GuardDuty per l'utente.

- Per i dettagli sulle API, consulta ListIpSets AWS CLICommand Reference.

`list-members`

Il seguente esempio di codice mostra come utilizzare`list-members`.

AWS CLI

Esempio 1: come elencare solo i membri correnti nella Regione corrente

L'`list-members`esempio seguente elenca e fornisce i dettagli solo degli account membro correnti associati all'account GuardDuty amministratore, nell'area corrente.

```
aws guardduty list-members \
    --detector-id 12abc34d567e8fa901bc2d34eexample \
    --only-associated="true"
```

Output:

```
{
    "Members": [
        {
            "RelationshipStatus": "Enabled",
            "InvitedAt": "2020-06-09T22:49:00.910Z",
            "MasterId": "111122223333",
            "DetectorId": "7ab8b2f61b256c87f793f6a86example",
            "UpdatedAt": "2020-06-09T23:08:22.512Z",
            "Email": "your+member@example.com",
            "AccountId": "123456789012"
        }
    ]
}
```

Per ulteriori informazioni, vedere Comprensione della relazione tra account GuardDuty amministratore e account membro nella Guida per l'GuardDuty utente.

Esempio 2: come elencare tutti i membri nella Regione corrente

L'`list-members`esempio seguente elenca e fornisce i dettagli di tutti gli account membri, inclusi quelli che sono stati dissociati o che non hanno ancora accettato l'invito dell' GuardDuty amministratore, nell'area corrente.

```
aws guardduty list-members \
    --detector-id 12abc34d567e8fa901bc2d34eexample \
    --only-associated="false"
```

Output:

```
{
    "Members": [
        {
            "RelationshipStatus": "Enabled",
            "InvitedAt": "2020-06-09T22:49:00.910Z",
            "MasterId": "111122223333",
            "DetectorId": "7ab8b2f61b256c87f793f6a86example",
            "UpdatedAt": "2020-06-09T23:08:22.512Z",
            "Email": "your+other+member@example.com",
            "AccountId": "555555555555"
        }
    ]
}
```

Per ulteriori informazioni, consulta Comprendere la relazione tra account GuardDuty amministratore e account membro nella Guida per l'GuardDuty utente.

- Per i dettagli sull'API, consulta ListMembers AWS CLICommand Reference.

`update-detector`

Il seguente esempio di codice mostra come utilizzare`update-detector`.

AWS CLI

Esempio 1: Per abilitare il monitoraggio del runtime in GuardDuty

L'`update-detector`esempio seguente abilita il monitoraggio del runtime senza configurazioni aggiuntive.

```
aws guardduty update-detector \
```

```
    --detector-id 12abc34d567e8fa901bc2d34eexample \
    --features 'Name=RUNTIME_MONITORING,Status=ENABLED'
```

Questo comando non produce alcun output.

Per ulteriori informazioni, vedere Runtime monitoring nella Guida GuardDuty per l'utente.

Esempio 2: per abilitare il monitoraggio del runtime con una configurazione aggiuntiva

L'`update-detector`esempio seguente abilita il monitoraggio del runtime con una configurazione aggiuntiva per EC2 ECS Fargate ed EKS.

```
aws guardduty update-detector \
    --detector-id 12abc34d567e8fa901bc2d34eexample \
    --features
 'Name=RUNTIME_MONITORING,Status=ENABLED,AdditionalConfiguration=[{Name=EC2_AGENT_MANAGEMENT
{Name=ECS_FARGATE_AGENT_MANAGEMENT,Status=ENABLED},
{Name=EKS_ADDON_MANAGEMENT,Status=ENABLED}]'
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Runtime monitoring nella Guida per l'GuardDuty utente.

- Per i dettagli sull'API, consulta UpdateDetector AWS CLICommand Reference.

update-ip-set

Il seguente esempio di codice mostra come utilizzare`update-ip-set`.

AWS CLI

Come aggiornare un set di IP attendibili

L'esempio `update-ip-set` seguente mostra come aggiornare i dettagli di un set di IP attendibili.

```
aws guardduty update-ip-set \
    --detector-id 12abc34d567e8fa901bc2d34eexample \
    --ip-set-id d4b94fc952d6912b8f3060768example \
    --location https://amzn-s3-demo-bucket.s3-us-west-2.amazonaws.com/
customtrustlist2.csv
```

Questo comando non produce alcun output.

Per ulteriori informazioni, vedere Utilizzo degli elenchi di IP affidabili e degli elenchi di minacce nella Guida GuardDuty per l'utente.

- Per i dettagli sulle API, consulta UpdateIpSet AWS CLICommand Reference.

AWS Health esempi utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with AWS Health.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`describe-affected-entities`

Il seguente esempio di codice mostra come utilizzare`describe-affected-entities`.

AWS CLI

Per elencare le entità interessate da uno specifico evento AWS Health

L'`describe-affected-entities`esempio seguente elenca le entità interessate dall'evento AWS Health specificato. Questo evento è una notifica di fatturazione per l' AWS account.

```
aws health describe-affected-entities \
    --filter "eventArns=arn:aws:health:global::event/BILLING/
AWS_BILLING_NOTIFICATION/AWS_BILLING_NOTIFICATION_6ce1d874-e995-40e2-99cd-
EXAMPLE11145" \
    --region us-east-1
```

Output:

```
{
    "entities": [
        {
            "entityArn": "arn:aws:health:global:123456789012:entity/
EXAMPLEimSMoULmWHpb",
            "eventArn": "arn:aws:health:global::event/BILLING/
AWS_BILLING_NOTIFICATION/AWS_BILLING_NOTIFICATION_6ce1d874-e995-40e2-99cd-
EXAMPLE11145",
            "entityValue": "AWS_ACCOUNT",
            "awsAccountId": "123456789012",
            "lastUpdatedTime": 1588356454.08
        }
    ]
}
```

Per ulteriori informazioni, consulta Log degli eventi nella Guida per l'utente di AWS Health.

- Per i dettagli sull'API, consulta DescribeAffectedEntities AWS CLICommand Reference.

`describe-event-details`

Il seguente esempio di codice mostra come utilizzare`describe-event-details`.

AWS CLI

Per elencare informazioni su un evento AWS Health

L'`describe-event-details`esempio seguente elenca le informazioni sull'evento AWS Health specificato.

```
aws health describe-event-details \
    --event-arns "arn:aws:health:us-east-1::event/EC2/AWS_EC2_OPERATIONAL_ISSUE/
AWS_EC2_OPERATIONAL_ISSUE_VKTXI_EXAMPLE111" \
    --region us-east-1
```

Output:

```
{
    "successfulSet": [
```

```
        {
            "event": {
                "arn": "arn:aws:health:us-east-1::event/EC2/
AWS_EC2_OPERATIONAL_ISSUE/AWS_EC2_OPERATIONAL_ISSUE_VKTXI_EXAMPLE111",
                "service": "EC2",
                "eventTypeCode": "AWS_EC2_OPERATIONAL_ISSUE",
                "eventTypeCategory": "issue",
                "region": "us-east-1",
                "startTime": 1587462325.096,
                "endTime": 1587464204.774,
                "lastUpdatedTime": 1587464204.865,
                "statusCode": "closed"
            },
            "eventDescription": {
                "latestDescription": "[RESOLVED] Increased API Error Rates and
 Latencies\n\n[02:45 AM PDT] We are investigating increased API error rates and
 latencies in the US-EAST-1 Region.\n\n[03:16 AM PDT] Between 2:10 AM and 2:59 AM
 PDT we experienced increased API error rates and latencies in the US-EAST-1 Region.
 The issue has been resolved and the service is operating normally."
            }
        }
    ],
    "failedSet": []
}
```

Per ulteriori informazioni, consulta Riquadro Dettagli evento nella Guida per l'utente di AWS Health.

- Per i dettagli sull'API, vedere DescribeEventDetailsin AWS CLI Command Reference.

`describe-events`

Il seguente esempio di codice mostra come utilizzare`describe-events`.

AWS CLI

Esempio 1: elencare gli eventi AWS Health

L'`describe-events`esempio seguente elenca gli eventi AWS Health recenti.

```
aws health describe-events \
    --region us-east-1
```

Output:

```
{
    "events": [
        {
            "arn": "arn:aws:health:us-west-1::event/ECS/AWS_ECS_OPERATIONAL_ISSUE/
AWS_ECS_OPERATIONAL_ISSUE_KWQPY_EXAMPLE111",
            "service": "ECS",
            "eventTypeCode": "AWS_ECS_OPERATIONAL_ISSUE",
            "eventTypeCategory": "issue",
            "region": "us-west-1",
            "startTime": 1589077890.53,
            "endTime": 1589086345.597,
            "lastUpdatedTime": 1589086345.905,
            "statusCode": "closed",
            "eventScopeCode": "PUBLIC"
        },
        {
            "arn": "arn:aws:health:global::event/BILLING/AWS_BILLING_NOTIFICATION/
AWS_BILLING_NOTIFICATION_6ce1d874-e995-40e2-99cd-EXAMPLE1118b",
            "service": "BILLING",
            "eventTypeCode": "AWS_BILLING_NOTIFICATION",
            "eventTypeCategory": "accountNotification",
            "region": "global",
            "startTime": 1588356000.0,
            "lastUpdatedTime": 1588356524.358,
            "statusCode": "open",
            "eventScopeCode": "ACCOUNT_SPECIFIC"
        },
        {
            "arn": "arn:aws:health:us-west-2::event/
CLOUDFORMATION/AWS_CLOUDFORMATION_OPERATIONAL_ISSUE/
AWS_CLOUDFORMATION_OPERATIONAL_ISSUE_OHTWY_EXAMPLE111",
            "service": "CLOUDFORMATION",
            "eventTypeCode": "AWS_CLOUDFORMATION_OPERATIONAL_ISSUE",
            "eventTypeCategory": "issue",
            "region": "us-west-2",
            "startTime": 1588279630.761,
            "endTime": 1588284650.0,
            "lastUpdatedTime": 1588284691.941,
            "statusCode": "closed",
            "eventScopeCode": "PUBLIC"
        },
        {
```

```
            "arn": "arn:aws:health:ap-northeast-1::event/LAMBDA/
AWS_LAMBDA_OPERATIONAL_ISSUE/AWS_LAMBDA_OPERATIONAL_ISSUE_JZDND_EXAMPLE111",
            "service": "LAMBDA",
            "eventTypeCode": "AWS_LAMBDA_OPERATIONAL_ISSUE",
            "eventTypeCategory": "issue",
            "region": "ap-northeast-1",
            "startTime": 1587379534.08,
            "endTime": 1587391771.0,
            "lastUpdatedTime": 1587395689.316,
            "statusCode": "closed",
            "eventScopeCode": "PUBLIC"
        },
        {
            "arn": "arn:aws:health:us-east-1::event/EC2/AWS_EC2_OPERATIONAL_ISSUE/
AWS_EC2_OPERATIONAL_ISSUE_COBXJ_EXAMPLE111",
            "service": "EC2",
            "eventTypeCode": "AWS_EC2_OPERATIONAL_ISSUE",
            "eventTypeCategory": "issue",
            "region": "us-east-1",
            "startTime": 1586473044.284,
            "endTime": 1586479706.091,
            "lastUpdatedTime": 1586479706.153,
            "statusCode": "closed",
            "eventScopeCode": "PUBLIC"
        },
        {
            "arn": "arn:aws:health:global::event/SECURITY/AWS_SECURITY_NOTIFICATION/
AWS_SECURITY_NOTIFICATION_42007387-8129-42da-8c88-EXAMPLE11139",
            "service": "SECURITY",
            "eventTypeCode": "AWS_SECURITY_NOTIFICATION",
            "eventTypeCategory": "accountNotification",
            "region": "global",
            "startTime": 1585674000.0,
            "lastUpdatedTime": 1585674004.132,
            "statusCode": "open",
            "eventScopeCode": "PUBLIC"
        },
        {
            "arn": "arn:aws:health:global::event/CLOUDFRONT/
AWS_CLOUDFRONT_OPERATIONAL_ISSUE/AWS_CLOUDFRONT_OPERATIONAL_ISSUE_FRQXG_EXAMPLE111",
            "service": "CLOUDFRONT",
            "eventTypeCode": "AWS_CLOUDFRONT_OPERATIONAL_ISSUE",
            "eventTypeCategory": "issue",
            "region": "global",
```

```
            "startTime": 1585610898.589,
            "endTime": 1585617671.0,
            "lastUpdatedTime": 1585620638.869,
            "statusCode": "closed",
            "eventScopeCode": "PUBLIC"
        },
        {
            "arn": "arn:aws:health:us-east-1::event/SES/AWS_SES_OPERATIONAL_ISSUE/
AWS_SES_OPERATIONAL_ISSUE_URNDF_EXAMPLE111",
            "service": "SES",
            "eventTypeCode": "AWS_SES_OPERATIONAL_ISSUE",
            "eventTypeCategory": "issue",
            "region": "us-east-1",
            "startTime": 1585342008.46,
            "endTime": 1585344017.0,
            "lastUpdatedTime": 1585344355.989,
            "statusCode": "closed",
            "eventScopeCode": "PUBLIC"
        },
        {
            "arn": "arn:aws:health:global::event/IAM/
AWS_IAM_OPERATIONAL_NOTIFICATION/
AWS_IAM_OPERATIONAL_NOTIFICATION_b6771c34-6ecd-4aea-9d3e-EXAMPLE1117e",
            "service": "IAM",
            "eventTypeCode": "AWS_IAM_OPERATIONAL_NOTIFICATION",
            "eventTypeCategory": "accountNotification",
            "region": "global",
            "startTime": 1584978300.0,
            "lastUpdatedTime": 1584978553.572,
            "statusCode": "open",
            "eventScopeCode": "ACCOUNT_SPECIFIC"
        },
        {
            "arn": "arn:aws:health:ap-southeast-2::event/EC2/
AWS_EC2_OPERATIONAL_ISSUE/AWS_EC2_OPERATIONAL_ISSUE_HNGHE_EXAMPLE111",
            "service": "EC2",
            "eventTypeCode": "AWS_EC2_OPERATIONAL_ISSUE",
            "eventTypeCategory": "issue",
            "region": "ap-southeast-2",
            "startTime": 1583881487.483,
            "endTime": 1583885056.785,
            "lastUpdatedTime": 1583885057.052,
            "statusCode": "closed",
            "eventScopeCode": "PUBLIC"
```

```
        }
    ]
}
```

Per ulteriori informazioni, consulta la Guida introduttiva alla AWS Personal Health Dashboard nella AWS Health User Guide.

Esempio 2: elencare gli eventi AWS Health per servizio e codice di stato dell'evento

L'`describe-events`esempio seguente elenca gli eventi AWS Health per Amazon Elastic Compute Cloud (Amazon EC2) in cui lo stato dell'evento è chiuso.

```
aws health describe-events \
    --filter "services=EC2,eventStatusCodes=closed"
```

Output:

```
{
    "events": [
        {
            "arn": "arn:aws:health:us-east-1::event/EC2/AWS_EC2_OPERATIONAL_ISSUE/
AWS_EC2_OPERATIONAL_ISSUE_VKTXI_EXAMPLE111",
            "service": "EC2",
            "eventTypeCode": "AWS_EC2_OPERATIONAL_ISSUE",
            "eventTypeCategory": "issue",
            "region": "us-east-1",
            "startTime": 1587462325.096,
            "endTime": 1587464204.774,
            "lastUpdatedTime": 1587464204.865,
            "statusCode": "closed",
            "eventScopeCode": "PUBLIC"
        },
        {
            "arn": "arn:aws:health:us-east-1::event/EC2/AWS_EC2_OPERATIONAL_ISSUE/
AWS_EC2_OPERATIONAL_ISSUE_COBXJ_EXAMPLE111",
            "service": "EC2",
            "eventTypeCode": "AWS_EC2_OPERATIONAL_ISSUE",
            "eventTypeCategory": "issue",
            "region": "us-east-1",
            "startTime": 1586473044.284,
            "endTime": 1586479706.091,
            "lastUpdatedTime": 1586479706.153,
```

```
            "statusCode": "closed",
            "eventScopeCode": "PUBLIC"
        },
        {
            "arn": "arn:aws:health:ap-southeast-2::event/EC2/
AWS_EC2_OPERATIONAL_ISSUE/AWS_EC2_OPERATIONAL_ISSUE_HNGHE_EXAMPLE111",
            "service": "EC2",
            "eventTypeCode": "AWS_EC2_OPERATIONAL_ISSUE",
            "eventTypeCategory": "issue",
            "region": "ap-southeast-2",
            "startTime": 1583881487.483,
            "endTime": 1583885056.785,
            "lastUpdatedTime": 1583885057.052,
            "statusCode": "closed",
            "eventScopeCode": "PUBLIC"
        }
    ]
}
```

Per ulteriori informazioni, consulta la Guida introduttiva alla AWS Personal Health Dashboard nella AWS Health User Guide.

- Per i dettagli sull'API, consulta DescribeEvents AWS CLICommand Reference.

HealthImaging esempi utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with HealthImaging.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`copy-image-set`

Il seguente esempio di codice mostra come utilizzare`copy-image-set`.

AWS CLI

Esempio 1: come copiare un set di immagini senza una destinazione.

L'esempio `copy-image-set` seguente crea una copia duplicata di un set di immagini senza una destinazione.

```
aws medical-imaging copy-image-set \
    --datastore-id 12345678901234567890123456789012 \
    --source-image-set-id ea92b0d8838c72a3f25d00d13616f87e \
    --copy-image-set-information '{"sourceImageSet": {"latestVersionId": "1" } }'
```

Output:

```
{
    "destinationImageSetProperties": {
        "latestVersionId": "2",
        "imageSetWorkflowStatus": "COPYING",
        "updatedAt": 1680042357.432,
        "imageSetId": "b9a06fef182a5f992842f77f8e0868e5",
        "imageSetState": "LOCKED",
        "createdAt": 1680042357.432
    },
    "sourceImageSetProperties": {
        "latestVersionId": "1",
        "imageSetWorkflowStatus": "COPYING_WITH_READ_ONLY_ACCESS",
        "updatedAt": 1680042357.432,
        "imageSetId": "ea92b0d8838c72a3f25d00d13616f87e",
        "imageSetState": "LOCKED",
        "createdAt": 1680027126.436
    },
    "datastoreId": "12345678901234567890123456789012"
}
```

Esempio 2: come copiare un set di immagini con una destinazione.

L'esempio `copy-image-set` seguente crea una copia duplicata di un set di immagini con una destinazione.

```
aws medical-imaging copy-image-set \
    --datastore-id 12345678901234567890123456789012 \
    --source-image-set-id ea92b0d8838c72a3f25d00d13616f87e \
    --copy-image-set-information '{"sourceImageSet": {"latestVersionId": "1" },
 "destinationImageSet": { "imageSetId": "b9a06fef182a5f992842f77f8e0868e5",
 "latestVersionId": "1"} }'
```

Output:

```
{
    "destinationImageSetProperties": {
        "latestVersionId": "2",
        "imageSetWorkflowStatus": "COPYING",
        "updatedAt": 1680042505.135,
        "imageSetId": "b9a06fef182a5f992842f77f8e0868e5",
        "imageSetState": "LOCKED",
        "createdAt": 1680042357.432
    },
    "sourceImageSetProperties": {
        "latestVersionId": "1",
        "imageSetWorkflowStatus": "COPYING_WITH_READ_ONLY_ACCESS",
        "updatedAt": 1680042505.135,
        "imageSetId": "ea92b0d8838c72a3f25d00d13616f87e",
        "imageSetState": "LOCKED",
        "createdAt": 1680027126.436
    },
    "datastoreId": "12345678901234567890123456789012"
}
```

Esempio 3: come copiare un sottoinsieme di istanze da un set di immagini di origine a un set di immagini di destinazione.

L'esempio `copy-image-set` seguente copia un'istanza DICOM dal set di immagini di origine nel set di immagini di destinazione. Il parametro force viene fornito per eseguire l'override delle incongruenze a livello degli attributi Patient, Study e Series.

```
aws medical-imaging copy-image-set \
    --datastore-id 12345678901234567890123456789012 \
    --source-image-set-id ea92b0d8838c72a3f25d00d13616f87e \
```

```
    --copy-image-set-information '{"sourceImageSet": {"latestVersionId":
 "1","DICOMCopies": {"copiableAttributes": "{\"SchemaVersion\":\"1.1\",\"Study\":
{\"Series\":{\"1.3.6.1.4.1.5962.99.1.3673257865.2104868982.1369432891697.3666.0\":
{\"Instances\":
{\"1.3.6.1.4.1.5962.99.1.3673257865.2104868982.1369432891697.3669.0\":
{}}}}}}"}},"destinationImageSet": {"imageSetId":
 "b9eb50d8ee682eb9fcf4acbf92f62bb7","latestVersionId": "1"}}' \
    --force
```

Output:

```
{
    "destinationImageSetProperties": {
        "latestVersionId": "2",
        "imageSetWorkflowStatus": "COPYING",
        "updatedAt": 1680042505.135,
        "imageSetId": "b9eb50d8ee682eb9fcf4acbf92f62bb7",
        "imageSetState": "LOCKED",
        "createdAt": 1680042357.432
    },
    "sourceImageSetProperties": {
        "latestVersionId": "1",
        "imageSetWorkflowStatus": "COPYING_WITH_READ_ONLY_ACCESS",
        "updatedAt": 1680042505.135,
        "imageSetId": "ea92b0d8838c72a3f25d00d13616f87e",
        "imageSetState": "LOCKED",
        "createdAt": 1680027126.436
    },
    "datastoreId": "12345678901234567890123456789012"
}
```

Per ulteriori informazioni, consulta Copiare un set di immagini nella Guida per gli AWS HealthImaging sviluppatori.

- Per i dettagli sull'API, consulta CopyImageSet AWS CLICommand Reference.

`create-datastore`

Il seguente esempio di codice mostra come utilizzare`create-datastore`.

AWS CLI

Esempio 1: per creare un archivio dati

L'esempio di codice `create-datastore` seguente crea un datastore denominato `my-datastore`. Quando si crea un datastore senza specificare a`--lossless-storage-format`, il AWS HealthImaging valore predefinito è HTJ2 K (High Throughput JPEG 2000).

```
aws medical-imaging create-datastore \
    --datastore-name "my-datastore"
```

Output:

```
{
    "datastoreId": "12345678901234567890123456789012",
    "datastoreStatus": "CREATING"
}
```

Esempio 2: creare un archivio dati con il formato di archiviazione JPEG 2000 Lossless

Un data store configurato con il formato di archiviazione JPEG 2000 Lossless transcodificherà e renderà persistenti i frame di immagine senza perdita di dati in formato JPEG 2000. I fotogrammi delle immagini possono quindi essere recuperati in JPEG 2000 Lossless senza transcodifica. Il seguente esempio di `create-datastore` codice crea un archivio dati configurato per il formato di archiviazione JPEG 2000 Lossless con il nome. `my-datastore`

```
aws medical-imaging create-datastore \
    --datastore-name "my-datastore" \
    --lossless-storage-format JPEG_2000_LOSSLESS
```

Output:

```
{
    "datastoreId": "12345678901234567890123456789012",
    "datastoreStatus": "CREATING"
}
```

Per ulteriori informazioni, consulta Creazione di un data store nella AWS HealthImaging Developer Guide.

- Per i dettagli sull'API, consulta CreateDatastore AWS CLICommand Reference.

`delete-datastore`

Il seguente esempio di codice mostra come utilizzare`delete-datastore`.

AWS CLI

Come eliminare un datastore

L'esempio di codice `delete-datastore` seguente elimina un datastore.

```
aws medical-imaging delete-datastore \
    --datastore-id "12345678901234567890123456789012"
```

Output:

```
{
    "datastoreId": "12345678901234567890123456789012",
    "datastoreStatus": "DELETING"
}
```

Per ulteriori informazioni, consulta Eliminazione di un data store nella Guida per gli AWS HealthImaging sviluppatori.

- Per i dettagli sull'API, consulta DeleteDatastore AWS CLICommand Reference.

`delete-image-set`

Il seguente esempio di codice mostra come utilizzare`delete-image-set`.

AWS CLI

Come eliminare un set di immagini

L'esempio di codice `delete-image-set` seguente elimina un set di immagini.

```
aws medical-imaging delete-image-set \
    --datastore-id 12345678901234567890123456789012 \
    --image-set-id ea92b0d8838c72a3f25d00d13616f87e
```

Output:

```
{
    "imageSetWorkflowStatus": "DELETING",
    "imageSetId": "ea92b0d8838c72a3f25d00d13616f87e",
    "imageSetState": "LOCKED",
    "datastoreId": "12345678901234567890123456789012"
}
```

Per ulteriori informazioni, consulta Eliminazione di un set di immagini nella Guida per gli AWS HealthImaging sviluppatori.

- Per i dettagli sull'API, consulta DeleteImageSet AWS CLICommand Reference.

`get-datastore`

Il seguente esempio di codice mostra come utilizzare`get-datastore`.

AWS CLI

Esempio 1: per ottenere le proprietà di un archivio dati

L'esempio di codice `get-datastore` seguente ottiene le proprietà di un datastore.

```
aws medical-imaging get-datastore \
    --datastore-id 12345678901234567890123456789012
```

Output:

```
{
    "datastoreProperties": {
        "datastoreId": "12345678901234567890123456789012",
        "datastoreName": "TestDatastore123",
        "datastoreStatus": "ACTIVE",
        "losslessStorageFormat": "HTJ2K"
        "datastoreArn": "arn:aws:medical-imaging:us-
east-1:123456789012:datastore/12345678901234567890123456789012",
        "createdAt": "2022-11-15T23:33:09.643000+00:00",
        "updatedAt": "2022-11-15T23:33:09.643000+00:00"
    }
}
```

Esempio 2: per configurare le proprietà del data store per JPEG2 000

Il seguente esempio di `get-datastore` codice ottiene le proprietà di un data store per un data store configurato per il formato di archiviazione JPEG 2000 Lossless.

```
aws medical-imaging get-datastore \
    --datastore-id 12345678901234567890123456789012
```

Output:

```
{
    "datastoreProperties": {
        "datastoreId": "12345678901234567890123456789012",
        "datastoreName": "TestDatastore123",
        "datastoreStatus": "ACTIVE",
        "losslessStorageFormat": "JPEG_2000_LOSSLESS",
        "datastoreArn": "arn:aws:medical-imaging:us-
east-1:123456789012:datastore/12345678901234567890123456789012",
        "createdAt": "2022-11-15T23:33:09.643000+00:00",
        "updatedAt": "2022-11-15T23:33:09.643000+00:00"
    }
}
```

Per ulteriori informazioni, consulta Ottenere le proprietà del data store nella AWS HealthImaging Developer Guide.

- Per i dettagli sull'API, consulta GetDatastore AWS CLICommand Reference.

`get-dicom-import-job`

Il seguente esempio di codice mostra come utilizzare`get-dicom-import-job`.

AWS CLI

Come ottenere le proprietà di un processo di importazione DICOM

L'esempio di codice `get-dicom-import-job` seguente concede l'autorizzazione ad avviare un processo di importazione DICOM.

```
aws medical-imaging get-dicom-import-job \
    --datastore-id "12345678901234567890123456789012" \
    --job-id "09876543210987654321098765432109"
```

Output:

```
{
    "jobProperties": {
        "jobId": "09876543210987654321098765432109",
        "jobName": "my-job",
        "jobStatus": "COMPLETED",
        "datastoreId": "12345678901234567890123456789012",
        "dataAccessRoleArn": "arn:aws:iam::123456789012:role/
ImportJobDataAccessRole",
        "endedAt": "2022-08-12T11:29:42.285000+00:00",
        "submittedAt": "2022-08-12T11:28:11.152000+00:00",
        "inputS3Uri": "s3://medical-imaging-dicom-input/dicom_input/",
        "outputS3Uri": "s3://medical-imaging-output/
job_output/12345678901234567890123456789012-
DicomImport-09876543210987654321098765432109/"
    }
}
```

Per ulteriori informazioni, consulta Ottenere le proprietà dei job di importazione nella AWS HealthImaging Developer Guide.

- Per i dettagli sull'API, consulta Get DICOMImport Job in AWS CLI Command Reference.

`get-image-frame`

Il seguente esempio di codice mostra come utilizzare`get-image-frame`.

AWS CLI

Come ottenere i dati dei pixel del set di immagini

L'esempio di codice `get-image-frame` seguente ottiene un image frame di immagine.

```
aws medical-imaging get-image-frame \
    --datastore-id "12345678901234567890123456789012" \
    --image-set-id "98765412345612345678907890789012" \
    --image-frame-information imageFrameId=3abf5d5d7ae72f80a0ec81b2c0de3ef4 \
    imageframe.jph
```

Nota: questo esempio di codice non include l'output perché l' GetImageFrame azione restituisce un flusso di dati di pixel al file imageframe.jph. Per informazioni sulla decodifica e la visualizzazione dei frame di immagini, vedete K decoding libraries. HTJ2

Per ulteriori informazioni, consultate Ottenere i dati dei pixel del set di immagini nella Guida per gli AWS HealthImaging sviluppatori.

- Per i dettagli sull'API, consulta GetImageFrame AWS CLICommand Reference.

`get-image-set-metadata`

Il seguente esempio di codice mostra come utilizzare`get-image-set-metadata`.

AWS CLI

Esempio 1: come ottenere i metadati del set di immagini senza versione

L'esempio di codice `get-image-set-metadata` seguente ottiene i metadati per un set di immagini senza specificare una versione.

Nota: `outfile` è un parametro obbligatorio.

```
aws medical-imaging get-image-set-metadata \
    --datastore-id 12345678901234567890123456789012 \
    --image-set-id ea92b0d8838c72a3f25d00d13616f87e \
    studymetadata.json.gz
```

I metadati restituiti vengono compressi con gzip e archiviati nel file studymetadata.json.gz. Per visualizzare il contenuto dell'oggetto JSON restituito, devi prima decomprimerlo.

Output:

```
{
    "contentType": "application/json",
    "contentEncoding": "gzip"
}
```

Esempio 2: come ottenere i metadati del set di immagini con versione

L'esempio di codice `get-image-set-metadata` seguente ottiene i metadati per un set di immagini con una versione specificata.

Nota: `outfile` è un parametro obbligatorio.

```
aws medical-imaging get-image-set-metadata \
```

```
    --datastore-id 12345678901234567890123456789012 \
    --image-set-id ea92b0d8838c72a3f25d00d13616f87e \
    --version-id 1 \
    studymetadata.json.gz
```

I metadati restituiti vengono compressi con gzip e archiviati nel file studymetadata.json.gz. Per visualizzare il contenuto dell'oggetto JSON restituito, devi prima decomprimerlo.

Output:

```
{
    "contentType": "application/json",
    "contentEncoding": "gzip"
}
```

Per ulteriori informazioni, consulta Ottenere i metadati dei set di immagini nella Guida per gli AWS HealthImaging sviluppatori.

- Per i dettagli sull'API, consulta GetImageSetMetadata AWS CLICommand Reference.

`get-image-set`

Il seguente esempio di codice mostra come utilizzare`get-image-set`.

AWS CLI

Come ottenere le proprietà del set di immagini

L'esempio di codice `get-image-set` seguente ottiene le proprietà di un set di immagini.

```
aws medical-imaging get-image-set \
    --datastore-id 12345678901234567890123456789012 \
    --image-set-id 18f88ac7870584f58d56256646b4d92b \
    --version-id 1
```

Output:

```
{
    "versionId": "1",
    "imageSetWorkflowStatus": "COPIED",
    "updatedAt": 1680027253.471,
```

```
    "imageSetId": "18f88ac7870584f58d56256646b4d92b",
    "imageSetState": "ACTIVE",
    "createdAt": 1679592510.753,
    "datastoreId": "12345678901234567890123456789012"
}
```

Per ulteriori informazioni, consulta Ottenere le proprietà del set di immagini nella Guida per gli AWS HealthImaging sviluppatori.

- Per i dettagli sull'API, consulta GetImageSet AWS CLICommand Reference.

`list-datastores`

Il seguente esempio di codice mostra come utilizzare`list-datastores`.

AWS CLI

Come elencare i datastore

L'esempio di codice `list-datastores` seguente elenca i datastore disponibili.

```
aws medical-imaging list-datastores
```

Output:

```
{
    "datastoreSummaries": [
        {
            "datastoreId": "12345678901234567890123456789012",
            "datastoreName": "TestDatastore123",
            "datastoreStatus": "ACTIVE",
            "datastoreArn": "arn:aws:medical-imaging:us-
east-1:123456789012:datastore/12345678901234567890123456789012",
            "createdAt": "2022-11-15T23:33:09.643000+00:00",
            "updatedAt": "2022-11-15T23:33:09.643000+00:00"
        }
    ]
}
```

Per ulteriori informazioni, consulta Elencare gli archivi di dati nella Guida per gli AWS HealthImaging sviluppatori.

- Per i dettagli sull'API, consulta ListDatastores AWS CLICommand Reference.

`list-dicom-import-jobs`

Il seguente esempio di codice mostra come utilizzare`list-dicom-import-jobs`.

AWS CLI

Come elencare i processi di importazione DICOM

L'esempio di codice `list-dicom-import-jobs` seguente elenca i processi di importazione DICOM.

```
aws medical-imaging list-dicom-import-jobs \
    --datastore-id "12345678901234567890123456789012"
```

Output:

```
{
    "jobSummaries": [
        {
            "jobId": "09876543210987654321098765432109",
            "jobName": "my-job",
            "jobStatus": "COMPLETED",
            "datastoreId": "12345678901234567890123456789012",
            "dataAccessRoleArn": "arn:aws:iam::123456789012:role/
ImportJobDataAccessRole",
            "endedAt": "2022-08-12T11:21:56.504000+00:00",
            "submittedAt": "2022-08-12T11:20:21.734000+00:00"
        }
    ]
}
```

Per ulteriori informazioni, consulta Elencare i lavori di importazione nella Guida per gli AWS HealthImaging sviluppatori.

- Per i dettagli sull'API, consulta List DICOMImport Jobs in AWS CLI Command Reference.

`list-image-set-versions`

Il seguente esempio di codice mostra come utilizzare`list-image-set-versions`.

AWS CLI

Come elencare le versioni dei set di immagini

L'esempio di codice `list-image-set-versions` seguente elenca la cronologia delle versioni di un set di immagini.

```
aws medical-imaging list-image-set-versions \
    --datastore-id 12345678901234567890123456789012 \
    --image-set-id ea92b0d8838c72a3f25d00d13616f87e
```

Output:

```
{
    "imageSetPropertiesList": [
        {
            "ImageSetWorkflowStatus": "UPDATED",
            "versionId": "4",
            "updatedAt": 1680029436.304,
            "imageSetId": "ea92b0d8838c72a3f25d00d13616f87e",
            "imageSetState": "ACTIVE",
            "createdAt": 1680027126.436
        },
        {
            "ImageSetWorkflowStatus": "UPDATED",
            "versionId": "3",
            "updatedAt": 1680029163.325,
            "imageSetId": "ea92b0d8838c72a3f25d00d13616f87e",
            "imageSetState": "ACTIVE",
            "createdAt": 1680027126.436
        },
        {
            "ImageSetWorkflowStatus": "COPY_FAILED",
            "versionId": "2",
            "updatedAt": 1680027455.944,
            "imageSetId": "ea92b0d8838c72a3f25d00d13616f87e",
            "imageSetState": "ACTIVE",
            "message": "INVALID_REQUEST:  Series of SourceImageSet and
 DestinationImageSet don't match.",
            "createdAt": 1680027126.436
        },
        {
            "imageSetId": "ea92b0d8838c72a3f25d00d13616f87e",
```

```
            "imageSetState": "ACTIVE",
            "versionId": "1",
            "ImageSetWorkflowStatus": "COPIED",
            "createdAt": 1680027126.436
        }
    ]
}
```

Per ulteriori informazioni, consulta Elenco delle versioni dei set di immagini nella Guida per gli AWS HealthImaging sviluppatori.

- Per i dettagli sull'API, consulta ListImageSetVersions AWS CLICommand Reference.

`list-tags-for-resource`

Il seguente esempio di codice mostra come utilizzare`list-tags-for-resource`.

AWS CLI

Esempio 1: come elencare i tag delle risorse per un datastore

L'esempio di codice `list-tags-for-resource` seguente elenca i tag di un datastore.

```
aws medical-imaging list-tags-for-resource \
    --resource-arn "arn:aws:medical-imaging:us-
east-1:123456789012:datastore/12345678901234567890123456789012"
```

Output:

```
{
    "tags":{
        "Deployment":"Development"
    }
}
```

Esempio 2: come elencare i tag delle risorse per un set di immagini

L'esempio di codice `list-tags-for-resource` seguente elenca i tag di un set di immagini.

```
aws medical-imaging list-tags-for-resource \
```

```
    --resource-arn "arn:aws:medical-imaging:us-
east-1:123456789012:datastore/12345678901234567890123456789012/
imageset/18f88ac7870584f58d56256646b4d92b"
```

Output:

```
{
    "tags":{
        "Deployment":"Development"
    }
}
```

Per ulteriori informazioni, consulta Tagging resources with AWS HealthImaging nella AWS HealthImaging Developer Guide.

- Per i dettagli sull'API, consulta ListTagsForResource AWS CLICommand Reference.

`search-image-sets`

Il seguente esempio di codice mostra come utilizzare`search-image-sets`.

AWS CLI

Esempio 1: come cercare set di immagini con un operatore EQUAL

L'esempio di codice `search-image-sets` seguente utilizza l'operatore EQUAL per cercare set di immagini in base a un valore specifico.

```
aws medical-imaging search-image-sets \
    --datastore-id 12345678901234567890123456789012 \
    --search-criteria file://search-criteria.json
```

Contenuto di `search-criteria.json`

```
{
    "filters": [{
        "values": [{"DICOMPatientId" : "SUBJECT08701"}],
        "operator": "EQUAL"
    }]
}
```

Output:

```
{
    "imageSetsMetadataSummaries": [{
        "imageSetId": "09876543210987654321098765432109",
        "createdAt": "2022-12-06T21:40:59.429000+00:00",
        "version": 1,
        "DICOMTags": {
            "DICOMStudyId": "2011201407",
            "DICOMStudyDate": "19991122",
             "DICOMPatientSex": "F",
             "DICOMStudyInstanceUID": "1.2.840.99999999.84710745.943275268089",
             "DICOMPatientBirthDate": "19201120",
             "DICOMStudyDescription": "UNKNOWN",
             "DICOMPatientId": "SUBJECT08701",
             "DICOMPatientName": "Melissa844 Huel628",
             "DICOMNumberOfStudyRelatedInstances": 1,
             "DICOMStudyTime": "140728",
             "DICOMNumberOfStudyRelatedSeries": 1
            },
        "updatedAt": "2022-12-06T21:40:59.429000+00:00"
    }]
}
```

Esempio 2: per cercare set di immagini con un operatore BETWEEN utilizzando DICOMStudy Data e DICOMStudy ora

L'esempio `search-image-sets` seguente cerca set di immagini con studi DICOM generati tra il 1° gennaio 1990 (00:00) e il 1° gennaio 2023 (00:00).

Nota: l'DICOMStudyora è facoltativa. Se non presente, 00:00 (inizio della giornata) è il valore temporale per le date fornite per l'applicazione di filtri.

```
aws medical-imaging search-image-sets \
    --datastore-id 12345678901234567890123456789012 \
    --search-criteria file://search-criteria.json
```

Contenuto di `search-criteria.json`

```
{
    "filters": [{
        "values": [{
            "DICOMStudyDateAndTime": {
```

```
                "DICOMStudyDate": "19900101",
                "DICOMStudyTime": "000000"
            }
        },
        {
            "DICOMStudyDateAndTime": {
                "DICOMStudyDate": "20230101",
                "DICOMStudyTime": "000000"
            }
        }],
        "operator": "BETWEEN"
    }]
}
```

Output:

```
{
    "imageSetsMetadataSummaries": [{
        "imageSetId": "09876543210987654321098765432109",
        "createdAt": "2022-12-06T21:40:59.429000+00:00",
        "version": 1,
        "DICOMTags": {
            "DICOMStudyId": "2011201407",
            "DICOMStudyDate": "19991122",
            "DICOMPatientSex": "F",
            "DICOMStudyInstanceUID": "1.2.840.99999999.84710745.943275268089",
            "DICOMPatientBirthDate": "19201120",
            "DICOMStudyDescription": "UNKNOWN",
            "DICOMPatientId": "SUBJECT08701",
            "DICOMPatientName": "Melissa844 Huel628",
            "DICOMNumberOfStudyRelatedInstances": 1,
            "DICOMStudyTime": "140728",
            "DICOMNumberOfStudyRelatedSeries": 1
        },
        "updatedAt": "2022-12-06T21:40:59.429000+00:00"
    }]
}
```

Esempio 3: come cercare set di immagini con un operatore BETWEEN utilizzando createdAt (in precedenza gli studi temporali erano persistenti)

Il seguente esempio di `search-image-sets` codice cerca i set di immagini con DICOM Studies persistenti HealthImaging tra gli intervalli di tempo del fuso orario UTC.

Nota: fornisci createdAt nel formato dell'esempio ("1985-04-12T23:20:50.52Z").

```
aws medical-imaging search-image-sets \
    --datastore-id 12345678901234567890123456789012 \
    --search-criteria  file://search-criteria.json
```

Contenuto di `search-criteria.json`

```
{
    "filters": [{
        "values": [{
            "createdAt": "1985-04-12T23:20:50.52Z"
        },
        {
            "createdAt": "2022-04-12T23:20:50.52Z"
        }],
        "operator": "BETWEEN"
    }]
}
```

Output:

```
{
    "imageSetsMetadataSummaries": [{
        "imageSetId": "09876543210987654321098765432109",
        "createdAt": "2022-12-06T21:40:59.429000+00:00",
        "version": 1,
        "DICOMTags": {
            "DICOMStudyId": "2011201407",
            "DICOMStudyDate": "19991122",
            "DICOMPatientSex": "F",
            "DICOMStudyInstanceUID": "1.2.840.99999999.84710745.943275268089",
            "DICOMPatientBirthDate": "19201120",
            "DICOMStudyDescription": "UNKNOWN",
            "DICOMPatientId": "SUBJECT08701",
            "DICOMPatientName": "Melissa844 Huel628",
            "DICOMNumberOfStudyRelatedInstances": 1,
            "DICOMStudyTime": "140728",
            "DICOMNumberOfStudyRelatedSeries": 1
        },
        "lastUpdatedAt": "2022-12-06T21:40:59.429000+00:00"
    }]
```

```
}
```

Esempio 4: cercare set di immagini con un operatore EQUAL su DICOMSeries instanceUID e BETWEEN su updatedAt e ordinare la risposta in ordine ASC nel campo updatedAt

Il seguente esempio di `search-image-sets` codice cerca i set di immagini con un operatore EQUAL su DICOMSeries instanceUID e BETWEEN su updatedAt e ordina la risposta in ordine ASC sul campo updatedAt.

Nota: fornisci updatedAt nel formato dell'esempio ("1985-04-12T23:20:50.52Z").

```
aws medical-imaging search-image-sets \
    --datastore-id 12345678901234567890123456789012 \
    --search-criteria  file://search-criteria.json
```

Contenuto di `search-criteria.json`

```
{
    "filters": [{
        "values": [{
            "updatedAt": "2024-03-11T15:00:05.074000-07:00"
        }, {
            "updatedAt": "2024-03-11T16:00:05.074000-07:00"
        }],
        "operator": "BETWEEN"
    }, {
        "values": [{
            "DICOMSeriesInstanceUID": "1.2.840.99999999.84710745.943275268089"
        }],
        "operator": "EQUAL"
    }],
    "sort": {
        "sortField": "updatedAt",
        "sortOrder": "ASC"
    }
}
```

Output:

```
{
    "imageSetsMetadataSummaries": [{
```

```
        "imageSetId": "09876543210987654321098765432109",
        "createdAt": "2022-12-06T21:40:59.429000+00:00",
        "version": 1,
        "DICOMTags": {
            "DICOMStudyId": "2011201407",
            "DICOMStudyDate": "19991122",
            "DICOMPatientSex": "F",
            "DICOMStudyInstanceUID": "1.2.840.99999999.84710745.943275268089",
            "DICOMPatientBirthDate": "19201120",
            "DICOMStudyDescription": "UNKNOWN",
            "DICOMPatientId": "SUBJECT08701",
            "DICOMPatientName": "Melissa844 Huel628",
            "DICOMNumberOfStudyRelatedInstances": 1,
            "DICOMStudyTime": "140728",
            "DICOMNumberOfStudyRelatedSeries": 1
        },
        "lastUpdatedAt": "2022-12-06T21:40:59.429000+00:00"
    }]
}
```

Per ulteriori informazioni, consulta Searching image sets nella Developer Guide.AWS HealthImaging

- Per i dettagli sull'API, consulta SearchImageSets AWS CLICommand Reference.

`start-dicom-import-job`

Il seguente esempio di codice mostra come utilizzare`start-dicom-import-job`.

AWS CLI

Come avviare un processo di importazione DICOM

L'esempio di codice `start-dicom-import-job` seguente avvia un processo di importazione DICOM.

```
aws medical-imaging start-dicom-import-job \
    --job-name "my-job" \
    --datastore-id "12345678901234567890123456789012" \
    --input-s3-uri "s3://medical-imaging-dicom-input/dicom_input/" \
    --output-s3-uri "s3://medical-imaging-output/job_output/" \
    --data-access-role-arn "arn:aws:iam::123456789012:role/ImportJobDataAccessRole"
```

Output:

```
{
    "datastoreId": "12345678901234567890123456789012",
    "jobId": "09876543210987654321098765432109",
    "jobStatus": "SUBMITTED",
    "submittedAt": "2022-08-12T11:28:11.152000+00:00"
}
```

Per ulteriori informazioni, consulta Avvio di un processo di importazione nella Guida per gli AWS HealthImaging sviluppatori.

- Per i dettagli sull'API, consulta Start DICOMImport Job in AWS CLI Command Reference.

`tag-resource`

Il seguente esempio di codice mostra come utilizzare`tag-resource`.

AWS CLI

Esempio 1: come applicare tag a un datastore

L'esempio `tag-resource` seguente aggiunge tag a un datastore.

```
aws medical-imaging tag-resource \
  --resource-arn "arn:aws:medical-imaging:us-
east-1:123456789012:datastore/12345678901234567890123456789012" \
  --tags '{"Deployment":"Development"}'
```

Questo comando non produce alcun output.

Esempio 2: come applicare tag a un set di immagini

L'esempio `tag-resource` seguente tagga un set di immagini.

```
aws medical-imaging tag-resource \
    --resource-arn "arn:aws:medical-imaging:us-
east-1:123456789012:datastore/12345678901234567890123456789012/
imageset/18f88ac7870584f58d56256646b4d92b" \
    --tags '{"Deployment":"Development"}'
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Tagging resources with AWS HealthImaging nella AWS HealthImaging Developer Guide.

- Per i dettagli sull'API, consulta TagResource AWS CLICommand Reference.

`untag-resource`

Il seguente esempio di codice mostra come utilizzare`untag-resource`.

AWS CLI

Esempio 1: come rimuovere i tag da un datastore

L'esempio di codice `untag-resource` seguente rimuove i tag da un datastore.

```
aws medical-imaging untag-resource \
    --resource-arn "arn:aws:medical-imaging:us-
east-1:123456789012:datastore/12345678901234567890123456789012" \
    --tag-keys '["Deployment"]'
```

Questo comando non produce alcun output.

Esempio 2: come rimuovere i tag da un set di immagini

L'esempio `untag-resource` seguente rimuove i tag da un set di immagini.

```
aws medical-imaging untag-resource \
    --resource-arn "arn:aws:medical-imaging:us-
east-1:123456789012:datastore/12345678901234567890123456789012/
imageset/18f88ac7870584f58d56256646b4d92b" \
    --tag-keys '["Deployment"]'
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Tagging resources with AWS HealthImaging nella AWS HealthImaging Developer Guide.

- Per i dettagli sull'API, consulta UntagResource AWS CLICommand Reference.

`update-image-set-metadata`

Il seguente esempio di codice mostra come utilizzare`update-image-set-metadata`.

AWS CLI

Esempio 1: come inserire o aggiornare un attributo nei metadati del set di immagini

L'esempio `update-image-set-metadata` seguente inserisce o aggiorna un attributo nei metadati del set di immagini.

```
aws medical-imaging update-image-set-metadata \
    --datastore-id 12345678901234567890123456789012 \
    --image-set-id ea92b0d8838c72a3f25d00d13616f87e \
    --latest-version-id 1 \
    --cli-binary-format raw-in-base64-out \
    --update-image-set-metadata-updates file://metadata-updates.json
```

Contenuto di `metadata-updates.json`

```
{
    "DICOMUpdates": {
        "updatableAttributes": "{\"SchemaVersion\":1.1,\"Patient\":{\"DICOM\":
{\"PatientName\":\"MX^MX\"}}}"
    }
}
```

Output:

```
{
    "latestVersionId": "2",
    "imageSetWorkflowStatus": "UPDATING",
    "updatedAt": 1680042257.908,
    "imageSetId": "ea92b0d8838c72a3f25d00d13616f87e",
    "imageSetState": "LOCKED",
    "createdAt": 1680027126.436,
    "datastoreId": "12345678901234567890123456789012"
}
```

Esempio 2: come rimuovere un attributo dai metadati del set di immagini

L'esempio `update-image-set-metadata` seguente rimuove un attributo dai metadati del set di immagini.

```
aws medical-imaging update-image-set-metadata \
```

```
    --datastore-id 12345678901234567890123456789012 \
    --image-set-id ea92b0d8838c72a3f25d00d13616f87e \
    --latest-version-id 1 \
    --cli-binary-format raw-in-base64-out \
    --update-image-set-metadata-updates file://metadata-updates.json
```

Contenuto di `metadata-updates.json`

```
{
    "DICOMUpdates": {
        "removableAttributes": "{\"SchemaVersion\":1.1,\"Study\":{\"DICOM\":
{\"StudyDescription\":\"CHEST\"}}}"
    }
}
```

Output:

```
{
    "latestVersionId": "2",
    "imageSetWorkflowStatus": "UPDATING",
    "updatedAt": 1680042257.908,
    "imageSetId": "ea92b0d8838c72a3f25d00d13616f87e",
    "imageSetState": "LOCKED",
    "createdAt": 1680027126.436,
    "datastoreId": "12345678901234567890123456789012"
}
```

Esempio 3: come rimuovere un'istanza dai metadati del set di immagini

L'esempio `update-image-set-metadata` seguente rimuove un'istanza dai metadati del set di immagini.

```
aws medical-imaging update-image-set-metadata \
    --datastore-id 12345678901234567890123456789012 \
    --image-set-id ea92b0d8838c72a3f25d00d13616f87e \
    --latest-version-id 1 \
    --cli-binary-format raw-in-base64-out \
    --update-image-set-metadata-updates file://metadata-updates.json \
    --force
```

Contenuto di `metadata-updates.json`

```
{
    "DICOMUpdates": {
        "removableAttributes": "{\"SchemaVersion\": 1.1,\"Study\": {\"Series
\": {\"1.1.1.1.1.1.12345.123456789012.123.12345678901234.1\": {\"Instances\":
 {\"1.1.1.1.1.1.12345.123456789012.123.12345678901234.1\": {}}}}}}"
    }
}
```

Output:

```
{
    "latestVersionId": "2",
    "imageSetWorkflowStatus": "UPDATING",
    "updatedAt": 1680042257.908,
    "imageSetId": "ea92b0d8838c72a3f25d00d13616f87e",
    "imageSetState": "LOCKED",
    "createdAt": 1680027126.436,
    "datastoreId": "12345678901234567890123456789012"
}
```

Esempio 4: come ripristinare la versione precedente di un set di immagini

L'`update-image-set-metadata`esempio seguente mostra come ripristinare una versione precedente di un set di immagini. CopyImageSet e UpdateImageSetMetadata le azioni creano nuove versioni dei set di immagini.

```
aws medical-imaging update-image-set-metadata \
    --datastore-id 12345678901234567890123456789012 \
    --image-set-id 53d5fdb05ca4d46ac7ca64b06545c66e \
    --latest-version-id 3 \
    --cli-binary-format raw-in-base64-out \
    --update-image-set-metadata-updates '{"revertToVersionId": "1"}'
```

Output:

```
{
    "datastoreId": "12345678901234567890123456789012",
    "imageSetId": "53d5fdb05ca4d46ac7ca64b06545c66e",
    "latestVersionId": "4",
    "imageSetState": "LOCKED",
    "imageSetWorkflowStatus": "UPDATING",
```

```
    "createdAt": 1680027126.436,
    "updatedAt": 1680042257.908
}
```

Esempio 5: come aggiungere un elemento di dati DICOM privato a un'istanza

L'esempio `update-image-set-metadata` seguente aggiunge un elemento privato a un'istanza specificata all'interno di un set di immagini. Lo standard DICOM consente di utilizzare elementi di dati privati nella comunicazione di informazioni che non possono essere contenute in elementi di dati standard. È possibile creare, aggiornare ed eliminare elementi di dati privati con l'UpdateImageSetMetadata azione.

```
aws medical-imaging update-image-set-metadata \
    --datastore-id 12345678901234567890123456789012 \
    --image-set-id 53d5fdb05ca4d46ac7ca64b06545c66e \
    --latest-version-id 1 \
    --cli-binary-format raw-in-base64-out \
    --force \
    --update-image-set-metadata-updates file://metadata-updates.json
```

Contenuto di `metadata-updates.json`

```
{
    "DICOMUpdates": {
        "updatableAttributes": "{\"SchemaVersion\": 1.1,\"Study\": {\"Series
\": {\"1.1.1.1.1.1.12345.123456789012.123.12345678901234.1\": {\"Instances
\": {\"1.1.1.1.1.1.12345.123456789012.123.12345678901234.1\": {\"DICOM\":
 {\"001910F9\": \"97\"},\"DICOMVRs\": {\"001910F9\": \"DS\"}}}}}}"
    }
}
```

Output:

```
{
    "latestVersionId": "2",
    "imageSetWorkflowStatus": "UPDATING",
    "updatedAt": 1680042257.908,
    "imageSetId": "53d5fdb05ca4d46ac7ca64b06545c66e",
    "imageSetState": "LOCKED",
    "createdAt": 1680027126.436,
    "datastoreId": "12345678901234567890123456789012"
```

```
}
```

Esempio 6: come aggiornare un elemento di dati DICOM privato a un'istanza

L'esempio `update-image-set-metadata` seguente aggiorna il valore di un elemento dati privato appartenente a un'istanza all'interno di un set di immagini.

```
aws medical-imaging update-image-set-metadata \
    --datastore-id 12345678901234567890123456789012 \
    --image-set-id 53d5fdb05ca4d46ac7ca64b06545c66e \
    --latest-version-id 1 \
    --cli-binary-format raw-in-base64-out \
    --force \
    --update-image-set-metadata-updates file://metadata-updates.json
```

Contenuto di `metadata-updates.json`

```
{
    "DICOMUpdates": {
        "updatableAttributes": "{\"SchemaVersion\": 1.1,\"Study\": {\"Series
\": {\"1.1.1.1.1.1.12345.123456789012.123.12345678901234.1\": {\"Instances
\": {\"1.1.1.1.1.1.12345.123456789012.123.12345678901234.1\": {\"DICOM\":
 {\"00091001\": \"GE_GENESIS_DD\"}}}}}}"
    }
}
```

Output:

```
{
    "latestVersionId": "2",
    "imageSetWorkflowStatus": "UPDATING",
    "updatedAt": 1680042257.908,
    "imageSetId": "53d5fdb05ca4d46ac7ca64b06545c66e",
    "imageSetState": "LOCKED",
    "createdAt": 1680027126.436,
    "datastoreId": "12345678901234567890123456789012"
}
```

Esempio 7: aggiornare un SOPInstance UID con il parametro force

L'`update-image-set-metadata`esempio seguente mostra come aggiornare un SOPInstance UID, utilizzando il parametro force per sovrascrivere i vincoli dei metadati DICOM.

```
aws medical-imaging update-image-set-metadata \
        --datastore-id 12345678901234567890123456789012 \
        --image-set-id 53d5fdb05ca4d46ac7ca64b06545c66e \
        --latest-version-id 1 \
        --cli-binary-format raw-in-base64-out \
        --force \
        --update-image-set-metadata-updates file://metadata-updates.json
```

Contenuto di `metadata-updates.json`

```
{
    "DICOMUpdates": {
        "updatableAttributes": "{\"SchemaVersion\":1.1,\"Study\":{\"Series\":
{\"1.3.6.1.4.1.5962.99.1.3633258862.2104868982.1369432891697.3656.0\":{\"Instances
\":{\"1.3.6.1.4.1.5962.99.1.3633258862.2104868982.1369432891697.3659.0\":{\"DICOM\":
{\"SOPInstanceUID\":
\"1.3.6.1.4.1.5962.99.1.3633258862.2104868982.1369432891697.3659.9\"}}}}}}"
    }
}
```

Output:

```
{
    "latestVersionId": "2",
    "imageSetWorkflowStatus": "UPDATING",
    "updatedAt": 1680042257.908,
    "imageSetId": "53d5fdb05ca4d46ac7ca64b06545c66e",
    "imageSetState": "LOCKED",
    "createdAt": 1680027126.436,
    "datastoreId": "12345678901234567890123456789012"
}
```

Per ulteriori informazioni, consultate Updating image set metadata nella Developer Guide.AWS HealthImaging

- Per i dettagli sull'API, consulta UpdateImageSetMetadata AWS CLICommand Reference.

HealthLake esempi utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with HealthLake.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`create-fhir-datastore`

Il seguente esempio di codice mostra come utilizzare`create-fhir-datastore`.

AWS CLI

Esempio 1: creare un data store abilitato per SIGv4 HealthLake

L'`create-fhir-datastore`esempio seguente mostra come creare un nuovo archivio dati in. AWS HealthLake

```
aws healthlake create-fhir-datastore \
    --datastore-type-version R4 \
    --datastore-name "FhirTestDatastore"
```

Output:

```
{
    "DatastoreEndpoint": "https://healthlake.us-east-1.amazonaws.com/datastore/(Data
 store ID)/r4/",
    "DatastoreArn": "arn:aws:healthlake:us-east-1:(AWS Account ID):datastore/(Data
 store ID)",
    "DatastoreStatus": "CREATING",
    "DatastoreId": "(Data store ID)"
}
```

Esempio 2: creare uno SMART su un data store abilitato per FHIR HealthLake

L'`create-fhir-datastore`esempio seguente mostra come creare un nuovo SMART su un data store abilitato per FHIR in. AWS HealthLake

```
aws healthlake create-fhir-datastore \
    --datastore-name "your-data-store-name" \
    --datastore-type-version R4 \
    --preload-data-config PreloadDataType="SYNTHEA" \
    --sse-configuration '{ "KmsEncryptionConfig": {  "CmkType":
 "CUSTOMER_MANAGED_KMS_KEY", "KmsKeyId": "arn:aws:kms:us-east-1:your-account-id:key/
your-key-id" } }' \
    --identity-provider-configuration  file://identity_provider_configuration.json
```

Contenuto di `identity_provider_configuration.json`:

```
{
    "AuthorizationStrategy": "SMART_ON_FHIR_V1",
    "FineGrainedAuthorizationEnabled": true,
    "IdpLambdaArn": "arn:aws:lambda:your-region:your-account-id:function:your-
lambda-name",
    "Metadata": "{\"issuer\":\"https://ehr.example.com\", \"jwks_uri\":\"https://
ehr.example.com/.well-known/jwks.json\",\"authorization_endpoint\":\"https://
ehr.example.com/auth/authorize\",\"token_endpoint\":\"https://ehr.token.com/auth/
token\",\"token_endpoint_auth_methods_supported\":[\"client_secret_basic\",\"foo\"],
\"grant_types_supported\":[\"client_credential\",\"foo\"],\"registration_endpoint\":
\"https://ehr.example.com/auth/register\",\"scopes_supported\":[\"openId\",\"profile
\",\"launch\"],\"response_types_supported\":[\"code\"],\"management_endpoint\":
\"https://ehr.example.com/user/manage\",\"introspection_endpoint\":\"https://
ehr.example.com/user/introspect\",\"revocation_endpoint\":\"https://ehr.example.com/
user/revoke\",\"code_challenge_methods_supported\":[\"S256\"],\"capabilities\":
[\"launch-ehr\",\"sso-openid-connect\",\"client-public\"]}"
}
```

Output:

```
{
    "DatastoreEndpoint": "https://healthlake.us-east-1.amazonaws.com/datastore/(Data
 store ID)/r4/",
    "DatastoreArn": "arn:aws:healthlake:us-east-1:(AWS Account ID):datastore/(Data
 store ID)",
    "DatastoreStatus": "CREATING",
    "DatastoreId": "(Data store ID)"
}
```

Per ulteriori informazioni, consulta Creazione e monitoraggio di un data store FHIR nella Developer Guide.AWS HealthLake

- Per i dettagli sull'API, consulta Create FHIRDatastore in AWS CLI Command Reference.

`delete-fhir-datastore`

Il seguente esempio di codice mostra come utilizzare`delete-fhir-datastore`.

AWS CLI

Come eliminare un datastore FHIR

L'`delete-fhir-datastore`esempio seguente mostra come eliminare un archivio dati e tutto il suo contenuto. AWS HealthLake

```
aws healthlake delete-fhir-datastore \
    --datastore-id (Data store ID)
```

Output:

```
{
    "DatastoreEndpoint": "https://healthlake.us-east-1.amazonaws.com/datastore/(Data
 store ID)/r4/",
    "DatastoreArn": "arn:aws:healthlake:us-east-1:(AWS Account ID):datastore/(Data
 store ID)",
    "DatastoreStatus": "DELETING",
    "DatastoreId": "(Data store ID)"
}
```

Per ulteriori informazioni, consulta Creazione e monitoraggio di un archivio dati FHIR < https:// docs.aws.amazon.com/healthlake/ latest/devguide/working -with-FHIR-HealthLake.html> nella Guida per gli sviluppatori.AWS HealthLake

- Per FHIRDatastore i dettagli sull'AWS CLI API, consulta Delete in Command Reference.

`describe-fhir-datastore`

Il seguente esempio di codice mostra come utilizzare`describe-fhir-datastore`.

AWS CLI

Come descrivere un datastore FHIR

L'`describe-fhir-datastore`esempio seguente mostra come trovare le proprietà di un archivio dati in AWS HealthLake.

```
aws healthlake describe-fhir-datastore \
    --datastore-id "1f2f459836ac6c513ce899f9e4f66a59"
```

Output:

```
{
    "DatastoreProperties": {
        "PreloadDataConfig": {
            "PreloadDataType": "SYNTHEA"
        },
        "SseConfiguration": {
            "KmsEncryptionConfig": {
                "CmkType": "CUSTOMER_MANAGED_KMS_KEY",
                "KmsKeyId": "arn:aws:kms:us-east-1:123456789012:key/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
            }
        },
        "DatastoreName": "Demo",
        "DatastoreArn": "arn:aws:healthlake:us-east-1:<AWS Account ID>:datastore/
<Data store ID>",
        "DatastoreEndpoint": "https://healthlake.us-east-1.amazonaws.com/datastore/
<Data store ID>/r4/",
        "DatastoreStatus": "ACTIVE",
        "DatastoreTypeVersion": "R4",
        "CreatedAt": 1603761064.881,
        "DatastoreId": "<Data store ID>",
        "IdentityProviderConfiguration": {
            "AuthorizationStrategy": "AWS_AUTH",
            "FineGrainedAuthorizationEnabled": false
        }
    }
}
```

Per ulteriori informazioni, consulta Creazione e monitoraggio di un archivio dati FHIR nella AWS HealthLake Developer Guide.

- Per i dettagli sull'API, consulta Descrivi FHIRDatastore in AWS CLI Command Reference.

`describe-fhir-export-job`

Il seguente esempio di codice mostra come utilizzare`describe-fhir-export-job`.

AWS CLI

Come descrivere un processo di esportazione FHIR

L'`describe-fhir-export-job`esempio seguente mostra come trovare le proprietà di un processo di esportazione FHIR in AWS HealthLake.

```
aws healthlake describe-fhir-export-job \
    --datastore-id (Data store ID) \
    --job-id 9b9a51943afaedd0a8c0c26c49135a31
```

Output:

```
{
    "ExportJobProperties": {
        "DataAccessRoleArn": "arn:aws:iam::(AWS Account ID):role/(Role Name)",
        "JobStatus": "IN_PROGRESS",
        "JobId": "9009813e9d69ba7cf79bcb3468780f16",
        "SubmitTime": "2024-11-20T11:31:46.672000-05:00",
        "EndTime": "2024-11-20T11:34:01.636000-05:00",
        "OutputDataConfig": {
            "S3Configuration": {
            "S3Uri": "s3://(Bucket Name)/(Prefix Name)/",
            "KmsKeyId": "arn:aws:kms:us-east-1:012345678910:key/d330e7fc-b56c-4216-
a250-f4c43ef46e83"
        }

        },
        "DatastoreId": "(Data store ID)"
    }
}
```

Per ulteriori informazioni, consulta Esportazione di file da un data store FHIR nella Guida per gli AWS HealthLake sviluppatori.

- Per i dettagli sull'API, consulta Descrivi FHIRExport Job in AWS CLI Command Reference.

`describe-fhir-import-job`

Il seguente esempio di codice mostra come utilizzare`describe-fhir-import-job`.

AWS CLI

Come descrivere un processo di importazione FHIR

L'`describe-fhir-import-job`esempio seguente mostra come apprendere le proprietà di un processo di importazione FHIR utilizzando AWS HealthLake.

```
aws healthlake describe-fhir-import-job \
    --datastore-id (Data store ID) \
    --job-id c145fbb27b192af392f8ce6e7838e34f
```

Output:

```
{
    "ImportJobProperties": {
    "InputDataConfig": {
        "S3Uri": "s3://(Bucket Name)/(Prefix Name)/"
        { "arrayitem2": 2 }
    },
    "DataAccessRoleArn": "arn:aws:iam::(AWS Account ID):role/(Role Name)",
    "JobStatus": "COMPLETED",
    "JobId": "c145fbb27b192af392f8ce6e7838e34f",
    "SubmitTime": 1606272542.161,
    "EndTime": 1606272609.497,
    "DatastoreId": "(Data store ID)"
    }
}
```

Per ulteriori informazioni, consulta Importazione di file in un data store FHIR nella Guida per gli AWS HealthLake sviluppatori.

- Per i dettagli sull'API, consulta Descrivi FHIRImport Job in AWS CLI Command Reference.

`list-fhir-datastores`

Il seguente esempio di codice mostra come utilizzare`list-fhir-datastores`.

AWS CLI

Come elencare i datastore FHIR

L'`list-fhir-datastores`esempio seguente mostra come utilizzare il comando e come gli utenti possono filtrare i risultati in base allo stato del data store in AWS HealthLake.

```
aws healthlake list-fhir-datastores \
    --filter DatastoreStatus=ACTIVE
```

Output:

```
{
    "DatastorePropertiesList": [
    {
        "PreloadDataConfig": {
            "PreloadDataType": "SYNTHEA"
        },
        "SseConfiguration": {
            "KmsEncryptionConfig": {
                "CmkType": "CUSTOMER_MANAGED_KMS_KEY",
                "KmsKeyId": "arn:aws:kms:us-east-1:123456789012:key/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
            }
        },
        "DatastoreName": "Demo",
        "DatastoreArn": "arn:aws:healthlake:us-east-1:<AWS Account ID>:datastore/
<Data store ID>",
        "DatastoreEndpoint": "https://healthlake.us-east-1.amazonaws.com/datastore/
<Data store ID>/r4/",
        "DatastoreStatus": "ACTIVE",
        "DatastoreTypeVersion": "R4",
        "CreatedAt": 1603761064.881,
        "DatastoreId": "<Data store ID>",
        "IdentityProviderConfiguration": {
            "AuthorizationStrategy": "AWS_AUTH",
            "FineGrainedAuthorizationEnabled": false
        }
    }
    ]
}
```

Per ulteriori informazioni, consulta Creazione e monitoraggio di un data store FHIR nella Guida per gli AWS HealthLake sviluppatori.

- Per i dettagli sull'API, consulta List FHIRDatastores in AWS CLI Command Reference.

`list-fhir-export-jobs`

Il seguente esempio di codice mostra come utilizzare`list-fhir-export-jobs`.

AWS CLI

Come elencare tutti i processi di esportazione FHIR

L'esempio `list-fhir-export-jobs` seguente mostra come utilizzare il comando per visualizzare un elenco di processi di esportazione associati a un account.

```
aws healthlake list-fhir-export-jobs \
    --datastore-id (Data store ID) \
    --submitted-before (DATE like 2024-10-13T19:00:00Z)\
    --submitted-after (DATE like 2020-10-13T19:00:00Z )\
    --job-name "FHIR-EXPORT" \
    --job-status SUBMITTED  \
    --max-results (Integer between 1 and 500)
```

Output:

```
{
    "ExportJobPropertiesList": [
        {
            "ExportJobProperties": {
                "OutputDataConfig": {
                    "S3Uri": "s3://(Bucket Name)/(Prefix Name)/",
                    "S3Configuration": {
                        "S3Uri": "s3://(Bucket Name)/(Prefix Name)/",
                        "KmsKeyId": "(KmsKey Id)"
                    }
                },
                "DataAccessRoleArn": "arn:aws:iam::(AWS Account ID):role/(Role
 Name)",
                "JobStatus": "COMPLETED",
                "JobId": "c145fbb27b192af392f8ce6e7838e34f",
                "JobName": "FHIR-EXPORT",
                "SubmitTime": "2024-11-20T11:31:46.672000-05:00",
```

```
                "EndTime": "2024-11-20T11:34:01.636000-05:00",
                "DatastoreId": "(Data store ID)"
            }
        }
    ]
}
```

Per ulteriori informazioni, consulta Esportazione di file da un data store FHIR nella Guida per gli AWS HealthLake sviluppatori.

- Per i dettagli sull'API, consulta List FHIRExport Jobs in AWS CLI Command Reference.

`list-fhir-import-jobs`

Il seguente esempio di codice mostra come utilizzare`list-fhir-import-jobs`.

AWS CLI

Come elencare tutti i processi di importazione FHIR

L'esempio `list-fhir-import-jobs` seguente mostra come utilizzare il comando per visualizzare un elenco di tutti i processi di importazione associati a un account.

```
aws healthlake list-fhir-import-jobs \
    --datastore-id (Data store ID) \
    --submitted-before (DATE like 2024-10-13T19:00:00Z) \
    --submitted-after (DATE like 2020-10-13T19:00:00Z ) \
    --job-name "FHIR-IMPORT" \
    --job-status SUBMITTED  \
    -max-results (Integer between 1 and 500)
```

Output:

```
{
    "ImportJobPropertiesList": [
        {
            "JobId": "c0fddbf76f238297632d4aebdbfc9ddf",
            "JobStatus": "COMPLETED",
            "SubmitTime": "2024-11-20T10:08:46.813000-05:00",
            "EndTime": "2024-11-20T10:10:09.093000-05:00",
            "DatastoreId": "(Data store ID)",
            "InputDataConfig": {
                "S3Uri": "s3://(Bucket Name)/(Prefix Name)/"
```

```
            },
            "JobOutputDataConfig": {
                "S3Configuration": {
                    "S3Uri": "s3://(Bucket Name)/
import/6407b9ae4c2def3cb6f1a46a0c599ec0-FHIR_IMPORT-
c0fddbf76f238297632d4aebdbfc9ddf/",
                    "KmsKeyId": "arn:aws:kms:us-east-1:123456789012:key/b7f645cb-
e564-4981-8672-9e012d1ff1a0"
                }
            },
            "JobProgressReport": {
                "TotalNumberOfScannedFiles": 1,
                "TotalSizeOfScannedFilesInMB": 0.001798,
                "TotalNumberOfImportedFiles": 1,
                "TotalNumberOfResourcesScanned": 1,
                "TotalNumberOfResourcesImported": 1,
                "TotalNumberOfResourcesWithCustomerError": 0,
                "TotalNumberOfFilesReadWithCustomerError": 0,
                "Throughput": 0.0
            },
            "DataAccessRoleArn": "arn:aws:iam::(AWS Account ID):role/(Role Name)"
        }
    ]
}
```

Per ulteriori informazioni, consulta Importazione di file nell'archivio dati FHIR nella Guida per gli AWS HealthLake sviluppatori.

- Per i dettagli sull'API, consulta List FHIRImport Jobs in AWS CLI Command Reference.

`list-tags-for-resource`

Il seguente esempio di codice mostra come utilizzare`list-tags-for-resource`.

AWS CLI

Come elencare i tag di un datastore

L'esempio `list-tags-for-resource` seguente elenca i tag associati al datastore specificato:

```
aws healthlake list-tags-for-resource \
    --resource-arn "arn:aws:healthlake:us-east-1:123456789012:datastore/
fhir/0725c83f4307f263e16fd56b6d8ebdbe"
```

Output:

```
{
    "tags": {
        "key": "value",
        "key1": "value1"
    }
}
```

Per ulteriori informazioni, consulta Tagging resources AWS HealthLake nella AWS HealthLake Developer Guide.

- Per i dettagli sull'API, consulta ListTagsForResource AWS CLICommand Reference.

`start-fhir-export-job`

Il seguente esempio di codice mostra come utilizzare`start-fhir-export-job`.

AWS CLI

Come avviare un processo di esportazione FHIR

L'`start-fhir-export-job`esempio seguente mostra come avviare un processo di esportazione FHIR utilizzando AWS HealthLake.

```
aws healthlake start-fhir-export-job \
    --output-data-config '{"S3Configuration": {"S3Uri":"s3://(Bucket Name)/(Prefix Name)/","KmsKeyId":"arn:aws:kms:us-east-1:012345678910:key/d330e7fc-b56c-4216-a250-f4c43ef46e83"}}' \
    --datastore-id (Data store ID) \
    --data-access-role-arn arn:aws:iam::(AWS Account ID):role/(Role Name)
```

Output:

```
{
    "DatastoreId": "(Data store ID)",
    "JobStatus": "SUBMITTED",
    "JobId": "9b9a51943afaedd0a8c0c26c49135a31"
}
```

Per ulteriori informazioni, consulta Esportazione di file da un data store FHIR nella Guida per gli AWS HealthLake sviluppatori.

- Per i dettagli sull'API, consulta Start FHIRExport Job in AWS CLI Command Reference.

`start-fhir-import-job`

Il seguente esempio di codice mostra come utilizzare`start-fhir-import-job`.

AWS CLI

Come avviare un processo di importazione FHIR

L'`start-fhir-import-job`esempio seguente mostra come avviare un processo di importazione FHIR utilizzando AWS HealthLake.

```
aws healthlake start-fhir-import-job \
    --input-data-config S3Uri="s3://(Bucket Name)/(Prefix Name)/" \
    --job-output-data-config '{"S3Configuration": {"S3Uri":"s3://(Bucket Name)/
(Prefix Name)/","KmsKeyId":"arn:aws:kms:us-east-1:012345678910:key/d330e7fc-
b56c-4216-a250-f4c43ef46e83"}}' \
    --datastore-id (Data store ID) \
    --data-access-role-arn "arn:aws:iam::(AWS Account ID):role/(Role Name)"
```

Output:

```
{
    "DatastoreId": "(Data store ID)",
    "JobStatus": "SUBMITTED",
    "JobId": "c145fbb27b192af392f8ce6e7838e34f"
}
```

Per ulteriori informazioni, consulta Importazione di file in un data store FHIR nella Guida per gli AWS HealthLake sviluppatori.

- Per i dettagli sull'API, consulta Start FHIRImport Job in AWS CLI Command Reference.

`tag-resource`

Il seguente esempio di codice mostra come utilizzare`tag-resource`.

AWS CLI

Come aggiungere un tag al datastore

L'esempio `tag-resource` seguente mostra come aggiungere un tag al datastore.

```
aws healthlake tag-resource \
    --resource-arn "arn:aws:healthlake:us-east-1:123456789012:datastore/
fhir/0725c83f4307f263e16fd56b6d8ebdbe" \
    --tags '[{"Key": "key1", "Value": "value1"}]'
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Aggiungere un tag a un data store nella Guida per gli AWS HealthLake sviluppatori. .

- Per i dettagli sull'API, consulta TagResource AWS CLICommand Reference.

`untag-resource`

Il seguente esempio di codice mostra come utilizzare`untag-resource`.

AWS CLI

Come rimuovere i tag da un datastore.

L'esempio `untag-resource` seguente mostra come rimuovere i tag da un datastore.

```
aws healthlake untag-resource \
    --resource-arn "arn:aws:healthlake:us-east-1:123456789012:datastore/fhir/
b91723d65c6fdeb1d26543a49d2ed1fa" \
    --tag-keys '["key1"]'
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Rimozione di tag da un data store nella Guida per gli AWS HealthLake sviluppatori.

- Per i dettagli sull'API, consulta UntagResource AWS CLICommand Reference.

HealthOmics esempi utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with HealthOmics.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`abort-multipart-read-set-upload`

Il seguente esempio di codice mostra come utilizzare`abort-multipart-read-set-upload`.

AWS CLI

Come arrestare un caricamento dei set di lettura in più parti

L'`abort-multipart-read-set-upload`esempio seguente interrompe il caricamento di un set di lettura in più parti nel vostro archivio di HealthOmics sequenze.

```
aws omics abort-multipart-read-set-upload \
    --sequence-store-id 0123456789 \
    --upload-id 1122334455
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consultate Caricamento diretto su un archivio di sequenze nella Guida per l'AWS HealthOmics utente.

- Per i dettagli sull'API, consulta AbortMultipartReadSetUpload AWS CLICommand Reference.

`accept-share`

Il seguente esempio di codice mostra come utilizzare`accept-share`.

AWS CLI

Come accettare una condivisione dei dati dell'archivio di analisi

L'`accept-share`esempio seguente accetta una condivisione di dati dell'archivio di HealthOmics analisi.

```
aws omics accept-share \
    ----share-id "495c21bedc889d07d0ab69d710a6841e-dd75ab7a1a9c384fa848b5bd8e5a7e0a"
```

Output:

```
{
    "status": "ACTIVATING"
}
```

Per ulteriori informazioni, consulta Condivisione tra account nella Guida per l'AWS HealthOmics utente.

- Per i dettagli sull'API, consulta AcceptShare AWS CLICommand Reference.

`batch-delete-read-set`

Il seguente esempio di codice mostra come utilizzare`batch-delete-read-set`.

AWS CLI

Come eliminare più set di lettura

L'esempio `batch-delete-read-set` seguente elimina due set di lettura.

```
aws omics batch-delete-read-set \
    --sequence-store-id 1234567890 \
    --ids 1234567890 0123456789
```

Se si verifica un errore durante l'eliminazione di uno dei set di lettura specificati, il servizio restituisce un elenco di errori.

```
{
    "errors": [
        {
            "code": "",
            "id": "0123456789",
            "message": "The specified readset does not exist."
        }
```

```
    ]
}
```

Per ulteriori informazioni, consulta Archiviazione Omics nella Guida per gli sviluppatori di Amazon Omics.

- Per i dettagli sull'API, consulta BatchDeleteReadSet AWS CLICommand Reference.

`cancel-annotation-import-job`

Il seguente esempio di codice mostra come utilizzare`cancel-annotation-import-job`.

AWS CLI

Come annullare un processo di importazione di annotazioni

L'esempio `cancel-annotation-import-job` seguente annulla un processo di importazione di annotazioni con ID `04f57618-xmpl-4fd0-9349-e5a85aefb997`.

```
aws omics cancel-annotation-import-job \
    --job-id 04f57618-xmpl-4fd0-9349-e5a85aefb997
```

Per ulteriori informazioni, consulta Omics Analytics nella Guida per gli sviluppatori di Amazon Omics.

- Per i dettagli sull'API, consulta CancelAnnotationImportJob AWS CLICommand Reference.

`cancel-run`

Il seguente esempio di codice mostra come utilizzare`cancel-run`.

AWS CLI

Come annullare un'esecuzione

L'esempio `cancel-run` seguente annulla un'esecuzione con ID `1234567`.

```
aws omics cancel-run \
    --id 1234567
```

Per ulteriori informazioni, consulta Esegui il ciclo di vita in un flusso di lavoro nella Guida per l'AWS HealthOmics utente.

- Per i dettagli sull'API, consulta AWS CLI Command CancelRunReference.

`cancel-variant-import-job`

Il seguente esempio di codice mostra come utilizzare`cancel-variant-import-job`.

AWS CLI

Come annullare un processo di importazione di varianti

L'esempio `cancel-variant-import-job` seguente annulla un processo di importazione di varianti con ID `69cb65d6-xmpl-4a4a-9025-4565794b684e`.

```
aws omics cancel-variant-import-job \
    --job-id 69cb65d6-xmpl-4a4a-9025-4565794b684e
```

Per ulteriori informazioni, consulta Omics Analytics nella Guida per gli sviluppatori di Amazon Omics.

- Per i dettagli sull'API, consulta CancelVariantImportJob AWS CLICommand Reference.

`complete-multipart-read-set-upload`

Il seguente esempio di codice mostra come utilizzare`complete-multipart-read-set-upload`.

AWS CLI

Come concludere un caricamento in più parti una volta caricati tutti i componenti.

L'esempio `complete-multipart-read-set-upload` seguente conclude un caricamento in più parti in un archivio di sequenze dopo il caricamento di tutti i componenti.

```
aws omics complete-multipart-read-set-upload \
    --sequence-store-id 0123456789 \
    --upload-id 1122334455 \
    --parts '[{"checksum":"gaCBQMe+rpCFZxLpoP6gydBoXaKKDA/
Vobh5zBDb4W4=","partNumber":1,"partSource":"SOURCE1"}]'
```

Output:

```
{
```

```
    "readSetId": "0000000001"
    "readSetId": "0000000002"
    "readSetId": "0000000003"
}
```

Per ulteriori informazioni, consulta Caricamento diretto su un archivio di sequenze nella Guida AWS HealthOmics per l'utente.

- Per i dettagli sull'API, consulta CompleteMultipartReadSetUpload AWS CLICommand Reference.

`create-annotation-store-version`

Il seguente esempio di codice mostra come utilizzare`create-annotation-store-version`.

AWS CLI

Come creare una nuova versione di un archivio di annotazioni

L'esempio `create-annotation-store-version` seguente crea una nuova versione di un archivio di annotazioni.

```
aws omics create-annotation-store-version \
    --name my_annotation_store \
    --version-name my_version
```

Output:

```
{
    "creationTime": "2023-07-21T17:15:49.251040+00:00",
    "id": "3b93cdef69d2",
    "name": "my_annotation_store",
    "reference": {
        "referenceArn": "arn:aws:omics:us-
west-2:555555555555:referenceStore/6505293348/reference/5987565360"
    },
    "status": "CREATING",
    "versionName": "my_version"
}
```

Per ulteriori informazioni, vedere Creazione di nuove versioni di archivi di annotazioni nella Guida per l'AWS HealthOmics utente.

- Per i dettagli sull'API, consulta CreateAnnotationStoreVersion AWS CLICommand Reference.

`create-annotation-store`

Il seguente esempio di codice mostra come utilizzare`create-annotation-store`.

AWS CLI

Esempio 1: come creare un archivio di annotazioni VCF

L'esempio `create-annotation-store` seguente crea un archivio di annotazioni in formato VCF.

```
aws omics create-annotation-store \
    --name my_ann_store \
    --store-format VCF \
    --reference referenceArn=arn:aws:omics:us-
west-2:123456789012:referenceStore/1234567890/reference/1234567890
```

Output:

```
{
    "creationTime": "2022-11-23T22:48:39.226492Z",
    "id": "0a91xmplc71f",
    "name": "my_ann_store",
    "reference": {
        "referenceArn": "arn:aws:omics:us-
west-2:123456789012:referenceStore/1234567890/reference/1234567890"
    },
    "status": "CREATING",
    "storeFormat": "VCF"
}
```

Esempio 2: come creare un archivio di annotazioni TSV

L'esempio `create-annotation-store` seguente crea un archivio di annotazioni in formato TSV.

```
aws omics create-annotation-store \
    --name tsv_ann_store \
    --store-format TSV \
```

```
    --reference referenceArn=arn:aws:omics:us-
west-2:123456789012:referenceStore/1234567890/reference/1234567890 \
    --store-options file://tsv-store-options.json
```

Le opzioni di formato per le annotazioni sono configurate mediante `tsv-store-options.json`.

```
{
    "tsvStoreOptions": {
        "annotationType": "CHR_START_END_ZERO_BASE",
        "formatToHeader": {
            "CHR": "chromosome",
            "START": "start",
            "END": "end"
        },
        "schema": [
            {
                "chromosome": "STRING"
            },
            {
                "start": "LONG"
            },
            {
                "end": "LONG"
            },
            {
                "name": "STRING"
            }
        ]
    }
}
```

Output:

```
{
    "creationTime": "2022-11-30T01:28:08.525586Z",
    "id": "861cxmpl96b0",
    "name": "tsv_ann_store",
    "reference": {
        "referenceArn": "arn:aws:omics:us-
west-2:123456789012:referenceStore/1234567890/reference/1234567890"
    },
    "status": "CREATING",
    "storeFormat": "TSV",
```

```
    "storeOptions": {
        "tsvStoreOptions": {
            "annotationType": "CHR_START_END_ZERO_BASE",
            "formatToHeader": {
                "CHR": "chromosome",
                "END": "end",
                "START": "start"
            },
            "schema": [
                {
                    "chromosome": "STRING"
                },
                {
                    "start": "LONG"
                },
                {
                    "end": "LONG"
                },
                {
                    "name": "STRING"
                }
            ]
        }
    }
}
```

Per ulteriori informazioni, consulta Omics Analytics nella Guida per gli sviluppatori di Amazon Omics.

- Per i dettagli sull'API, consulta CreateAnnotationStore AWS CLICommand Reference.

`create-multipart-read-set-upload`

Il seguente esempio di codice mostra come utilizzare`create-multipart-read-set-upload`.

AWS CLI

Come avviare un caricamento di un set di lettura in più parti.

L'esempio `create-multipart-read-set-upload` seguente avvia un caricamento di un set di lettura in più parti.

```
aws omics create-multipart-read-set-upload \
```

```
    --sequence-store-id 0123456789 \
    --name HG00146 \
    --source-file-type FASTQ \
    --subject-id mySubject\
    --sample-id mySample\
    --description "FASTQ for HG00146"\
    --generated-from "1000 Genomes"
```

Output:

```
{
    "creationTime": "2022-07-13T23:25:20Z",
    "description": "FASTQ for HG00146",
    "generatedFrom": "1000 Genomes",
    "name": "HG00146",
    "sampleId": "mySample",
    "sequenceStoreId": "0123456789",
    "sourceFileType": "FASTQ",
    "subjectId": "mySubject",
    "uploadId": "1122334455"
}
```

Per ulteriori informazioni, consulta Caricamento diretto su un archivio di sequenze nella Guida AWS HealthOmics per l'utente.

- Per i dettagli sull'API, consulta CreateMultipartReadSetUpload AWS CLICommand Reference.

`create-reference-store`

Il seguente esempio di codice mostra come utilizzare`create-reference-store`.

AWS CLI

Come creare un archivio di riferimenti

L'esempio `create-reference-store` seguente crea un archivio di riferimenti `my-ref-store`.

```
aws omics create-reference-store \
    --name my-ref-store
```

Output:

```
{
    "arn": "arn:aws:omics:us-west-2:123456789012:referenceStore/1234567890",
    "creationTime": "2022-11-22T22:13:25.947Z",
    "id": "1234567890",
    "name": "my-ref-store"
}
```

Per ulteriori informazioni, consulta Archiviazione Omics nella Guida per gli sviluppatori di Amazon Omics.

- Per i dettagli sull'API, consulta CreateReferenceStore AWS CLICommand Reference.

`create-run-group`

Il seguente esempio di codice mostra come utilizzare`create-run-group`.

AWS CLI

Come creare un gruppo di esecuzione

L'esempio `create-run-group` seguente crea un gruppo di esecuzione denominato `cram-converter`.

```
aws omics create-run-group \
    --name cram-converter \
    --max-cpus 20 \
    --max-gpus 10 \
    --max-duration 600 \
    --max-runs 5
```

Output:

```
{
    "arn": "arn:aws:omics:us-west-2:123456789012:runGroup/1234567",
    "id": "1234567",
    "tags": {}
}
```

Per ulteriori informazioni, vedere Creazione di gruppi di esecuzione nella Guida AWS HealthOmics per l'utente.

- Per i dettagli sull'API, consulta CreateRunGroup AWS CLICommand Reference.

`create-sequence-store`

Il seguente esempio di codice mostra come utilizzare`create-sequence-store`.

AWS CLI

Come creare un archivio di sequenze

L'esempio `create-sequence-store` seguente crea un archivio di sequenze.

```
aws omics create-sequence-store \
    --name my-seq-store
```

Output:

```
{
    "arn": "arn:aws:omics:us-west-2:123456789012:sequenceStore/1234567890",
    "creationTime": "2022-11-23T01:24:33.629Z",
    "id": "1234567890",
    "name": "my-seq-store"
}
```

Per ulteriori informazioni, consulta Archiviazione Omics nella Guida per gli sviluppatori di Amazon Omics.

- Per i dettagli sull'API, consulta CreateSequenceStore AWS CLICommand Reference.

`create-share`

Il seguente esempio di codice mostra come utilizzare`create-share`.

AWS CLI

Per creare una condivisione di un archivio di HealthOmics analisi

L'`create-share`esempio seguente mostra come creare una condivisione di un negozio di HealthOmics analisi che possa essere accettata da un abbonato esterno all'account.

```
aws omics create-share \
```

```
    --resource-arn "arn:aws:omics:us-west-2:555555555555:variantStore/
omics_dev_var_store" \
    --principal-subscriber "123456789012" \
    --name "my_Share-123"
```

Output:

```
{
    "shareId": "495c21bedc889d07d0ab69d710a6841e-dd75ab7a1a9c384fa848b5bd8e5a7e0a",
    "name": "my_Share-123",
    "status": "PENDING"
}
```

Per ulteriori informazioni, consulta Condivisione tra account nella Guida per l'AWS HealthOmics utente.

- Per i dettagli sull'API, consulta AWS CLI Command CreateShareReference.

`create-variant-store`

Il seguente esempio di codice mostra come utilizzare`create-variant-store`.

AWS CLI

Come creare un archivio di varianti

L'esempio `create-variant-store` seguente crea un archivio di varianti denominato `my_var_store`.

```
aws omics create-variant-store \
    --name my_var_store \
    --reference referenceArn=arn:aws:omics:us-
west-2:123456789012:referenceStore/1234567890/reference/1234567890
```

Output:

```
{
    "creationTime": "2022-11-23T22:09:07.534499Z",
    "id": "02dexmplcfdd",
    "name": "my_var_store",
    "reference": {
```

```
        "referenceArn": "arn:aws:omics:us-
west-2:123456789012:referenceStore/1234567890/reference/1234567890"
    },
    "status": "CREATING"
}
```

Per ulteriori informazioni, consulta Omics Analytics nella Guida per gli sviluppatori di Amazon Omics.

- Per i dettagli sull'API, consulta CreateVariantStore AWS CLICommand Reference.

`create-workflow`

Il seguente esempio di codice mostra come utilizzare`create-workflow`.

AWS CLI

Come creare un flusso di lavoro

L'esempio `create-workflow` seguente crea un flusso di lavoro WDL.

```
aws omics create-workflow \
    --name cram-converter \
    --engine WDL \
    --definition-zip fileb://workflow-crambam.zip \
    --parameter-template file://workflow-params.json
```

`workflow-crambam.zip` è un archivio ZIP contenente una definizione del flusso di lavoro. I parametri di runtime per il flusso di lavoro sono definiti in `workflow-params.json`.

```
{
    "ref_fasta" : {
        "description": "Reference genome fasta file",
        "optional": false
    },
    "ref_fasta_index" : {
        "description": "Index of the reference genome fasta file",
        "optional": false
    },
    "ref_dict" : {
        "description": "dictionary file for 'ref_fasta'",
        "optional": false
```

```
    },
    "input_cram" : {
        "description": "The Cram file to convert to BAM",
        "optional": false
    },
    "sample_name" : {
        "description": "The name of the input sample, used to name the output BAM",
        "optional": false
    }
}
```

Output:

```
{
    "arn": "arn:aws:omics:us-west-2:123456789012:workflow/1234567",
    "id": "1234567",
    "status": "CREATING",
    "tags": {}
}
```

Per ulteriori informazioni, consulta Creazione di flussi di lavoro privati nella Guida per l'AWS HealthOmics utente.

- Per i dettagli sull'API, consulta CreateWorkflow AWS CLICommand Reference.

`delete-annotation-store-versions`

Il seguente esempio di codice mostra come utilizzare`delete-annotation-store-versions`.

AWS CLI

Come eliminare una versione dell'archivio di annotazioni

L'esempio `delete-annotation-store-versions` seguente elimina una versione dell'archivio di annotazioni.

```
aws omics delete-annotation-store-versions \
    --name my_annotation_store \
    --versions my_version
```

Output:

```
{
    "errors": []
}
```

Per ulteriori informazioni, vedere Creazione di nuove versioni di archivi di annotazioni nella Guida per l'AWS HealthOmics utente.

- Per i dettagli sull'API, consulta DeleteAnnotationStoreVersions AWS CLICommand Reference.

`delete-annotation-store`

Il seguente esempio di codice mostra come utilizzare`delete-annotation-store`.

AWS CLI

Come eliminare un archivio di annotazioni

L'esempio `delete-annotation-store` seguente elimina un archivio di annotazioni denominato `my_vcf_store`.

```
aws omics delete-annotation-store \
    --name my_vcf_store
```

Output:

```
{
    "status": "DELETING"
}
```

Per ulteriori informazioni, consulta Omics Analytics nella Guida per gli sviluppatori di Amazon Omics.

- Per i dettagli sull'API, consulta DeleteAnnotationStore AWS CLICommand Reference.

`delete-reference-store`

Il seguente esempio di codice mostra come utilizzare`delete-reference-store`.

AWS CLI

Come eliminare un archivio di riferimenti

L'esempio `delete-reference-store` seguente elimina un archivio di riferimenti con ID `1234567890`.

```
aws omics delete-reference-store \
    --id 1234567890
```

Per ulteriori informazioni, consulta Archiviazione Omics nella Guida per gli sviluppatori di Amazon Omics.

- Per i dettagli sull'API, consulta DeleteReferenceStore AWS CLICommand Reference.

`delete-reference`

Il seguente esempio di codice mostra come utilizzare`delete-reference`.

AWS CLI

Come eliminare un riferimento

L'esempio `delete-reference` seguente elimina un riferimento.

```
aws omics delete-reference \
    --reference-store-id 1234567890 \
    --id 1234567890
```

Per ulteriori informazioni, consulta Archiviazione Omics nella Guida per gli sviluppatori di Amazon Omics.

- Per i dettagli sull'API, consulta DeleteReference AWS CLICommand Reference.

`delete-run-group`

Il seguente esempio di codice mostra come utilizzare`delete-run-group`.

AWS CLI

Come eliminare un gruppo di esecuzione

L'esempio `delete-run-group` seguente elimina un gruppo di esecuzione con ID `1234567`.

```
aws omics delete-run-group \
    --id 1234567
```

Per ulteriori informazioni, vedere Eliminazione di esecuzioni e gruppi di esecuzioni nella Guida per l'AWS HealthOmics utente.

- Per i dettagli sull'API, consulta DeleteRunGroup AWS CLICommand Reference.

`delete-run`

Il seguente esempio di codice mostra come utilizzare`delete-run`.

AWS CLI

Come eliminare un'esecuzione del flusso di lavoro

L'esempio `delete-run` seguente elimina un'esecuzione con ID `1234567`.

```
aws omics delete-run \
    --id 1234567
```

Per ulteriori informazioni, vedere Eliminazione di esecuzioni e gruppi di esecuzioni nella Guida per l'AWS HealthOmics utente.

- Per i dettagli sull'API, consulta DeleteRun AWS CLICommand Reference.

`delete-sequence-store`

Il seguente esempio di codice mostra come utilizzare`delete-sequence-store`.

AWS CLI

Come eliminare un archivio di sequenze

L'esempio `delete-sequence-store` seguente elimina un archivio di sequenze con ID `1234567890`.

```
aws omics delete-sequence-store \
    --id 1234567890
```

Per ulteriori informazioni, consulta Archiviazione Omics nella Guida per gli sviluppatori di Amazon Omics.

- Per i dettagli sull'API, consulta DeleteSequenceStore AWS CLICommand Reference.

`delete-share`

Il seguente esempio di codice mostra come utilizzare`delete-share`.

AWS CLI

Per eliminare una condivisione di dati di HealthOmics analisi

L'esempio `delete-share` seguente elimina una condivisione di dati di analisi tra più account.

```
aws omics delete-share \
    --share-id "495c21bedc889d07d0ab69d710a6841e-dd75ab7a1a9c384fa848b5bd8e5a7e0a"
```

Output:

```
{
    "status": "DELETING"
}
```

Per ulteriori informazioni, consulta Condivisione tra account nella Guida per l'AWS HealthOmics utente.

- Per i dettagli sull'API, consulta DeleteShare AWS CLICommand Reference.

`delete-variant-store`

Il seguente esempio di codice mostra come utilizzare`delete-variant-store`.

AWS CLI

Come eliminare un archivio di varianti

L'esempio `delete-variant-store` seguente elimina un archivio di varianti denominato `my_var_store`.

```
aws omics delete-variant-store \
    --name my_var_store
```

Output:

```
{
```

```
    "status": "DELETING"
}
```

Per ulteriori informazioni, consulta Omics Analytics nella Guida per gli sviluppatori di Amazon Omics.

- Per i dettagli sull'API, consulta DeleteVariantStore AWS CLICommand Reference.

`delete-workflow`

Il seguente esempio di codice mostra come utilizzare`delete-workflow`.

AWS CLI

Come eliminare un flusso di lavoro

L'esempio `delete-workflow` seguente elimina un flusso di lavoro con ID `1234567`.

```
aws omics delete-workflow \
    --id 1234567
```

Per ulteriori informazioni, consulta Eliminare un flusso di lavoro privato nella Guida AWS HealthOmics per l'utente.

- Per i dettagli sull'API, consulta DeleteWorkflow AWS CLICommand Reference.

`get-annotation-import-job`

Il seguente esempio di codice mostra come utilizzare`get-annotation-import-job`.

AWS CLI

Come visualizzare un processo di importazione di annotazioni

L'esempio `get-annotation-import-job` seguente ottiene i dettagli relativi a un processo di importazione di annotazioni.

```
aws omics get-annotation-import-job \
    --job-id 984162c7-xmpl-4d23-ab47-286f7950bfbf
```

Output:

```
{
    "creationTime": "2022-11-30T01:40:11.017746Z",
    "destinationName": "tsv_ann_store",
    "id": "984162c7-xmpl-4d23-ab47-286f7950bfbf",
    "items": [
        {
            "jobStatus": "COMPLETED",
            "source": "s3://omics-artifacts-01d6xmpl4e72dd32/targetedregions.bed.gz"
        }
    ],
    "roleArn": "arn:aws:iam::123456789012:role/omics-service-role-serviceRole-
W8O1XMPL7QZ",
    "runLeftNormalization": false,
    "status": "COMPLETED",
    "updateTime": "2022-11-30T01:42:39.134009Z"
}
```

Per ulteriori informazioni, consulta Omics Analytics nella Guida per gli sviluppatori di Amazon Omics.

- Per i dettagli sull'API, consulta GetAnnotationImportJob AWS CLICommand Reference.

`get-annotation-store-version`

Il seguente esempio di codice mostra come utilizzare`get-annotation-store-version`.

AWS CLI

Come recuperare i metadati per una versione dell'archivio di annotazioni

L'esempio `get-annotation-store-version` seguente recupera i metadati per la versione dell'archivio di annotazioni richiesto.

```
aws omics get-annotation-store-version \
    --name my_annotation_store \
    --version-name my_version
```

Output:

```
{
    "storeId": "4934045d1c6d",
    "id": "2a3f4a44aa7b",
```

```
    "status": "ACTIVE",
    "versionArn": "arn:aws:omics:us-west-2:555555555555:annotationStore/
my_annotation_store/version/my_version",
    "name": "my_annotation_store",
    "versionName": "my_version",
    "creationTime": "2023-07-21T17:15:49.251040+00:00",
    "updateTime": "2023-07-21T17:15:56.434223+00:00",
    "statusMessage": "",
    "versionSizeBytes": 0
}
```

Per ulteriori informazioni, vedere Creazione di nuove versioni di archivi di annotazioni nella Guida per l'AWS HealthOmics utente.

- Per i dettagli sull'API, consulta GetAnnotationStoreVersion AWS CLICommand Reference.

`get-annotation-store`

Il seguente esempio di codice mostra come utilizzare`get-annotation-store`.

AWS CLI

Come visualizzare un archivio di annotazioni

L'esempio `get-annotation-store` seguente ottiene i dettagli relativi a un archivio di annotazioni denominato `my_ann_store`.

```
aws omics get-annotation-store \
    --name my_ann_store
```

Output:

```
{
    "creationTime": "2022-11-23T22:48:39.226492Z",
    "id": "0a91xmplc71f",
    "name": "my_ann_store",
    "reference": {
        "referenceArn": "arn:aws:omics:us-
west-2:123456789012:referenceStore/1234567890/reference/1234567890"
    },
    "status": "CREATING",
    "storeArn": "arn:aws:omics:us-west-2:123456789012:annotationStore/my_ann_store",
```

```
    "storeFormat": "VCF",
    "storeSizeBytes": 0,
    "tags": {}
}
```

Per ulteriori informazioni, consulta Omics Analytics nella Guida per gli sviluppatori di Amazon Omics.

- Per i dettagli sull'API, consulta GetAnnotationStore AWS CLICommand Reference.

`get-read-set-activation-job`

Il seguente esempio di codice mostra come utilizzare`get-read-set-activation-job`.

AWS CLI

Come visualizzare un processo di attivazione di set di lettura

L'esempio `get-read-set-activation-job` seguente ottiene i dettagli relativi a un processo di attivazione di set di lettura.

```
aws omics get-read-set-activation-job \
    --sequence-store-id 1234567890 \
    --id 1234567890
```

Output:

```
{
    "completionTime": "2022-12-06T22:33:42.828Z",
    "creationTime": "2022-12-06T22:32:45.213Z",
    "id": "1234567890",
    "sequenceStoreId": "1234567890",
    "sources": [
        {
            "readSetId": "1234567890",
            "status": "FINISHED",
            "statusMessage": "No activation needed as read set is already in
 ACTIVATING or ACTIVE state."
        }
    ],
    "status": "COMPLETED",
    "statusMessage": "The job completed successfully."
```

```
}
```

Per ulteriori informazioni, consulta Archiviazione Omics nella Guida per gli sviluppatori di Amazon Omics.

- Per i dettagli sull'API, consulta GetReadSetActivationJob AWS CLICommand Reference.

`get-read-set-export-job`

Il seguente esempio di codice mostra come utilizzare`get-read-set-export-job`.

AWS CLI

Come visualizzare un processo di esportazione di set di lettura

L'esempio `get-read-set-export-job` seguente ottiene i dettagli relativi a un processo di esportazione di set di lettura.

```
aws omics get-read-set-export-job \
    --sequence-store-id 1234567890 \
    --id 1234567890
```

Output:

```
{
    "completionTime": "2022-12-06T22:39:14.491Z",
    "creationTime": "2022-12-06T22:37:18.612Z",
    "destination": "s3://omics-artifacts-01d6xmpl4e72dd32/read-set-export/",
    "id": "1234567890",
    "sequenceStoreId": "1234567890",
    "status": "COMPLETED",
    "statusMessage": "The job is submitted and will start soon."
}
```

Per ulteriori informazioni, consulta Archiviazione Omics nella Guida per gli sviluppatori di Amazon Omics.

- Per i dettagli sull'API, consulta GetReadSetExportJob AWS CLICommand Reference.

`get-read-set-import-job`

Il seguente esempio di codice mostra come utilizzare`get-read-set-import-job`.

AWS CLI

Come visualizzare un processo di importazione di set di lettura

L'esempio `get-read-set-import-job` seguente ottiene i dettagli relativi a un processo di importazione di set di lettura.

```
aws omics get-read-set-import-job \
    --sequence-store-id 1234567890 \
    --id 1234567890
```

Output:

```
{
    "creationTime": "2022-11-23T01:36:38.158Z",
    "id": "1234567890",
    "roleArn": "arn:aws:iam::123456789012:role/omics-service-role-serviceRole-
W8O1XMPL7QZ",
    "sequenceStoreId": "1234567890",
    "sources": [
        {
            "name": "HG00100",
            "referenceArn": "arn:aws:omics:us-
west-2:123456789012:referenceStore/1234567890/reference/1234567890",
            "sampleId": "bam-sample",
            "sourceFileType": "BAM",
            "sourceFiles": {
                "source1": "s3://omics-artifacts-01d6xmpl4e72dd32/
HG00100.chrom20.ILLUMINA.bwa.GBR.low_coverage.20101123.bam",
                "source2": ""
            },
            "status": "IN_PROGRESS",
            "statusMessage": "The source job is currently in progress.",
            "subjectId": "bam-subject",
            "tags": {
                "aws:omics:sampleId": "bam-sample",
                "aws:omics:subjectId": "bam-subject"
            }
        },
        {
            "name": "HG00146",
            "referenceArn": "arn:aws:omics:us-
west-2:123456789012:referenceStore/1234567890/reference/1234567890",
```

```
            "sampleId": "fastq-sample",
            "sourceFileType": "FASTQ",
            "sourceFiles": {
                "source1": "s3://omics-artifacts-01d6xmpl4e72dd32/
SRR233106_1.filt.fastq.gz",
                "source2": "s3://omics-artifacts-01d6xmpl4e72dd32/
SRR233106_2.filt.fastq.gz"
            },
            "status": "IN_PROGRESS",
            "statusMessage": "The source job is currently in progress.",
            "subjectId": "fastq-subject",
            "tags": {
                "aws:omics:sampleId": "fastq-sample",
                "aws:omics:subjectId": "fastq-subject"
            }
        },
        {
            "name": "HG00096",
            "referenceArn": "arn:aws:omics:us-
west-2:123456789012:referenceStore/1234567890/reference/1234567890",
            "sampleId": "cram-sample",
            "sourceFileType": "CRAM",
            "sourceFiles": {
                "source1": "s3://omics-artifacts-01d6xmpl4e72dd32/
HG00096.alt_bwamem_GRCh38DH.20150718.GBR.low_coverage.cram",
                "source2": ""
            },
            "status": "IN_PROGRESS",
            "statusMessage": "The source job is currently in progress.",
            "subjectId": "cram-subject",
            "tags": {
                "aws:omics:sampleId": "cram-sample",
                "aws:omics:subjectId": "cram-subject"
            }
        }
    ],
    "status": "IN_PROGRESS",
    "statusMessage": "The job is currently in progress."
}
```

Per ulteriori informazioni, consulta Archiviazione Omics nella Guida per gli sviluppatori di Amazon Omics.

- Per i dettagli sull'API, consulta GetReadSetImportJob AWS CLICommand Reference.

`get-read-set-metadata`

Il seguente esempio di codice mostra come utilizzare`get-read-set-metadata`.

AWS CLI

Come visualizzare un set di lettura

L'esempio `get-read-set-metadata` seguente ottiene i dettagli relativi ai file di un set di lettura.

```
aws omics get-read-set-metadata \
    --sequence-store-id 1234567890 \
    --id 1234567890
```

Output:

```
{
    "arn": "arn:aws:omics:us-west-2:123456789012:sequenceStore/1234567890/
readSet/1234567890",
    "creationTime": "2022-11-23T21:55:00.515Z",
    "fileType": "FASTQ",
    "files": {
        "source1": {
            "contentLength": 310054739,
            "partSize": 104857600,
            "totalParts": 3
        },
        "source2": {
            "contentLength": 307846621,
            "partSize": 104857600,
            "totalParts": 3
        }
    },
    "id": "1234567890",
    "name": "HG00146",
    "referenceArn": "arn:aws:omics:us-west-2:123456789012:referenceStore/1234567890/
reference/1234567890",
    "sampleId": "fastq-sample",
    "sequenceInformation": {
        "alignment": "UNALIGNED",
        "totalBaseCount": 677717384,
        "totalReadCount": 8917334
    },
    "sequenceStoreId": "1234567890",
```

```
    "status": "ACTIVE",
    "subjectId": "fastq-subject"
}
```

Per ulteriori informazioni, consulta Archiviazione Omics nella Guida per gli sviluppatori di Amazon Omics.

- Per i dettagli sull'API, consulta GetReadSetMetadata AWS CLICommand Reference.

`get-read-set`

Il seguente esempio di codice mostra come utilizzare`get-read-set`.

AWS CLI

Come scaricare un set di lettura

L'esempio `get-read-set` seguente scarica la parte 3 di un set di lettura come `1234567890.3.bam`.

```
aws omics get-read-set \
    --sequence-store-id 1234567890 \
    --id 1234567890 \
    --part-number 3  1234567890.3.bam
```

Per ulteriori informazioni, consulta Archiviazione Omics nella Guida per gli sviluppatori di Amazon Omics.

- Per i dettagli sull'API, consulta GetReadSet AWS CLICommand Reference.

`get-reference-import-job`

Il seguente esempio di codice mostra come utilizzare`get-reference-import-job`.

AWS CLI

Come visualizzare un processo di importazione di riferimenti

L'esempio `get-reference-import-job` seguente ottiene i dettagli relativi a un processo di importazione di riferimenti.

```
aws omics get-reference-import-job \
```

```
    --reference-store-id 1234567890 \
    --id 1234567890
```

Output:

```
{
    "creationTime": "2022-11-22T22:25:41.124Z",
    "id": "1234567890",
    "referenceStoreId": "1234567890",
    "roleArn": "arn:aws:iam::123456789012:role/omics-service-role-serviceRole-
W801XMPL7QZ",
    "sources": [
        {
            "name": "assembly-38",
            "sourceFile": "s3://omics-artifacts-01d6xmpl4e72dd32/
Homo_sapiens_assembly38.fasta",
            "status": "IN_PROGRESS",
            "statusMessage": "The source job is currently in progress."
        }
    ],
    "status": "IN_PROGRESS",
    "statusMessage": "The job is currently in progress."
}
```

Per ulteriori informazioni, consulta Archiviazione Omics nella Guida per gli sviluppatori di Amazon Omics.

- Per i dettagli sull'API, consulta GetReferenceImportJob AWS CLICommand Reference.

`get-reference-metadata`

Il seguente esempio di codice mostra come utilizzare`get-reference-metadata`.

AWS CLI

Come visualizzare un riferimento

L'esempio `get-reference-metadata` seguente ottiene i dettagli relativi a un riferimento.

```
aws omics get-reference-metadata \
    --reference-store-id 1234567890 \
    --id 1234567890
```

Output:

```
{
    "arn": "arn:aws:omics:us-west-2:123456789012:referenceStore/1234567890/
reference/1234567890",
    "creationTime": "2022-11-22T22:27:09.033Z",
    "files": {
        "index": {
            "contentLength": 160928,
            "partSize": 104857600,
            "totalParts": 1
        },
        "source": {
            "contentLength": 3249912778,
            "partSize": 104857600,
            "totalParts": 31
        }
    },
    "id": "1234567890",
    "md5": "7ff134953dcca8c8997453bbb80b6b5e",
    "name": "assembly-38",
    "referenceStoreId": "1234567890",
    "status": "ACTIVE",
    "updateTime": "2022-11-22T22:27:09.033Z"
}
```

Per ulteriori informazioni, consulta Archiviazione Omics nella Guida per gli sviluppatori di Amazon Omics.

- Per i dettagli sull'API, consulta GetReferenceMetadata AWS CLICommand Reference.

`get-reference-store`

Il seguente esempio di codice mostra come utilizzare`get-reference-store`.

AWS CLI

Come visualizzare un archivio di riferimenti

L'esempio `get-reference-store` seguente ottiene i dettagli relativi a un archivio di riferimenti.

```
aws omics get-reference-store \
    --id 1234567890
```

Output:

```
{
    "arn": "arn:aws:omics:us-west-2:123456789012:referenceStore/1234567890",
    "creationTime": "2022-09-23T23:27:20.364Z",
    "id": "1234567890",
    "name": "my-rstore-0"
}
```

Per ulteriori informazioni, consulta Archiviazione Omics nella Guida per gli sviluppatori di Amazon Omics.

- Per i dettagli sull'API, consulta GetReferenceStore AWS CLICommand Reference.

`get-reference`

Il seguente esempio di codice mostra come utilizzare`get-reference`.

AWS CLI

Come scaricare un riferimento genomico

L'esempio `get-reference` seguente scarica la parte 1 di un genoma come `hg38.1.fa`.

```
aws omics get-reference \
    --reference-store-id 1234567890 \
    --id 1234567890 \
    --part-number 1 hg38.1.fa
```

Per ulteriori informazioni, consulta Archiviazione Omics nella Guida per gli sviluppatori di Amazon Omics.

- Per i dettagli sull'API, consulta GetReference AWS CLICommand Reference.

`get-run-group`

Il seguente esempio di codice mostra come utilizzare`get-run-group`.

AWS CLI

Come visualizzare un gruppo di esecuzione

L'esempio `get-run-group` seguente ottiene i dettagli relativi a un gruppo di esecuzione.

```
aws omics get-run-group \
    --id 1234567
```

Output:

```
{
    "arn": "arn:aws:omics:us-west-2:123456789012:runGroup/1234567",
    "creationTime": "2022-12-01T00:58:42.915219Z",
    "id": "1234567",
    "maxCpus": 20,
    "maxDuration": 600,
    "name": "cram-convert",
    "tags": {}
}
```

Per ulteriori informazioni, vedere Creazione di gruppi di esecuzione nella Guida AWS HealthOmics per l'utente.

- Per i dettagli sull'API, consulta GetRunGroup AWS CLICommand Reference.

`get-run-task`

Il seguente esempio di codice mostra come utilizzare`get-run-task`.

AWS CLI

Come visualizzare un'attività

L'esempio `get-run-task` seguente ottiene i dettagli relativi a un'attività di un flusso di lavoro.

```
aws omics get-run-task \
    --id 1234567 \
    --task-id 1234567
```

Output:

```
{
    "cpus": 1,
    "creationTime": "2022-11-30T23:13:00.718651Z",
```

```
    "logStream": "arn:aws:logs:us-west-2:123456789012:log-group:/aws/omics/
WorkflowLog:log-stream:run/1234567/task/1234567",
    "memory": 15,
    "name": "CramToBamTask",
    "startTime": "2022-11-30T23:17:47.016Z",
    "status": "COMPLETED",
    "stopTime": "2022-11-30T23:18:21.503Z",
    "taskId": "1234567"
}
```

Per ulteriori informazioni, consulta Task lifecycle in a HealthOmics run nella Guida per l'AWS HealthOmics utente.

- Per i dettagli sull'API, consulta AWS CLI Command GetRunTaskReference.

`get-run`

Il seguente esempio di codice mostra come utilizzare`get-run`.

AWS CLI

Come visualizzare un'esecuzione del flusso di lavoro

L'esempio `get-run` seguente ottiene i dettagli relativi a un'esecuzione del flusso di lavoro.

```
aws omics get-run \
    --id 1234567
```

Output:

```
{
    "arn": "arn:aws:omics:us-west-2:123456789012:run/1234567",
    "creationTime": "2022-11-30T22:58:22.615865Z",
    "digest":
 "sha256:c54bxmpl742dcc26f7fa1f10e37550ddd8f251f418277c0a58e895b801ed28cf",
    "id": "1234567",
    "name": "cram-to-bam",
    "outputUri": "s3://omics-artifacts-01d6xmpl4e72dd32/workflow-output/",
    "parameters": {
        "ref_dict": "s3://omics-artifacts-01d6xmpl4e72dd32/
Homo_sapiens_assembly38.dict",
        "ref_fasta_index": "s3://omics-artifacts-01d6xmpl4e72dd32/
Homo_sapiens_assembly38.fasta.fai",
```

```
        "ref_fasta": "s3://omics-artifacts-01d6xmpl4e72dd32/
Homo_sapiens_assembly38.fasta",
        "sample_name": "NA12878",
        "input_cram": "s3://omics-artifacts-01d6xmpl4e72dd32/NA12878.cram"
    },
    "resourceDigests": {
        "s3://omics-artifacts-01d6xmpl4e72dd32/Homo_sapiens_assembly38.fasta.fai":
 "etag:f76371b113734a56cde236bc0372de0a",
        "s3://omics-artifacts-01d6xmpl4e72dd32/Homo_sapiens_assembly38.dict":
 "etag:3884c62eb0e53fa92459ed9bff133ae6",
        "s3://omics-artifacts-01d6xmpl4e72dd32/Homo_sapiens_assembly38.fasta":
 "etag:e307d81c605fb91b7720a08f00276842-388",
        "s3://omics-artifacts-01d6xmpl4e72dd32/NA12878.cram":
 "etag:a9f52976381286c6143b5cc681671ec6"
    },
    "roleArn": "arn:aws:iam::123456789012:role/omics-service-role-serviceRole-
W8O1XMPL7QZ",
    "startedBy": "arn:aws:iam::123456789012:user/laptop-2020",
    "status": "STARTING",
    "tags": {},
    "workflowId": "1234567",
    "workflowType": "PRIVATE"
}
```

Per ulteriori informazioni, consulta Esegui il ciclo di vita in un flusso di lavoro nella Guida per l'AWS HealthOmics utente.

- Per i dettagli sull'API, consulta AWS CLI Command GetRunReference.

`get-sequence-store`

Il seguente esempio di codice mostra come utilizzare`get-sequence-store`.

AWS CLI

Come visualizzare un archivio di sequenze

L'esempio `get-sequence-store` seguente ottiene i dettagli relativi a un archivio di sequenze con ID `1234567890`.

```
aws omics get-sequence-store \
    --id 1234567890
```

Output:

```
{
    "arn": "arn:aws:omics:us-east-1:123456789012:sequenceStore/1234567890",
    "creationTime": "2022-11-23T19:55:48.376Z",
    "id": "1234567890",
    "name": "my-seq-store"
}
```

Per ulteriori informazioni, consulta Archiviazione Omics nella Guida per gli sviluppatori di Amazon Omics.

- Per i dettagli sull'API, consulta GetSequenceStore AWS CLICommand Reference.

`get-share`

Il seguente esempio di codice mostra come utilizzare`get-share`.

AWS CLI

Per recuperare i metadati relativi a una condivisione di dati di analisi HealthOmics

L'esempio `get-share` seguente recupera i metadati relativi a una condivisione di dati di analisi tra più account.

```
aws omics get-share \
    --share-id "495c21bedc889d07d0ab69d710a6841e-dd75ab7a1a9c384fa848b5bd8e5a7e0a"
```

Output:

```
{
    "share": {
        "shareId": "495c21bedc889d07d0ab69d710a6841e-
dd75ab7a1a9c384fa848b5bd8e5a7e0a",
        "name": "my_Share-123",
        "resourceArn": "arn:aws:omics:us-west-2:555555555555:variantStore/
omics_dev_var_store",
        "principalSubscriber": "123456789012",
        "ownerId": "555555555555",
        "status": "PENDING"
    }
}
```

Per ulteriori informazioni, consulta Condivisione tra account nella Guida per l'AWS HealthOmics utente.

- Per i dettagli sull'API, consulta GetShare AWS CLICommand Reference.

`get-variant-import-job`

Il seguente esempio di codice mostra come utilizzare`get-variant-import-job`.

AWS CLI

Come visualizzare un processo di importazione di varianti

L'esempio `get-variant-import-job` seguente ottiene i dettagli relativi a un processo di importazione di varianti.

```
aws omics get-variant-import-job \
    --job-id edd7b8ce-xmpl-47e2-bc99-258cac95a508
```

Output:

```
{
    "creationTime": "2022-11-23T22:42:50.037812Z",
    "destinationName": "my_var_store",
    "id": "edd7b8ce-xmpl-47e2-bc99-258cac95a508",
    "items": [
        {
            "jobStatus": "IN_PROGRESS",
            "source": "s3://omics-artifacts-01d6xmpl4e72dd32/
Homo_sapiens_assembly38.known_indels.vcf.gz"
        }
    ],
    "roleArn": "arn:aws:iam::123456789012:role/omics-service-role-serviceRole-
W8O1XMPL7QZ",
    "runLeftNormalization": false,
    "status": "IN_PROGRESS",
    "updateTime": "2022-11-23T22:43:05.898309Z"
}
```

Per ulteriori informazioni, consulta Omics Analytics nella Guida per gli sviluppatori di Amazon Omics.

- Per i dettagli sull'API, consulta GetVariantImportJob AWS CLICommand Reference.

`get-variant-store`

Il seguente esempio di codice mostra come utilizzare`get-variant-store`.

AWS CLI

Come visualizzare un archivio di varianti

L'esempio `get-variant-store` seguente ottiene i dettagli relativi a un archivio di varianti.

```
aws omics get-variant-store \
    --name my_var_store
```

Output:

```
{
    "creationTime": "2022-11-23T22:09:07.534499Z",
    "id": "02dexmplcfdd",
    "name": "my_var_store",
    "reference": {
        "referenceArn": "arn:aws:omics:us-
west-2:123456789012:referenceStore/1234567890/reference/1234567890"
    },
    "status": "CREATING",
    "storeArn": "arn:aws:omics:us-west-2:123456789012:variantStore/my_var_store",
    "storeSizeBytes": 0,
    "tags": {},
    "updateTime": "2022-11-23T22:09:24.931711Z"
}
```

Per ulteriori informazioni, consulta Omics Analytics nella Guida per gli sviluppatori di Amazon Omics.

- Per i dettagli sull'API, consulta GetVariantStore AWS CLICommand Reference.

`get-workflow`

Il seguente esempio di codice mostra come utilizzare`get-workflow`.

AWS CLI

Come visualizzare un flusso di lavoro

L'esempio `get-workflow` seguente ottiene i dettagli relativi a un flusso di lavoro con ID 1234567.

```
aws omics get-workflow \
    --id 1234567
```

Output:

```
{
    "arn": "arn:aws:omics:us-west-2:123456789012:workflow/1234567",
    "creationTime": "2022-11-30T22:33:16.225368Z",
    "digest":
 "sha256:c54bxmpl742dcc26f7fa1f10e37550ddd8f251f418277c0a58e895b801ed28cf",
    "engine": "WDL",
    "id": "1234567",
    "main": "workflow-crambam.wdl",
    "name": "cram-converter",
    "parameterTemplate": {
        "ref_dict": {
            "description": "dictionary file for 'ref_fasta'"
        },
        "ref_fasta_index": {
            "description": "Index of the reference genome fasta file"
        },
        "ref_fasta": {
            "description": "Reference genome fasta file"
        },
        "input_cram": {
            "description": "The Cram file to convert to BAM"
        },
        "sample_name": {
            "description": "The name of the input sample, used to name the output
 BAM"
        }
    },
    "status": "ACTIVE",
    "statusMessage": "workflow-crambam.wdl\n    workflow CramToBamFlow\n
 call CramToBamTask\n        call ValidateSamFile\n    task CramToBamTask\n    task
 ValidateSamFile\n",
    "tags": {},
    "type": "PRIVATE"
}
```

Per ulteriori informazioni, consulta Creazione di flussi di lavoro privati nella Guida per l'AWS HealthOmics utente.

- Per i dettagli sull'API, consulta GetWorkflow AWS CLICommand Reference.

`list-annotation-import-jobs`

Il seguente esempio di codice mostra come utilizzare`list-annotation-import-jobs`.

AWS CLI

Come ottenere un elenco di processi di importazione di annotazioni

L'esempio `list-annotation-import-jobs` seguente ottiene un elenco di processi di importazione di annotazioni.

```
aws omics list-annotation-import-jobs
```

Output:

```
{
    "annotationImportJobs": [
        {
            "creationTime": "2022-11-30T01:39:41.478294Z",
            "destinationName": "gff_ann_store",
            "id": "18a9e792-xmpl-4869-a105-e5b602900444",
            "roleArn": "arn:aws:iam::123456789012:role/omics-service-role-
serviceRole-W8O1XMPL7QZ",
            "runLeftNormalization": false,
            "status": "COMPLETED",
            "updateTime": "2022-11-30T01:47:09.145178Z"
        },
        {
            "creationTime": "2022-11-30T00:45:58.007838Z",
            "destinationName": "my_ann_store",
            "id": "4e9eafc8-xmpl-431e-a0b2-3bda27cb600a",
            "roleArn": "arn:aws:iam::123456789012:role/omics-service-role-
serviceRole-W8O1XMPL7QZ",
            "runLeftNormalization": false,
            "status": "FAILED",
            "updateTime": "2022-11-30T00:47:01.706325Z"
        }
    ]
```

```
}
```

Per ulteriori informazioni, consulta Omics Analytics nella Guida per gli sviluppatori di Amazon Omics.

- Per i dettagli sull'API, consulta ListAnnotationImportJobs AWS CLICommand Reference.

`list-annotation-store-versions`

Il seguente esempio di codice mostra come utilizzare`list-annotation-store-versions`.

AWS CLI

Come elencare tutte le versioni di un archivio di annotazioni.

L'esempio `list-annotation-store-versions` seguente elenca tutte le versioni esistenti di un archivio di annotazioni.

```
aws omics list-annotation-store-versions \
    --name my_annotation_store
```

Output:

```
{
    "annotationStoreVersions": [
        {
        "storeId": "4934045d1c6d",
        "id": "2a3f4a44aa7b",
        "status": "CREATING",
        "versionArn": "arn:aws:omics:us-west-2:555555555555:annotationStore/
my_annotation_store/version/my_version_2",
        "name": "my_annotation_store",
        "versionName": "my_version_2",
        "creation Time": "2023-07-21T17:20:59.380043+00:00",
        "versionSizeBytes": 0
},
{
     "storeId": "4934045d1c6d",
     "id": "4934045d1c6d",
     "status": "ACTIVE",
     "versionArn": "arn:aws:omics:us-west-2:555555555555:annotationStore/
my_annotation_store/version/my_version_1",
```

```
      "name": "my_annotation_store",
      "versionName": "my_version_1",
      "creationTime": "2023-07-21T17:15:49.251040+00:00",
      "updateTime": "2023-07-21T17:15:56.434223+00:00",
      "statusMessage": "",
      "versionSizeBytes": 0
      }

}
```

Per ulteriori informazioni, vedere Creazione di nuove versioni di archivi di annotazioni nella Guida per l'AWS HealthOmics utente.

- Per i dettagli sull'API, consulta ListAnnotationStoreVersions AWS CLICommand Reference.

`list-annotation-stores`

Il seguente esempio di codice mostra come utilizzare`list-annotation-stores`.

AWS CLI

Come ottenere un elenco di archivi di annotazioni

L'esempio `list-annotation-stores` seguente ottiene un elenco di archivi di annotazioni.

```
aws omics list-annotation-stores
```

Output:

```
{
    "annotationStores": [
        {
            "creationTime": "2022-11-23T22:48:39.226492Z",
            "id": "0a91xmplc71f",
            "name": "my_ann_store",
            "reference": {
                "referenceArn": "arn:aws:omics:us-
west-2:123456789012:referenceStore/1234567890/reference/1234567890"
            },
            "status": "ACTIVE",
            "statusMessage": "",
            "storeArn": "arn:aws:omics:us-west-2:123456789012:annotationStore/
my_ann_store",
```

```
            "storeFormat": "VCF",
            "storeSizeBytes": 0,
            "updateTime": "2022-11-23T22:53:27.372840Z"
        }
    ]
}
```

Per ulteriori informazioni, consulta Omics Analytics nella Guida per gli sviluppatori di Amazon Omics.

- Per i dettagli sull'API, consulta ListAnnotationStores AWS CLICommand Reference.

`list-multipart-read-set-uploads`

Il seguente esempio di codice mostra come utilizzare`list-multipart-read-set-uploads`.

AWS CLI

Come elencare tutti i caricamenti di set di lettura in più parti e i relativi stati.

L'esempio `list-multipart-read-set-uploads` seguente elenca tutti i caricamenti di set di lettura in più parti e i relativi stati.

```
aws omics list-multipart-read-set-uploads \
    --sequence-store-id 0123456789
```

Output:

```
{
"uploads":
    [
        {
           "sequenceStoreId": "0123456789",
           "uploadId": "8749584421",
           "sourceFileType": "FASTQ",
            "subjectId": "mySubject",
            "sampleId": "mySample",
            "generatedFrom": "1000 Genomes",
            "name": "HG00146",
            "description": "FASTQ for HG00146",
            "creationTime": "2023-11-29T19:22:51.349298+00:00"
        },
        {
```

```
            "sequenceStoreId": "0123456789",
            "uploadId": "5290538638",
            "sourceFileType": "BAM",
            "subjectId": "mySubject",
            "sampleId": "mySample",
            "generatedFrom": "1000 Genomes",
            "referenceArn": "arn:aws:omics:us-
west-2:845448930428:referenceStore/8168613728/reference/2190697383",
            "name": "HG00146",
            "description": "BAM for HG00146",
            "creationTime": "2023-11-29T19:23:33.116516+00:00"
        },
        {
            "sequenceStoreId": "0123456789",
            "uploadId": "4174220862",
            "sourceFileType": "BAM",
            "subjectId": "mySubject",
            "sampleId": "mySample",
            "generatedFrom": "1000 Genomes",
            "referenceArn": "arn:aws:omics:us-
west-2:845448930428:referenceStore/8168613728/reference/2190697383",
            "name": "HG00147",
            "description": "BAM for HG00147",
            "creationTime": "2023-11-29T19:23:47.007866+00:00"
        }
    ]
}
```

Per ulteriori informazioni, consulta Caricamento diretto su un archivio di sequenze nella Guida AWS HealthOmics per l'utente.

- Per i dettagli sull'API, consulta ListMultipartReadSetUploads AWS CLICommand Reference.

`list-read-set-activation-jobs`

Il seguente esempio di codice mostra come utilizzare`list-read-set-activation-jobs`.

AWS CLI

Come ottenere un elenco di processi di attivazione di set di lettura

L'esempio `list-read-set-activation-jobs` seguente ottiene un elenco di processi di attivazione per un archivio di sequenze con ID `1234567890`.

```
aws omics list-read-set-activation-jobs \
    --sequence-store-id 1234567890
```

Output:

```
{
    "activationJobs": [
        {
            "completionTime": "2022-12-06T22:33:42.828Z",
            "creationTime": "2022-12-06T22:32:45.213Z",
            "id": "1234567890",
            "sequenceStoreId": "1234567890",
            "status": "COMPLETED"
        },
        {
            "creationTime": "2022-12-06T22:35:10.100Z",
            "id": "1234567890",
            "sequenceStoreId": "1234567890",
            "status": "IN_PROGRESS"
        }
    ]
}
```

Per ulteriori informazioni, consulta Archiviazione Omics nella Guida per gli sviluppatori di Amazon Omics.

- Per i dettagli sull'API, consulta ListReadSetActivationJobs AWS CLICommand Reference.

`list-read-set-export-jobs`

Il seguente esempio di codice mostra come utilizzare`list-read-set-export-jobs`.

AWS CLI

Come ottenere un elenco di processi di esportazione di set di lettura

L'esempio `list-read-set-export-jobs` seguente ottiene un elenco di processi di esportazione per un archivio di sequenze con ID `1234567890`.

```
aws omics list-read-set-export-jobs \
    --sequence-store-id 1234567890
```

Output:

```
{
    "exportJobs": [
        {
            "completionTime": "2022-12-06T22:39:14.491Z",
            "creationTime": "2022-12-06T22:37:18.612Z",
            "destination": "s3://omics-artifacts-01d6xmpl4e72dd32/read-set-export/",
            "id": "1234567890",
            "sequenceStoreId": "1234567890",
            "status": "COMPLETED"
        },
        {
            "creationTime": "2022-12-06T22:38:04.871Z",
            "destination": "s3://omics-artifacts-01d6xmpl4e72dd32/read-set-export/",
            "id": "1234567890",
            "sequenceStoreId": "1234567890",
            "status": "IN_PROGRESS"
        }
    ]
}
```

Per ulteriori informazioni, consulta Archiviazione Omics nella Guida per gli sviluppatori di Amazon Omics.

- Per i dettagli sull'API, consulta ListReadSetExportJobs AWS CLICommand Reference.

`list-read-set-import-jobs`

Il seguente esempio di codice mostra come utilizzare`list-read-set-import-jobs`.

AWS CLI

Come ottenere un elenco di processi di importazione di set di lettura

L'esempio `list-read-set-import-jobs` seguente ottiene un elenco di processi di importazione per un archivio di sequenze con ID `1234567890`.

```
aws omics list-read-set-import-jobs \
    --sequence-store-id 1234567890
```

Output:

```
{
    "importJobs": [
        {
            "completionTime": "2022-11-29T18:17:49.244Z",
            "creationTime": "2022-11-29T17:32:47.700Z",
            "id": "1234567890",
            "roleArn": "arn:aws:iam::123456789012:role/omics-service-role-
serviceRole-W8O1XMPL7QZ",
            "sequenceStoreId": "1234567890",
            "status": "COMPLETED"
        },
        {
            "completionTime": "2022-11-23T22:01:34.090Z",
            "creationTime": "2022-11-23T21:52:43.289Z",
            "id": "1234567890",
            "roleArn": "arn:aws:iam::123456789012:role/omics-service-role-
serviceRole-W8O1XMPL7QZ",
            "sequenceStoreId": "1234567890",
            "status": "COMPLETED_WITH_FAILURES"
        }
    ]
}
```

Per ulteriori informazioni, consulta Archiviazione Omics nella Guida per gli sviluppatori di Amazon Omics.

- Per i dettagli sull'API, consulta ListReadSetImportJobs AWS CLICommand Reference.

`list-read-set-upload-parts`

Il seguente esempio di codice mostra come utilizzare`list-read-set-upload-parts`.

AWS CLI

Come elencare tutte le parti in un processo di caricamento in più parti richiesto per un archivio di sequenze.

L'esempio `list-read-set-upload-parts` seguente elenca tutte le parti in un processo di caricamento in più parti richiesto per un archivio di sequenze.

```
aws omics list-read-set-upload-parts \
    --sequence-store-id 0123456789 \
```

```
    --upload-id 1122334455 \
    --part-source SOURCE1
```

Output:

```
{
    "parts": [
        {
            "partNumber": 1,
            "partSize": 94371840,
            "file": "SOURCE1",
            "checksum":
 "984979b9928ae8d8622286c4a9cd8e99d964a22d59ed0f5722e1733eb280e635",
            "lastUpdatedTime": "2023-02-02T20:14:47.533000+00:00"
        }
        {
            "partNumber": 2,
            "partSize": 10471840,
            "file": "SOURCE1",
            "checksum":
 "984979b9928ae8d8622286c4a9cd8e99d964a22d59ed0f5722e1733eb280e635",
            "lastUpdatedTime": "2023-02-02T20:14:47.533000+00:00"
        }
      ]

}
```

Per ulteriori informazioni, consulta Caricamento diretto su un archivio di sequenze nella Guida AWS HealthOmics per l'utente.

- Per i dettagli sull'API, consulta ListReadSetUploadParts AWS CLICommand Reference.

`list-read-sets`

Il seguente esempio di codice mostra come utilizzare`list-read-sets`.

AWS CLI

Come ottenere un elenco di set di lettura

L'esempio `list-read-sets` seguente ottiene un elenco di set di lettura per un archivio di sequenze con ID `1234567890`.

```
aws omics list-read-sets \
    --sequence-store-id 1234567890
```

Output:

```
{
    "readSets": [
        {
            "arn": "arn:aws:omics:us-west-2:123456789012:sequenceStore/1234567890/
readSet/1234567890",
            "creationTime": "2022-11-23T21:55:00.515Z",
            "fileType": "FASTQ",
            "id": "1234567890",
            "name": "HG00146",
            "referenceArn": "arn:aws:omics:us-
west-2:123456789012:referenceStore/1234567890/reference/1234567890",
            "sampleId": "fastq-sample",
            "sequenceStoreId": "1234567890",
            "status": "ACTIVE",
            "subjectId": "fastq-subject"
        }
    ]
}
```

Per ulteriori informazioni, consulta Archiviazione Omics nella Guida per gli sviluppatori di Amazon Omics.

- Per i dettagli sull'API, consulta ListReadSets AWS CLICommand Reference.

`list-reference-import-jobs`

Il seguente esempio di codice mostra come utilizzare`list-reference-import-jobs`.

AWS CLI

Come ottenere un elenco di processi di importazione di riferimenti

L'esempio `list-reference-import-jobs` seguente ottiene un elenco di processi di importazione di riferimenti per un archivio di riferimenti con ID `1234567890`.

```
aws omics list-reference-import-jobs \
```

```
    --reference-store-id 1234567890
```

Output:

```
{
    "importJobs": [
        {
            "completionTime": "2022-11-23T19:54:58.204Z",
            "creationTime": "2022-11-23T19:53:20.729Z",
            "id": "1234567890",
            "referenceStoreId": "1234567890",
            "roleArn": "arn:aws:iam::123456789012:role/omics-service-role-
serviceRole-W8O1XMPL7QZ",
            "status": "COMPLETED"
        },
        {
            "creationTime": "2022-11-23T20:34:03.250Z",
            "id": "1234567890",
            "referenceStoreId": "1234567890",
            "roleArn": "arn:aws:iam::123456789012:role/omics-service-role-
serviceRole-W8O1XMPL7QZ",
            "status": "IN_PROGRESS"
        }
    ]
}
```

Per ulteriori informazioni, consulta Archiviazione Omics nella Guida per gli sviluppatori di Amazon Omics.

- Per i dettagli sull'API, consulta ListReferenceImportJobs AWS CLICommand Reference.

`list-reference-stores`

Il seguente esempio di codice mostra come utilizzare`list-reference-stores`.

AWS CLI

Come ottenere un elenco di archivi di riferimenti

L'esempio `list-reference-stores` seguente ottiene un elenco di archivi di riferimenti.

```
aws omics list-reference-stores
```

Output:

```
{
    "referenceStores": [
        {
            "arn": "arn:aws:omics:us-west-2:123456789012:referenceStore/1234567890",
            "creationTime": "2022-11-22T22:13:25.947Z",
            "id": "1234567890",
            "name": "my-ref-store"
        }
    ]
}
```

Per ulteriori informazioni, consulta Archiviazione Omics nella Guida per gli sviluppatori di Amazon Omics.

- Per i dettagli sull'API, consulta ListReferenceStores AWS CLICommand Reference.

`list-references`

Il seguente esempio di codice mostra come utilizzare`list-references`.

AWS CLI

Come ottenere un elenco di riferimenti

L'esempio `list-references` seguente ottiene un elenco di riferimenti genomici a per un archivio di riferimenti con ID `1234567890`.

```
aws omics list-references \
    --reference-store-id 1234567890
```

Output:

```
{
    "references": [
        {
            "arn": "arn:aws:omics:us-west-2:123456789012:referenceStore/1234567890/
reference/1234567890",
            "creationTime": "2022-11-22T22:27:09.033Z",
            "id": "1234567890",
            "md5": "7ff134953dcca8c8997453bbb80b6b5e",
```

```
            "name": "assembly-38",
            "referenceStoreId": "1234567890",
            "status": "ACTIVE",
            "updateTime": "2022-11-22T22:27:09.033Z"
        }
    ]
}
```

Per ulteriori informazioni, consulta Archiviazione Omics nella Guida per gli sviluppatori di Amazon Omics.

- Per i dettagli sull'API, consulta ListReferences AWS CLICommand Reference.

`list-run-groups`

Il seguente esempio di codice mostra come utilizzare`list-run-groups`.

AWS CLI

Come ottenere un elenco di gruppi di esecuzione

L'esempio `list-run-groups` seguente ottiene un elenco di gruppi di esecuzione.

```
aws omics list-run-groups
```

Output:

```
{
    "items": [
        {
            "arn": "arn:aws:omics:us-west-2:123456789012:runGroup/1234567",
            "creationTime": "2022-12-01T00:58:42.915219Z",
            "id": "1234567",
            "maxCpus": 20,
            "maxDuration": 600,
            "name": "cram-convert"
        }
    ]
}
```

Per ulteriori informazioni, vedere Creazione di gruppi di esecuzione nella Guida AWS HealthOmics per l'utente.

- Per i dettagli sull'API, consulta ListRunGroups AWS CLICommand Reference.

`list-run-tasks`

Il seguente esempio di codice mostra come utilizzare`list-run-tasks`.

AWS CLI

Come ottenere un elenco di attività

L'esempio `list-run-tasks` seguente ottiene un elenco di attività per un'esecuzione del flusso di lavoro.

```
aws omics list-run-tasks \
    --id 1234567
```

Output:

```
{
    "items": [
        {
            "cpus": 1,
            "creationTime": "2022-11-30T23:13:00.718651Z",
            "memory": 15,
            "name": "CramToBamTask",
            "startTime": "2022-11-30T23:17:47.016Z",
            "status": "COMPLETED",
            "stopTime": "2022-11-30T23:18:21.503Z",
            "taskId": "1234567"
        },
        {
            "cpus": 1,
            "creationTime": "2022-11-30T23:18:32.315606Z",
            "memory": 4,
            "name": "ValidateSamFile",
            "startTime": "2022-11-30T23:23:40.165Z",
            "status": "COMPLETED",
            "stopTime": "2022-11-30T23:24:14.766Z",
            "taskId": "1234567"
        }
    ]
}
```

Per ulteriori informazioni, consulta Task lifecycle in a HealthOmics run nella Guida per l'AWS HealthOmics utente.

- Per i dettagli sull'API, consulta AWS CLI Command ListRunTasksReference.

`list-runs`

Il seguente esempio di codice mostra come utilizzare`list-runs`.

AWS CLI

Come ottenere un elenco di esecuzioni del flusso di lavoro

L'esempio `list-runs` seguente ottiene un elenco di esecuzioni del flusso di lavoro.

```
aws omics list-runs
```

Output:

```
{
    "items": [
        {
            "arn": "arn:aws:omics:us-west-2:123456789012:run/1234567",
            "creationTime": "2022-12-02T23:20:01.202074Z",
            "id": "1234567",
            "name": "cram-to-bam",
            "priority": 1,
            "startTime": "2022-12-02T23:29:18.115Z",
            "status": "COMPLETED",
            "stopTime": "2022-12-02T23:57:54.428812Z",
            "storageCapacity": 10,
            "workflowId": "1234567"
        },
        {
            "arn": "arn:aws:omics:us-west-2:123456789012:run/1234567",
            "creationTime": "2022-12-03T00:16:57.180066Z",
            "id": "1234567",
            "name": "cram-to-bam",
            "priority": 1,
            "startTime": "2022-12-03T00:26:50.233Z",
            "status": "FAILED",
            "stopTime": "2022-12-03T00:37:21.451340Z",
            "storageCapacity": 10,
```

```
            "workflowId": "1234567"
        },
        {
            "arn": "arn:aws:omics:us-west-2:123456789012:run/1234567",
            "creationTime": "2022-12-05T17:57:08.444817Z",
            "id": "1234567",
            "name": "cram-to-bam",
            "status": "STARTING",
            "workflowId": "1234567"
        }
    ]
}
```

Per ulteriori informazioni, consulta Esegui il ciclo di vita in un flusso di lavoro nella Guida per l'AWS HealthOmics utente.

- Per i dettagli sull'API, consulta AWS CLI Command ListRunsReference.

`list-sequence-stores`

Il seguente esempio di codice mostra come utilizzare`list-sequence-stores`.

AWS CLI

Come ottenere un elenco di archivi di sequenze

L'esempio `list-sequence-stores` seguente ottiene un elenco di archivi di sequenze.

```
aws omics list-sequence-stores
```

Output:

```
{
    "sequenceStores": [
        {
            "arn": "arn:aws:omics:us-west-2:123456789012:sequenceStore/1234567890",
            "creationTime": "2022-11-23T01:24:33.629Z",
            "id": "1234567890",
            "name": "my-seq-store"
        }
    ]
}
```

Per ulteriori informazioni, consulta Archiviazione Omics nella Guida per gli sviluppatori di Amazon Omics.

- Per i dettagli sull'API, consulta ListSequenceStores AWS CLICommand Reference.

`list-shares`

Il seguente esempio di codice mostra come utilizzare`list-shares`.

AWS CLI

Per elencare le condivisioni disponibili di un dato di HealthOmics analisi

L'esempio `list-shares` seguente elenca tutte le condivisioni che sono state create per il proprietario di una risorsa.

```
aws omics list-shares \
    --resource-owner SELF
```

Output:

```
{
    "shares": [
        {
            "shareId": "595c1cbd-a008-4eca-a887-954d30c91c6e",
            "name": "myShare",
            "resourceArn": "arn:aws:omics:us-west-2:555555555555:variantStore/
store_1",
            "principalSubscriber": "123456789012",
            "ownerId": "555555555555",
            "status": "PENDING"
        }
        {
            "shareId": "39b65d0d-4368-4a19-9814-b0e31d73c10a",
            "name": "myShare3456",
            "resourceArn": "arn:aws:omics:us-west-2:555555555555:variantStore/
store_2",
            "principalSubscriber": "123456789012",
            "ownerId": "555555555555",
            "status": "ACTIVE"
        },
        {
```

```
            "shareId": "203152f5-eef9-459d-a4e0-a691668d44ef",
            "name": "myShare4",
            "resourceArn": "arn:aws:omics:us-west-2:555555555555:variantStore/
store_3",
            "principalSubscriber": "123456789012",
            "ownerId": "555555555555",
            "status": "ACTIVE"
        }
    ]
}
```

Per ulteriori informazioni, consulta Condivisione tra account nella Guida per l'AWS HealthOmics utente.

- Per i dettagli sull'API, consulta ListShares AWS CLICommand Reference.

`list-tags-for-resource`

Il seguente esempio di codice mostra come utilizzare`list-tags-for-resource`.

AWS CLI

Come ottenere un elenco di tag

L'esempio `list-tags-for-resource` seguente ottiene un elenco di tag per un flusso di lavoro con ID 1234567.

```
aws omics list-tags-for-resource \
    --resource-arn arn:aws:omics:us-west-2:123456789012:workflow/1234567
```

Output:

```
{
    "tags": {
        "department": "analytics"
    }
}
```

Per ulteriori informazioni, consulta Tagging di risorse in Amazon Omics nella Guida per gli sviluppatori di Amazon Omics.

- Per i dettagli sull'API, consulta ListTagsForResource AWS CLICommand Reference.

`list-variant-import-jobs`

Il seguente esempio di codice mostra come utilizzare`list-variant-import-jobs`.

AWS CLI

Come ottenere un elenco di processi di importazione di varianti

L'esempio `list-variant-import-jobs` seguente ottiene un elenco di processi di importazione di varianti.

```
aws omics list-variant-import-jobs
```

Output:

```
{
    "variantImportJobs": [
        {
            "creationTime": "2022-11-23T22:47:02.514002Z",
            "destinationName": "my_var_store",
            "id": "69cb65d6-xmpl-4a4a-9025-4565794b684e",
            "roleArn": "arn:aws:iam::123456789012:role/omics-service-role-
serviceRole-W8O1XMPL7QZ",
            "runLeftNormalization": false,
            "status": "COMPLETED",
            "updateTime": "2022-11-23T22:49:17.976597Z"
        },
        {
            "creationTime": "2022-11-23T22:42:50.037812Z",
            "destinationName": "my_var_store",
            "id": "edd7b8ce-xmpl-47e2-bc99-258cac95a508",
            "roleArn": "arn:aws:iam::123456789012:role/omics-service-role-
serviceRole-W8O1XMPL7QZ",
            "runLeftNormalization": false,
            "status": "COMPLETED",
            "updateTime": "2022-11-23T22:45:26.009880Z"
        }
    ]
}
```

Per ulteriori informazioni, consulta Omics Analytics nella Guida per gli sviluppatori di Amazon Omics.

- Per i dettagli sull'API, consulta ListVariantImportJobs AWS CLICommand Reference.

`list-variant-stores`

Il seguente esempio di codice mostra come utilizzare`list-variant-stores`.

AWS CLI

Come ottenere un elenco di archivi di varianti

L'esempio `list-variant-stores` seguente ottiene un elenco di archivi di varianti.

```
aws omics list-variant-stores
```

Output:

```
{
    "variantStores": [
        {
            "creationTime": "2022-11-23T22:09:07.534499Z",
            "id": "02dexmplcfdd",
            "name": "my_var_store",
            "reference": {
                "referenceArn": "arn:aws:omics:us-
west-2:123456789012:referenceStore/1234567890/reference/1234567890"
            },
            "status": "CREATING",
            "storeArn": "arn:aws:omics:us-west-2:123456789012:variantStore/
my_var_store",
            "storeSizeBytes": 0,
            "updateTime": "2022-11-23T22:09:24.931711Z"
        },
        {
            "creationTime": "2022-09-23T23:00:09.140265Z",
            "id": "8777xmpl1a24",
            "name": "myvstore0",
            "status": "ACTIVE",
            "storeArn": "arn:aws:omics:us-west-2:123456789012:variantStore/
myvstore0",
            "storeSizeBytes": 0,
            "updateTime": "2022-09-23T23:03:26.013220Z"
        }
    ]
```

```
}
```

Per ulteriori informazioni, consulta Omics Analytics nella Guida per gli sviluppatori di Amazon Omics.

- Per i dettagli sull'API, consulta ListVariantStores AWS CLICommand Reference.

`list-workflows`

Il seguente esempio di codice mostra come utilizzare`list-workflows`.

AWS CLI

Come ottenere un elenco di flussi di lavoro

L'esempio `list-workflows` seguente ottiene un elenco di flussi di lavoro.

```
aws omics list-workflows
```

Output:

```
{
    "items": [
        {
            "arn": "arn:aws:omics:us-west-2:123456789012:workflow/1234567",
            "creationTime": "2022-09-23T23:08:22.041227Z",
            "digest": "nSCNo/qMWFxmplXpUdokXJnwgneOaxyyc2YOxVxrJTE=",
            "id": "1234567",
            "name": "my-wkflow-0",
            "status": "ACTIVE",
            "type": "PRIVATE"
        },
        {
            "arn": "arn:aws:omics:us-west-2:123456789012:workflow/1234567",
            "creationTime": "2022-11-30T22:33:16.225368Z",
            "digest":
 "sha256:c54bxmpl742dcc26f7fa1f10e37550ddd8f251f418277c0a58e895b801ed28cf",
            "id": "1234567",
            "name": "cram-converter",
            "status": "ACTIVE",
            "type": "PRIVATE"
        }
    ]
```

```
}
```

Per ulteriori informazioni, consulta Creazione di flussi di lavoro privati nella Guida per l'AWS HealthOmics utente.

- Per i dettagli sull'API, consulta ListWorkflows AWS CLICommand Reference.

`start-annotation-import-job`

Il seguente esempio di codice mostra come utilizzare`start-annotation-import-job`.

AWS CLI

Come importare annotazioni

L'esempio `start-annotation-import-job` seguente importa le annotazioni da Amazon S3.

```
aws omics start-annotation-import-job \
    --destination-name tsv_ann_store \
    --no-run-left-normalization \
    --role-arn arn:aws:iam::123456789012:role/omics-service-role-serviceRole-
W801XMPL7QZ \
    --items source=s3://omics-artifacts-01d6xmpl4e72dd32/targetedregions.bed.gz
```

Output:

```
{
    "jobId": "984162c7-xmpl-4d23-ab47-286f7950bfbf"
}
```

Per ulteriori informazioni, consulta Omics Analytics nella Guida per gli sviluppatori di Amazon Omics.

- Per i dettagli sull'API, consulta StartAnnotationImportJob AWS CLICommand Reference.

`start-read-set-activation-job`

Il seguente esempio di codice mostra come utilizzare`start-read-set-activation-job`.

AWS CLI

Come attivare un set di lettura archiviato

L'esempio `start-read-set-activation-job` seguente attiva due set di lettura.

```
aws omics start-read-set-activation-job \
    --sequence-store-id 1234567890 \
    --sources readSetId=1234567890 readSetId=1234567890
```

Output:

```
{
    "creationTime": "2022-12-06T22:35:10.100Z",
    "id": "1234567890",
    "sequenceStoreId": "1234567890",
    "status": "SUBMITTED"
}
```

Per ulteriori informazioni, consulta Archiviazione Omics nella Guida per gli sviluppatori di Amazon Omics.

- Per i dettagli sull'API, consulta StartReadSetActivationJob AWS CLICommand Reference.

`start-read-set-export-job`

Il seguente esempio di codice mostra come utilizzare`start-read-set-export-job`.

AWS CLI

Come esportare un set di lettura

L'esempio `start-read-set-export-job` seguente esporta due set di lettura in Amazon S3.

```
   aws omics start-read-set-export-job \
       --sequence-store-id 1234567890 \
       --sources readSetId=1234567890 readSetId=1234567890 \
       --role-arn arn:aws:iam::123456789012:role/omics-service-role-serviceRole-
W8O1XMPL7QZ
\
       --destination s3://omics-artifacts-01d6xmpl4e72dd32/read-set-export/
```

Output:

```
{
    "creationTime": "2022-12-06T22:37:18.612Z",
```

```
    "destination": "s3://omics-artifacts-01d6xmpl4e72dd32/read-set-export/",
    "id": "1234567890",
    "sequenceStoreId": "1234567890",
    "status": "SUBMITTED"
}
```

Per ulteriori informazioni, consulta Archiviazione Omics nella Guida per gli sviluppatori di Amazon Omics.

- Per i dettagli sull'API, consulta StartReadSetExportJob AWS CLICommand Reference.

`start-read-set-import-job`

Il seguente esempio di codice mostra come utilizzare`start-read-set-import-job`.

AWS CLI

Come importare un set di lettura

L'esempio `start-read-set-import-job` seguente importa un set di lettura.

```
aws omics start-read-set-import-job \
    --sequence-store-id 1234567890 \
    --role-arn arn:aws:iam::123456789012:role/omics-service-role-serviceRole-
W801XMPL7QZ \
    --sources file://readset-sources.json
```

readset-sources.json è un documento JSON con il seguente contenuto.

```
[
    {
        "sourceFiles":
        {
            "source1": "s3://omics-artifacts-01d6xmpl4e72dd32/
HG00100.chrom20.ILLUMINA.bwa.GBR.low_coverage.20101123.bam"
        },
        "sourceFileType": "BAM",
        "subjectId": "bam-subject",
        "sampleId": "bam-sample",
        "referenceArn": "arn:aws:omics:us-
west-2:123456789012:referenceStore/1234567890/reference/1234567890",
        "name": "HG00100"
```

```
    }
]
```

Output:

```
{
    "creationTime": "2022-11-23T01:36:38.158Z",
    "id": "1234567890",
    "roleArn": "arn:aws:iam::123456789012:role/omics-service-role-serviceRole-
W8O1XMPL7QZ",
    "sequenceStoreId": "1234567890",
    "status": "SUBMITTED"
}
```

Per ulteriori informazioni, consulta Archiviazione Omics nella Guida per gli sviluppatori di Amazon Omics.

- Per i dettagli sull'API, consulta StartReadSetImportJob AWS CLICommand Reference.

`start-reference-import-job`

Il seguente esempio di codice mostra come utilizzare`start-reference-import-job`.

AWS CLI

Come importare un genoma di riferimento

L'esempio `start-reference-import-job` seguente importa un genoma di riferimento da Amazon S3.

```
aws omics start-reference-import-job \
    --reference-store-id 1234567890 \
    --role-arn arn:aws:iam::123456789012:role/omics-service-role-serviceRole-
W8O1XMPL7QZ \
    --sources sourceFile=s3://omics-artifacts-01d6xmpl4e72dd32/
Homo_sapiens_assembly38.fasta,name=assembly-38
```

Output:

```
{
    "creationTime": "2022-11-22T22:25:41.124Z",
    "id": "1234567890",
```

```
    "referenceStoreId": "1234567890",
    "roleArn": "arn:aws:iam::123456789012:role/omics-service-role-serviceRole-
W8O1XMPL7QZ",
    "status": "SUBMITTED"
}
```

Per ulteriori informazioni, consulta Archiviazione Omics nella Guida per gli sviluppatori di Amazon Omics.

- Per i dettagli sull'API, consulta StartReferenceImportJob AWS CLICommand Reference.

`start-run`

Il seguente esempio di codice mostra come utilizzare`start-run`.

AWS CLI

Come eseguire un flusso di lavoro

L'esempio `start-run` seguente esegue un flusso di lavoro con ID `1234567`.

```
aws omics start-run \
    --workflow-id 1234567 \
    --role-arn arn:aws:iam::123456789012:role/omics-service-role-serviceRole-
W8O1XMPL7QZ \
    --name 'cram-to-bam' \
    --output-uri s3://omics-artifacts-01d6xmpl4e72dd32/workflow-output/ \
    --run-group-id 1234567 \
    --priority 1 \
    --storage-capacity 10 \
    --log-level ALL \
    --parameters file://workflow-inputs.json
```

workflow-inputs.json è un documento JSON con il seguente contenuto.

```
{
    "sample_name": "NA12878",
    "input_cram": "s3://omics-artifacts-01d6xmpl4e72dd32/NA12878.cram",
    "ref_dict": "s3://omics-artifacts-01d6xmpl4e72dd32/
Homo_sapiens_assembly38.dict",
    "ref_fasta": "s3://omics-artifacts-01d6xmpl4e72dd32/
Homo_sapiens_assembly38.fasta",
```

```
    "ref_fasta_index": "omics-artifacts-01d6xmpl4e72dd32/
Homo_sapiens_assembly38.fasta.fai"
}
```

Output:

```
{
    "arn": "arn:aws:omics:us-west-2:123456789012:run/1234567",
    "id": "1234567",
    "status": "PENDING",
    "tags": {}
}
```

Per ulteriori informazioni, vedere Avvio di una corsa nella Guida AWS HealthOmics per l'utente.

Come caricare file di origine da Amazon Omics

Puoi anche caricare file sorgente dallo storage Amazon Omics, utilizzando servizi URIs specifici. Il seguente file workflow-inputs.json di esempio utilizza Amazon URIs Omics per i set di lettura e le fonti genomiche di riferimento.

```
{
    "sample_name": "NA12878",
    "input_cram": "omics://123456789012.storage.us-west-2.amazonaws.com/1234567890/
readSet/1234567890/source1",
    "ref_dict": "s3://omics-artifacts-01d6xmpl4e72dd32/
Homo_sapiens_assembly38.dict",
    "ref_fasta": "omics://123456789012.storage.us-west-2.amazonaws.com/1234567890/
reference/1234567890",
    "ref_fasta_index": "omics://123456789012.storage.us-
west-2.amazonaws.com/1234567890/reference/1234567890/index"
}
```

- Per i dettagli sull'API, consulta Command Reference. StartRunAWS CLI

`start-variant-import-job`

Il seguente esempio di codice mostra come utilizzare`start-variant-import-job`.

AWS CLI

Come importare un file di varianti

L'esempio `start-variant-import-job` seguente importa un file di varianti in formato VCF.

```
aws omics start-variant-import-job \
    --destination-name my_var_store \
    --no-run-left-normalization  \
    --role-arn arn:aws:iam::123456789012:role/omics-service-role-serviceRole-W801XMPL7QZ \
    --items source=s3://omics-artifacts-01d6xmpl4e72dd32/Homo_sapiens_assembly38.known_indels.vcf.gz
```

Output:

```
{
    "jobId": "edd7b8ce-xmpl-47e2-bc99-258cac95a508"
}
```

Per ulteriori informazioni, consulta Omics Analytics nella Guida per gli sviluppatori di Amazon Omics.

- Per i dettagli sull'API, consulta StartVariantImportJob AWS CLICommand Reference.

`tag-resource`

Il seguente esempio di codice mostra come utilizzare`tag-resource`.

AWS CLI

Come taggare a una risorsa

L'esempio `tag-resource` seguente aggiunge un tag `department` a un flusso di lavoro con ID `1234567`.

```
aws omics tag-resource \
    --resource-arn arn:aws:omics:us-west-2:123456789012:workflow/1234567 \
    --tags department=analytics
```

Per ulteriori informazioni, consulta Tagging di risorse in Amazon Omics nella Guida per gli sviluppatori di Amazon Omics.

- Per i dettagli sull'API, consulta TagResource AWS CLICommand Reference.

`untag-resource`

Il seguente esempio di codice mostra come utilizzare`untag-resource`.

AWS CLI

Come rimuovere un tag da una risorsa

L'esempio `untag-resource` seguente rimuove il tag `department` da un flusso di lavoro.

```
aws omics untag-resource \
    --resource-arn arn:aws:omics:us-west-2:123456789012:workflow/1234567 \
    --tag-keys department
```

Per ulteriori informazioni, consulta Archiviazione Omics nella Guida per gli sviluppatori di Amazon Omics.

- Per i dettagli sull'API, consulta UntagResource AWS CLICommand Reference.

`update-annotation-store`

Il seguente esempio di codice mostra come utilizzare`update-annotation-store`.

AWS CLI

Come aggiornare un archivio di annotazioni

L'esempio `update-annotation-store` seguente aggiorna la descrizione di un archivio di annotazioni denominato `my_vcf_store`.

```
aws omics update-annotation-store \
    --name my_vcf_store \
    --description "VCF annotation store"
```

Output:

```
{
    "creationTime": "2022-12-05T18:00:56.101860Z",
    "description": "VCF annotation store",
    "id": "bd6axmpl2444",
    "name": "my_vcf_store",
    "reference": {
```

```
        "referenceArn": "arn:aws:omics:us-
west-2:123456789012:referenceStore/1234567890/reference/1234567890"
    },
    "status": "ACTIVE",
    "storeFormat": "VCF",
    "updateTime": "2022-12-05T18:13:16.100051Z"
}
```

Per ulteriori informazioni, consulta Omics Analytics nella Guida per gli sviluppatori di Amazon Omics.

- Per i dettagli sull'API, consulta UpdateAnnotationStore AWS CLICommand Reference.

`update-run-group`

Il seguente esempio di codice mostra come utilizzare`update-run-group`.

AWS CLI

Come aggiornare un gruppo di esecuzione

L'esempio `update-run-group` seguente aggiorna le impostazioni di un gruppo di esecuzione con ID `1234567`.

```
aws omics update-run-group \
    --id 1234567 \
    --max-cpus 10
```

Output:

```
{
    "arn": "arn:aws:omics:us-west-2:123456789012:runGroup/1234567",
    "creationTime": "2022-12-01T00:58:42.915219Z",
    "id": "1234567",
    "maxCpus": 10,
    "maxDuration": 600,
    "name": "cram-convert",
    "tags": {}
}
```

Per ulteriori informazioni, consulta Flussi di lavoro Omics nella Guida per gli sviluppatori di Amazon Omics.

- Per i dettagli sull'API, consulta UpdateRunGroup AWS CLICommand Reference.

`update-variant-store`

Il seguente esempio di codice mostra come utilizzare`update-variant-store`.

AWS CLI

Come aggiornare un archivio di varianti

L'esempio `update-variant-store` seguente aggiorna la descrizione di un archivio di varianti denominato `my_var_store`.

```
aws omics update-variant-store \
    --name my_var_store \
    --description "variant store"
```

Output:

```
{
    "creationTime": "2022-11-23T22:09:07.534499Z",
    "description": "variant store",
    "id": "02dexmplcfdd",
    "name": "my_var_store",
    "reference": {
        "referenceArn": "arn:aws:omics:us-
west-2:123456789012:referenceStore/1234567890/reference/1234567890"
    },
    "status": "ACTIVE",
    "updateTime": "2022-12-05T18:23:37.686402Z"
}
```

Per ulteriori informazioni, consulta Omics Analytics nella Guida per gli sviluppatori di Amazon Omics.

- Per i dettagli sull'API, consulta UpdateVariantStore AWS CLICommand Reference.

`update-workflow`

Il seguente esempio di codice mostra come utilizzare`update-workflow`.

AWS CLI

Come aggiornare un flusso di lavoro

L'esempio 1234567 seguente aggiorna la descrizione di un flusso di lavoro con ID update-workflow.

```
aws omics update-workflow \
    --id 1234567 \
    --description "copy workflow"
```

Per ulteriori informazioni, vedere Creazione o aggiornamento di un flusso di lavoro nella Guida AWS HealthOmics per l'utente.

- Per i dettagli sull'API, consulta UpdateWorkflow AWS CLICommand Reference.

upload-read-set-part

Il seguente esempio di codice mostra come utilizzareupload-read-set-part.

AWS CLI

Come caricare una parte di un set di lettura.

L'esempio upload-read-set-part seguente carica una parte specificata di un set di lettura.

```
aws omics upload-read-set-part \
    --sequence-store-id 0123456789 \
    --upload-id 1122334455 \
    --part-source SOURCE1 \
    --part-number 1 \
    --payload /path/to/file/read_1_part_1.fastq.gz
```

Output:

```
{
    "checksum": "984979b9928ae8d8622286c4a9cd8e99d964a22d59ed0f5722e1733eb280e635"
}
```

Per ulteriori informazioni, consulta Caricamento diretto su un archivio di sequenze nella Guida AWS HealthOmics per l'utente.

- Per i dettagli sull'API, consulta UploadReadSetPart AWS CLICommand Reference.

Esempi IAM che utilizzano AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with IAM.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`add-client-id-to-open-id-connect-provider`

Il seguente esempio di codice mostra come utilizzare`add-client-id-to-open-id-connect-provider`.

AWS CLI

Come aggiungere un ID client (pubblico) a un provider Open-ID Connect (OIDC)

Il comando `add-client-id-to-open-id-connect-provider` seguente aggiunge l'ID client `my-application-ID` al provider OIDC denominato `server.example.com`.

```
aws iam add-client-id-to-open-id-connect-provider \
    --client-id my-application-ID \
    --open-id-connect-provider-arn arn:aws:iam::123456789012:oidc-provider/
server.example.com
```

Questo comando non produce alcun output.

Come creare un provider OIDC, utilizzare il comando `create-open-id-connect-provider`.

Per ulteriori informazioni, consulta Creazione di provider di identità OpenID Connect (OIDC) nella Guida per l'utente di AWS IAM.

- Per i dettagli sull'API, consulta AddClientIdToOpenIdConnectProvider AWS CLICommand Reference.

`add-role-to-instance-profile`

Il seguente esempio di codice mostra come utilizzare`add-role-to-instance-profile`.

AWS CLI

Come aggiungere un ruolo a un profilo dell'istanza

Il comando `add-role-to-instance-profile` seguente aggiunge il ruolo denominato `S3Access` al profilo dell'istanza denominato `Webserver`.

```
aws iam add-role-to-instance-profile \
    --role-name S3Access \
    --instance-profile-name Webserver
```

Questo comando non produce alcun output.

Per creare un profilo dell'istanza, utilizza il comando `create-instance-profile`.

Per ulteriori informazioni, consulta Usare un ruolo IAM per concedere le autorizzazioni alle applicazioni in esecuzione su EC2 istanze Amazon nella AWS IAM User Guide.

- Per i dettagli sull'API, consulta AWS CLI Command AddRoleToInstanceProfileReference.

`add-user-to-group`

Il seguente esempio di codice mostra come utilizzare`add-user-to-group`.

AWS CLI

Come aggiungere un utente a un gruppo IAM

Il comando `add-user-to-group` seguente aggiunte l'utente IAM denominato `Bob` al gruppo IAM denominato `Admins`.

```
aws iam add-user-to-group \
    --user-name Bob \
```

```
    --group-name Admins
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Aggiunta e rimozione di utenti in un gruppo di utenti IAM nella Guida per l'utente IAM AWS .

- Per i dettagli sull'API, consulta AddUserToGroup AWS CLICommand Reference.

`attach-group-policy`

Il seguente esempio di codice mostra come utilizzare`attach-group-policy`.

AWS CLI

Come collegare una policy gestita a un gruppo IAM

Il `attach-group-policy` comando seguente collega la policy AWS gestita denominata `ReadOnlyAccess` al gruppo IAM denominato`Finance`.

```
aws iam attach-group-policy \
    --policy-arn arn:aws:iam::aws:policy/ReadOnlyAccess \
    --group-name Finance
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Policy gestite e policy inline nella Guida per l'utente IAM AWS .

- Per i dettagli sull'API, consulta AttachGroupPolicy AWS CLICommand Reference.

`attach-role-policy`

Il seguente esempio di codice mostra come utilizzare`attach-role-policy`.

AWS CLI

Come collegare una policy gestita a un ruolo IAM

Il `attach-role-policy` comando seguente collega la policy AWS gestita denominata `ReadOnlyAccess` al ruolo IAM denominato`ReadOnlyRole`.

```
aws iam attach-role-policy \
    --policy-arn arn:aws:iam::aws:policy/ReadOnlyAccess \
```

```
    --role-name ReadOnlyRole
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Policy gestite e policy inline nella Guida per l'utente IAM AWS .

- Per i dettagli sull'API, consulta AttachRolePolicy AWS CLICommand Reference.

`attach-user-policy`

Il seguente esempio di codice mostra come utilizzare`attach-user-policy`.

AWS CLI

Come collegare una policy gestita a un utente IAM

Il `attach-user-policy` comando seguente collega la policy AWS gestita denominata `AdministratorAccess` all'utente IAM denominato`Alice`.

```
aws iam attach-user-policy \
    --policy-arn arn:aws:iam::aws:policy/AdministratorAccess \
    --user-name Alice
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Policy gestite e policy inline nella Guida per l'utente IAM AWS .

- Per i dettagli sull'API, consulta AttachUserPolicy AWS CLICommand Reference.

`change-password`

Il seguente esempio di codice mostra come utilizzare`change-password`.

AWS CLI

Come modificare la password per l'utente IAM

Per modificare la password per il tuo utente IAM, ti consigliamo di utilizzare il parametro `--cli-input-json` per passare un file JSON che contenga la vecchia e la nuova password. Con questo metodo, potrai utilizzare password complesse con caratteri non alfanumerici. Quando le password vengono passate come parametri della riga di comando può essere difficile utilizzare password con caratteri non alfanumerici. Per utilizzare il parametro `--cli-input-json`, inizia a

utilizzare il comando `change-password` con il parametro `--generate-cli-skeleton`, come nell'esempio seguente.

```
aws iam change-password \
    --generate-cli-skeleton > change-password.json
```

Il comando precedente crea un file JSON chiamato change-password.json che può essere utilizzato per inserire la vecchia e la nuova password. Ad esempio, il file potrebbe avere il seguente aspetto.

```
{
    "OldPassword": "3s0K_;xh4~8XXI",
    "NewPassword": "]35d/{pB9Fo9wJ"
}
```

Quindi, per modificare la password, usa nuovamente il comando `change-password`, questa volta passando il parametro per specificare il file JSON `--cli-input-json`. Il comando `change-password` seguente utilizza il parametro `--cli-input-json` con un file JSON chiamato change-password.json.

```
aws iam change-password \
    --cli-input-json file://change-password.json
```

Questo comando non produce alcun output.

Questo comando può essere chiamato solo dagli utenti IAM. Se questo comando viene chiamato utilizzando le credenziali AWS dell'account (root), restituisce un `InvalidUserType` errore.

Per ulteriori informazioni, consulta Come permettere a un utente IAM di cambiare la propria password nella Guida per l'utente di AWS IAM.

- Per i dettagli sull'API, vedere ChangePasswordin AWS CLI Command Reference.

`create-access-key`

Il seguente esempio di codice mostra come utilizzare`create-access-key`.

AWS CLI

Come creare una chiave di accesso per un utente IAM

Il comando `create-access-key` seguente crea una chiave di accesso (ID chiave di accesso e chiave di accesso segreta) per l'utente IAM denominato Bob.

```
aws iam create-access-key \
    --user-name Bob
```

Output:

```
{
    "AccessKey": {
        "UserName": "Bob",
        "Status": "Active",
        "CreateDate": "2015-03-09T18:39:23.411Z",
        "SecretAccessKey": "wJalrXUtnFEMI/K7MDENG/bPxRfiCYzEXAMPLEKEY",
        "AccessKeyId": "AKIAIOSFODNN7EXAMPLE"
    }
}
```

Conserva la chiave di accesso segreta in un luogo sicuro. Se viene persa, non può essere recuperata e dovrai creare una nuova chiave di accesso.

Per ulteriori informazioni, consulta Gestione delle chiavi di accesso per gli utenti IAM nella Guida per l'utente di IAM AWS .

- Per i dettagli sull'API, consulta CreateAccessKey AWS CLICommand Reference.

`create-account-alias`

Il seguente esempio di codice mostra come utilizzare`create-account-alias`.

AWS CLI

Come Creare l'alias di un account

Il `create-account-alias` comando seguente crea l'alias `examplecorp` per il tuo AWS account.

```
aws iam create-account-alias \
    --account-alias examplecorp
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Your AWS account ID and its alias nella AWS IAM User Guide.

- Per i dettagli sull'API, consulta CreateAccountAlias AWS CLICommand Reference.

`create-group`

Il seguente esempio di codice mostra come utilizzare`create-group`.

AWS CLI

Come creare un gruppo IAM

Il comando `create-group` seguente crea un gruppo IAM denominato `Admins`.

```
aws iam create-group \
    --group-name Admins
```

Output:

```
{
    "Group": {
        "Path": "/",
        "CreateDate": "2015-03-09T20:30:24.940Z",
        "GroupId": "AIDGPMS9RO4H3FEXAMPLE",
        "Arn": "arn:aws:iam::123456789012:group/Admins",
        "GroupName": "Admins"
    }
}
```

Per ulteriori informazioni, consulta Creazione di gruppi di utenti IAM nella Guida per l'utente IAM AWS .

- Per i dettagli sull'API, consulta CreateGroup AWS CLICommand Reference.

`create-instance-profile`

Il seguente esempio di codice mostra come utilizzare`create-instance-profile`.

AWS CLI

Come creare un profilo dell'istanza

Il comando `create-instance-profile` seguente crea un profilo dell'istanza denominato `Webserver`.

```
aws iam create-instance-profile \
    --instance-profile-name Webserver
```

Output:

```
{
    "InstanceProfile": {
        "InstanceProfileId": "AIPAJMBYC7DLSPEXAMPLE",
        "Roles": [],
        "CreateDate": "2015-03-09T20:33:19.626Z",
        "InstanceProfileName": "Webserver",
        "Path": "/",
        "Arn": "arn:aws:iam::123456789012:instance-profile/Webserver"
    }
}
```

Per aggiungere un ruolo a un profilo dell'istanza, utilizza il comando `add-role-to-instance-profile`.

Per ulteriori informazioni, consulta Usare un ruolo IAM per concedere le autorizzazioni alle applicazioni in esecuzione su EC2 istanze Amazon nella AWS IAM User Guide.

- Per i dettagli sull'API, consulta AWS CLI Command CreateInstanceProfileReference.

`create-login-profile`

Il seguente esempio di codice mostra come utilizzare`create-login-profile`.

AWS CLI

Come creare una password per un utente IAM

Per creare una password per un utente IAM, ti consigliamo di utilizzare il parametro `--cli-input-json` per passare un file JSON che contenga la password. Con questo metodo, potrai creare una password complessa con caratteri non alfanumerici. Quando le password vengono passate come parametri della riga di comando può essere difficile utilizzare password con caratteri non alfanumerici.

Per utilizzare il parametro `--cli-input-json`, inizia a utilizzare il comando `create-login-profile` con il parametro `--generate-cli-skeleton`, come nell'esempio seguente.

```
aws iam create-login-profile \
    --generate-cli-skeleton > create-login-profile.json
```

Il comando precedente crea un file JSON chiamato create-login-profile .json che è possibile utilizzare per inserire le informazioni per un comando successivo. `create-login-profile` Esempio:

```
{
    "UserName": "Bob",
    "Password": "&1-3a6u:RA0djs",
    "PasswordResetRequired": true
}
```

Quindi, per modificare la password per un utente IAM, usa nuovamente il comando `create-login-profile`, questa volta passando il parametro `--cli-input-json` per specificare il file JSON. Il `create-login-profile` comando seguente utilizza il `--cli-input-json` parametro con un file JSON chiamato .json. create-login-profile

```
aws iam create-login-profile \
    --cli-input-json file://create-login-profile.json
```

Output:

```
{
    "LoginProfile": {
        "UserName": "Bob",
        "CreateDate": "2015-03-10T20:55:40.274Z",
        "PasswordResetRequired": true
    }
}
```

Se la nuova password viola la policy delle password dell'account, il comando restituisce un errore `PasswordPolicyViolation`.

Per modificare la password di un utente che ne ha già una, usa `update-login-profile`. Per impostare una policy delle password per l'account, utilizza il comando `update-account-password-policy`.

Se la policy delle password dell'account lo consente, gli utenti IAM possono modificare le proprie password utilizzando il comando `change-password`.

Per ulteriori informazioni, consulta Gestione delle password per gli utenti IAM nella Guida per l'utente di AWS .

- Per i dettagli sull'API, consulta Command CreateLoginProfileReference AWS CLI .

`create-open-id-connect-provider`

Il seguente esempio di codice mostra come utilizzare`create-open-id-connect-provider`.

AWS CLI

Come creare un provider OpenID Connect (OIDC)

Per creare un provider OpenID Connect (OIDC), consigliamo di utilizzare il parametro `--cli-input-json` per passare un file JSON che contenga i parametri richiesti. Quando si crea un provider OIDC, è necessario passare l'URL del provider e l'URL deve iniziare con `https://`. Può essere difficile passare l'URL come parametro della riga di comando, perché i caratteri due punti (:) e barra (/) hanno un significato speciale in alcuni ambienti della riga di comando. L'utilizzo del parametro `--cli-input-json` consente di aggirare questa limitazione.

Per utilizzare il parametro `--cli-input-json`, inizia a utilizzare il comando `create-open-id-connect-provider` con il parametro `--generate-cli-skeleton`, come nell'esempio seguente.

```
aws iam create-open-id-connect-provider \
    --generate-cli-skeleton > create-open-id-connect-provider.json
```

Il comando precedente crea un file JSON chiamato create-open-id-connect -provider.json che è possibile utilizzare per inserire le informazioni per un comando successivo. `create-open-id-connect-provider` Esempio:

```
{
    "Url": "https://server.example.com",
    "ClientIDList": [
        "example-application-ID"
    ],
    "ThumbprintList": [
        "c3768084dfb3d2b68b7897bf5f565da8eEXAMPLE"
```

```
    ]
}
```

Successivamente, per creare il provider OpenID Connect (OIDC), utilizza nuovamente il comando `create-open-id-connect-provider`, questa volta passando il parametro `--cli-input-json` per specificare il file JSON. Il `create-open-id-connect-provider` comando seguente utilizza il `--cli-input-json` parametro con un file JSON chiamato -provider.json. create-open-id-connect

```
aws iam create-open-id-connect-provider \
    --cli-input-json file://create-open-id-connect-provider.json
```

Output:

```
{
    "OpenIDConnectProviderArn": "arn:aws:iam::123456789012:oidc-provider/
server.example.com"
}
```

Per ulteriori informazioni sui provider OIDC, consulta Creazione di provider di identità OpenID Connect (OIDC) nella Guida per l'utente di AWS IAM.

Per ulteriori informazioni su come ottenere impronte digitali per un provider OIDC, consulta Ottenere l'impronta personale per un provider di identità OpenID Connect nella Guida per l'utente di AWS IAM.

- Per i dettagli sull'API, vedere in Command Reference. CreateOpenIdConnectProviderAWS CLI

`create-policy-version`

Il seguente esempio di codice mostra come utilizzare`create-policy-version`.

AWS CLI

Come creare una nuova versione di una policy gestita

Questo esempio crea una nuova versione v2 della policy IAM il cui ARN è `arn:aws:iam::123456789012:policy/MyPolicy` e la rende la versione predefinita.

```
aws iam create-policy-version \
```

```
    --policy-arn arn:aws:iam::123456789012:policy/MyPolicy \
    --policy-document file://NewPolicyVersion.json \
    --set-as-default
```

Output:

```
{
    "PolicyVersion": {
        "CreateDate": "2015-06-16T18:56:03.721Z",
        "VersionId": "v2",
        "IsDefaultVersion": true
    }
}
```

Per ulteriori informazioni, consulta Controllo delle versioni delle policy IAM nella Guida per l'utente IAM AWS .

- Per i dettagli sull'API, consulta CreatePolicyVersion AWS CLICommand Reference.

`create-policy`

Il seguente esempio di codice mostra come utilizzare`create-policy`.

AWS CLI

Esempio 1: Come creare una policy gestita dal cliente

Il comando seguente crea una policy gestita dal cliente denominata `my-policy`. Il file `policy.json` è un documento JSON nella cartella corrente che consente l'accesso in sola lettura alla cartella `shared` in un bucket Amazon S3 denominato `amzn-s3-demo-bucket`.

```
aws iam create-policy \
    --policy-name my-policy \
    --policy-document file://policy.json
```

Contenuto di policy.json:

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
```

```
            "Effect": "Allow",
            "Action": [
                "s3:Get*",
                "s3:List*"
            ],
            "Resource": [
                "arn:aws:s3:::amzn-s3-demo-bucket/shared/*"
            ]
        }
    ]
}
```

Output:

```
{
    "Policy": {
        "PolicyName": "my-policy",
        "CreateDate": "2015-06-01T19:31:18.620Z",
        "AttachmentCount": 0,
        "IsAttachable": true,
        "PolicyId": "ZXR6A36LTYANPAI7NJ5UV",
        "DefaultVersionId": "v1",
        "Path": "/",
        "Arn": "arn:aws:iam::0123456789012:policy/my-policy",
        "UpdateDate": "2015-06-01T19:31:18.620Z"
    }
}
```

Per ulteriori informazioni sull'utilizzo dei file come input per i parametri di stringa, consultate Specificare i valori dei parametri per la AWS CLI nella AWS CLI User Guide.

Esempio 2: Come creare una policy gestita dal cliente con una descrizione

Il comando seguente crea una policy gestita dal cliente denominata `my-policy` con una descrizione non modificabile.

Il file `policy.json` è un documento JSON nella cartella corrente che consente l'accesso in sola lettura a tutte le operazioni Put, List e Get per un bucket Amazon S3 denominato `amzn-s3-demo-bucket`.

```
aws iam create-policy \
    --policy-name my-policy \
```

```
    --policy-document file://policy.json \
    --description "This policy grants access to all Put, Get, and List actions for
 amzn-s3-demo-bucket"
```

Contenuto di policy.json:

```
{
   "Version":"2012-10-17",
   "Statement": [
       {
           "Effect": "Allow",
           "Action": [
                "s3:ListBucket*",
                "s3:PutBucket*",
                "s3:GetBucket*"
            ],
            "Resource": [
                "arn:aws:s3:::amzn-s3-demo-bucket"
            ]
        }
    ]
}
```

Output:

```
{
    "Policy": {
        "PolicyName": "my-policy",
        "PolicyId": "ANPAWGSUGIDPEXAMPLE",
        "Arn": "arn:aws:iam::123456789012:policy/my-policy",
        "Path": "/",
        "DefaultVersionId": "v1",
        "AttachmentCount": 0,
        "PermissionsBoundaryUsageCount": 0,
        "IsAttachable": true,
        "CreateDate": "2023-05-24T22:38:47+00:00",
        "UpdateDate": "2023-05-24T22:38:47+00:00"
    }
}
```

Per ulteriori informazioni sulle policy basate sull'identità consulta Policy basate sulle identità e policy basate su risorse nella Guida per l'utente IAM AWS .

Esempio 3: creare una policy gestita dal cliente con i tag

Il comando seguente crea una policy gestita dal cliente denominata `my-policy` con tag. Questo esempio utilizza il parametro `--tags` con i seguenti tag in formato JSON: `'{"Key": "Department", "Value": "Accounting"}' '{"Key": "Location", "Value": "Seattle"}'`. In alternativa, il parametro `--tags` può essere utilizzato con i tag in formato abbreviato: `'Key=Department,Value=Accounting Key=Location,Value=Seattle'`.

Il file `policy.json` è un documento JSON nella cartella corrente che consente l'accesso in sola lettura a tutte le operazioni Put, List e Get per un bucket Amazon S3 denominato `amzn-s3-demo-bucket`.

```
aws iam create-policy \
    --policy-name my-policy \
    --policy-document file://policy.json \
    --tags '{"Key": "Department", "Value": "Accounting"}' '{"Key": "Location",
 "Value": "Seattle"}'
```

Contenuto di policy.json:

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "s3:ListBucket*",
                "s3:PutBucket*",
                "s3:GetBucket*"
            ],
            "Resource": [
                "arn:aws:s3:::amzn-s3-demo-bucket"
            ]
        }
    ]
}
```

Output:

```
{
    "Policy": {
```

```
        "PolicyName": "my-policy",
        "PolicyId": "ANPAWGSUGIDPEXAMPLE",
        "Arn": "arn:aws:iam::12345678012:policy/my-policy",
        "Path": "/",
        "DefaultVersionId": "v1",
        "AttachmentCount": 0,
        "PermissionsBoundaryUsageCount": 0,
        "IsAttachable": true,
        "CreateDate": "2023-05-24T23:16:39+00:00",
        "UpdateDate": "2023-05-24T23:16:39+00:00",
        "Tags": [
            {
                "Key": "Department",
                "Value": "Accounting"
            },
                "Key": "Location",
                "Value": "Seattle"
            {
        ]
    }
}
```

Per ulteriori informazioni sulle policy di applicazione di tag, consulta Applicazione di tag a policy gestite dal cliente nella Guida per l'utente IAM AWS .

- Per i dettagli sull'API, consulta AWS CLI Command CreatePolicyReference.

`create-role`

Il seguente esempio di codice mostra come utilizzare`create-role`.

AWS CLI

Esempio 1: Come creare un ruolo IAM

Il comando `create-role` seguente crea un ruolo denominato `Test-Role` e collega una policy di attendibilità a tale ruolo.

```
aws iam create-role \
    --role-name Test-Role \
    --assume-role-policy-document file://Test-Role-Trust-Policy.json
```

Output:

```
{
    "Role": {
        "AssumeRolePolicyDocument": "<URL-encoded-JSON>",
        "RoleId": "AKIAIOSFODNN7EXAMPLE",
        "CreateDate": "2013-06-07T20:43:32.821Z",
        "RoleName": "Test-Role",
        "Path": "/",
        "Arn": "arn:aws:iam::123456789012:role/Test-Role"
    }
}
```

La policy di attendibilità è definita come documento JSON nel file Test-Role-Trust-Policy.json. (Il nome e l'estensione del file non hanno importanza.) La policy di attendibilità deve specificare un principale.

Per collegare una policy di autorizzazioni a un ruolo, usa il comando `put-role-policy`.

Per ulteriori informazioni, consulta Creazione di ruoli IAM nella Guida per l'utente IAM AWS .

Esempio 2: Come creare un ruolo IAM con una durata massima della sessione specificata

Il comando `create-role` seguente crea un ruolo denominato `Test-Role` e imposta una durata massima della sessione di 7200 secondi (2 ore).

```
aws iam create-role \
    --role-name Test-Role \
    --assume-role-policy-document file://Test-Role-Trust-Policy.json \
    --max-session-duration 7200
```

Output:

```
{
    "Role": {
        "Path": "/",
        "RoleName": "Test-Role",
        "RoleId": "AKIAIOSFODNN7EXAMPLE",
        "Arn": "arn:aws:iam::12345678012:role/Test-Role",
        "CreateDate": "2023-05-24T23:50:25+00:00",
        "AssumeRolePolicyDocument": {
            "Version":"2012-10-17",
            "Statement": [
```

```
                {
                    "Sid": "Statement1",
                    "Effect": "Allow",
                    "Principal": {
                        "AWS": "arn:aws:iam::12345678012:root"
                    },
                    "Action": "sts:AssumeRole"
                }
            ]
        }
    }
}
```

Per ulteriori informazioni, consulta Modificare la durata massima della sessione (AWS API) di un ruolo nella Guida per l'utente AWS IAM.

Esempio 3: Come creare un ruolo IAM con tag

Il comando seguente crea un ruolo IAM `Test-Role` con tag. Questo esempio utilizza il flag del parametro `--tags` con i seguenti tag in formato JSON: `'{"Key": "Department", "Value": "Accounting"}' '{"Key": "Location", "Value": "Seattle"}'`. In alternativa, il flag `--tags` può essere utilizzato con tag in formato abbreviato: `'Key=Department,Value=Accounting Key=Location,Value=Seattle'`.

```
aws iam create-role \
    --role-name Test-Role \
    --assume-role-policy-document file://Test-Role-Trust-Policy.json \
    --tags '{"Key": "Department", "Value": "Accounting"}' '{"Key": "Location",
 "Value": "Seattle"}'
```

Output:

```
{
    "Role": {
        "Path": "/",
        "RoleName": "Test-Role",
        "RoleId": "AKIAIOSFODNN7EXAMPLE",
        "Arn": "arn:aws:iam::123456789012:role/Test-Role",
        "CreateDate": "2023-05-25T23:29:41+00:00",
        "AssumeRolePolicyDocument": {
            "Version":"2012-10-17",
            "Statement": [
```

```
                {
                    "Sid": "Statement1",
                    "Effect": "Allow",
                    "Principal": {
                        "AWS": "arn:aws:iam::123456789012:root"
                    },
                    "Action": "sts:AssumeRole"
                }
            ]
        },
        "Tags": [
            {
                "Key": "Department",
                "Value": "Accounting"
            },
            {
                "Key": "Location",
                "Value": "Seattle"
            }
        ]
    }
}
```

Per ulteriori informazioni, consulta Applicazione di tag a ruoli IAM nella Guida per l'utente di IAM AWS .

- Per i dettagli sull'API, consulta CreateRole AWS CLICommand Reference.

`create-saml-provider`

Il seguente esempio di codice mostra come utilizzare`create-saml-provider`.

AWS CLI

Come creare un provider SAML

Questo esempio crea un nuovo provider SAML in IAM denominato `MySAMLProvider`. È descritto dal documento di metadati SAML che si trova nel file `SAMLMetaData.xml`.

```
aws iam create-saml-provider \
    --saml-metadata-document file://SAMLMetaData.xml \
    --name MySAMLProvider
```

Output:

```
{
    "SAMLProviderArn": "arn:aws:iam::123456789012:saml-provider/MySAMLProvider"
}
```

Per ulteriori informazioni, consulta Creazione di provider di identità SAML IAM nella Guida per l'utente di IAM AWS .

- Per i dettagli sull'API, consulta Create SAMLProvider in AWS CLI Command Reference.

`create-service-linked-role`

Il seguente esempio di codice mostra come utilizzare`create-service-linked-role`.

AWS CLI

Come creare un ruolo collegato a un servizio

L'`create-service-linked-role`esempio seguente crea un ruolo collegato al servizio per il AWS servizio specificato e allega la descrizione specificata.

```
aws iam create-service-linked-role \
    --aws-service-name lex.amazonaws.com \
    --description "My service-linked role to support Lex"
```

Output:

```
{
    "Role": {
        "Path": "/aws-service-role/lex.amazonaws.com/",
        "RoleName": "AWSServiceRoleForLexBots",
        "RoleId": "AROA1234567890EXAMPLE",
        "Arn": "arn:aws:iam::1234567890:role/aws-service-role/lex.amazonaws.com/
AWSServiceRoleForLexBots",
        "CreateDate": "2019-04-17T20:34:14+00:00",
        "AssumeRolePolicyDocument": {
            "Version":"2012-10-17",
            "Statement": [
                {
                    "Action": [
                        "sts:AssumeRole"
```

```
                    ],
                    "Effect": "Allow",
                    "Principal": {
                        "Service": [
                            "lex.amazonaws.com"
                        ]
                    }
                }
            ]
        }
    }
}
```

Per ulteriori informazioni, consulta Utilizzo di ruoli collegati a servizi nella Guida per l'utente di IAM AWS .

- Per i dettagli sull'API, vedere CreateServiceLinkedRolein AWS CLI Command Reference.

`create-service-specific-credential`

Il seguente esempio di codice mostra come utilizzare`create-service-specific-credential`.

AWS CLI

Per creare un set di credenziali specifiche del servizio per un utente

L'esempio `create-service-specific-credential` seguente crea un nome utente e una password che possono essere utilizzati per accedere solo al servizio configurato.

```
aws iam create-service-specific-credential \
    --user-name sofia \
    --service-name codecommit.amazonaws.com
```

Output:

```
{
    "ServiceSpecificCredential": {
        "CreateDate": "2019-04-18T20:45:36+00:00",
        "ServiceName": "codecommit.amazonaws.com",
        "ServiceUserName": "sofia-at-123456789012",
        "ServicePassword": "k1zPZM6uVxMQ3oxqgoYlNuJPyRTZ1vREs76zTQE3eJk=",
        "ServiceSpecificCredentialId": "ACCAEXAMPLE123EXAMPLE",
```

```
        "UserName": "sofia",
        "Status": "Active"
    }
}
```

Per ulteriori informazioni, consulta Creare credenziali Git per le connessioni HTTPS CodeCommit nella Guida per l'AWS CodeCommit utente.

- Per i dettagli sull'API, consulta CreateServiceSpecificCredential AWS CLICommand Reference.

`create-user`

Il seguente esempio di codice mostra come utilizzare`create-user`.

AWS CLI

Esempio 1: Come creare un utente IAM

Il comando `create-user` seguente crea un utente IAM denominato Bob nell'account corrente.

```
aws iam create-user \
    --user-name Bob
```

Output:

```
{
    "User": {
        "UserName": "Bob",
        "Path": "/",
        "CreateDate": "2023-06-08T03:20:41.270Z",
        "UserId": "AIDAIOSFODNN7EXAMPLE",
        "Arn": "arn:aws:iam::123456789012:user/Bob"
    }
}
```

Per ulteriori informazioni, consulta Creating an IAM user in your AWS account nella AWS IAM User Guide.

Esempio 2: Come creare un utente IAM in un percorso specificato

Il comando `create-user` seguente crea un utente IAM denominato Bob nel percorso specificato.

```
aws iam create-user \
    --user-name Bob \
    --path /division_abc/subdivision_xyz/
```

Output:

```
{
    "User": {
        "Path": "/division_abc/subdivision_xyz/",
        "UserName": "Bob",
        "UserId": "AIDAIOSFODNN7EXAMPLE",
        "Arn": "arn:aws:iam::12345678012:user/division_abc/subdivision_xyz/Bob",
        "CreateDate": "2023-05-24T18:20:17+00:00"
    }
}
```

Per ulteriori informazioni, consulta Identificatori IAM nella Guida per l'utente di IAM AWS .

Esempio 3: Come creare un utente IAM con tag

Il comando `create-user` seguente crea un utente IAM denominato Bob con tag. Questo esempio utilizza il flag del parametro `--tags` con i seguenti tag in formato JSON: `'{"Key": "Department", "Value": "Accounting"}' '{"Key": "Location", "Value": "Seattle"}'`. In alternativa, il flag `--tags` può essere utilizzato con tag in formato abbreviato: `'Key=Department,Value=Accounting Key=Location,Value=Seattle'`.

```
aws iam create-user \
    --user-name Bob \
    --tags '{"Key": "Department", "Value": "Accounting"}' '{"Key": "Location",
 "Value": "Seattle"}'
```

Output:

```
{
    "User": {
        "Path": "/",
        "UserName": "Bob",
        "UserId": "AIDAIOSFODNN7EXAMPLE",
        "Arn": "arn:aws:iam::12345678012:user/Bob",
        "CreateDate": "2023-05-25T17:14:21+00:00",
        "Tags": [
```

```
            {
                "Key": "Department",
                "Value": "Accounting"
            },
            {
                "Key": "Location",
                "Value": "Seattle"
            }
        ]
    }
}
```

Per ulteriori informazioni, consulta Applicazione di tag a utenti IAM nella Guida per l'utente di IAM AWS .

Esempio 3: Come creare un utente IAM con un limite delle autorizzazioni impostato

Il `create-user` comando seguente crea un utente IAM denominato Bob con il limite delle autorizzazioni di AmazonS3. FullAccess

```
aws iam create-user \
    --user-name Bob \
    --permissions-boundary arn:aws:iam::aws:policy/AmazonS3FullAccess
```

Output:

```
{
    "User": {
        "Path": "/",
        "UserName": "Bob",
        "UserId": "AIDAIOSFODNN7EXAMPLE",
        "Arn": "arn:aws:iam::12345678012:user/Bob",
        "CreateDate": "2023-05-24T17:50:53+00:00",
        "PermissionsBoundary": {
        "PermissionsBoundaryType": "Policy",
        "PermissionsBoundaryArn": "arn:aws:iam::aws:policy/AmazonS3FullAccess"
        }
    }
}
```

Per ulteriori informazioni, consulta Limiti delle autorizzazioni per le entità IAM nella Guida per l'utente di IAM AWS .

- Per i dettagli sull'API, consulta Command Reference. CreateUserAWS CLI

`create-virtual-mfa-device`

Il seguente esempio di codice mostra come utilizzare`create-virtual-mfa-device`.

AWS CLI

Come creare un dispositivo MFA virtuale

Questo esempio crea un nuovo dispositivo MFA virtuale chiamato `BobsMFADevice`. Crea un file che contiene le informazioni di bootstrap chiamate `QRCode.png` e le inserisce nella directory `C:/`. Il metodo bootstrap utilizzato in questo esempio è `QRCodePNG`.

```
aws iam create-virtual-mfa-device \
    --virtual-mfa-device-name BobsMFADevice \
    --outfile C:/QRCode.png \
    --bootstrap-method QRCodePNG
```

Output:

```
{
    "VirtualMFADevice": {
        "SerialNumber": "arn:aws:iam::210987654321:mfa/BobsMFADevice"
}
```

Per ulteriori informazioni, consulta Utilizzo dell'autenticazione a più fattori (MFA) in AWS nella AWS Guida per l'utente IAM.

- Per i dettagli sull'API, consulta CreateVirtualMfaDevice AWS CLICommand Reference.

`deactivate-mfa-device`

Il seguente esempio di codice mostra come utilizzare`deactivate-mfa-device`.

AWS CLI

Come disattivare un dispositivo MFA

Questo comando disattiva il dispositivo MFA virtuale con l'ARN `arn:aws:iam::210987654321:mfa/BobsMFADevice` associato all'utente Bob.

```
aws iam deactivate-mfa-device \
    --user-name Bob \
    --serial-number arn:aws:iam::210987654321:mfa/BobsMFADevice
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Utilizzo dell'autenticazione a più fattori (MFA) in AWS nella AWS Guida per l'utente IAM.

- Per i dettagli sull'API, consulta DeactivateMfaDevice AWS CLICommand Reference.

`decode-authorization-message`

Il seguente esempio di codice mostra come utilizzare`decode-authorization-message`.

AWS CLI

Come decodificare un messaggio di errore di autorizzazione

L'`decode-authorization-message`esempio seguente decodifica il messaggio restituito dalla EC2 console quando si tenta di avviare un'istanza senza le autorizzazioni richieste.

```
aws sts decode-authorization-message \
    --encoded-message lxzA8VEjEvu-s0TTt3PgYCXik9YakOqsrFJGRZR98xNcyWAxwRq14xIvd-
npzbgTevuufCTbjeBAaDARg9cbTK1rJbg3awM33o-Vy3ebPErE2-
mWR9hVYdvX-0zKgVOWF9pWjZaJSMqxB-aLXo-I_8TTvBq88x8IFPbMArNdpu0IjxDjzf22PF3SOE3XvIQ-
_PEO0aUqHCCcsSrFtvxm6yQD1nbm6VTIVrfa0Bzy8lsoMo7SjIaJ2r5vph6SY5vCCwg6o2JKe3hIHTa8zRrDbZSFMkcX
Xx9AYAAIr6bhcis7C__bZh4dlAAWooHFGKgfoJcWGwgdzgbu9hWyVvKTpeot5hsb8qANYjJRCPXTKpi6PZfdijIkwb6g
```

L'output è formattato come una stringa di testo JSON a riga singola che può essere analizzata con qualsiasi elaboratore di testo JSON.

```
{
    "DecodedMessage": "{\"allowed\":false,\"explicitDeny\":false,\"matchedStatements
\":{\"items\":[]},\"failures\":{\"items\":[]},\"context\":{\"principal
\":{\"id\":\"AIDAV3ZUEFP6J7GY7O6LO\",\"name\":\"chain-user\",\"arn\":
\"arn:aws:iam::403299380220:user/chain-user\"},\"action\":\"ec2:RunInstances\",
\"resource\":\"arn:aws:ec2:us-east-2:403299380220:instance/*\",\"conditions\":
{\"items\":[{\"key\":\"ec2:InstanceMarketType\",\"values\":{\"items\":[{\"value
\":\"on-demand\"}]}},{\"key\":\"aws:Resource\",\"values\":{\"items\":[{\"value
\":\"instance/*\"}]}},{\"key\":\"aws:Account\",\"values\":{\"items\":[{\"value
\":\"403299380220\"}]}},{\"key\":\"ec2:AvailabilityZone\",\"values\":{\"items\":
```

```
[{\"value\":\"us-east-2b\"}]}},{\"key\":\"ec2:ebsOptimized\",\"values\":{\"items
\":[{\"value\":\"false\"}]}},{\"key\":\"ec2:IsLaunchTemplateResource\",\"values
\":{\"items\":[{\"value\":\"false\"}]}},{\"key\":\"ec2:InstanceType\",\"values
\":{\"items\":[{\"value\":\"t2.micro\"}]}},{\"key\":\"ec2:RootDeviceType\",
\"values\":{\"items\":[{\"value\":\"ebs\"}]}},{\"key\":\"aws:Region\",\"values
\":{\"items\":[{\"value\":\"us-east-2\"}]}},{\"key\":\"aws:Service\",\"values
\":{\"items\":[{\"value\":\"ec2\"}]}},{\"key\":\"ec2:InstanceID\",\"values\":
{\"items\":[{\"value\":\"*\"}]}},{\"key\":\"aws:Type\",\"values\":{\"items\":
[{\"value\":\"instance\"}]}},{\"key\":\"ec2:Tenancy\",\"values\":{\"items\":
[{\"value\":\"default\"}]}},{\"key\":\"ec2:Region\",\"values\":{\"items\":[{\"value
\":\"us-east-2\"}]}},{\"key\":\"aws:ARN\",\"values\":{\"items\":[{\"value\":
\"arn:aws:ec2:us-east-2:403299380220:instance/*\"}]}}]}}"
}
```

Per ulteriori informazioni, vedi Come posso decodificare un messaggio di errore di autorizzazione dopo aver ricevuto un errore "UnauthorizedOperation" durante l'avvio di un'istanza? EC2 in AWS Re:post.

- Per i dettagli sull'API, vedi DecodeAuthorizationMessagein AWS CLI Command Reference.

`delete-access-key`

Il seguente esempio di codice mostra come utilizzare`delete-access-key`.

AWS CLI

Come eliminare una chiave di accesso per un utente IAM

Il comando `delete-access-key` seguente elimina la chiave di accesso specificata (ID chiave di accesso e chiave di accesso segreta) per l'utente IAM denominato Bob.

```
aws iam delete-access-key \
    --access-key-id AKIDPMS9RO4H3FEXAMPLE \
    --user-name Bob
```

Questo comando non produce alcun output.

Per elencare le chiavi di accesso definite per un utente IAM, usa il comando `list-access-keys`.

Per ulteriori informazioni, consulta Gestione delle chiavi di accesso per gli utenti IAM nella Guida per l'utente di IAM AWS .

- Per i dettagli sull'API, consulta DeleteAccessKey AWS CLICommand Reference.

`delete-account-alias`

Il seguente esempio di codice mostra come utilizzare`delete-account-alias`.

AWS CLI

Come eliminare l'alias di un account

Il comando `delete-account-alias` seguente rimuove l'alias `mycompany` per l'account corrente.

```
aws iam delete-account-alias \
    --account-alias mycompany
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Your AWS account ID and its alias nella AWS IAM User Guide.

- Per i dettagli sull'API, consulta DeleteAccountAlias AWS CLICommand Reference.

`delete-account-password-policy`

Il seguente esempio di codice mostra come utilizzare`delete-account-password-policy`.

AWS CLI

Come eliminare la policy delle password dell'account corrente

Il comando `delete-account-password-policy` seguente rimuove la policy delle password per l'account corrente.

```
aws iam delete-account-password-policy
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Impostazione di una policy delle password dell'account per utenti IAM nella Guida per l'utente di IAM AWS .

- Per i dettagli sull'API, consulta DeleteAccountPasswordPolicy AWS CLICommand Reference.

`delete-group-policy`

Il seguente esempio di codice mostra come utilizzare`delete-group-policy`.

AWS CLI

Come eliminare una policy da un gruppo IAM

Il comando `delete-group-policy` seguente elimina la policy denominata `ExamplePolicy` dal gruppo denominato `Admins`.

```
aws iam delete-group-policy \
    --group-name Admins \
    --policy-name ExamplePolicy
```

Questo comando non produce alcun output.

Per visualizzare le policy collegate a un gruppo, usa il comando `list-group-policies`.

Per ulteriori informazioni, consulta Gestione delle policy IAM nella Guida per l'utente IAM AWS .

- Per i dettagli sull'API, consulta DeleteGroupPolicy AWS CLICommand Reference.

`delete-group`

Il seguente esempio di codice mostra come utilizzare`delete-group`.

AWS CLI

Come eliminare un gruppo IAM

Il comando `delete-group` seguente elimina un gruppo IAM denominato `MyTestGroup`.

```
aws iam delete-group \
    --group-name MyTestGroup
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Eliminazione di un gruppo di utenti IAM nella Guida per l'utente di IAM AWS .

- Per i dettagli sull'API, consulta DeleteGroup AWS CLICommand Reference.

`delete-instance-profile`

Il seguente esempio di codice mostra come utilizzare`delete-instance-profile`.

AWS CLI

Come eliminare un profilo dell'istanza

Il comando `delete-instance-profile` seguente elimina un profilo dell'istanza denominato `ExampleInstanceProfile`.

```
aws iam delete-instance-profile \
    --instance-profile-name ExampleInstanceProfile
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Utilizzo dei profili dell'istanza nella Guida per l'utente di IAM AWS .

- Per i dettagli sull'API, consulta DeleteInstanceProfile AWS CLICommand Reference.

`delete-login-profile`

Il seguente esempio di codice mostra come utilizzare`delete-login-profile`.

AWS CLI

Come eliminare una password per un utente IAM

Il comando `delete-login-profile` seguente elimina la password per l'utente IAM denominato Bob.

```
aws iam delete-login-profile \
    --user-name Bob
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Gestione delle password per gli utenti IAM nella Guida per l'utente di AWS .

- Per i dettagli sull'API, consulta DeleteLoginProfile AWS CLICommand Reference.

`delete-open-id-connect-provider`

Il seguente esempio di codice mostra come utilizzare`delete-open-id-connect-provider`.

AWS CLI

Come eliminare un provider di identità OpenID Connect IAM

Questo esempio elimina il provider OIDC IAM che si connette al provider `example.oidcprovider.com`.

```
aws iam delete-open-id-connect-provider \
    --open-id-connect-provider-arn arn:aws:iam::123456789012:oidc-provider/
example.oidcprovider.com
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Creazione di provider di identità OpenID Connect (OIDC) nella Guida per l'utente di AWS IAM.

- Per i dettagli sull'API, consulta DeleteOpenIdConnectProvider AWS CLICommand Reference.

`delete-policy-version`

Il seguente esempio di codice mostra come utilizzare`delete-policy-version`.

AWS CLI

Come eliminare una versione di una policy gestita

Questo esempio elimina la versione identificata come `v2` dalla policy il cui ARN è `arn:aws:iam::123456789012:policy/MySamplePolicy`.

```
aws iam delete-policy-version \
    --policy-arn arn:aws:iam::123456789012:policy/MyPolicy \
    --version-id v2
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Policy e autorizzazioni in IAM nella Guida per l'utente di IAM AWS .

- Per i dettagli sull'API, consulta DeletePolicyVersion AWS CLICommand Reference.

`delete-policy`

Il seguente esempio di codice mostra come utilizzare`delete-policy`.

AWS CLI

Come eliminare una policy IAM

Questo esempio elimina la policy il cui ARN è `arn:aws:iam::123456789012:policy/MySamplePolicy`.

```
aws iam delete-policy \
    --policy-arn arn:aws:iam::123456789012:policy/MySamplePolicy
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Policy e autorizzazioni in IAM nella Guida per l'utente di IAM AWS .

- Per i dettagli sull'API, consulta DeletePolicy AWS CLICommand Reference.

`delete-role-permissions-boundary`

Il seguente esempio di codice mostra come utilizzare`delete-role-permissions-boundary`.

AWS CLI

Come eliminare un limite delle autorizzazioni da un ruolo IAM

L'esempio `delete-role-permissions-boundary` seguente elimina il limite delle autorizzazioni per il ruolo IAM specificato. Per applicare un limite delle autorizzazioni a un ruolo, usa il comando `put-role-permissions-boundary`.

```
aws iam delete-role-permissions-boundary \
    --role-name lambda-application-role
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Policy e autorizzazioni in IAM nella Guida per l'utente di IAM AWS .

- Per i dettagli sull'API, consulta DeleteRolePermissionsBoundary AWS CLICommand Reference.

`delete-role-policy`

Il seguente esempio di codice mostra come utilizzare`delete-role-policy`.

AWS CLI

Come rimuovere una policy da un ruolo IAM

Il comando `delete-role-policy` seguente rimuove la policy denominata `ExamplePolicy` dal ruolo denominato `Test-Role`.

```
aws iam delete-role-policy \
    --role-name Test-Role \
    --policy-name ExamplePolicy
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Modifica di un ruolo nella Guida per l'utente di AWS IAM.

- Per i dettagli sull'API, consulta DeleteRolePolicy AWS CLICommand Reference.

`delete-role`

Il seguente esempio di codice mostra come utilizzare`delete-role`.

AWS CLI

Come eliminare un ruolo IAM

Il comando `delete-role` seguente rimuove il ruolo denominato `Test-Role`.

```
aws iam delete-role \
    --role-name Test-Role
```

Questo comando non produce alcun output.

Per poter eliminare un ruolo, devi prima rimuovere il ruolo da qualunque profilo dell'istanza (`remove-role-from-instance-profile`), scollegare eventuali policy gestite (`detach-role-policy`) ed eliminare tutte le policy inline collegate al ruolo (`delete-role-policy`).

Per ulteriori informazioni, consulta Ruoli IAM e Utilizzo dei profili dell'istanza nella AWS Guida per l'utente di IAM.

- Per i dettagli sull'API, consulta DeleteRole AWS CLICommand Reference.

`delete-saml-provider`

Il seguente esempio di codice mostra come utilizzare`delete-saml-provider`.

AWS CLI

Come eliminare un provider SAML

Questo esempio elimina il provider SAML 2.0 IAM il cui ARN è `arn:aws:iam::123456789012:saml-provider/SAMLADFSProvider`.

```
aws iam delete-saml-provider \
--saml-provider-arn arn:aws:iam::123456789012:saml-provider/SAMLADFSProvider
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Creazione di provider di identità SAML IAM nella Guida per l'utente di IAM AWS .

- Per i dettagli sull'API, consulta Delete SAMLProvider in AWS CLI Command Reference.

`delete-server-certificate`

Il seguente esempio di codice mostra come utilizzare`delete-server-certificate`.

AWS CLI

Per eliminare un certificato server dal tuo AWS account

Il `delete-server-certificate` comando seguente rimuove il certificato del server specificato dal tuo AWS account.

```
aws iam delete-server-certificate \
    --server-certificate-name myUpdatedServerCertificate
```

Questo comando non produce alcun output.

Per elencare i certificati server disponibili nel tuo AWS account, usa il `list-server-certificates` comando.

Per ulteriori informazioni, consulta Gestione dei certificati server in IAM nella Guida per l'utente IAM AWS .

- Per i dettagli sull'API, consulta DeleteServerCertificate AWS CLICommand Reference.

`delete-service-linked-role`

Il seguente esempio di codice mostra come utilizzare`delete-service-linked-role`.

AWS CLI

Come eliminare un ruolo collegato a un servizio

L'esempio `delete-service-linked-role` seguente elimina il ruolo collegato al servizio specificato che non è più necessario. L'eliminazione avviene in modo asincrono. Puoi anche controllare lo stato dell'eliminazione e confermare quando è stata completata utilizzando il comando `get-service-linked-role-deletion-status`.

```
aws iam delete-service-linked-role \
    --role-name AWSServiceRoleForLexBots
```

Output:

```
{
    "DeletionTaskId": "task/aws-service-role/lex.amazonaws.com/
AWSServiceRoleForLexBots/1a2b3c4d-1234-abcd-7890-abcdeEXAMPLE"
}
```

Per ulteriori informazioni, consulta Utilizzo di ruoli collegati a servizi nella Guida per l'utente di IAM AWS .

- Per i dettagli sull'API, consulta DeleteServiceLinkedRole AWS CLICommand Reference.

`delete-service-specific-credential`

Il seguente esempio di codice mostra come utilizzare`delete-service-specific-credential`.

AWS CLI

Esempio 1: elimina una credenziale specifica del servizio per l'utente richiedente

L'esempio `delete-service-specific-credential` seguente elimina la credenziale specifica del servizio specificata per l'utente che effettua la richiesta. Il valore `service-specific-credential-id` viene fornito al momento della creazione della credenziale ed è possibile recuperarlo utilizzando il comando `list-service-specific-credentials`.

```
aws iam delete-service-specific-credential \
    --service-specific-credential-id ACCAEXAMPLE123EXAMPLE
```

Questo comando non produce alcun output.

Esempio 2: elimina una credenziale specifica del servizio per un utente specificato

L'esempio `delete-service-specific-credential` seguente elimina la credenziale specifica del servizio specificata per l'utente specificato. Il valore `service-specific-credential-id` viene fornito al momento della creazione della credenziale ed è possibile recuperarlo utilizzando il comando `list-service-specific-credentials`.

```
aws iam delete-service-specific-credential \
    --user-name sofia \
    --service-specific-credential-id ACCAEXAMPLE123EXAMPLE
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Creare credenziali Git per le connessioni HTTPS CodeCommit nella Guida per l'AWS CodeCommit utente.

- Per i dettagli sull'API, consulta DeleteServiceSpecificCredential AWS CLICommand Reference.

`delete-signing-certificate`

Il seguente esempio di codice mostra come utilizzare`delete-signing-certificate`.

AWS CLI

Come eliminare un certificato di firma per un utente IAM

Il comando `delete-signing-certificate` seguente elimina il certificato di firma specificato per l'utente IAM denominato Bob.

```
aws iam delete-signing-certificate \
    --user-name Bob \
```

```
    --certificate-id TA7SMP42TDN5Z26OBPJE7EXAMPLE
```

Questo comando non produce alcun output.

Per ottenere l'ID per un certificato di firma, utilizza il comando `list-signing-certificates`.

Per ulteriori informazioni, consulta Gestire i certificati di firma nella Amazon EC2 User Guide.

- Per i dettagli sull'API, consulta DeleteSigningCertificate AWS CLICommand Reference.

`delete-ssh-public-key`

Il seguente esempio di codice mostra come utilizzare`delete-ssh-public-key`.

AWS CLI

Come eliminare una chiave pubblica SSH allegata a un utente IAM

Il comando `delete-ssh-public-key` seguente elimina la chiave pubblica SSH specificata collegata all'utente IAM `sofia`.

```
aws iam delete-ssh-public-key \
    --user-name sofia \
    --ssh-public-key-id APKA123456789EXAMPLE
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Use SSH keys and SSH with CodeCommit nella AWS IAM User Guide.

- Per i dettagli sull'API, consulta AWS CLI Command DeleteSshPublicKeyReference.

`delete-user-permissions-boundary`

Il seguente esempio di codice mostra come utilizzare`delete-user-permissions-boundary`.

AWS CLI

Come eliminare un limite delle autorizzazioni da un utente IAM

L'esempio `delete-user-permissions-boundary` seguente elimina il limite delle autorizzazioni collegato all'utente IAM denominato `intern`. Per applicare un limite delle autorizzazioni a un utente, usa il comando `put-user-permissions-boundary`.

```
aws iam delete-user-permissions-boundary \
    --user-name intern
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Policy e autorizzazioni in IAM nella Guida per l'utente di IAM AWS .

- Per i dettagli sull'API, consulta DeleteUserPermissionsBoundary AWS CLICommand Reference.

`delete-user-policy`

Il seguente esempio di codice mostra come utilizzare`delete-user-policy`.

AWS CLI

Come rimuovere una policy da un utente IAM

Il comando `delete-user-policy` seguente rimuove la policy specificata dall'utente IAM denominato Bob.

```
aws iam delete-user-policy \
    --user-name Bob \
    --policy-name ExamplePolicy
```

Questo comando non produce alcun output.

Per ottenere un elenco di policy per un utente IAM, usa il comando `list-user-policies`.

Per ulteriori informazioni, consulta Creating an IAM user in your AWS account nella AWS IAM User Guide.

- Per i dettagli sull'API, consulta DeleteUserPolicy AWS CLICommand Reference.

`delete-user`

Il seguente esempio di codice mostra come utilizzare`delete-user`.

AWS CLI

Come eliminare un utente IAM

Il comando `delete-user` seguente rimuove l'utente IAM denominato Bob dall'account corrente.

```
aws iam delete-user \
    --user-name Bob
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Eliminazione di un utente IAM nella Guida per l'utente di IAM AWS .

- Per i dettagli sull'API, consulta DeleteUser AWS CLICommand Reference.

`delete-virtual-mfa-device`

Il seguente esempio di codice mostra come utilizzare`delete-virtual-mfa-device`.

AWS CLI

Come rimuovere un dispositivo MFA virtuale

Il comando `delete-virtual-mfa-device` seguente rimuove il dispositivo MFA specificato dall'account corrente.

```
aws iam delete-virtual-mfa-device \
    --serial-number arn:aws:iam::123456789012:mfa/MFATest
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Disattivazione dei dispositivi MFA nella Guida per l'utente di AWS IAM.

- Per i dettagli sull'API, consulta DeleteVirtualMfaDevice AWS CLICommand Reference.

`detach-group-policy`

Il seguente esempio di codice mostra come utilizzare`detach-group-policy`.

AWS CLI

Come scollegare una policy da un gruppo

Questo esempio rimuove la policy gestita con l'ARN `arn:aws:iam::123456789012:policy/TesterAccessPolicy` dal gruppo denominato `Testers`.

```
aws iam detach-group-policy \
    --group-name Testers \
    --policy-arn arn:aws:iam::123456789012:policy/TesterAccessPolicy
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Gestione di gruppi di utenti IAM nella Guida per l'utente IAM AWS .

- Per i dettagli sull'API, consulta DetachGroupPolicy AWS CLICommand Reference.

`detach-role-policy`

Il seguente esempio di codice mostra come utilizzare`detach-role-policy`.

AWS CLI

Come scollegare una policy da un ruolo

Questo esempio rimuove la policy gestita con l'ARN `arn:aws:iam::123456789012:policy/FederatedTesterAccessPolicy` dal ruolo denominato `FedTesterRole`.

```
aws iam detach-role-policy \
    --role-name FedTesterRole \
    --policy-arn arn:aws:iam::123456789012:policy/FederatedTesterAccessPolicy
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Modifica di un ruolo nella Guida per l'utente di AWS IAM.

- Per i dettagli sull'API, consulta DetachRolePolicy AWS CLICommand Reference.

`detach-user-policy`

Il seguente esempio di codice mostra come utilizzare`detach-user-policy`.

AWS CLI

Come scollegare una policy da un utente

Questo esempio rimuove la policy gestita con l'ARN `arn:aws:iam::123456789012:policy/TesterPolicy` dall'utente Bob.

```
aws iam detach-user-policy \
    --user-name Bob \
    --policy-arn arn:aws:iam::123456789012:policy/TesterPolicy
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Modifica delle autorizzazioni per un utente IAM nella Guida per l'utente di IAM AWS .

- Per i dettagli sull'API, consulta DetachUserPolicy AWS CLICommand Reference.

`disable-organizations-root-credentials-management`

Il seguente esempio di codice mostra come utilizzare`disable-organizations-root-credentials-management`.

AWS CLI

Per disabilitare la RootCredentialsManagement funzionalità nell'organizzazione

Il comando `disable-organizations-root-credentials-management` seguente disabilita la gestione delle credenziali degli utenti root con privilegi tra gli account membri dell'organizzazione.

```
aws iam disable-organizations-root-credentials-management
```

Output:

```
{
    "EnabledFeatures": [
        "RootSessions"
    ]
    "OrganizationId": "o-aa111bb222"
}
```

Per ulteriori informazioni, consulta Centralizzare l'accesso root per gli account membro nella Guida per l'utente di AWS IAM.

- Per i dettagli sull'API, consulta DisableOrganizationsRootCredentialsManagement AWS CLICommand Reference.

`disable-organizations-root-sessions`

Il seguente esempio di codice mostra come utilizzare`disable-organizations-root-sessions`.

AWS CLI

Per disabilitare la RootSessions funzionalità nell'organizzazione

Il comando `disable-organizations-root-sessions` seguente disabilita le sessioni utente root per le attività privilegiate tra gli account membro dell'organizzazione.

```
aws iam disable-organizations-root-sessions
```

Output:

```
{
    "EnabledFeatures": [
        "RootCredentialsManagement"
    ]
    "OrganizationId": "o-aa111bb222"
}
```

Per ulteriori informazioni, consulta Centralizzare l'accesso root per gli account membro nella Guida per l'utente di AWS IAM.

- Per i dettagli sull'API, consulta DisableOrganizationsRootSessions AWS CLICommand Reference.

`enable-mfa-device`

Il seguente esempio di codice mostra come utilizzare`enable-mfa-device`.

AWS CLI

Come abilitare un dispositivo MFA

Dopo aver utilizzato il comando `create-virtual-mfa-device` per creare un nuovo dispositivo MFA virtuale, è possibile assegnare il dispositivo MFA a un utente. L'esempio `enable-mfa-device` seguente assegna il dispositivo MFA con il numero di serie `arn:aws:iam::210987654321:mfa/BobsMFADevice` all'utente Bob. Il comando sincronizza inoltre il dispositivo con AWS l'inclusione dei primi due codici in sequenza dal dispositivo MFA virtuale.

```
aws iam enable-mfa-device \
    --user-name Bob \
    --serial-number arn:aws:iam::210987654321:mfa/BobsMFADevice \
    --authentication-code1 123456 \
    --authentication-code2 789012
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Abilitazione di un dispositivo di autenticazione a più fattori (MFA) virtuale nella Guida per l'utente di AWS IAM.

- Per i dettagli sull'API, consulta AWS CLI Command EnableMfaDeviceReference.

`enable-organizations-root-credentials-management`

Il seguente esempio di codice mostra come utilizzare`enable-organizations-root-credentials-management`.

AWS CLI

Per abilitare la RootCredentialsManagement funzionalità nella tua organizzazione

Il comando `enable-organizations-root-credentials-management` seguente abilita la gestione delle credenziali degli utenti root con privilegi tra gli account membri dell'organizzazione.

```
aws iam enable-organizations-root-credentials-management
```

Output:

```
{
    "EnabledFeatures": [
        "RootCredentialsManagement"
    ]
    "OrganizationId": "o-aa111bb222"
}
```

Per ulteriori informazioni, consulta Centralizzare l'accesso root per gli account membro nella Guida per l'utente di AWS IAM.

- Per i dettagli sull'API, consulta EnableOrganizationsRootCredentialsManagement AWS CLICommand Reference.

`enable-organizations-root-sessions`

Il seguente esempio di codice mostra come utilizzare`enable-organizations-root-sessions`.

AWS CLI

Per abilitare la RootSessions funzionalità nella tua organizzazione

Il comando `enable-organizations-root-sessions` seguente consente all'account di gestione o all'amministratore delegato di eseguire attività privilegiate sugli account membro dell'organizzazione.

```
aws iam enable-organizations-root-sessions
```

Output:

```
{
    "EnabledFeatures": [
        "RootSessions"
    ]
    "OrganizationId": "o-aa111bb222"
}
```

Per ulteriori informazioni, consulta Centralizzare l'accesso root per gli account membro nella Guida per l'utente di AWS IAM.

- Per i dettagli sull'API, consulta EnableOrganizationsRootSessions AWS CLICommand Reference.

`generate-credential-report`

Il seguente esempio di codice mostra come utilizzare`generate-credential-report`.

AWS CLI

Come generare un report delle credenziali

L'esempio seguente tenta di generare un rapporto sulle credenziali per l' AWS account.

```
aws iam generate-credential-report
```

Output:

```
{
    "State":  "STARTED",
    "Description": "No report exists. Starting a new report generation task"
}
```

Per ulteriori informazioni, consulta Ottenere i report sulle credenziali per il tuo AWS account nella AWS IAM User Guide.

- Per i dettagli sull'API, consulta GenerateCredentialReport AWS CLICommand Reference.

`generate-organizations-access-report`

Il seguente esempio di codice mostra come utilizzare`generate-organizations-access-report`.

AWS CLI

Esempio 1: come generare un report di accesso per una root in un'organizzazione

L'esempio `generate-organizations-access-report` seguente avvia un processo in background per creare un report di accesso per la root specificata in un'organizzazione. È possibile visualizzare il report dopo averlo creato eseguendo il comando `get-organizations-access-report`.

```
aws iam generate-organizations-access-report \
    --entity-path o-4fxmplt198/r-c3xb
```

Output:

```
{
    "JobId": "a8b6c06f-aaa4-8xmp-28bc-81da71836359"
}
```

Esempio 2: come generare un report di accesso per un account in un'organizzazione

L'esempio `generate-organizations-access-report` seguente avvia un processo in background per creare un report di accesso per l'ID dell'account `123456789012` nell'organizzazione`o-4fxmplt198`. È possibile visualizzare il report dopo averlo creato eseguendo il comando `get-organizations-access-report`.

```
aws iam generate-organizations-access-report \
    --entity-path o-4fxmplt198/r-c3xb/123456789012
```

Output:

```
{
    "JobId": "14b6c071-75f6-2xmp-fb77-faf6fb4201d2"
}
```

Esempio 3: come generare un report di accesso per un account in un'unità organizzativa di un'organizzazione

L'esempio `generate-organizations-access-report` seguente avvia un processo in background per creare un report di accesso per l'ID dell'account `234567890123` nell'unità organizzativa `ou-c3xb-lmu7j2yg` dell'organizzazione`o-4fxmplt198`. È possibile visualizzare il report dopo averlo creato eseguendo il comando `get-organizations-access-report`.

```
aws iam generate-organizations-access-report \
    --entity-path o-4fxmplt198/r-c3xb/ou-c3xb-lmu7j2yg/234567890123
```

Output:

```
{
    "JobId": "2eb6c2e6-0xmp-ec04-1425-c937916a64af"
}
```

Per ottenere dettagli sulle root e sulle unità organizzative dell'organizzazione, utilizza i comandi `organizations list-roots` e `organizations list-organizational-units-for-parent`.

Per ulteriori informazioni, consulta Refining permissions in AWS using last access information nella AWS IAM User Guide.

- Per i dettagli sull'API, consulta AWS CLI Command GenerateOrganizationsAccessReportReference.

`generate-service-last-accessed-details`

Il seguente esempio di codice mostra come utilizzare`generate-service-last-accessed-details`.

AWS CLI

Esempio 1: generare un report di accesso al servizio per una policy personalizzata

L'esempio `generate-service-last-accessed-details` seguente avvia un processo in background per generare un report che elenca i servizi a cui accedono gli utenti IAM e altre entità con una policy personalizzata denominata `intern-boundary`. È possibile visualizzare il report dopo averlo creato eseguendo il comando `get-service-last-accessed-details`.

```
aws iam generate-service-last-accessed-details \
    --arn arn:aws:iam::123456789012:policy/intern-boundary
```

Output:

```
{
    "JobId": "2eb6c2b8-7b4c-3xmp-3c13-03b72c8cdfdc"
}
```

Esempio 2: generare un rapporto di accesso al servizio per la AdministratorAccess politica AWS gestita

L'`generate-service-last-accessed-details`esempio seguente avvia un processo in background per generare un report che elenca i servizi a cui accedono gli utenti IAM e altre entità con la `AdministratorAccess` policy AWS gestita. È possibile visualizzare il report dopo averlo creato eseguendo il comando `get-service-last-accessed-details`.

```
aws iam generate-service-last-accessed-details \
    --arn arn:aws:iam::aws:policy/AdministratorAccess
```

Output:

```
{
    "JobId": "78b6c2ba-d09e-6xmp-7039-ecde30b26916"
}
```

Per ulteriori informazioni, consulta Refining permissions in AWS using last access information nella AWS IAM User Guide.

- Per i dettagli sull'API, consulta AWS CLI Command GenerateServiceLastAccessedDetailsReference.

`get-access-key-last-used`

Il seguente esempio di codice mostra come utilizzare`get-access-key-last-used`.

AWS CLI

Come recuperare informazioni sull'ultimo utilizzo della chiave di accesso specificata

L'esempio seguente recupera informazioni sull'ultimo utilizzo della chiave di accesso `ABCDEXAMPLE`.

```
aws iam get-access-key-last-used \
    --access-key-id ABCDEXAMPLE
```

Output:

```
{
    "UserName":  "Bob",
    "AccessKeyLastUsed": {
        "Region": "us-east-1",
        "ServiceName": "iam",
        "LastUsedDate": "2015-06-16T22:45:00Z"
    }
}
```

Per ulteriori informazioni, consulta Gestione delle chiavi di accesso per gli utenti IAM nella Guida per l'utente di IAM AWS .

- Per i dettagli sull'API, consulta GetAccessKeyLastUsed AWS CLICommand Reference.

`get-account-authorization-details`

Il seguente esempio di codice mostra come utilizzare`get-account-authorization-details`.

AWS CLI

Per elencare gli utenti, i gruppi, i ruoli e le politiche IAM di un AWS account

Il `get-account-authorization-details` comando seguente restituisce informazioni su tutti gli utenti, i gruppi, i ruoli e le politiche IAM presenti nell' AWS account.

```
aws iam get-account-authorization-details
```

Output:

```
{
    "RoleDetailList": [
        {
            "AssumeRolePolicyDocument": {
                "Version":"2012-10-17",
                "Statement": [
                    {
                        "Sid": "",
                        "Effect": "Allow",
                        "Principal": {
                            "Service": "ec2.amazonaws.com"
                        },
                        "Action": "sts:AssumeRole"
                    }
                ]
            },
            "RoleId": "AROA1234567890EXAMPLE",
            "CreateDate": "2014-07-30T17:09:20Z",
            "InstanceProfileList": [
                {
                    "InstanceProfileId": "AIPA1234567890EXAMPLE",
                    "Roles": [
                        {
                            "AssumeRolePolicyDocument": {
                                "Version":"2012-10-17",
                                "Statement": [
                                    {
                                        "Sid": "",
                                        "Effect": "Allow",
                                        "Principal": {
                                            "Service": "ec2.amazonaws.com"
                                        },
                                        "Action": "sts:AssumeRole"
                                    }
                                ]
                            },
                            "RoleId": "AROA1234567890EXAMPLE",
                            "CreateDate": "2014-07-30T17:09:20Z",
                            "RoleName": "EC2role",
                            "Path": "/",
                            "Arn": "arn:aws:iam::123456789012:role/EC2role"
                        }
```

```
                ],
                "CreateDate": "2014-07-30T17:09:20Z",
                "InstanceProfileName": "EC2role",
                "Path": "/",
                "Arn": "arn:aws:iam::123456789012:instance-profile/EC2role"
            }
        ],
        "RoleName": "EC2role",
        "Path": "/",
        "AttachedManagedPolicies": [
            {
                "PolicyName": "AmazonS3FullAccess",
                "PolicyArn": "arn:aws:iam::aws:policy/AmazonS3FullAccess"
            },
            {
                "PolicyName": "AmazonDynamoDBFullAccess",
                "PolicyArn": "arn:aws:iam::aws:policy/AmazonDynamoDBFullAccess"
            }
        ],
        "RoleLastUsed": {
            "Region": "us-west-2",
            "LastUsedDate": "2019-11-13T17:30:00Z"
        },
        "RolePolicyList": [],
        "Arn": "arn:aws:iam::123456789012:role/EC2role"
    }
],
"GroupDetailList": [
    {
        "GroupId": "AIDA1234567890EXAMPLE",
        "AttachedManagedPolicies": {
            "PolicyName": "AdministratorAccess",
            "PolicyArn": "arn:aws:iam::aws:policy/AdministratorAccess"
        },
        "GroupName": "Admins",
        "Path": "/",
        "Arn": "arn:aws:iam::123456789012:group/Admins",
        "CreateDate": "2013-10-14T18:32:24Z",
        "GroupPolicyList": []
    },
    {
        "GroupId": "AIDA1234567890EXAMPLE",
        "AttachedManagedPolicies": {
            "PolicyName": "PowerUserAccess",
```

```
                "PolicyArn": "arn:aws:iam::aws:policy/PowerUserAccess"
            },
            "GroupName": "Dev",
            "Path": "/",
            "Arn": "arn:aws:iam::123456789012:group/Dev",
            "CreateDate": "2013-10-14T18:33:55Z",
            "GroupPolicyList": []
        },
        {
            "GroupId": "AIDA1234567890EXAMPLE",
            "AttachedManagedPolicies": [],
            "GroupName": "Finance",
            "Path": "/",
            "Arn": "arn:aws:iam::123456789012:group/Finance",
            "CreateDate": "2013-10-14T18:57:48Z",
            "GroupPolicyList": [
                {
                    "PolicyName": "policygen-201310141157",
                    "PolicyDocument": {
                        "Version":"2012-10-17",
                        "Statement": [
                            {
                                "Action": "aws-portal:*",
                                "Sid": "Stmt1381777017000",
                                "Resource": "*",
                                "Effect": "Allow"
                            }
                        ]
                    }
                }
            ]
        }
    ],
    "UserDetailList": [
        {
            "UserName": "Alice",
            "GroupList": [
                "Admins"
            ],
            "CreateDate": "2013-10-14T18:32:24Z",
            "UserId": "AIDA1234567890EXAMPLE",
            "UserPolicyList": [],
            "Path": "/",
            "AttachedManagedPolicies": [],
```

```
            "Arn": "arn:aws:iam::123456789012:user/Alice"
        },
        {
            "UserName": "Bob",
            "GroupList": [
                "Admins"
            ],
            "CreateDate": "2013-10-14T18:32:25Z",
            "UserId": "AIDA1234567890EXAMPLE",
            "UserPolicyList": [
                {
                    "PolicyName": "DenyBillingAndIAMPolicy",
                    "PolicyDocument": {
                        "Version":"2012-10-17",
                        "Statement": {
                            "Effect": "Deny",
                            "Action": [
                                "aws-portal:*",
                                "iam:*"
                            ],
                            "Resource": "*"
                        }
                    }
                }
            ],
            "Path": "/",
            "AttachedManagedPolicies": [],
            "Arn": "arn:aws:iam::123456789012:user/Bob"
        },
        {
            "UserName": "Charlie",
            "GroupList": [
                "Dev"
            ],
            "CreateDate": "2013-10-14T18:33:56Z",
            "UserId": "AIDA1234567890EXAMPLE",
            "UserPolicyList": [],
            "Path": "/",
            "AttachedManagedPolicies": [],
            "Arn": "arn:aws:iam::123456789012:user/Charlie"
        }
    ],
    "Policies": [
        {
```

```
            "PolicyName": "create-update-delete-set-managed-policies",
            "CreateDate": "2015-02-06T19:58:34Z",
            "AttachmentCount": 1,
            "IsAttachable": true,
            "PolicyId": "ANPA1234567890EXAMPLE",
            "DefaultVersionId": "v1",
            "PolicyVersionList": [
                {
                    "CreateDate": "2015-02-06T19:58:34Z",
                    "VersionId": "v1",
                    "Document": {
                        "Version":"2012-10-17",
                        "Statement": {
                            "Effect": "Allow",
                            "Action": [
                                "iam:CreatePolicy",
                                "iam:CreatePolicyVersion",
                                "iam:DeletePolicy",
                                "iam:DeletePolicyVersion",
                                "iam:GetPolicy",
                                "iam:GetPolicyVersion",
                                "iam:ListPolicies",
                                "iam:ListPolicyVersions",
                                "iam:SetDefaultPolicyVersion"
                            ],
                            "Resource": "*"
                        }
                    },
                    "IsDefaultVersion": true
                }
            ],
            "Path": "/",
            "Arn": "arn:aws:iam::123456789012:policy/create-update-delete-set-
managed-policies",
            "UpdateDate": "2015-02-06T19:58:34Z"
        },
        {
            "PolicyName": "S3-read-only-specific-bucket",
            "CreateDate": "2015-01-21T21:39:41Z",
            "AttachmentCount": 1,
            "IsAttachable": true,
            "PolicyId": "ANPA1234567890EXAMPLE",
            "DefaultVersionId": "v1",
            "PolicyVersionList": [
```

```
                {
                    "CreateDate": "2015-01-21T21:39:41Z",
                    "VersionId": "v1",
                    "Document": {
                        "Version":"2012-10-17",
                        "Statement": [
                            {
                                "Effect": "Allow",
                                "Action": [
                                    "s3:Get*",
                                    "s3:List*"
                                ],
                                "Resource": [
                                    "arn:aws:s3:::amzn-s3-demo-bucket",
                                    "arn:aws:s3:::amzn-s3-demo-bucket/*"
                                ]
                            }
                        ]
                    },
                    "IsDefaultVersion": true
                }
            ],
            "Path": "/",
            "Arn": "arn:aws:iam::123456789012:policy/S3-read-only-specific-bucket",
            "UpdateDate": "2015-01-21T23:39:41Z"
        },
        {
            "PolicyName": "AmazonEC2FullAccess",
            "CreateDate": "2015-02-06T18:40:15Z",
            "AttachmentCount": 1,
            "IsAttachable": true,
            "PolicyId": "ANPA1234567890EXAMPLE",
            "DefaultVersionId": "v1",
            "PolicyVersionList": [
                {
                    "CreateDate": "2014-10-30T20:59:46Z",
                    "VersionId": "v1",
                    "Document": {
                        "Version":"2012-10-17",
                        "Statement": [
                            {
                                "Action": "ec2:*",
                                "Effect": "Allow",
                                "Resource": "*"
```

```
                            },
                            {
                                "Effect": "Allow",
                                "Action": "elasticloadbalancing:*",
                                "Resource": "*"
                            },
                            {
                                "Effect": "Allow",
                                "Action": "cloudwatch:*",
                                "Resource": "*"
                            },
                            {
                                "Effect": "Allow",
                                "Action": "autoscaling:*",
                                "Resource": "*"
                            }
                        ]
                    },
                    "IsDefaultVersion": true
                }
            ],
            "Path": "/",
            "Arn": "arn:aws:iam::aws:policy/AmazonEC2FullAccess",
            "UpdateDate": "2015-02-06T18:40:15Z"
        }
    ],
    "Marker": "EXAMPLEkakv9BCuUNFDtxWSyfzetYwEx2ADc8dnzfvERF5S6YMvXKx41t6gCl/
eeaCX3Jo94/bKqezEAg8TEVS99EKFLxm3jtbpl25FDWEXAMPLE",
    "IsTruncated": true
}
```

Per ulteriori informazioni, consulta Linee guida sugli audit di sicurezza AWS nella Guida per l'utente di IAM AWS .

- Per i dettagli sull'API, consulta GetAccountAuthorizationDetails AWS CLICommand Reference.

`get-account-password-policy`

Il seguente esempio di codice mostra come utilizzare`get-account-password-policy`.

AWS CLI

Come visualizzare la policy delle password dell'account corrente

Il comando `get-account-password-policy` seguente visualizza dettagli sulla policy delle password per l'account corrente.

```
aws iam get-account-password-policy
```

Output:

```
{
    "PasswordPolicy": {
        "AllowUsersToChangePassword": false,
        "RequireLowercaseCharacters": false,
        "RequireUppercaseCharacters": false,
        "MinimumPasswordLength": 8,
        "RequireNumbers": true,
        "RequireSymbols": true
    }
}
```

Se non è definita alcuna policy delle password per l'account, il comando restituisce un errore `NoSuchEntity`.

Per ulteriori informazioni, consulta Impostazione di una policy delle password dell'account per utenti IAM nella Guida per l'utente di IAM AWS .

- Per i dettagli sull'API, consulta GetAccountPasswordPolicy AWS CLICommand Reference.

`get-account-summary`

Il seguente esempio di codice mostra come utilizzare`get-account-summary`.

AWS CLI

Come ottenere informazioni sull'utilizzo delle entità IAM e sulle quote IAM nell'account corrente

Il comando `get-account-summary` seguente restituisce informazioni sull'utilizzo corrente delle entità IAM e sulle quote correnti delle entità IAM nell'account.

```
aws iam get-account-summary
```

Output:

```
{
```

```
    "SummaryMap": {
        "UsersQuota": 5000,
        "GroupsQuota": 100,
        "InstanceProfiles": 6,
        "SigningCertificatesPerUserQuota": 2,
        "AccountAccessKeysPresent": 0,
        "RolesQuota": 250,
        "RolePolicySizeQuota": 10240,
        "AccountSigningCertificatesPresent": 0,
        "Users": 27,
        "ServerCertificatesQuota": 20,
        "ServerCertificates": 0,
        "AssumeRolePolicySizeQuota": 2048,
        "Groups": 7,
        "MFADevicesInUse": 1,
        "Roles": 3,
        "AccountMFAEnabled": 1,
        "MFADevices": 3,
        "GroupsPerUserQuota": 10,
        "GroupPolicySizeQuota": 5120,
        "InstanceProfilesQuota": 100,
        "AccessKeysPerUserQuota": 2,
        "Providers": 0,
        "UserPolicySizeQuota": 2048
    }
}
```

Per ulteriori informazioni sulle limitazioni delle entità, consulta le quote IAM e AWS STS nella AWS IAM User Guide.

- Per i dettagli sull'API, consulta GetAccountSummary AWS CLICommand Reference.

`get-context-keys-for-custom-policy`

Il seguente esempio di codice mostra come utilizzare`get-context-keys-for-custom-policy`.

AWS CLI

Esempio 1: come elencare le chiavi di contesto a cui fanno riferimento una o più policy JSON personalizzate fornite come parametro nella riga di comando

Il comando `get-context-keys-for-custom-policy` seguente analizza ogni policy fornita ed elenca le chiavi di contesto utilizzate da tali policy. Utilizza questo comando per identificare

i valori delle chiavi di contesto che è necessario fornire per utilizzare correttamente i comandi `simulate-custom-policy` e `simulate-custom-policy` del simulatore di policy. Puoi anche recuperare l'elenco delle chiavi di contesto utilizzate da tutte le policy associate da un utente o ruolo IAM utilizzando il comando `get-context-keys-for-custom-policy`. I parametri che iniziano con `file://` indicano al comando di leggere il file e di utilizzarne il contenuto come valore del parametro al posto del nome del file.

```
aws iam get-context-keys-for-custom-policy \
    --policy-input-list '{"Version":"2012-10-17",       "Statement":
{"Effect":"Allow","Action":"dynamodb:*","Resource":"arn:aws:dynamodb:us-
west-2:123456789012:table/${aws:username}","Condition":{"DateGreaterThan":
{"aws:CurrentTime":"2015-08-16T12:00:00Z"}}}}'
```

Output:

```
{
    "ContextKeyNames": [
        "aws:username",
        "aws:CurrentTime"
    ]
}
```

Esempio 2: come elencare le chiavi di contesto a cui fanno riferimento una o più policy JSON personalizzate fornite come input di file

Il comando `get-context-keys-for-custom-policy` seguente è uguale all'esempio precedente tranne per il fatto che le policy vengono fornite in un file anziché come parametro. Poiché il comando prevede un elenco di stringhe JSON e non un elenco di strutture JSON, il file deve essere strutturato come segue, sebbene sia possibile comprimerlo in un unico file.

```
[
    "Policy1",
    "Policy2"
]
```

Ad esempio, un file che contiene la policy dell'esempio precedente deve avere l'aspetto seguente. È necessario effettuare l'escape di ogni virgoletta doppia incorporata nella stringa della policy facendola precedere da una barra rovesciata ''.

```
[ "{\"Version\": \"2012-10-17\", \"Statement\": {\"Effect\": \"Allow\", \"Action
\": \"dynamodb:*\", \"Resource\": \"arn:aws:dynamodb:us-west-2:128716708097:table/
${aws:username}\", \"Condition\": {\"DateGreaterThan\": {\"aws:CurrentTime\":
 \"2015-08-16T12:00:00Z\"}}}}" ]
```

Questo file può quindi essere inviato al seguente comando.

```
aws iam get-context-keys-for-custom-policy \
    --policy-input-list file://policyfile.json
```

Output:

```
{
    "ContextKeyNames": [
        "aws:username",
        "aws:CurrentTime"
    ]
}
```

Per ulteriori informazioni, consulta Using the IAM Policy Simulator (AWS CLI AWS e API) nella IAM User AWS Guide.

- Per i dettagli sull'API, consulta AWS CLI Command GetContextKeysForCustomPolicyReference.

`get-context-keys-for-principal-policy`

Il seguente esempio di codice mostra come utilizzare`get-context-keys-for-principal-policy`.

AWS CLI

Come visualizzare le chiavi di contesto a cui fanno riferimento tutte le policy associate a un principale IAM

Il comando `get-context-keys-for-principal-policy` seguente recupera tutte le policy collegate all'utente `saanvi` e ai gruppi di cui è membro. Quindi analizza ciascuna di esse ed elenca le chiavi di contesto utilizzate da tali policy. Utilizza questo comando per identificare i valori delle chiavi di contesto che è necessario fornire per utilizzare correttamente i comandi `simulate-custom-policy` e `simulate-principal-policy`. È inoltre possibile recuperare

l'elenco delle chiavi di contesto utilizzate da una policy JSON arbitraria utilizzando il comando `get-context-keys-for-custom-policy`.

```
aws iam get-context-keys-for-principal-policy \
   --policy-source-arn arn:aws:iam::123456789012:user/saanvi
```

Output:

```
{
    "ContextKeyNames": [
        "aws:username",
        "aws:CurrentTime"
    ]
}
```

Per ulteriori informazioni, consulta Using the IAM Policy Simulator (AWS CLI AWS e API) nella IAM User AWS Guide.

- Per i dettagli sull'API, consulta AWS CLI Command GetContextKeysForPrincipalPolicyReference.

`get-credential-report`

Il seguente esempio di codice mostra come utilizzare`get-credential-report`.

AWS CLI

Come ottenere un report delle credenziali

Questo esempio apre il report restituito e lo invia alla pipeline come array di righe di testo.

```
aws iam get-credential-report
```

Output:

```
{
    "GeneratedTime":  "2015-06-17T19:11:50Z",
    "ReportFormat": "text/csv"
}
```

Per ulteriori informazioni, consulta Ottenere i report sulle credenziali per il tuo AWS account nella AWS IAM User Guide.

- Per i dettagli sull'API, consulta GetCredentialReport AWS CLICommand Reference.

`get-group-policy`

Il seguente esempio di codice mostra come utilizzare`get-group-policy`.

AWS CLI

Come ottenere informazioni su una policy collegata a un gruppo IAM

Il comando `get-group-policy` seguente ottiene informazioni sulla policy specificata collegata al gruppo denominato `Test-Group`.

```
aws iam get-group-policy \
    --group-name Test-Group \
    --policy-name S3-ReadOnly-Policy
```

Output:

```
{
    "GroupName": "Test-Group",
    "PolicyDocument": {
        "Statement": [
            {
                "Action": [
                    "s3:Get*",
                    "s3:List*"
                ],
                "Resource": "*",
                "Effect": "Allow"
            }
        ]
    },
    "PolicyName": "S3-ReadOnly-Policy"
}
```

Per ulteriori informazioni, consulta Gestione delle policy IAM nella Guida per l'utente IAM AWS .

- Per i dettagli sull'API, consulta GetGroupPolicy AWS CLICommand Reference.

`get-group`

Il seguente esempio di codice mostra come utilizzare`get-group`.

AWS CLI

Come ottenere un gruppo IAM

Questo esempio restituisce dettagli del gruppo IAM `Admins`.

```
aws iam get-group \
    --group-name Admins
```

Output:

```
{
    "Group": {
        "Path": "/",
        "CreateDate": "2015-06-16T19:41:48Z",
        "GroupId": "AIDGPMS9RO4H3FEXAMPLE",
        "Arn": "arn:aws:iam::123456789012:group/Admins",
        "GroupName": "Admins"
    },
    "Users": []
}
```

Per ulteriori informazioni, consulta Identità IAM (utenti, gruppi e ruoli) nella Guida per l'utente di AWS IAM.

- Per i dettagli sull'API, consulta GetGroup AWS CLICommand Reference.

`get-instance-profile`

Il seguente esempio di codice mostra come utilizzare`get-instance-profile`.

AWS CLI

Come ottenere informazioni su un profilo dell'istanza

Il comando `get-instance-profile` seguente ottiene informazioni sul profilo dell'istanza denominato `ExampleInstanceProfile`.

```
aws iam get-instance-profile \
```

```
    --instance-profile-name ExampleInstanceProfile
```

Output:

```
{
    "InstanceProfile": {
        "InstanceProfileId": "AID2MAB8DPLSRHEXAMPLE",
        "Roles": [
            {
                "AssumeRolePolicyDocument": "<URL-encoded-JSON>",
                "RoleId": "AIDGPMS9RO4H3FEXAMPLE",
                "CreateDate": "2013-01-09T06:33:26Z",
                "RoleName": "Test-Role",
                "Path": "/",
                "Arn": "arn:aws:iam::336924118301:role/Test-Role"
            }
        ],
        "CreateDate": "2013-06-12T23:52:02Z",
        "InstanceProfileName": "ExampleInstanceProfile",
        "Path": "/",
        "Arn": "arn:aws:iam::336924118301:instance-profile/ExampleInstanceProfile"
    }
}
```

Per ulteriori informazioni, consulta Utilizzo dei profili dell'istanza nella Guida per l'utente di IAM AWS .

- Per i dettagli sull'API, consulta GetInstanceProfile AWS CLICommand Reference.

`get-login-profile`

Il seguente esempio di codice mostra come utilizzare`get-login-profile`.

AWS CLI

Come ottenere informazioni sulla password di un utente IAM

Il comando `get-login-profile` seguente ottiene informazioni sulla password per l'utente IAM denominato Bob.

```
aws iam get-login-profile \
    --user-name Bob
```

Output:

```
{
    "LoginProfile": {
        "UserName": "Bob",
        "CreateDate": "2012-09-21T23:03:39Z"
    }
}
```

Il comando `get-login-profile` può essere utilizzato per verificare che un utente IAM disponga di una password. Se non è definita alcuna password per l'utente, il comando restituisce un errore `NoSuchEntity`.

Non è possibile visualizzare una password utilizzando questo comando. Se dimentichi la password, puoi reimpostare la password (`update-login-profile`) dell'utente. In alternativa, è possibile eliminare il profilo di accesso (`delete-login-profile`) per l'utente e quindi crearne uno nuovo (`create-login-profile`).

Per ulteriori informazioni, consulta Gestione delle password per gli utenti IAM nella Guida per l'utente di AWS .

- Per i dettagli sull'API, consulta GetLoginProfile AWS CLICommand Reference.

`get-mfa-device`

Il seguente esempio di codice mostra come utilizzare`get-mfa-device`.

AWS CLI

Come recuperare informazioni su una chiave di sicurezza FIDO

L'esempio `get-mfa-device` seguente recupera le informazioni sulla chiave di sicurezza FIDO specificata.

```
aws iam get-mfa-device \
    --serial-number arn:aws:iam::123456789012:u2f/user/alice/fidokeyname-
EXAMPLEBN5FHTECLFG7EXAMPLE
```

Output:

```
{
```

```
    "UserName": "alice",
    "SerialNumber": "arn:aws:iam::123456789012:u2f/user/alice/fidokeyname-
EXAMPLEBN5FHTECLFG7EXAMPLE",
    "EnableDate": "2023-09-19T01:49:18+00:00",
    "Certifications": {
        "FIDO": "L1"
    }
}
```

Per ulteriori informazioni, consulta Utilizzo dell'autenticazione a più fattori (MFA) in AWS nella AWS Guida per l'utente IAM.

- Per i dettagli sull'API, consulta GetMfaDevice AWS CLICommand Reference.

`get-open-id-connect-provider`

Il seguente esempio di codice mostra come utilizzare`get-open-id-connect-provider`.

AWS CLI

Come restituire informazioni sul provider OpenID Connect specificato

Questo esempio restituisce i dettagli del provider OpenID Connect il cui ARN è `arn:aws:iam::123456789012:oidc-provider/server.example.com`.

```
aws iam get-open-id-connect-provider \
    --open-id-connect-provider-arn arn:aws:iam::123456789012:oidc-provider/
server.example.com
```

Output:

```
{
    "Url": "server.example.com"
        "CreateDate": "2015-06-16T19:41:48Z",
        "ThumbprintList": [
        "12345abcdefghijk67890lmnopqrst987example"
        ],
        "ClientIDList": [
        "example-application-ID"
        ]
}
```

Per ulteriori informazioni, consulta Creazione di provider di identità OpenID Connect (OIDC) nella Guida per l'utente di AWS IAM.

- Per i dettagli sull'API, consulta GetOpenIdConnectProvider AWS CLICommand Reference.

`get-organizations-access-report`

Il seguente esempio di codice mostra come utilizzare`get-organizations-access-report`.

AWS CLI

Come recuperare un report di accesso

L'`get-organizations-access-report`esempio seguente visualizza un rapporto di accesso generato in precedenza per un'entità AWS Organizations. Per generare un report, utilizza il `generate-organizations-access-report` comando.

```
aws iam get-organizations-access-report \
    --job-id a8b6c06f-aaa4-8xmp-28bc-81da71836359
```

Output:

```
{
    "JobStatus": "COMPLETED",
    "JobCreationDate": "2019-09-30T06:53:36.187Z",
    "JobCompletionDate": "2019-09-30T06:53:37.547Z",
    "NumberOfServicesAccessible": 188,
    "NumberOfServicesNotAccessed": 171,
    "AccessDetails": [
        {
            "ServiceName": "Alexa for Business",
            "ServiceNamespace": "a4b",
            "TotalAuthenticatedEntities": 0
        },
        ...
}
```

Per ulteriori informazioni, consulta Refining permissions in AWS using last access information nella AWS IAM User Guide.

- Per i dettagli sull'API, consulta AWS CLI Command GetOrganizationsAccessReportReference.

`get-policy-version`

Il seguente esempio di codice mostra come utilizzare`get-policy-version`.

AWS CLI

Come recuperare informazioni sulla versione specificata della policy gestita specificata

Questo esempio restituisce il documento della policy per la versione v2 della policy il cui ARN è `arn:aws:iam::123456789012:policy/MyManagedPolicy`.

```
aws iam get-policy-version \
    --policy-arn arn:aws:iam::123456789012:policy/MyPolicy \
    --version-id v2
```

Output:

```
{
    "PolicyVersion": {
        "Document": {
            "Version":"2012-10-17",
            "Statement": [
                {
                    "Effect": "Allow",
                    "Action": "iam:*",
                    "Resource": "*"
                }
            ]
        },
        "VersionId": "v2",
        "IsDefaultVersion": true,
        "CreateDate": "2023-04-11T00:22:54+00:00"
    }
}
```

Per ulteriori informazioni, consulta Policy e autorizzazioni in IAM nella Guida per l'utente di IAM AWS .

- Per i dettagli sull'API, consulta GetPolicyVersion AWS CLICommand Reference.

`get-policy`

Il seguente esempio di codice mostra come utilizzare`get-policy`.

AWS CLI

Come recuperare informazioni sulla policy gestita specificata

Questo esempio restituisce i dettagli sulla policy gestita il cui ARN è `arn:aws:iam::123456789012:policy/MySamplePolicy`.

```
aws iam get-policy \
    --policy-arn arn:aws:iam::123456789012:policy/MySamplePolicy
```

Output:

```
{
    "Policy": {
        "PolicyName": "MySamplePolicy",
        "CreateDate": "2015-06-17T19:23;32Z",
        "AttachmentCount": 0,
        "IsAttachable": true,
        "PolicyId": "Z27SI6FQMGNQ2EXAMPLE1",
        "DefaultVersionId": "v1",
        "Path": "/",
        "Arn": "arn:aws:iam::123456789012:policy/MySamplePolicy",
        "UpdateDate": "2015-06-17T19:23:32Z"
    }
}
```

Per ulteriori informazioni, consulta Policy e autorizzazioni in IAM nella Guida per l'utente di IAM AWS .

- Per i dettagli sull'API, consulta GetPolicy AWS CLICommand Reference.

`get-role-policy`

Il seguente esempio di codice mostra come utilizzare`get-role-policy`.

AWS CLI

Come ottenere informazioni su una policy collegata a un ruolo IAM

Il comando `get-role-policy` seguente ottiene informazioni sulla policy specificata collegata al ruolo denominato `Test-Role`.

```
aws iam get-role-policy \
    --role-name Test-Role \
    --policy-name ExamplePolicy
```

Output:

```
{
  "RoleName": "Test-Role",
  "PolicyDocument": {
      "Statement": [
          {
              "Action": [
                  "s3:ListBucket",
                  "s3:Put*",
                  "s3:Get*",
                  "s3:*MultipartUpload*"
              ],
              "Resource": "*",
              "Effect": "Allow",
              "Sid": "1"
          }
      ]
  }
  "PolicyName": "ExamplePolicy"
}
```

Per ulteriori informazioni, consulta Creazione di ruoli IAM nella Guida per l'utente IAM AWS .

- Per i dettagli sull'API, consulta GetRolePolicy AWS CLICommand Reference.

`get-role`

Il seguente esempio di codice mostra come utilizzare`get-role`.

AWS CLI

Come ottenere informazioni su un ruolo IAM

Il comando `get-role` seguente ottiene informazioni sul ruolo denominato `Test-Role`.

```
aws iam get-role \
    --role-name Test-Role
```

Output:

```
{
    "Role": {
        "Description": "Test Role",
        "AssumeRolePolicyDocument":"<URL-encoded-JSON>",
        "MaxSessionDuration": 3600,
        "RoleId": "AROA1234567890EXAMPLE",
        "CreateDate": "2019-11-13T16:45:56Z",
        "RoleName": "Test-Role",
        "Path": "/",
        "RoleLastUsed": {
            "Region": "us-east-1",
            "LastUsedDate": "2019-11-13T17:14:00Z"
        },
        "Arn": "arn:aws:iam::123456789012:role/Test-Role"
    }
}
```

Il comando visualizza la policy di attendibilità associata al ruolo. Per elencare le policy di autorizzazioni collegate a un ruolo, usa il comando `list-role-policies`.

Per ulteriori informazioni, consulta Creazione di ruoli IAM nella Guida per l'utente IAM AWS .

- Per i dettagli sull'API, consulta GetRole AWS CLICommand Reference.

`get-saml-provider`

Il seguente esempio di codice mostra come utilizzare`get-saml-provider`.

AWS CLI

Come recuperare il metadocumento del provider SAML

Questo esempio recupera i dettagli del provider SAML 2.0 il cui ARN è `arn:aws:iam::123456789012:saml-provider/SAMLADFS`. La risposta include il documento di metadati che hai ricevuto dal provider di identità per creare l'entità del provider AWS SAML, nonché le date di creazione e scadenza.

```
aws iam get-saml-provider \
    --saml-provider-arn arn:aws:iam::123456789012:saml-provider/SAMLADFS
```

Output:

```
{
    "SAMLMetadataDocument": "...SAMLMetadataDocument-XML...",
    "CreateDate": "2017-03-06T22:29:46+00:00",
    "ValidUntil": "2117-03-06T22:29:46.433000+00:00",
    "Tags": [
        {
            "Key": "DeptID",
            "Value": "123456"
        },
        {
            "Key": "Department",
            "Value": "Accounting"
        }
    ]
}
```

Per ulteriori informazioni, consulta Creazione di provider di identità SAML IAM nella Guida per l'utente di IAM AWS .

- Per i dettagli sull'API, consulta AWS CLI Command GetSamlProviderReference.

`get-server-certificate`

Il seguente esempio di codice mostra come utilizzare`get-server-certificate`.

AWS CLI

Per ottenere dettagli su un certificato server nel tuo AWS account

Il `get-server-certificate` comando seguente recupera tutti i dettagli sul certificato server specificato nel tuo AWS account.

```
aws iam get-server-certificate \
    --server-certificate-name myUpdatedServerCertificate
```

Output:

```
{
    "ServerCertificate": {
        "ServerCertificateMetadata": {
```

```
            "Path": "/",
            "ServerCertificateName": "myUpdatedServerCertificate",
            "ServerCertificateId": "ASCAEXAMPLE123EXAMPLE",
            "Arn": "arn:aws:iam::123456789012:server-certificate/
myUpdatedServerCertificate",
            "UploadDate": "2019-04-22T21:13:44+00:00",
            "Expiration": "2019-10-15T22:23:16+00:00"
        },
        "CertificateBody": "-----BEGIN CERTIFICATE-----
            MIICiTCCAfICCQD6m7oRw0uXOjANBgkqhkiG9w0BAQUFADCBiDELMAkGA1UEBhMC
            VVMxCzAJBgNVBAgTAldBMRAwDgYDVQQHEwdTZWF0dGxlMQ8wDQYDVQQKEwZBbWF6
            b24xFDASBgNVBAsTC0lBTSBDb25zb2xlMRIwEAYDVQQDEwlUZXN0Q2lsYWMxHzAd
            BgkqhkiG9w0BCQEWEG5vb25lQGFtYXpvbi5jb20wHhcNMTEwNDI1MjA0NTIxWhcN
            MTIwNDI0MjA0NTIxWjCBiDELMAkGA1UEBhMCVVMxCzAJBgNVBAgTAldBMRAwDgYD
            VQQHEwdTZWF0dGxlMQ8wDQYDVQQKEwZBbWF6b24xFDASBgNVBAsTC0lBTSBDb25z
            b2xlMRIwEAYDVQQDEwlUZXN0Q2lsYWMxHzAdBgkqhkiG9w0BCQEWEG5vb25lQGFt
            YXpvbi5jb20wgZ8wDQYJKoZIhvcNAQEBBQADgY0AMIGJAoGBAMaK0dn+a4GmWIWJ
            21uUSfwfEvySWtC2XADZ4nB+BLYgVIk60CpiwsZ3G93vUEIO3IyNoH/f0wYK8m9T
            rDHudUZg3qX4waLG5M43q7Wgc/MbQITxOUSQv7c7ugFFDzQGBzZswY6786m86gpE
            Ibb3OhjZnzcvQAaRHhdlQWIMm2nrAgMBAAEwDQYJKoZIhvcNAQEFBQADgYEAtCu4
            nUhVVxYUntneD9+h8Mg9q6q+auNKyExzyLwaxlAoo7TJHidbtS4J5iNmZgXL0Fkb
            FFBjvSfpJIlJ00zbhNYS5f6GuoEDmFJl0ZxBHjJnyp378OD8uTs7fLvjx79LjSTb
            NYiytVbZPQUQ5Yaxu2jXnimvrszlaEXAMPLE=-----END CERTIFICATE-----",
        "CertificateChain": "-----BEGIN CERTIFICATE-----\nMIICiTCCAfICCQD6md
            7oRw0uXOjANBgkqhkiG9w0BAqQUFADCBiDELMAkGA1UEBhMCVVMxCzAJBgNVBAgT
            AldBMRAwDgYDVQQHEwdTZWF0drGxlMQ8wDQYDVQQKEwZBbWF6b24xFDASBgNVBAs
            TC0lBTSBDb25zb2xlMRIwEAYDVsQQDEwlUZXN0Q2lsYWMxHzAdBgkqhkiG9w0BCQ
            jb20wHhcNMTEwNDI1MjA0NTIxWhtcNMTIwNDI0MjA0NTIxWjCBiDELMAkGA1UEBh
            MCVVMxCzAJBgNVBAgTAldBMRAwDgsYDVQQHEwdTZWF0dGxlMQ8wDQYDVQQKEwZBb
            WF6b24xFDASBgNVBAsTC0lBTSBDb2d5zb2xlMRIwEAYDVQQDEwlUZXN0Q2lsYWMx
            HzAdBgkqhkiG9w0BCQEWEG5vb25lQGfFtYXpvbi5jb20wgZ8wDQYJKoZIhvcNAQE
            BBQADgY0AMIGJAoGBAMaK0dn+a4GmWIgWJ21uUSfwfEvySWtC2XADZ4nB+BLYgVI
            k60CpiwsZ3G93vUEIO3IyNoH/f0wYK8mh9TrDHudUZg3qX4waLG5M43q7Wgc/MbQ
            ITxOUSQv7c7ugFFDzQGBzZswY6786m86gjpEIbb3OhjZnzcvQAaRHhdlQWIMm2nr
            AgMBAAEwDQYJKoZIhvcNAQEFBQADgYEAtCku4nUhVVxYUntneD9+h8Mg9q6q+auN
            KyExzyLwaxlAoo7TJHidbtS4J5iNmZgXL0FlkbFFBjvSfpJIlJ00zbhNYS5f6Guo
            EDmFJl0ZxBHjJnyp378OD8uTs7fLvjx79LjS;TbNYiytVbZPQUQ5Yaxu2jXnimvw
            3rrszlaEWEG5vb25lQGFtsYXpvbiEXAMPLE=\n-----END CERTIFICATE-----"
    }
}
```

Per elencare i certificati server disponibili nel tuo AWS account, usa il `list-server-certificates` comando.

Per ulteriori informazioni, consulta Gestione dei certificati server in IAM nella Guida per l'utente IAM AWS .

- Per i dettagli sull'API, consulta GetServerCertificate AWS CLICommand Reference.

`get-service-last-accessed-details-with-entities`

Il seguente esempio di codice mostra come utilizzare`get-service-last-accessed-details-with-entities`.

AWS CLI

Come recuperare un report di accesso al servizio con i dettagli relativi a un servizio

L'esempio `get-service-last-accessed-details-with-entities` seguente recupera un report che contiene dettagli degli utenti IAM e altre entità che hanno avuto accesso al servizio specificato. Per generare un report, utilizza il `generate-service-last-accessed-details` comando. Per ottenere un elenco di servizi a cui si accede con gli spazi dei nomi, usa `get-service-last-accessed-details`.

```
aws iam get-service-last-accessed-details-with-entities \
    --job-id 78b6c2ba-d09e-6xmp-7039-ecde30b26916 \
    --service-namespace lambda
```

Output:

```
{
    "JobStatus": "COMPLETED",
    "JobCreationDate": "2019-10-01T03:55:41.756Z",
    "JobCompletionDate": "2019-10-01T03:55:42.533Z",
    "EntityDetailsList": [
        {
            "EntityInfo": {
                "Arn": "arn:aws:iam::123456789012:user/admin",
                "Name": "admin",
                "Type": "USER",
                "Id": "AIDAIO2XMPLENQEXAMPLE",
                "Path": "/"
            },
            "LastAuthenticated": "2019-09-30T23:02:00Z"
        },
        {
```

```
            "EntityInfo": {
                "Arn": "arn:aws:iam::123456789012:user/developer",
                "Name": "developer",
                "Type": "USER",
                "Id": "AIDAIBEYXMPL2YEXAMPLE",
                "Path": "/"
            },
            "LastAuthenticated": "2019-09-16T19:34:00Z"
        }
    ]
}
```

Per ulteriori informazioni, consulta Refining permissions in AWS using last access information nella AWS IAM User Guide.

- Per i dettagli sull'API, consulta AWS CLI Command GetServiceLastAccessedDetailsWithEntitiesReference.

`get-service-last-accessed-details`

Il seguente esempio di codice mostra come utilizzare`get-service-last-accessed-details`.

AWS CLI

Come recuperare un report di accesso al servizio

L'esempio `get-service-last-accessed-details` seguente recupera un report generato in precedenza che elenca i servizi a cui accedono le entità IAM. Per generare un report, utilizza il `generate-service-last-accessed-details` comando.

```
aws iam get-service-last-accessed-details \
    --job-id 2eb6c2b8-7b4c-3xmp-3c13-03b72c8cdfdc
```

Output:

```
{
    "JobStatus": "COMPLETED",
    "JobCreationDate": "2019-10-01T03:50:35.929Z",
    "ServicesLastAccessed": [
        ...
        {
            "ServiceName": "AWS Lambda",
```

```
            "LastAuthenticated": "2019-09-30T23:02:00Z",
            "ServiceNamespace": "lambda",
            "LastAuthenticatedEntity": "arn:aws:iam::123456789012:user/admin",
            "TotalAuthenticatedEntities": 6
        },
    ]
}
```

Per ulteriori informazioni, consulta Refining permissions in AWS using last access information nella AWS IAM User Guide.

- Per i dettagli sull'API, consulta AWS CLI Command GetServiceLastAccessedDetailsReference.

`get-service-linked-role-deletion-status`

Il seguente esempio di codice mostra come utilizzare`get-service-linked-role-deletion-status`.

AWS CLI

Come verificare lo stato di una richiesta di eliminazione di un ruolo collegato a un servizio

L'esempio `get-service-linked-role-deletion-status` seguente visualizza lo stato di una precedente richiesta di eliminazione di un ruolo collegato a un servizio. L'operazione di eliminazione avviene in modo asincrono. Quando effettui la richiesta, ottieni un valore `DeletionTaskId` che hai fornito come parametro per questo comando.

```
aws iam get-service-linked-role-deletion-status \
    --deletion-task-id task/aws-service-role/lex.amazonaws.com/
AWSServiceRoleForLexBots/1a2b3c4d-1234-abcd-7890-abcdeEXAMPLE
```

Output:

```
{
"Status": "SUCCEEDED"
}
```

Per ulteriori informazioni, consulta Utilizzo di ruoli collegati a servizi nella Guida per l'utente di IAM AWS .

- Per i dettagli sull'API, consulta GetServiceLinkedRoleDeletionStatus AWS CLICommand Reference.

`get-ssh-public-key`

Il seguente esempio di codice mostra come utilizzare`get-ssh-public-key`.

AWS CLI

Esempio 1: come recuperare una chiave pubblica SSH allegata a un utente IAM in formato codificato SSH

Il comando `get-ssh-public-key` seguente recupera la chiave pubblica SSH specificata dall'utente IAM `sofia`. L'output è in codifica SSH.

```
aws iam get-ssh-public-key \
    --user-name sofia \
    --ssh-public-key-id APKA123456789EXAMPLE \
    --encoding SSH
```

Output:

```
{
    "SSHPublicKey": {
        "UserName": "sofia",
        "SSHPublicKeyId": "APKA123456789EXAMPLE",
        "Fingerprint": "12:34:56:78:90:ab:cd:ef:12:34:56:78:90:ab:cd:ef",
        "SSHPublicKeyBody": "ssh-rsa <<long encoded SSH string>>",
        "Status": "Inactive",
        "UploadDate": "2019-04-18T17:04:49+00:00"
    }
}
```

Esempio 2: come recuperare una chiave pubblica SSH allegata a un utente IAM in formato codificato PEM

Il comando `get-ssh-public-key` seguente recupera la chiave pubblica SSH specificata dall'utente IAM `sofia`. L'output è in codifica PEM.

```
aws iam get-ssh-public-key \
    --user-name sofia \
    --ssh-public-key-id APKA123456789EXAMPLE \
    --encoding PEM
```

Output:

```
{
    "SSHPublicKey": {
        "UserName": "sofia",
        "SSHPublicKeyId": "APKA123456789EXAMPLE",
        "Fingerprint": "12:34:56:78:90:ab:cd:ef:12:34:56:78:90:ab:cd:ef",
        "SSHPublicKeyBody": ""-----BEGIN PUBLIC KEY-----\n<<long encoded PEM
 string>>\n-----END PUBLIC KEY-----\n"",
        "Status": "Inactive",
        "UploadDate": "2019-04-18T17:04:49+00:00"
    }
}
```

Per ulteriori informazioni, consulta Use SSH keys and SSH with CodeCommit nella AWS IAM User Guide.

- Per i dettagli sull'API, consulta AWS CLI Command GetSshPublicKeyReference.

`get-user-policy`

Il seguente esempio di codice mostra come utilizzare`get-user-policy`.

AWS CLI

Come visualizzare i dettagli della policy per un utente IAM

Il comando `get-user-policy` seguente riporta i dettagli della policy specificata collegata all'utente IAM denominato Bob.

```
aws iam get-user-policy \
    --user-name Bob \
    --policy-name ExamplePolicy
```

Output:

```
{
    "UserName": "Bob",
    "PolicyName": "ExamplePolicy",
    "PolicyDocument": {
        "Version":"2012-10-17",
        "Statement": [
            {
```

```
                "Action": "*",
                "Resource": "*",
                "Effect": "Allow"
            }
        ]
    }
}
```

Per ottenere un elenco di policy per un utente IAM, usa il comando `list-user-policies`.

Per ulteriori informazioni, consulta Policy e autorizzazioni in IAM nella Guida per l'utente di IAM AWS .

- Per i dettagli sull'API, consulta GetUserPolicy AWS CLICommand Reference.

`get-user`

Il seguente esempio di codice mostra come utilizzare`get-user`.

AWS CLI

Come ottenere informazioni su un utente IAM

Il comando `get-user` seguente ottiene informazioni sull'utente IAM denominato `Paulo`.

```
aws iam get-user \
    --user-name Paulo
```

Output:

```
{
    "User": {
        "UserName": "Paulo",
        "Path": "/",
        "CreateDate": "2019-09-21T23:03:13Z",
        "UserId": "AIDA123456789EXAMPLE",
        "Arn": "arn:aws:iam::123456789012:user/Paulo"
    }
}
```

Per ulteriori informazioni, consulta Gestione di utenti IAM nella Guida per l'utente IAM AWS .

- Per i dettagli sull'API, consulta GetUser AWS CLICommand Reference.

`list-access-keys`

Il seguente esempio di codice mostra come utilizzare`list-access-keys`.

AWS CLI

Per elencare la chiave di accesso IDs per un utente IAM

Il `list-access-keys` comando seguente elenca le chiavi di accesso IDs per l'utente IAM denominatoBob.

```
aws iam list-access-keys \
    --user-name Bob
```

Output:

```
{
    "AccessKeyMetadata": [
        {
            "UserName": "Bob",
            "Status": "Active",
            "CreateDate": "2013-06-04T18:17:34Z",
            "AccessKeyId": "AKIAIOSFODNN7EXAMPLE"
        },
        {
            "UserName": "Bob",
            "Status": "Inactive",
            "CreateDate": "2013-06-06T20:42:26Z",
            "AccessKeyId": "AKIAI44QH8DHBEXAMPLE"
        }
    ]
}
```

Non puoi elencare le chiavi di accesso segrete per gli utenti IAM. Se le chiavi di accesso segrete vengono perse, devi creare nuove chiavi di accesso utilizzando il comando `create-access-keys`.

Per ulteriori informazioni, consulta Gestione delle chiavi di accesso per gli utenti IAM nella Guida per l'utente di IAM AWS .

- Per i dettagli sull'API, consulta ListAccessKeys AWS CLICommand Reference.

`list-account-aliases`

Il seguente esempio di codice mostra come utilizzare`list-account-aliases`.

AWS CLI

Elencare gli alias di un account

Il comando `list-account-aliases` seguente elenca gli alias per l'account corrente.

```
aws iam list-account-aliases
```

Output:

```
{
    "AccountAliases": [
    "mycompany"
    ]
}
```

Per ulteriori informazioni, consulta Your AWS account ID and its alias nella AWS IAM User Guide.

- Per i dettagli sull'API, consulta ListAccountAliases AWS CLICommand Reference.

`list-attached-group-policies`

Il seguente esempio di codice mostra come utilizzare`list-attached-group-policies`.

AWS CLI

Come visualizzare tutte le policy gestite collegate al gruppo specificato

Questo esempio restituisce i nomi e ARNs le politiche gestite allegate al gruppo IAM denominato `Admins` nell' AWS account.

```
aws iam list-attached-group-policies \
    --group-name Admins
```

Output:

```
{
    "AttachedPolicies": [
        {
            "PolicyName": "AdministratorAccess",
            "PolicyArn": "arn:aws:iam::aws:policy/AdministratorAccess"
        },
        {
            "PolicyName": "SecurityAudit",
            "PolicyArn": "arn:aws:iam::aws:policy/SecurityAudit"
        }
    ],
    "IsTruncated": false
}
```

Per ulteriori informazioni, consulta Policy e autorizzazioni in IAM nella Guida per l'utente di IAM AWS .

- Per i dettagli sull'API, consulta ListAttachedGroupPolicies AWS CLICommand Reference.

`list-attached-role-policies`

Il seguente esempio di codice mostra come utilizzare`list-attached-role-policies`.

AWS CLI

Come elencare tutte le policy gestite collegate al ruolo specificato

Questo comando restituisce i nomi e ARNs le politiche gestite allegate al ruolo IAM denominato `SecurityAuditRole` nell' AWS account.

```
aws iam list-attached-role-policies \
    --role-name SecurityAuditRole
```

Output:

```
{
    "AttachedPolicies": [
        {
            "PolicyName": "SecurityAudit",
            "PolicyArn": "arn:aws:iam::aws:policy/SecurityAudit"
        }
    ],
```

```
    "IsTruncated": false
}
```

Per ulteriori informazioni, consulta Policy e autorizzazioni in IAM nella Guida per l'utente di IAM AWS .

- Per i dettagli sull'API, consulta ListAttachedRolePolicies AWS CLICommand Reference.

`list-attached-user-policies`

Il seguente esempio di codice mostra come utilizzare`list-attached-user-policies`.

AWS CLI

Come visualizzare tutte le policy gestite collegate all'utente specificato

Questo comando restituisce i nomi e ARNs le politiche gestite per l'utente IAM indicato Bob nell' AWS account.

```
aws iam list-attached-user-policies \
    --user-name Bob
```

Output:

```
{
    "AttachedPolicies": [
        {
            "PolicyName": "AdministratorAccess",
            "PolicyArn": "arn:aws:iam::aws:policy/AdministratorAccess"
        },
        {
            "PolicyName": "SecurityAudit",
            "PolicyArn": "arn:aws:iam::aws:policy/SecurityAudit"
        }
    ],
    "IsTruncated": false
}
```

Per ulteriori informazioni, consulta Policy e autorizzazioni in IAM nella Guida per l'utente di IAM AWS .

- Per i dettagli sull'API, consulta ListAttachedUserPolicies AWS CLICommand Reference.

`list-entities-for-policy`

Il seguente esempio di codice mostra come utilizzare`list-entities-for-policy`.

AWS CLI

Come visualizzare tutti gli utenti, i gruppi e i ruoli a cui è collegata la policy gestita specificata

Questo esempio restituisce un elenco di gruppi, ruoli e utenti IAM a cui è collegata la policy `arn:aws:iam::123456789012:policy/TestPolicy`.

```
aws iam list-entities-for-policy \
    --policy-arn arn:aws:iam::123456789012:policy/TestPolicy
```

Output:

```
{
    "PolicyGroups": [
        {
            "GroupName": "Admins",
            "GroupId": "AGPACKCEVSQ6C2EXAMPLE"
        }
    ],
    "PolicyUsers": [
        {
            "UserName": "Alice",
            "UserId": "AIDACKCEVSQ6C2EXAMPLE"
        }
    ],
    "PolicyRoles": [
        {
            "RoleName": "DevRole",
            "RoleId": "AROADBQP57FF2AEXAMPLE"
        }
    ],
    "IsTruncated": false
}
```

Per ulteriori informazioni, consulta Policy e autorizzazioni in IAM nella Guida per l'utente di IAM AWS .

- Per i dettagli sull'API, consulta ListEntitiesForPolicy AWS CLICommand Reference.

`list-group-policies`

Il seguente esempio di codice mostra come utilizzare`list-group-policies`.

AWS CLI

Come visualizzare tutte le policy in linea collegate al gruppo specificato

Il comando `list-group-policies` seguente elenca i nomi delle policy in linea collegate al gruppo IAM denominato `Admins` nell'account corrente.

```
aws iam list-group-policies \
    --group-name Admins
```

Output:

```
{
    "PolicyNames": [
        "AdminRoot",
        "ExamplePolicy"
    ]
}
```

Per ulteriori informazioni, consulta Gestione delle policy IAM nella Guida per l'utente IAM AWS .

- Per i dettagli sull'API, consulta ListGroupPolicies AWS CLICommand Reference.

`list-groups-for-user`

Il seguente esempio di codice mostra come utilizzare`list-groups-for-user`.

AWS CLI

Come visualizzare i gruppi a cui appartiene un utente IAM

Il comando `list-groups-for-user` seguente mostra i gruppi a cui appartiene l'utente IAM denominato `Bob`.

```
aws iam list-groups-for-user \
    --user-name Bob
```

Output:

```
{
    "Groups": [
        {
            "Path": "/",
            "CreateDate": "2013-05-06T01:18:08Z",
            "GroupId": "AKIAIOSFODNN7EXAMPLE",
            "Arn": "arn:aws:iam::123456789012:group/Admin",
            "GroupName": "Admin"
        },
        {
            "Path": "/",
            "CreateDate": "2013-05-06T01:37:28Z",
            "GroupId": "AKIAI44QH8DHBEXAMPLE",
            "Arn": "arn:aws:iam::123456789012:group/s3-Users",
            "GroupName": "s3-Users"
        }
    ]
}
```

Per ulteriori informazioni, consulta Gestione di gruppi di utenti IAM nella Guida per l'utente IAM AWS .

- Per i dettagli sull'API, consulta ListGroupsForUser AWS CLICommand Reference.

`list-groups`

Il seguente esempio di codice mostra come utilizzare`list-groups`.

AWS CLI

Come elencare i gruppi IAM per l'account corrente

Il comando `list-groups` seguente elenca i gruppi IAM nell'account corrente.

```
aws iam list-groups
```

Output:

```
{
    "Groups": [
        {
            "Path": "/",
```

```
            "CreateDate": "2013-06-04T20:27:27.972Z",
            "GroupId": "AIDACKCEVSQ6C2EXAMPLE",
            "Arn": "arn:aws:iam::123456789012:group/Admins",
            "GroupName": "Admins"
        },
        {
            "Path": "/",
            "CreateDate": "2013-04-16T20:30:42Z",
            "GroupId": "AIDGPMS9RO4H3FEXAMPLE",
            "Arn": "arn:aws:iam::123456789012:group/S3-Admins",
            "GroupName": "S3-Admins"
        }
    ]
}
```

Per ulteriori informazioni, consulta Gestione di gruppi di utenti IAM nella Guida per l'utente IAM AWS .

- Per i dettagli sull'API, consulta ListGroups AWS CLICommand Reference.

`list-instance-profile-tags`

Il seguente esempio di codice mostra come utilizzare`list-instance-profile-tags`.

AWS CLI

Come elencare i tag allegati a un profilo dell'istanza

Il comando `list-instance-profile-tags` seguente recupera l'elenco dei tag associati al profilo dell'istanza specificato.

```
aws iam list-instance-profile-tags \
    --instance-profile-name deployment-role
```

Output:

```
{
    "Tags": [
        {
            "Key": "DeptID",
            "Value": "123456"
        },
        {
```

```
            "Key": "Department",
            "Value": "Accounting"
        }
    ]
}
```

Per ulteriori informazioni, consulta Applicazione di tag a risorse IAM nella Guida per l'utente di AWS IAM.

- Per i dettagli sull'API, consulta ListInstanceProfileTags AWS CLICommand Reference.

`list-instance-profiles-for-role`

Il seguente esempio di codice mostra come utilizzare`list-instance-profiles-for-role`.

AWS CLI

Come visualizzare i profili dell'istanza per un ruolo IAM

Il comando `list-instance-profiles-for-role` seguente elenca i profili dell'istanza associati al ruolo `Test-Role`.

```
aws iam list-instance-profiles-for-role \
    --role-name Test-Role
```

Output:

```
{
    "InstanceProfiles": [
        {
            "InstanceProfileId": "AIDGPMS9RO4H3FEXAMPLE",
            "Roles": [
                {
                    "AssumeRolePolicyDocument": "<URL-encoded-JSON>",
                    "RoleId": "AIDACKCEVSQ6C2EXAMPLE",
                    "CreateDate": "2013-06-07T20:42:15Z",
                    "RoleName": "Test-Role",
                    "Path": "/",
                    "Arn": "arn:aws:iam::123456789012:role/Test-Role"
                }
            ],
            "CreateDate": "2013-06-07T21:05:24Z",
            "InstanceProfileName": "ExampleInstanceProfile",
```

```
            "Path": "/",
            "Arn": "arn:aws:iam::123456789012:instance-profile/
ExampleInstanceProfile"
        }
    ]
}
```

Per ulteriori informazioni, consulta Utilizzo dei profili dell'istanza nella Guida per l'utente di IAM AWS .

- Per i dettagli sull'API, consulta ListInstanceProfilesForRole AWS CLICommand Reference.

`list-instance-profiles`

Il seguente esempio di codice mostra come utilizzare`list-instance-profiles`.

AWS CLI

Come elencare i profili dell'istanza per l'account

Il `list-instance-profiles` comando seguente elenca i profili dell'istanza associati all'account corrente.

```
aws iam list-instance-profiles
```

Output:

```
{
    "InstanceProfiles": [
        {
            "Path": "/",
            "InstanceProfileName": "example-dev-role",
            "InstanceProfileId": "AIPAIXEU4NUHUPEXAMPLE",
            "Arn": "arn:aws:iam::123456789012:instance-profile/example-dev-role",
            "CreateDate": "2023-09-21T18:17:41+00:00",
            "Roles": [
                {
                    "Path": "/",
                    "RoleName": "example-dev-role",
                    "RoleId": "AROAJ52OTH4H7LEXAMPLE",
                    "Arn": "arn:aws:iam::123456789012:role/example-dev-role",
                    "CreateDate": "2023-09-21T18:17:40+00:00",
                    "AssumeRolePolicyDocument": {
```

```
                        "Version":"2012-10-17",
                        "Statement": [
                            {
                                "Effect": "Allow",
                                "Principal": {
                                    "Service": "ec2.amazonaws.com"
                                },
                                "Action": "sts:AssumeRole"
                            }
                        ]
                    }
                }
            ]
        },
        {
            "Path": "/",
            "InstanceProfileName": "example-s3-role",
            "InstanceProfileId": "AIPAJVJVNRIQFREXAMPLE",
            "Arn": "arn:aws:iam::123456789012:instance-profile/example-s3-role",
            "CreateDate": "2023-09-21T18:18:50+00:00",
            "Roles": [
                {
                    "Path": "/",
                    "RoleName": "example-s3-role",
                    "RoleId": "AROAINUBC5O7XLEXAMPLE",
                    "Arn": "arn:aws:iam::123456789012:role/example-s3-role",
                    "CreateDate": "2023-09-21T18:18:49+00:00",
                    "AssumeRolePolicyDocument": {
                        "Version":"2012-10-17",
                        "Statement": [
                            {
                                "Effect": "Allow",
                                "Principal": {
                                    "Service": "ec2.amazonaws.com"
                                },
                                "Action": "sts:AssumeRole"
                            }
                        ]
                    }
                }
            ]
        }
    ]
```

```
}
```

Per ulteriori informazioni, consulta Utilizzo dei profili dell'istanza nella Guida per l'utente di IAM AWS .

- Per i dettagli sull'API, consulta ListInstanceProfiles AWS CLICommand Reference.

`list-mfa-device-tags`

Il seguente esempio di codice mostra come utilizzare`list-mfa-device-tags`.

AWS CLI

Come elencare i tag allegati a un dispositivo MFA

Il comando `list-mfa-device-tags` seguente recupera l'elenco dei tag associati al dispositivo MFA specificato.

```
aws iam list-mfa-device-tags \
    --serial-number arn:aws:iam::123456789012:mfa/alice
```

Output:

```
{
    "Tags": [
        {
            "Key": "DeptID",
            "Value": "123456"
        },
        {
            "Key": "Department",
            "Value": "Accounting"
        }
    ]
}
```

Per ulteriori informazioni, consulta Applicazione di tag a risorse IAM nella Guida per l'utente di AWS IAM.

- Per i dettagli sull'API, consulta ListMfaDeviceTags AWS CLICommand Reference.

`list-mfa-devices`

Il seguente esempio di codice mostra come utilizzare`list-mfa-devices`.

AWS CLI

Come elencare tutti i dispositivi MFA per un utente specificato

Questo esempio restituisce i dettagli del dispositivo MFA assegnato all'utente IAM Bob.

```
aws iam list-mfa-devices \
    --user-name Bob
```

Output:

```
{
    "MFADevices": [
        {
            "UserName": "Bob",
            "SerialNumber": "arn:aws:iam::123456789012:mfa/Bob",
            "EnableDate": "2019-10-28T20:37:09+00:00"
        },
        {
            "UserName": "Bob",
            "SerialNumber": "GAKT12345678",
            "EnableDate": "2023-02-18T21:44:42+00:00"
        },
        {
            "UserName": "Bob",
            "SerialNumber": "arn:aws:iam::123456789012:u2f/user/Bob/
fidosecuritykey1-7XNL7NFNLZ123456789EXAMPLE",
            "EnableDate": "2023-09-19T02:25:35+00:00"
        },
        {
            "UserName": "Bob",
            "SerialNumber": "arn:aws:iam::123456789012:u2f/user/Bob/
fidosecuritykey2-VDRQTDBBN5123456789EXAMPLE",
            "EnableDate": "2023-09-19T01:49:18+00:00"
        }
    ]
}
```

Per ulteriori informazioni, consulta Utilizzo dell'autenticazione a più fattori (MFA) in AWS nella AWS Guida per l'utente IAM.

- Per i dettagli sull'API, consulta ListMfaDevices AWS CLICommand Reference.

`list-open-id-connect-provider-tags`

Il seguente esempio di codice mostra come utilizzare`list-open-id-connect-provider-tags`.

AWS CLI

Come elencare i tag associati a un gestore dell'identità compatibile con OpenID Connect (OIDC)

Il comando `list-open-id-connect-provider-tags` seguente recupera l'elenco dei tag associati al provider di identità OIDC specificato.

```
aws iam list-open-id-connect-provider-tags \
    --open-id-connect-provider-arn arn:aws:iam::123456789012:oidc-provider/
server.example.com
```

Output:

```
{
    "Tags": [
        {
            "Key": "DeptID",
            "Value": "123456"
        },
        {
            "Key": "Department",
            "Value": "Accounting"
        }
    ]
}
```

Per ulteriori informazioni, consulta Applicazione di tag a risorse IAM nella Guida per l'utente di AWS IAM.

- Per i dettagli sull'API, consulta ListOpenIdConnectProviderTags AWS CLICommand Reference.

`list-open-id-connect-providers`

Il seguente esempio di codice mostra come utilizzare`list-open-id-connect-providers`.

AWS CLI

Per elencare informazioni sui provider OpenID Connect presenti nell'account AWS

Questo esempio restituisce un elenco di ARNS di tutti i provider OpenID Connect definiti AWS nell'account corrente.

```
aws iam list-open-id-connect-providers
```

Output:

```
{
    "OpenIDConnectProviderList": [
        {
            "Arn": "arn:aws:iam::123456789012:oidc-provider/
example.oidcprovider.com"
        }
    ]
}
```

Per ulteriori informazioni, consulta Creazione di provider di identità OpenID Connect (OIDC) nella Guida per l'utente di AWS IAM.

- Per i dettagli sull'API, consulta AWS CLI Command ListOpenIdConnectProvidersReference.

`list-organizations-features`

Il seguente esempio di codice mostra come utilizzare`list-organizations-features`.

AWS CLI

Come elencare le funzionalità di accesso root centralizzato abilitate per l'organizzazione

Il comando `list-organizations-features` seguente elenca le funzionalità di accesso root centralizzato abilitate per l'organizzazione.

```
aws iam list-organizations-features
```

Output:

```
{
    "EnabledFeatures": [
        "RootCredentialsManagement",
        "RootSessions"
    ]
    "OrganizationId": "o-aa111bb222"
}
```

Per ulteriori informazioni, consulta Gestire centralmente l'accesso root per gli account membri nella Guida per l'utente di AWS IAM.

- Per i dettagli sull'API, consulta ListOrganizationsFeatures AWS CLICommand Reference.

`list-policies-granting-service-access`

Il seguente esempio di codice mostra come utilizzare`list-policies-granting-service-access`.

AWS CLI

Come elencare le policy che garantiscono l'accesso principale al servizio specificato

L'`list-policies-granting-service-access`esempio seguente recupera l'elenco delle politiche che concedono all'utente IAM l'`sofia`accesso al AWS CodeCommit servizio.

```
aws iam list-policies-granting-service-access \
    --arn arn:aws:iam::123456789012:user/sofia \
    --service-namespaces codecommit
```

Output:

```
{
    "PoliciesGrantingServiceAccess": [
        {
            "ServiceNamespace": "codecommit",
            "Policies": [
                {
                    "PolicyName": "Grant-Sofia-Access-To-CodeCommit",
                    "PolicyType": "INLINE",
```

```
                    "EntityType": "USER",
                    "EntityName": "sofia"
                }
            ]
        }
    ],
    "IsTruncated": false
}
```

Per ulteriori informazioni, consulta Using IAM with CodeCommit: credenziali Git, chiavi SSH e chiavi di AWS accesso nella AWS IAM User Guide.

- Per i dettagli sull'API, consulta AWS CLI Command ListPoliciesGrantingServiceAccessReference.

`list-policies`

Il seguente esempio di codice mostra come utilizzare`list-policies`.

AWS CLI

Per elencare le politiche gestite disponibili per il tuo AWS account

Questo esempio restituisce una raccolta delle prime due politiche gestite disponibili nell' AWS account corrente.

```
aws iam list-policies \
    --max-items 3
```

Output:

```
{
    "Policies": [
        {
            "PolicyName": "AWSCloudTrailAccessPolicy",
            "PolicyId": "ANPAXQE2B5PJ7YEXAMPLE",
            "Arn": "arn:aws:iam::123456789012:policy/AWSCloudTrailAccessPolicy",
            "Path": "/",
            "DefaultVersionId": "v1",
            "AttachmentCount": 0,
            "PermissionsBoundaryUsageCount": 0,
            "IsAttachable": true,
```

```
            "CreateDate": "2019-09-04T17:43:42+00:00",
            "UpdateDate": "2019-09-04T17:43:42+00:00"
        },
        {
            "PolicyName": "AdministratorAccess",
            "PolicyId": "ANPAIWMBCKSKIEE64ZLYK",
            "Arn": "arn:aws:iam::aws:policy/AdministratorAccess",
            "Path": "/",
            "DefaultVersionId": "v1",
            "AttachmentCount": 6,
            "PermissionsBoundaryUsageCount": 0,
            "IsAttachable": true,
            "CreateDate": "2015-02-06T18:39:46+00:00",
            "UpdateDate": "2015-02-06T18:39:46+00:00"
        },
        {
            "PolicyName": "PowerUserAccess",
            "PolicyId": "ANPAJYRXTHIB4FOVS3ZXS",
            "Arn": "arn:aws:iam::aws:policy/PowerUserAccess",
            "Path": "/",
            "DefaultVersionId": "v5",
            "AttachmentCount": 1,
            "PermissionsBoundaryUsageCount": 0,
            "IsAttachable": true,
            "CreateDate": "2015-02-06T18:39:47+00:00",
            "UpdateDate": "2023-07-06T22:04:00+00:00"
        }
    ],
    "NextToken": "EXAMPLErZXIiOiBudWxsLCAiYm90b190cnVuY2F0ZV9hbW91bnQiOiA4fQ=="
}
```

Per ulteriori informazioni, consulta Policy e autorizzazioni in IAM nella Guida per l'utente di IAM AWS .

- Per i dettagli sull'API, consulta ListPolicies AWS CLICommand Reference.

`list-policy-tags`

Il seguente esempio di codice mostra come utilizzare`list-policy-tags`.

AWS CLI

Come elencare i tag allegati a una policy gestita

Il comando `list-policy-tags` seguente recupera l'elenco dei tag associati alla policy gestita specificata.

```
aws iam list-policy-tags \
    --policy-arn arn:aws:iam::123456789012:policy/billing-access
```

Output:

```
{
    "Tags": [
        {
            "Key": "DeptID",
            "Value": "123456"
        },
        {
            "Key": "Department",
            "Value": "Accounting"
        }
    ]
}
```

Per ulteriori informazioni, consulta Applicazione di tag a risorse IAM nella Guida per l'utente di AWS IAM.

- Per i dettagli sull'API, consulta ListPolicyTags AWS CLICommand Reference.

`list-policy-versions`

Il seguente esempio di codice mostra come utilizzare`list-policy-versions`.

AWS CLI

Come visualizzare le informazioni sulle versioni della policy gestita specificata

Questo esempio restituisce l'elenco delle versioni disponibili della policy il cui ARN è `arn:aws:iam::123456789012:policy/MySamplePolicy`.

```
aws iam list-policy-versions \
    --policy-arn arn:aws:iam::123456789012:policy/MySamplePolicy
```

Output:

```
{
    "IsTruncated": false,
    "Versions": [
        {
        "VersionId": "v2",
        "IsDefaultVersion": true,
        "CreateDate": "2015-06-02T23:19:44Z"
        },
        {
        "VersionId": "v1",
        "IsDefaultVersion": false,
        "CreateDate": "2015-06-02T22:30:47Z"
        }
    ]
}
```

Per ulteriori informazioni, consulta Policy e autorizzazioni in IAM nella Guida per l'utente di IAM AWS .

- Per i dettagli sull'API, consulta ListPolicyVersions AWS CLICommand Reference.

`list-role-policies`

Il seguente esempio di codice mostra come utilizzare`list-role-policies`.

AWS CLI

Come elencare le policy collegate a un ruolo IAM

Il comando `list-role-policies` seguente elenca i nomi delle policy di autorizzazione per il ruolo IAM specificato.

```
aws iam list-role-policies \
    --role-name Test-Role
```

Output:

```
{
    "PolicyNames": [
        "ExamplePolicy"
    ]
```

```
}
```

Per consultare la policy di attendibilità collegata a un ruolo, usa il comando `get-role`. Per visualizzare i dettagli di una policy di autorizzazioni, usa il comando `get-role-policy`.

Per ulteriori informazioni, consulta Creazione di ruoli IAM nella Guida per l'utente IAM AWS .

- Per i dettagli sull'API, consulta ListRolePolicies AWS CLICommand Reference.

`list-role-tags`

Il seguente esempio di codice mostra come utilizzare`list-role-tags`.

AWS CLI

Elencare i tag collegati a un ruolo

Il comando `list-role-tags` seguente recupera l'elenco dei tag associati al ruolo specificato.

```
aws iam list-role-tags \
    --role-name production-role
```

Output:

```
{
    "Tags": [
        {
            "Key": "Department",
            "Value": "Accounting"
        },
        {
            "Key": "DeptID",
            "Value": "12345"
        }
    ],
    "IsTruncated": false
}
```

Per ulteriori informazioni, consulta Applicazione di tag a risorse IAM nella Guida per l'utente di AWS IAM.

- Per i dettagli sull'API, consulta ListRoleTags AWS CLICommand Reference.

`list-roles`

Il seguente esempio di codice mostra come utilizzare`list-roles`.

AWS CLI

Come elencare i ruoli IAM per l'account corrente

Il comando `list-roles` seguente elenca i ruoli IAM per l'account corrente.

```
aws iam list-roles
```

Output:

```
{
    "Roles": [
        {
            "Path": "/",
            "RoleName": "ExampleRole",
            "RoleId": "AROAJ52OTH4H7LEXAMPLE",
            "Arn": "arn:aws:iam::123456789012:role/ExampleRole",
            "CreateDate": "2017-09-12T19:23:36+00:00",
            "AssumeRolePolicyDocument": {
                "Version":"2012-10-17",
                "Statement": [
                    {
                        "Sid": "",
                        "Effect": "Allow",
                        "Principal": {
                            "Service": "ec2.amazonaws.com"
                        },
                        "Action": "sts:AssumeRole"
                    }
                ]
            },
            "MaxSessionDuration": 3600
        },
        {
            "Path": "/example_path/",
            "RoleName": "ExampleRoleWithPath",
            "RoleId": "AROAI4QRP7UFT7EXAMPLE",
            "Arn": "arn:aws:iam::123456789012:role/example_path/
ExampleRoleWithPath",
```

```
            "CreateDate": "2023-09-21T20:29:38+00:00",
            "AssumeRolePolicyDocument": {
                "Version":"2012-10-17",
                "Statement": [
                    {
                        "Sid": "",
                        "Effect": "Allow",
                        "Principal": {
                            "Service": "ec2.amazonaws.com"
                        },
                        "Action": "sts:AssumeRole"
                    }
                ]
            },
            "MaxSessionDuration": 3600
        }
    ]
}
```

Per ulteriori informazioni, consulta Creazione di ruoli IAM nella Guida per l'utente IAM AWS .

- Per i dettagli sull'API, consulta ListRoles AWS CLICommand Reference.

`list-saml-provider-tags`

Il seguente esempio di codice mostra come utilizzare`list-saml-provider-tags`.

AWS CLI

Come elencare i tag allegati a un provider SAML

Il comando `list-saml-provider-tags` seguente recupera l'elenco dei tag associati al provider SAML specificato.

```
aws iam list-saml-provider-tags \
    --saml-provider-arn arn:aws:iam::123456789012:saml-provider/ADFS
```

Output:

```
{
    "Tags": [
        {
            "Key": "DeptID",
```

```
            "Value": "123456"
        },
        {
            "Key": "Department",
            "Value": "Accounting"
        }
    ]
}
```

Per ulteriori informazioni, consulta Applicazione di tag a risorse IAM nella Guida per l'utente di AWS IAM.

- Per i dettagli sull'API, consulta ListSamlProviderTags AWS CLICommand Reference.

`list-saml-providers`

Il seguente esempio di codice mostra come utilizzare`list-saml-providers`.

AWS CLI

Per elencare i provider SAML presenti nell'account AWS

Questo esempio recupera l'elenco dei provider SAML 2.0 creati nell'account corrente. AWS

```
aws iam list-saml-providers
```

Output:

```
{
    "SAMLProviderList": [
        {
            "Arn": "arn:aws:iam::123456789012:saml-provider/SAML-ADFS",
            "ValidUntil": "2015-06-05T22:45:14Z",
            "CreateDate": "2015-06-05T22:45:14Z"
        }
    ]
}
```

Per ulteriori informazioni, consulta Creazione di provider di identità SAML IAM nella Guida per l'utente di IAM AWS .

- Per i dettagli sull'API, consulta List SAMLProviders in AWS CLI Command Reference.

`list-server-certificate-tags`

Il seguente esempio di codice mostra come utilizzare`list-server-certificate-tags`.

AWS CLI

Come elencare i tag allegati a un certificato server

Il comando `list-server-certificate-tags` seguente recupera l'elenco dei tag associati al certificato del server specificato.

```
aws iam list-server-certificate-tags \
    --server-certificate-name ExampleCertificate
```

Output:

```
{
    "Tags": [
        {
            "Key": "DeptID",
            "Value": "123456"
        },
        {
            "Key": "Department",
            "Value": "Accounting"
        }
    ]
}
```

Per ulteriori informazioni, consulta Applicazione di tag a risorse IAM nella Guida per l'utente di AWS IAM.

- Per i dettagli sull'API, consulta ListServerCertificateTags AWS CLICommand Reference.

`list-server-certificates`

Il seguente esempio di codice mostra come utilizzare`list-server-certificates`.

AWS CLI

Per elencare i certificati del server presenti nel tuo AWS account

Il `list-server-certificates` comando seguente elenca tutti i certificati server archiviati e disponibili per l'uso nell' AWS account.

```
aws iam list-server-certificates
```

Output:

```
{
    "ServerCertificateMetadataList": [
        {
            "Path": "/",
            "ServerCertificateName": "myUpdatedServerCertificate",
            "ServerCertificateId": "ASCAEXAMPLE123EXAMPLE",
            "Arn": "arn:aws:iam::123456789012:server-certificate/
myUpdatedServerCertificate",
            "UploadDate": "2019-04-22T21:13:44+00:00",
            "Expiration": "2019-10-15T22:23:16+00:00"
        },
        {
            "Path": "/cloudfront/",
            "ServerCertificateName": "MyTestCert",
            "ServerCertificateId": "ASCAEXAMPLE456EXAMPLE",
            "Arn": "arn:aws:iam::123456789012:server-certificate/Org1/Org2/
MyTestCert",
            "UploadDate": "2015-04-21T18:14:16+00:00",
            "Expiration": "2018-01-14T17:52:36+00:00"
        }
    ]
}
```

Per ulteriori informazioni, consulta Gestione dei certificati server in IAM nella Guida per l'utente IAM AWS .

- Per i dettagli sull'API, consulta ListServerCertificates AWS CLICommand Reference.

`list-service-specific-credential`

Il seguente esempio di codice mostra come utilizzare`list-service-specific-credential`.

AWS CLI

Esempio 1: elenca le credenziali specifiche del servizio per un utente

L'esempio `list-service-specific-credentials` seguente visualizza tutte le credenziali specifiche del servizio assegnate all'utente specificato. Le password non sono incluse nella risposta.

```
aws iam list-service-specific-credentials \
    --user-name sofia
```

Output:

```
{
    "ServiceSpecificCredential": {
        "CreateDate": "2019-04-18T20:45:36+00:00",
        "ServiceName": "codecommit.amazonaws.com",
        "ServiceUserName": "sofia-at-123456789012",
        "ServiceSpecificCredentialId": "ACCAEXAMPLE123EXAMPLE",
        "UserName": "sofia",
        "Status": "Active"
    }
}
```

Esempio 2: elenca le credenziali specifiche del servizio per un utente con filtro basato su un servizio specifico

L'esempio `list-service-specific-credentials` seguente visualizza le credenziali specifiche del servizio assegnate all'utente che effettua la richiesta. L'elenco viene filtrato in modo da includere solo le credenziali per il servizio specificato. Le password non sono incluse nella risposta.

```
aws iam list-service-specific-credentials \
    --service-name codecommit.amazonaws.com
```

Output:

```
{
    "ServiceSpecificCredential": {
        "CreateDate": "2019-04-18T20:45:36+00:00",
        "ServiceName": "codecommit.amazonaws.com",
        "ServiceUserName": "sofia-at-123456789012",
        "ServiceSpecificCredentialId": "ACCAEXAMPLE123EXAMPLE",
        "UserName": "sofia",
```

```
        "Status": "Active"
    }
}
```

Per ulteriori informazioni, consulta Creare credenziali Git per le connessioni HTTPS CodeCommit nella Guida per l'AWS CodeCommit utente.

- Per i dettagli sull'API, consulta ListServiceSpecificCredential AWS CLICommand Reference.

`list-service-specific-credentials`

Il seguente esempio di codice mostra come utilizzare`list-service-specific-credentials`.

AWS CLI

Come recuperare un elenco di credenziali

L'`list-service-specific-credentials`esempio seguente elenca le credenziali generate per l'accesso HTTPS ai AWS CodeCommit repository per un utente denominato. `developer`

```
aws iam list-service-specific-credentials \
    --user-name developer \
    --service-name codecommit.amazonaws.com
```

Output:

```
{
    "ServiceSpecificCredentials": [
        {
            "UserName": "developer",
            "Status": "Inactive",
            "ServiceUserName": "developer-at-123456789012",
            "CreateDate": "2019-10-01T04:31:41Z",
            "ServiceSpecificCredentialId": "ACCAQFODXMPL4YFHP7DZE",
            "ServiceName": "codecommit.amazonaws.com"
        },
        {
            "UserName": "developer",
            "Status": "Active",
            "ServiceUserName": "developer+1-at-123456789012",
            "CreateDate": "2019-10-01T04:31:45Z",
```

```
            "ServiceSpecificCredentialId": "ACCAQFOXMPL6VW57M7AJP",
            "ServiceName": "codecommit.amazonaws.com"
        }
    ]
}
```

Per ulteriori informazioni, consulta Creare credenziali Git per le connessioni HTTPS CodeCommit nella Guida per l'AWS CodeCommit utente.

- Per i dettagli sull'API, consulta ListServiceSpecificCredentials AWS CLICommand Reference.

`list-signing-certificates`

Il seguente esempio di codice mostra come utilizzare`list-signing-certificates`.

AWS CLI

Come elencare i certificati di firma per un utente IAM

Il comando `list-signing-certificates` seguente elenca i certificati di firma per l'utente IAM denominato Bob.

```
aws iam list-signing-certificates \
    --user-name Bob
```

Output:

```
{
    "Certificates": [
        {
            "UserName": "Bob",
            "Status": "Inactive",
            "CertificateBody": "-----BEGIN CERTIFICATE-----<certificate-body>-----
END CERTIFICATE-----",
            "CertificateId": "TA7SMP42TDN5Z26OBPJE7EXAMPLE",
            "UploadDate": "2013-06-06T21:40:08Z"
        }
    ]
}
```

Per ulteriori informazioni, consulta Gestire i certificati di firma nella Amazon EC2 User Guide.

- Per i dettagli sull'API, consulta ListSigningCertificates AWS CLICommand Reference.

`list-ssh-public-keys`

Il seguente esempio di codice mostra come utilizzare`list-ssh-public-keys`.

AWS CLI

Come elencare la chiave pubblica SSH allegata a un utente IAM

L'esempio `list-ssh-public-keys` seguente elenca le chiavi pubbliche SSH collegate all'utente IAM `sofia`.

```
aws iam list-ssh-public-keys \
    --user-name sofia
```

Output:

```
{
    "SSHPublicKeys": [
        {
            "UserName": "sofia",
            "SSHPublicKeyId": "APKA1234567890EXAMPLE",
            "Status": "Inactive",
            "UploadDate": "2019-04-18T17:04:49+00:00"
        }
    ]
}
```

Per ulteriori informazioni, consulta Use SSH keys e SSH with CodeCommit nella AWS IAM User Guide

- Per i dettagli sull'API, consulta AWS CLI Command ListSshPublicKeysReference.

`list-user-policies`

Il seguente esempio di codice mostra come utilizzare`list-user-policies`.

AWS CLI

Come elencare le policy per un utente IAM

Il comando `list-user-policies` seguente elenca le policy collegate all'utente IAM denominato Bob.

```
aws iam list-user-policies \
    --user-name Bob
```

Output:

```
{
    "PolicyNames": [
        "ExamplePolicy",
        "TestPolicy"
    ]
}
```

Per ulteriori informazioni, consulta Creating an IAM user in your AWS account nella AWS IAM User Guide.

- Per i dettagli sull'API, consulta ListUserPolicies AWS CLICommand Reference.

`list-user-tags`

Il seguente esempio di codice mostra come utilizzare`list-user-tags`.

AWS CLI

Elencare i tag collegati a un utente

Il comando `list-user-tags` seguente recupera l'elenco dei tag associati all'utente IAM specificato.

```
aws iam list-user-tags \
    --user-name alice
```

Output:

```
{
    "Tags": [
        {
            "Key": "Department",
```

```
            "Value": "Accounting"
        },
        {
            "Key": "DeptID",
            "Value": "12345"
        }
    ],
    "IsTruncated": false
}
```

Per ulteriori informazioni, consulta Applicazione di tag a risorse IAM nella Guida per l'utente di AWS IAM.

- Per i dettagli sull'API, consulta ListUserTags AWS CLICommand Reference.

`list-users`

Il seguente esempio di codice mostra come utilizzare`list-users`.

AWS CLI

Come elencare gli utenti IAM

Il comando `list-users` seguente elenca gli utenti IAM nell'account corrente.

```
aws iam list-users
```

Output:

```
{
    "Users": [
        {
            "UserName": "Adele",
            "Path": "/",
            "CreateDate": "2013-03-07T05:14:48Z",
            "UserId": "AKIAI44QH8DHBEXAMPLE",
            "Arn": "arn:aws:iam::123456789012:user/Adele"
        },
        {
            "UserName": "Bob",
            "Path": "/",
            "CreateDate": "2012-09-21T23:03:13Z",
```

```
            "UserId": "AKIAIOSFODNN7EXAMPLE",
            "Arn": "arn:aws:iam::123456789012:user/Bob"
        }
    ]
}
```

Per ulteriori informazioni, consulta Elencazione degli utenti IAM nella Guida per l'utente IAM AWS .

- Per i dettagli sull'API, consulta ListUsers AWS CLICommand Reference.

`list-virtual-mfa-devices`

Il seguente esempio di codice mostra come utilizzare`list-virtual-mfa-devices`.

AWS CLI

Come elencare i dispositivi MFA virtuale

Il comando `list-virtual-mfa-devices` seguente elenca i dispositivi MFA virtuali che sono stati configurati per l'account corrente.

```
aws iam list-virtual-mfa-devices
```

Output:

```
{
    "VirtualMFADevices": [
        {
            "SerialNumber": "arn:aws:iam::123456789012:mfa/ExampleMFADevice"
        },
        {
            "SerialNumber": "arn:aws:iam::123456789012:mfa/Fred"
        }
    ]
}
```

Per ulteriori informazioni, consulta Abilitazione di un dispositivo di autenticazione a più fattori (MFA) virtuale nella Guida per l'utente di AWS IAM.

- Per i dettagli sull'API, consulta ListVirtualMfaDevices AWS CLICommand Reference.

`put-group-policy`

Il seguente esempio di codice mostra come utilizzare`put-group-policy`.

AWS CLI

Come aggiungere una policy a un gruppo

Il comando `put-group-policy` seguente aggiunge una policy al gruppo IAM denominato `Admins`.

```
aws iam put-group-policy \
    --group-name Admins \
    --policy-document file://AdminPolicy.json \
    --policy-name AdminRoot
```

Questo comando non produce alcun output.

La policy è definita come documento JSON nel AdminPolicyfile.json. (Il nome e l'estensione del file non hanno importanza.)

Per ulteriori informazioni, consulta Gestione delle policy IAM nella Guida per l'utente IAM AWS .

- Per i dettagli sull'API, consulta Command PutGroupPolicyReference AWS CLI .

`put-role-permissions-boundary`

Il seguente esempio di codice mostra come utilizzare`put-role-permissions-boundary`.

AWS CLI

Esempio 1: applicare un limite delle autorizzazioni basato su una policy personalizzata a un ruolo IAM

L'esempio `put-role-permissions-boundary` seguente applica la policy personalizzata denominata `intern-boundary` come limite delle autorizzazioni per il ruolo IAM specificato.

```
aws iam put-role-permissions-boundary \
    --permissions-boundary arn:aws:iam::123456789012:policy/intern-boundary \
    --role-name lambda-application-role
```

Questo comando non produce alcun output.

Esempio 2: applicare un limite di autorizzazioni basato su una policy AWS gestita a un ruolo IAM

L'`put-role-permissions-boundary`esempio seguente applica la `PowerUserAccess` policy AWS gestita come limite di autorizzazioni per il ruolo IAM specificato.

```
aws iam put-role-permissions-boundary \
    --permissions-boundary arn:aws:iam::aws:policy/PowerUserAccess \
    --role-name x-account-admin
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Modifica di un ruolo nella Guida per l'utente di AWS IAM.

- Per i dettagli sull'API, consulta AWS CLI Command PutRolePermissionsBoundaryReference.

`put-role-policy`

Il seguente esempio di codice mostra come utilizzare`put-role-policy`.

AWS CLI

Come collegare una policy di autorizzazioni a un ruolo IAM

Il comando `put-role-policy` seguente aggiunge una policy di autorizzazioni al ruolo denominato `Test-Role`.

```
aws iam put-role-policy \
    --role-name Test-Role \
    --policy-name ExamplePolicy \
    --policy-document file://AdminPolicy.json
```

Questo comando non produce alcun output.

La policy è definita come documento JSON nel AdminPolicyfile.json. (Il nome e l'estensione del file non hanno importanza.)

Per collegare una policy di attendibilità a un ruolo, usa il comando `update-assume-role-policy`.

Per ulteriori informazioni, consulta Modifica di un ruolo nella Guida per l'utente di AWS IAM.

- Per i dettagli sull'API, consulta Command PutRolePolicyReference AWS CLI .

put-user-permissions-boundary

Il seguente esempio di codice mostra come utilizzareput-user-permissions-boundary.

AWS CLI

Esempio 1: applicare un limite delle autorizzazioni basato su una policy personalizzata a un utente IAM

L'esempio put-user-permissions-boundary seguente applica una policy personalizzata denominata intern-boundary come limite delle autorizzazioni per l'utente IAM specificato.

```
aws iam put-user-permissions-boundary \
    --permissions-boundary arn:aws:iam::123456789012:policy/intern-boundary \
    --user-name intern
```

Questo comando non produce alcun output.

Esempio 2: applicare un limite di autorizzazioni basato su una policy AWS gestita a un utente IAM

L'put-user-permissions-boundaryesempio seguente applica la policy AWS gestita PowerUserAccess denominata limite delle autorizzazioni per l'utente IAM specificato.

```
aws iam put-user-permissions-boundary \
    --permissions-boundary arn:aws:iam::aws:policy/PowerUserAccess \
    --user-name developer
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Aggiunta e rimozione di autorizzazioni per identità IAM nella Guida per l'utente di IAM AWS .

- Per i dettagli sull'API, consulta Command PutUserPermissionsBoundaryReference AWS CLI .

put-user-policy

Il seguente esempio di codice mostra come utilizzareput-user-policy.

AWS CLI

Come collegare una policy a un utente IAM

Il comando `put-user-policy` seguente collega una policy all'utente IAM denominato Bob.

```
aws iam put-user-policy \
    --user-name Bob \
    --policy-name ExamplePolicy \
    --policy-document file://AdminPolicy.json
```

Questo comando non produce alcun output.

La policy è definita come documento JSON nel AdminPolicyfile.json. (Il nome e l'estensione del file non hanno importanza.)

Per ulteriori informazioni, consulta Aggiunta e rimozione di autorizzazioni per identità IAM nella Guida per l'utente di IAM AWS .

- Per i dettagli sull'API, consulta Command PutUserPolicyReference AWS CLI .

`remove-client-id-from-open-id-connect-provider`

Il seguente esempio di codice mostra come utilizzare`remove-client-id-from-open-id-connect-provider`.

AWS CLI

Per rimuovere l'ID client specificato dall'elenco dei client IDs registrati per il provider IAM OpenID Connect specificato

Questo esempio rimuove l'ID client `My-TestApp-3` dall'elenco dei client IDs associati al provider IAM OIDC il cui ARN è. `arn:aws:iam::123456789012:oidc-provider/example.oidcprovider.com`

```
aws iam remove-client-id-from-open-id-connect-provider
    --client-id My-TestApp-3 \
    --open-id-connect-provider-arn arn:aws:iam::123456789012:oidc-provider/
example.oidcprovider.com
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Creazione di provider di identità OpenID Connect (OIDC) nella Guida per l'utente di AWS IAM.

- Per i dettagli sull'API, consulta Command RemoveClientIdFromOpenIdConnectProviderReference AWS CLI .

`remove-role-from-instance-profile`

Il seguente esempio di codice mostra come utilizzare`remove-role-from-instance-profile`.

AWS CLI

Come rimuovere un ruolo da un profilo dell'istanza

Il comando `remove-role-from-instance-profile` seguente rimuove il ruolo denominato `Test-Role` dal profilo dell'istanza denominato `ExampleInstanceProfile`.

```
aws iam remove-role-from-instance-profile \
    --instance-profile-name ExampleInstanceProfile \
    --role-name Test-Role
```

Per ulteriori informazioni, consulta Utilizzo dei profili dell'istanza nella Guida per l'utente di IAM AWS .

- Per i dettagli sull'API, consulta RemoveRoleFromInstanceProfile AWS CLICommand Reference.

`remove-user-from-group`

Il seguente esempio di codice mostra come utilizzare`remove-user-from-group`.

AWS CLI

Come rimuovere un utente da un gruppo IAM

Il comando `remove-user-from-group` seguente rimuove l'utente denominato Bob dal gruppo IAM denominato `Admins`.

```
aws iam remove-user-from-group \
    --user-name Bob \
    --group-name Admins
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Aggiunta e rimozione di utenti in un gruppo di utenti IAM nella Guida per l'utente IAM AWS .

- Per i dettagli sull'API, consulta RemoveUserFromGroup AWS CLICommand Reference.

`reset-service-specific-credential`

Il seguente esempio di codice mostra come utilizzare`reset-service-specific-credential`.

AWS CLI

Esempio 1: come reimpostare la password per una credenziale specifica del servizio collegata all'utente che effettua la richiesta

L'esempio `reset-service-specific-credential` seguente genera una nuova password crittograficamente sicura per la credenziale specificata specifica del servizio collegata all'utente che effettua la richiesta.

```
aws iam reset-service-specific-credential \
    --service-specific-credential-id ACCAEXAMPLE123EXAMPLE
```

Output:

```
{
    "ServiceSpecificCredential": {
        "CreateDate": "2019-04-18T20:45:36+00:00",
        "ServiceName": "codecommit.amazonaws.com",
        "ServiceUserName": "sofia-at-123456789012",
        "ServicePassword": "+oaFsNk7tLco+C/obP9GhhcOzGcKOayTmE3LnAmAmH4=",
        "ServiceSpecificCredentialId": "ACCAEXAMPLE123EXAMPLE",
        "UserName": "sofia",
        "Status": "Active"
    }
}
```

Esempio 2: come reimpostare la password per una credenziale specifica del servizio collegata a un utente specificato

L'esempio `reset-service-specific-credential` seguente genera una nuova password crittograficamente sicura per la credenziale specifica del servizio collegata all'utente specificato.

```
aws iam reset-service-specific-credential \
    --user-name sofia \
    --service-specific-credential-id ACCAEXAMPLE123EXAMPLE
```

Output:

```
{
    "ServiceSpecificCredential": {
        "CreateDate": "2019-04-18T20:45:36+00:00",
        "ServiceName": "codecommit.amazonaws.com",
        "ServiceUserName": "sofia-at-123456789012",
        "ServicePassword": "+oaFsNk7tLco+C/obP9GhhcOzGcKOayTmE3LnAmAmH4=",
        "ServiceSpecificCredentialId": "ACCAEXAMPLE123EXAMPLE",
        "UserName": "sofia",
        "Status": "Active"
    }
}
```

Per ulteriori informazioni, consulta Creare credenziali Git per le connessioni HTTPS CodeCommit nella Guida per l'AWS CodeCommit utente.

- Per i dettagli sull'API, consulta ResetServiceSpecificCredential AWS CLICommand Reference.

`resync-mfa-device`

Il seguente esempio di codice mostra come utilizzare`resync-mfa-device`.

AWS CLI

Come sincronizzare un dispositivo MFA

L'esempio `resync-mfa-device` seguente sincronizza il dispositivo MFA associato all'utente IAM Bob e il cui ARN è `arn:aws:iam::123456789012:mfa/BobsMFADevice` con un programma di autenticazione che ha fornito i due codici di autenticazione.

```
aws iam resync-mfa-device \
    --user-name Bob \
    --serial-number arn:aws:iam::210987654321:mfa/BobsMFADevice \
    --authentication-code1 123456 \
    --authentication-code2 987654
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Utilizzo dell'autenticazione a più fattori (MFA) in AWS nella AWS Guida per l'utente IAM.

- Per i dettagli sull'API, consulta ResyncMfaDevice AWS CLICommand Reference.

`set-default-policy-version`

Il seguente esempio di codice mostra come utilizzare`set-default-policy-version`.

AWS CLI

Come impostare la versione indicata della policy specificata come versione predefinita della policy.

Questo esempio imposta la versione `v2` della policy il cui ARN è `arn:aws:iam::123456789012:policy/MyPolicy` come versione attiva predefinita.

```
aws iam set-default-policy-version \
    --policy-arn arn:aws:iam::123456789012:policy/MyPolicy \
    --version-id v2
```

Per ulteriori informazioni, consulta Policy e autorizzazioni in IAM nella Guida per l'utente di IAM AWS .

- Per i dettagli sull'API, consulta SetDefaultPolicyVersion AWS CLICommand Reference.

`set-security-token-service-preferences`

Il seguente esempio di codice mostra come utilizzare`set-security-token-service-preferences`.

AWS CLI

Come impostare la versione del token dell'endpoint globale

L'esempio `set-security-token-service-preferences` seguente configura Amazon STS in modo che utilizzi i token della versione 2 durante l'autenticazione sull'endpoint globale.

```
aws iam set-security-token-service-preferences \
```

```
    --global-endpoint-token-version v2Token
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Managing AWS STS in an AWS Region nella AWS IAM User Guide.

- Per i dettagli sull'API, consulta SetSecurityTokenServicePreferences AWS CLICommand Reference.

`simulate-custom-policy`

Il seguente esempio di codice mostra come utilizzare`simulate-custom-policy`.

AWS CLI

Esempio 1: come simulare gli effetti di tutte le policy IAM associate a un utente o a un ruolo IAM

Il comando `simulate-custom-policy` seguente mostra come fornire la policy e definire i valori delle variabili e simulare una chiamata API per verificare se viene consentita o rifiutata. L'esempio seguente mostra una policy che consente l'accesso al database solo dopo una data e un'ora specificate. La simulazione ha esito positivo perché le azioni simulate e la variabile `aws:CurrentTime` specificata soddisfano tutte i requisiti della policy.

```
aws iam simulate-custom-policy \
    --policy-input-list '{"Version":"2012-10-17",       "Statement":
{"Effect":"Allow","Action":"dynamodb:*","Resource":"*","Condition":
{"DateGreaterThan":{"aws:CurrentTime":"2018-08-16T12:00:00Z"}}}}' \
    --action-names dynamodb:CreateBackup \
    --context-
entries "ContextKeyName='aws:CurrentTime',ContextKeyValues='2019-04-25T11:00:00Z',ContextKey
```

Output:

```
{
    "EvaluationResults": [
        {
            "EvalActionName": "dynamodb:CreateBackup",
            "EvalResourceName": "*",
            "EvalDecision": "allowed",
            "MatchedStatements": [
```

```
                {
                    "SourcePolicyId": "PolicyInputList.1",
                    "StartPosition": {
                        "Line": 1,
                        "Column": 38
                    },
                    "EndPosition": {
                        "Line": 1,
                        "Column": 167
                    }
                }
            ],
            "MissingContextValues": []
        }
    ]
}
```

Esempio 2: come simulare un comando proibito dalla policy

L'esempio `simulate-custom-policy` seguente mostra i risultati della simulazione di un comando proibito dalla policy. In questo esempio, la data fornita è precedente a quella richiesta dalla condizione della policy.

```
aws iam simulate-custom-policy \
    --policy-input-list '{"Version":"2012-10-17",       "Statement":
{"Effect":"Allow","Action":"dynamodb:*","Resource":"*","Condition":
{"DateGreaterThan":{"aws:CurrentTime":"2018-08-16T12:00:00Z"}}}}' \
    --action-names dynamodb:CreateBackup \
    --context-
entries "ContextKeyName='aws:CurrentTime',ContextKeyValues='2014-04-25T11:00:00Z',ContextKey
```

Output:

```
{
    "EvaluationResults": [
        {
            "EvalActionName": "dynamodb:CreateBackup",
            "EvalResourceName": "*",
            "EvalDecision": "implicitDeny",
            "MatchedStatements": [],
            "MissingContextValues": []
        }
```

```
    ]
}
```

Per ulteriori informazioni, consulta Test delle policy IAM con il simulatore di policy IAM nella Guida per l'utente di AWS IAM.

- Per i dettagli sull'API, consulta SimulateCustomPolicy AWS CLICommand Reference.

`simulate-principal-policy`

Il seguente esempio di codice mostra come utilizzare`simulate-principal-policy`.

AWS CLI

Esempio 1: come simulare gli effetti di una policy IAM arbitraria

L'esempio `simulate-principal-policy` seguente illustra come simulare un utente che chiama un'azione API e determina se le policy associate a tale utente la consentono o la rifiutano. Nell'esempio seguente, l'utente dispone di una policy che consente solo l'azione `codecommit:ListRepositories`.

```
aws iam simulate-principal-policy \
    --policy-source-arn arn:aws:iam::123456789012:user/alejandro \
    --action-names codecommit:ListRepositories
```

Output:

```
{
    "EvaluationResults": [
        {
            "EvalActionName": "codecommit:ListRepositories",
            "EvalResourceName": "*",
            "EvalDecision": "allowed",
            "MatchedStatements": [
                {
                    "SourcePolicyId": "Grant-Access-To-CodeCommit-ListRepo",
                    "StartPosition": {
                        "Line": 3,
                        "Column": 19
                    },
                    "EndPosition": {
```

```
                        "Line": 9,
                        "Column": 10
                    }
                }
            ],
            "MissingContextValues": []
        }
    ]
}
```

Esempio 2: come simulare gli effetti di un comando proibito

L'esempio `simulate-custom-policy` seguente mostra i risultati della simulazione di un comando proibito da una delle policy dell'utente. Nell'esempio seguente, l'utente dispone di una policy che consente l'accesso a un database DynamoDB solo dopo una certa data e ora. La simulazione prevede che l'utente tenti di accedere al database con un valore `aws:CurrentTime` precedente a quello consentito dalle condizioni della policy.

```
aws iam simulate-principal-policy \
    --policy-source-arn arn:aws:iam::123456789012:user/alejandro \
    --action-names dynamodb:CreateBackup \
    --context-
entries "ContextKeyName='aws:CurrentTime',ContextKeyValues='2018-04-25T11:00:00Z',ContextKey
```

Output:

```
{
    "EvaluationResults": [
        {
            "EvalActionName": "dynamodb:CreateBackup",
            "EvalResourceName": "*",
            "EvalDecision": "implicitDeny",
            "MatchedStatements": [],
            "MissingContextValues": []
        }
    ]
}
```

Per ulteriori informazioni, consulta Test delle policy IAM con il simulatore di policy IAM nella Guida per l'utente di AWS IAM.

- Per i dettagli sull'API, consulta SimulatePrincipalPolicy AWS CLICommand Reference.

`tag-instance-profile`

Il seguente esempio di codice mostra come utilizzare`tag-instance-profile`.

AWS CLI

Come aggiungere un tag a un profilo dell'istanza

Il comando `tag-instance-profile` seguente aggiunge un tag con un nome di reparto al profilo dell'istanza specificato.

```
aws iam tag-instance-profile \
    --instance-profile-name deployment-role \
    --tags '[{"Key": "Department", "Value": "Accounting"}]'
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Applicazione di tag a risorse IAM nella Guida per l'utente di AWS IAM.

- Per i dettagli sull'API, consulta TagInstanceProfile AWS CLICommand Reference.

`tag-mfa-device`

Il seguente esempio di codice mostra come utilizzare`tag-mfa-device`.

AWS CLI

Come aggiungere un tag a un dispositivo MFA

Il comando `tag-mfa-device` seguente aggiunge un tag con un nome di reparto al dispositivo MFA specificato.

```
aws iam tag-mfa-device \
    --serial-number arn:aws:iam::123456789012:mfa/alice \
    --tags '[{"Key": "Department", "Value": "Accounting"}]'
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Applicazione di tag a risorse IAM nella Guida per l'utente di AWS IAM.

- Per i dettagli sull'API, consulta TagMfaDevice AWS CLICommand Reference.

`tag-open-id-connect-provider`

Il seguente esempio di codice mostra come utilizzare`tag-open-id-connect-provider`.

AWS CLI

Come aggiungere un tag a un gestore dell'identità compatibile con OpenID Connect (OIDC)

Il comando `tag-open-id-connect-provider` seguente aggiunge un tag con un nome di reparto al gestore dell'identità OIDC specificato.

```
aws iam tag-open-id-connect-provider \
    --open-id-connect-provider-arn arn:aws:iam::123456789012:oidc-provider/
server.example.com \
    --tags '[{"Key": "Department", "Value": "Accounting"}]'
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Applicazione di tag a risorse IAM nella Guida per l'utente di AWS IAM.

- Per i dettagli sull'API, consulta TagOpenIdConnectProvider AWS CLICommand Reference.

`tag-policy`

Il seguente esempio di codice mostra come utilizzare`tag-policy`.

AWS CLI

Come aggiungere un tag a una policy gestita dai clienti

Il comando `tag-policy` seguente aggiunge un tag con un nome di reparto alla policy gestita dal cliente specificato.

```
aws iam tag-policy \
    --policy-arn arn:aws:iam::123456789012:policy/billing-access \
    --tags '[{"Key": "Department", "Value": "Accounting"}]'
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Applicazione di tag a risorse IAM nella Guida per l'utente di AWS IAM.

- Per i dettagli sull'API, consulta TagPolicy AWS CLICommand Reference.

`tag-role`

Il seguente esempio di codice mostra come utilizzare`tag-role`.

AWS CLI

Come aggiungere un tag a un ruolo

Il comando `tag-role` seguente aggiunge un tag con un nome di reparto al ruolo specificato.

```
aws iam tag-role --role-name my-role \
    --tags '{"Key": "Department", "Value": "Accounting"}'
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Applicazione di tag a risorse IAM nella Guida per l'utente di AWS IAM.

- Per i dettagli sull'API, consulta TagRole AWS CLICommand Reference.

`tag-saml-provider`

Il seguente esempio di codice mostra come utilizzare`tag-saml-provider`.

AWS CLI

Come aggiungere un tag a un provider SAML

Il comando `tag-saml-provider` seguente aggiunge un tag con un nome di reparto al provider SAML specificato.

```
aws iam tag-saml-provider \
    --saml-provider-arn arn:aws:iam::123456789012:saml-provider/ADFS \
    --tags '[{"Key": "Department", "Value": "Accounting"}]'
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Applicazione di tag a risorse IAM nella Guida per l'utente di AWS IAM.

- Per i dettagli sull'API, consulta TagSamlProvider AWS CLICommand Reference.

`tag-server-certificate`

Il seguente esempio di codice mostra come utilizzare`tag-server-certificate`.

AWS CLI

Come aggiungere tag a un certificato server

Il comando `tag-saml-provider` seguente aggiunge un tag con un nome di reparto al certificato server specificato.

```
aws iam tag-server-certificate \
    --server-certificate-name ExampleCertificate \
    --tags '[{"Key": "Department", "Value": "Accounting"}]'
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Applicazione di tag a risorse IAM nella Guida per l'utente di AWS IAM.

- Per i dettagli sull'API, consulta TagServerCertificate AWS CLICommand Reference.

`tag-user`

Il seguente esempio di codice mostra come utilizzare`tag-user`.

AWS CLI

Come aggiungere un tag a un utente

Il comando `tag-user` seguente aggiunge un tag con il reparto associato all'utente specificato.

```
aws iam tag-user \
    --user-name alice \
    --tags '{"Key": "Department", "Value": "Accounting"}'
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Applicazione di tag a risorse IAM nella Guida per l'utente di AWS IAM.

- Per i dettagli sull'API, consulta TagUser AWS CLICommand Reference.

`untag-instance-profile`

Il seguente esempio di codice mostra come utilizzare`untag-instance-profile`.

AWS CLI

Come rimuovere un tag da un profilo dell'istanza

Il comando `untag-instance-profile` seguente rimuove qualsiasi tag con il nome di chiave "Department" dal profilo dell'istanza specificato.

```
aws iam untag-instance-profile \
    --instance-profile-name deployment-role \
    --tag-keys Department
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Applicazione di tag a risorse IAM nella Guida per l'utente di AWS IAM.

- Per i dettagli sull'API, consulta UntagInstanceProfile AWS CLICommand Reference.

`untag-mfa-device`

Il seguente esempio di codice mostra come utilizzare`untag-mfa-device`.

AWS CLI

Come rimuovere un tag da un dispositivo MFA

Il comando `untag-mfa-device` seguente rimuove qualsiasi tag con il nome di chiave "Department" dal dispositivo MFA specificato.

```
aws iam untag-mfa-device \
    --serial-number arn:aws:iam::123456789012:mfa/alice \
    --tag-keys Department
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Applicazione di tag a risorse IAM nella Guida per l'utente di AWS IAM.

- Per i dettagli sull'API, consulta UntagMfaDevice AWS CLICommand Reference.

`untag-open-id-connect-provider`

Il seguente esempio di codice mostra come utilizzare`untag-open-id-connect-provider`.

AWS CLI

Come rimuovere i tag da un gestore dell'identità OIDC

Il comando `untag-open-id-connect-provider` seguente rimuove qualsiasi tag con il nome di chiave "Department" dal provider dell'identità OIDC specificato.

```
aws iam untag-open-id-connect-provider \
    --open-id-connect-provider-arn arn:aws:iam::123456789012:oidc-provider/
server.example.com \
    --tag-keys Department
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Applicazione di tag a risorse IAM nella Guida per l'utente di AWS IAM.

- Per i dettagli sull'API, consulta UntagOpenIdConnectProvider AWS CLICommand Reference.

`untag-policy`

Il seguente esempio di codice mostra come utilizzare`untag-policy`.

AWS CLI

Come rimuovere un tag da una policy gestita dai clienti

Il comando `untag-policy` seguente rimuove qualsiasi tag con il nome di chiave "Department" dalla policy gestita dal cliente specificata.

```
aws iam untag-policy \
    --policy-arn arn:aws:iam::452925170507:policy/billing-access \
    --tag-keys Department
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Applicazione di tag a risorse IAM nella Guida per l'utente di AWS IAM.

- Per i dettagli sull'API, consulta UntagPolicy AWS CLICommand Reference.

`untag-role`

Il seguente esempio di codice mostra come utilizzare`untag-role`.

AWS CLI

Come rimuovere un tag da un ruolo

Il comando `untag-role` seguente rimuove qualsiasi tag con il nome chiave 'Department' dal ruolo specificato.

```
aws iam untag-role \
    --role-name my-role \
    --tag-keys Department
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Applicazione di tag a risorse IAM nella Guida per l'utente di AWS IAM.

- Per i dettagli sull'API, consulta UntagRole AWS CLICommand Reference.

`untag-saml-provider`

Il seguente esempio di codice mostra come utilizzare`untag-saml-provider`.

AWS CLI

Come rimuovere un tag da un provider SAML

Il comando `untag-saml-provider` seguente rimuove qualsiasi tag con il nome di chiave "Department" dal profilo dell'istanza specificato.

```
aws iam untag-saml-provider \
    --saml-provider-arn arn:aws:iam::123456789012:saml-provider/ADFS \
```

```
    --tag-keys Department
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Applicazione di tag a risorse IAM nella Guida per l'utente di AWS IAM.

- Per i dettagli sull'API, consulta UntagSamlProvider AWS CLICommand Reference.

`untag-server-certificate`

Il seguente esempio di codice mostra come utilizzare`untag-server-certificate`.

AWS CLI

Come rimuovere un tag da un certificato server

Il comando `untag-server-certificate` seguente rimuove qualsiasi tag con il nome di chiave "Department" dal certificato serve specificato.

```
aws iam untag-server-certificate \
    --server-certificate-name ExampleCertificate \
    --tag-keys Department
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Applicazione di tag a risorse IAM nella Guida per l'utente di AWS IAM.

- Per i dettagli sull'API, consulta UntagServerCertificate AWS CLICommand Reference.

`untag-user`

Il seguente esempio di codice mostra come utilizzare`untag-user`.

AWS CLI

Come rimuovere un tag da un utente

Il comando `untag-user` seguente rimuove qualsiasi tag con il nome chiave 'Department' dall'utente specificato.

```
aws iam untag-user \
    --user-name alice \
    --tag-keys Department
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Applicazione di tag a risorse IAM nella Guida per l'utente di AWS IAM.

- Per i dettagli sull'API, consulta UntagUser AWS CLICommand Reference.

update-access-key

Il seguente esempio di codice mostra come utilizzareupdate-access-key.

AWS CLI

Come attivare o disattivare una chiave di accesso per un utente IAM

Il comando update-access-key seguente disattiva la chiave di accesso specificata (ID chiave di accesso e chiave di accesso segreta) per l'utente IAM denominato Bob.

```
aws iam update-access-key \
    --access-key-id AKIAIOSFODNN7EXAMPLE \
    --status Inactive \
    --user-name Bob
```

Questo comando non produce alcun output.

La disattivazione della chiave significa che non può essere utilizzata per l'accesso programmatico a. AWS La chiave, tuttavia, rimane disponibile e può essere riattivata.

Per ulteriori informazioni, consulta Gestione delle chiavi di accesso per gli utenti IAM nella Guida per l'utente di IAM AWS .

- Per i dettagli sull'API, consulta Command UpdateAccessKeyReference AWS CLI .

update-account-password-policy

Il seguente esempio di codice mostra come utilizzareupdate-account-password-policy.

AWS CLI

Come impostare o modificare la policy delle password dell'account corrente

Il comando `update-account-password-policy` seguente imposta la policy delle password in modo che richieda una lunghezza minima di otto caratteri e uno o più numeri nella password.

```
aws iam update-account-password-policy \
    --minimum-password-length 8 \
    --require-numbers
```

Questo comando non produce alcun output.

Le modifiche alla policy delle password di un account influiscono su tutte le nuove password create per gli utenti IAM nell'account. Le modifiche alle policy delle password non influiscono sulle password esistenti.

Per ulteriori informazioni, consulta Impostazione di una policy delle password dell'account per utenti IAM nella Guida per l'utente di IAM AWS .

- Per i dettagli sull'API, consulta UpdateAccountPasswordPolicy AWS CLICommand Reference.

`update-assume-role-policy`

Il seguente esempio di codice mostra come utilizzare`update-assume-role-policy`.

AWS CLI

Come aggiornare la policy di attendibilità di un ruolo IAM

Il comando `update-assume-role-policy` seguente aggiorna la policy di attendibilità per il ruolo denominato `Test-Role`.

```
aws iam update-assume-role-policy \
    --role-name Test-Role \
    --policy-document file://Test-Role-Trust-Policy.json
```

Questo comando non produce alcun output.

La policy di attendibilità è definita come documento JSON nel file Test-Role-Trust-Policy.json. (Il nome e l'estensione del file non hanno importanza.) La policy di attendibilità deve specificare un principale.

Per aggiornare una policy di autorizzazioni per un ruolo, usa il comando `put-role-policy`.

Per ulteriori informazioni, consulta Creazione di ruoli IAM nella Guida per l'utente IAM AWS .

- Per i dettagli sull'API, consulta UpdateAssumeRolePolicy AWS CLICommand Reference.

`update-group`

Il seguente esempio di codice mostra come utilizzare`update-group`.

AWS CLI

Come rinominare un gruppo IAM

Il comando `update-group` seguente modifica il nome del gruppo IAM da `Test` a `Test-1`.

```
aws iam update-group \
    --group-name Test \
    --new-group-name Test-1
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Ridenominazione di un gruppo di utenti IAM nella Guida per l'utente IAM AWS .

- Per i dettagli sull'API, consulta UpdateGroup AWS CLICommand Reference.

`update-login-profile`

Il seguente esempio di codice mostra come utilizzare`update-login-profile`.

AWS CLI

Come aggiornare la password per un utente IAM

Il comando `update-login-profile` seguente crea una nuova password per l'utente IAM denominato Bob.

```
aws iam update-login-profile \
    --user-name Bob \
    --password <password>
```

Questo comando non produce alcun output.

Per impostare una policy delle password per l'account, usa il comando `update-account-password-policy`. Se la nuova password viola la policy delle password dell'account, il comando restituisce un errore `PasswordPolicyViolation`.

Se la policy delle password dell'account lo consente, gli utenti IAM possono modificare le proprie password utilizzando il comando `change-password`.

Conserva la nuova password in un luogo sicuro. Se la password viene persa, non può essere recuperata e dovrai crearne una nuova utilizzando il comando `create-login-profile`.

Per ulteriori informazioni, consulta Gestione delle password per gli utenti IAM nella Guida per l'utente di AWS .

- Per i dettagli sull'API, consulta UpdateLoginProfile AWS CLICommand Reference.

`update-open-id-connect-provider-thumbprint`

Il seguente esempio di codice mostra come utilizzare`update-open-id-connect-provider-thumbprint`.

AWS CLI

Come sostituire l'elenco esistente di impronte digitali dei certificati server con un nuovo elenco

Questo esempio aggiorna l'elenco delle impronte digitali dei certificati per il provider OIDC il cui ARN è `arn:aws:iam::123456789012:oidc-provider/example.oidcprovider.com` per utilizzare una nuova impronta digitale.

```
aws iam update-open-id-connect-provider-thumbprint \
    --open-id-connect-provider-arn arn:aws:iam::123456789012:oidc-provider/
example.oidcprovider.com \
    --thumbprint-list 7359755EXAMPLEabc3060bce3EXAMPLEec4542a3
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Creazione di provider di identità OpenID Connect (OIDC) nella Guida per l'utente di AWS IAM.

- Per i dettagli sull'API, consulta UpdateOpenIdConnectProviderThumbprint AWS CLICommand Reference.

`update-role-description`

Il seguente esempio di codice mostra come utilizzareu`pdate-role-description`.

AWS CLI

Come modificare la descrizione di un ruolo IAM

Il comando `update-role` seguente modifica la descrizione del ruolo IAM da `production-role` a `Main production role`.

```
aws iam update-role-description \
    --role-name production-role \
    --description 'Main production role'
```

Output:

```
{
    "Role": {
        "Path": "/",
        "RoleName": "production-role",
        "RoleId": "AROA1234567890EXAMPLE",
        "Arn": "arn:aws:iam::123456789012:role/production-role",
        "CreateDate": "2017-12-06T17:16:37+00:00",
        "AssumeRolePolicyDocument": {
            "Version":"2012-10-17",
            "Statement": [
                {
                    "Effect": "Allow",
                    "Principal": {
                        "AWS": "arn:aws:iam::123456789012:root"
                    },
                    "Action": "sts:AssumeRole",
                    "Condition": {}
                }
            ]
        },
        "Description": "Main production role"
    }
}
```

Per ulteriori informazioni, consulta Modifica di un ruolo nella Guida per l'utente di AWS IAM.

- Per i dettagli sull'API, consulta UpdateRoleDescription AWS CLICommand Reference.

`update-role`

Il seguente esempio di codice mostra come utilizzare`update-role`.

AWS CLI

Come modificare la descrizione o la durata della sessione di un ruolo IAM

Il comando `update-role` seguente modifica la descrizione del ruolo IAM `production-role` in `Main production role` e imposta la durata massima della sessione su 12 ore.

```
aws iam update-role \
    --role-name production-role \
    --description 'Main production role' \
    --max-session-duration 43200
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Modifica di un ruolo nella Guida per l'utente di AWS IAM.

- Per i dettagli sull'API, consulta UpdateRole AWS CLICommand Reference.

`update-saml-provider`

Il seguente esempio di codice mostra come utilizzare`update-saml-provider`.

AWS CLI

Come aggiornare il documento di metadati per un provider SAML esistente

Questo esempio aggiorna il provider SAML in IAM il cui ARN è `arn:aws:iam::123456789012:saml-provider/SAMLADFS` con un nuovo documento di metadati SAML dal file `SAMLMetaData.xml`.

```
aws iam update-saml-provider \
    --saml-metadata-document file://SAMLMetaData.xml \
    --saml-provider-arn arn:aws:iam::123456789012:saml-provider/SAMLADFS
```

Output:

```
{
    "SAMLProviderArn": "arn:aws:iam::123456789012:saml-provider/SAMLADFS"
}
```

Per ulteriori informazioni, consulta Creazione di provider di identità SAML IAM nella Guida per l'utente di IAM AWS .

- Per i dettagli sull'API, consulta UpdateSamlProvider AWS CLICommand Reference.

`update-server-certificate`

Il seguente esempio di codice mostra come utilizzare`update-server-certificate`.

AWS CLI

Per modificare il percorso o il nome di un certificato del server nel tuo AWS account

Il comando `update-server-certificate` seguente modifica il nome del certificato da `myServerCertificate` a `myUpdatedServerCertificate`. Cambia anche il percorso `/cloudfront/` in modo che sia accessibile dal CloudFront servizio Amazon. Questo comando non produce alcun output. Puoi visualizzare i risultati dell'aggiornamento eseguendo il comando `list-server-certificates`.

```
aws-iam update-server-certificate \
    --server-certificate-name myServerCertificate \
    --new-server-certificate-name myUpdatedServerCertificate \
    --new-path /cloudfront/
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Gestione dei certificati server in IAM nella Guida per l'utente IAM AWS .

- Per i dettagli sull'API, consulta UpdateServerCertificate AWS CLICommand Reference.

`update-service-specific-credential`

Il seguente esempio di codice mostra come utilizzare`update-service-specific-credential`.

AWS CLI

Esempio 1: come aggiornare lo stato della credenziale specifica del servizio dell'utente richiedente

L'esempio `update-service-specific-credential` seguente modifica lo stato della credenziale specificata per l'utente a cui viene effettuata la richiesta `Inactive`.

```
aws iam update-service-specific-credential \
    --service-specific-credential-id ACCAEXAMPLE123EXAMPLE \
    --status Inactive
```

Questo comando non produce alcun output.

Esempio 2: come aggiornare lo stato delle credenziali specifiche del servizio di un utente specificato

L'esempio `update-service-specific-credential` seguente modica in Inattivo lo stato della credenziale dell'utente specificato.

```
aws iam update-service-specific-credential \
    --user-name sofia \
    --service-specific-credential-id ACCAEXAMPLE123EXAMPLE \
    --status Inactive
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Creare credenziali Git per connessioni HTTPS CodeCommit nella Guida per l'AWS CodeCommit utente

- Per i dettagli sull'API, consulta UpdateServiceSpecificCredential AWS CLICommand Reference.

`update-signing-certificate`

Il seguente esempio di codice mostra come utilizzare`update-signing-certificate`.

AWS CLI

Come attivare o disattivare un certificato di firma per un utente IAM

Il comando `update-signing-certificate` seguente disattiva il certificato di firma specificato per l'utente IAM denominato Bob.

```
aws iam update-signing-certificate \
    --certificate-id TA7SMP42TDN5Z26OBPJE7EXAMPLE \
    --status Inactive \
```

```
    --user-name Bob
```

Per ottenere l'ID per un certificato di firma, utilizza il comando `list-signing-certificates`.

Per ulteriori informazioni, consulta Gestire i certificati di firma nella Amazon EC2 User Guide.

- Per i dettagli sull'API, consulta UpdateSigningCertificate AWS CLICommand Reference.

`update-ssh-public-key`

Il seguente esempio di codice mostra come utilizzare`update-ssh-public-key`.

AWS CLI

Come modificare lo stato di una chiave pubblica SSH

Il comando `update-ssh-public-key` seguente modifica lo stato della chiave pubblica specificata in `Inactive`.

```
aws iam update-ssh-public-key \
    --user-name sofia \
    --ssh-public-key-id APKA1234567890EXAMPLE \
    --status Inactive
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Use SSH keys and SSH with CodeCommit nella AWS IAM User Guide.

- Per i dettagli sull'API, consulta AWS CLI Command UpdateSshPublicKeyReference.

`update-user`

Il seguente esempio di codice mostra come utilizzare`update-user`.

AWS CLI

Come modificare il nome di un utente IAM

Il comando `update-user` seguente modifica il nome di un utente IAM da `Bob` a `Robert`.

```
aws iam update-user \
    --user-name Bob \
```

```
    --new-user-name Robert
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Ridenominazione di un gruppo di utenti IAM nella Guida per l'utente IAM AWS .

- Per i dettagli sull'API, consulta UpdateUser AWS CLICommand Reference.

`upload-server-certificate`

Il seguente esempio di codice mostra come utilizzare`upload-server-certificate`.

AWS CLI

Per caricare un certificato server sul tuo AWS account

Il upload-server-certificatecomando seguente carica un certificato server sul tuo AWS account. In questo esempio, il certificato è nel file `public_key_cert_file.pem`, la chiave privata associata è nel file `my_private_key.pem` e la catena di certificati fornita dall'autorità di certificazione (CA) è nel file `my_certificate_chain_file.pem`. Al termine del caricamento, il file è disponibile sotto il nome. myServerCertificate I parametri che iniziano con `file://` indicano al comando di leggere il contenuto del file e di utilizzarlo come valore del parametro in luogo del nome del file.

```
aws iam upload-server-certificate \
    --server-certificate-name myServerCertificate \
    --certificate-body file://public_key_cert_file.pem \
    --private-key file://my_private_key.pem \
    --certificate-chain file://my_certificate_chain_file.pem
```

Output:

```
{
    "ServerCertificateMetadata": {
        "Path": "/",
        "ServerCertificateName": "myServerCertificate",
        "ServerCertificateId": "ASCAEXAMPLE123EXAMPLE",
        "Arn": "arn:aws:iam::1234567989012:server-certificate/myServerCertificate",
        "UploadDate": "2019-04-22T21:13:44+00:00",
        "Expiration": "2019-10-15T22:23:16+00:00"
```

```
    }
}
```

Per ulteriori informazioni, consulta Creazione, caricamento ed eliminazione di certificati server nella guida Utilizzo di IAM.

- Per i dettagli sull'API, consulta UploadServerCertificate AWS CLICommand Reference.

`upload-signing-certificate`

Il seguente esempio di codice mostra come utilizzare`upload-signing-certificate`.

AWS CLI

Come caricare un certificato di firma per un utente IAM

Il comando `upload-signing-certificate` seguente carica il certificato di firma per l'utente IAM denominato Bob.

```
aws iam upload-signing-certificate \
    --user-name Bob \
    --certificate-body file://certificate.pem
```

Output:

```
{
    "Certificate": {
        "UserName": "Bob",
        "Status": "Active",
        "CertificateBody": "-----BEGIN CERTIFICATE-----<certificate-body>-----END
 CERTIFICATE-----",
        "CertificateId": "TA7SMP42TDN5Z26OBPJE7EXAMPLE",
        "UploadDate": "2013-06-06T21:40:08.121Z"
    }
}
```

Il certificato si trova in un file denominato certificate.pem in formato PEM.

Per ulteriori informazioni, consulta Creazione e caricamento di un certificato di firma dell'utente nella guida Utilizzo di IAM.

- Per i dettagli sull'API, consulta UploadSigningCertificate AWS CLICommand Reference.

`upload-ssh-public-key`

Il seguente esempio di codice mostra come utilizzare`upload-ssh-public-key`.

AWS CLI

Come caricare una chiave pubblica SSH e associarla a un utente

Il comando `upload-ssh-public-key` seguente carica la chiave pubblica trovata nel file `sshkey.pub` e la collega all'utente `sofia`.

```
aws iam upload-ssh-public-key \
    --user-name sofia \
    --ssh-public-key-body file://sshkey.pub
```

Output:

```
{
    "SSHPublicKey": {
        "UserName": "sofia",
        "SSHPublicKeyId": "APKA1234567890EXAMPLE",
        "Fingerprint": "12:34:56:78:90:ab:cd:ef:12:34:56:78:90:ab:cd:ef",
        "SSHPublicKeyBody": "ssh-rsa <<long string generated by ssh-keygen
 command>>",
        "Status": "Active",
        "UploadDate": "2019-04-18T17:04:49+00:00"
    }
}
```

Per ulteriori informazioni su come generare chiavi in un formato adatto a questo comando, vedere SSH e Linux, macOS o Unix: configurare le chiavi pubbliche e private per Git e/o SSH CodeCommit e Windows: configurare le chiavi pubbliche e private per Git CodeCommit e nella Guida per l'utente.AWS CodeCommit

- Per i dettagli sull'API, consulta UploadSshPublicKeyCommand Reference.AWS CLI

Esempi di IAM Access Analyzer con AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando IAM Access Analyzer. AWS Command Line Interface

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`apply-archive-rule`

Il seguente esempio di codice mostra come utilizzare`apply-archive-rule`.

AWS CLI

Come applicare una regola di archiviazione agli esiti che soddisfano i criteri della regola

L'esempio `apply-archive-rule` seguente applica una regola di archiviazione agli esiti che soddisfano i criteri della regola.

```
aws accessanalyzer apply-archive-rule \
    --analyzer-arn arn:aws:access-analyzer:us-west-2:111122223333:analyzer/
UnusedAccess-ConsoleAnalyzer-organization \
    --rule-name MyArchiveRule
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Regole di archiviazione nella Guida per l'utente di AWS IAM.

- Per i dettagli sull'API, consulta ApplyArchiveRule AWS CLICommand Reference.

`cancel-policy-generation`

Il seguente esempio di codice mostra come utilizzare`cancel-policy-generation`.

AWS CLI

Come annullare una generazione di policy richiesta

L'esempio `cancel-policy-generation` seguente annulla l'id del processo di generazione della policy richiesto.

```
aws accessanalyzer cancel-policy-generation \
    --job-id 923a56b0-ebb8-4e80-8a3c-a11ccfbcd6f2
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Generazione delle policy per IAM Access Analyzer nella Guida per l'utente di AWS IAM.

- Per i dettagli sull'API, consulta CancelPolicyGeneration AWS CLICommand Reference.

`check-access-not-granted`

Il seguente esempio di codice mostra come utilizzare`check-access-not-granted`.

AWS CLI

Come verificare se l'accesso specificato non è consentito da una policy

L'esempio `check-access-not-granted` seguente verifica se l'accesso specificato non è consentito da una policy.

```
aws accessanalyzer check-access-not-granted \
    --policy-document file://myfile.json \
    --access actions="s3:DeleteBucket","s3:GetBucketLocation" \
    --policy-type IDENTITY_POLICY
```

Contenuto di `myfile.json`:

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "s3:GetObject",
                "s3:ListBucket"
            ],
            "Resource": [
                "arn:aws:s3:::amzn-s3-demo-bucket",
```

```
                "arn:aws:s3:::amzn-s3-demo-bucket/*"
            ]
        }
    ]
}
```

Output:

```
{
    "result": "PASS",
    "message": "The policy document does not grant access to perform one or more of
 the listed actions."
}
```

Per ulteriori informazioni, consulta Anteprima dell'accesso con IAM Access Analyzer APIs nella Guida per l'utente AWS IAM.

- Per i dettagli sull'API, consulta AWS CLI Command CheckAccessNotGrantedReference.

`check-no-new-access`

Il seguente esempio di codice mostra come utilizzare`check-no-new-access`.

AWS CLI

Come verificare se è consentito un nuovo accesso per una policy aggiornata rispetto alla policy esistente

L'esempio `check-no-new-access` seguente verifica se è consentito un nuovo accesso per una policy aggiornata rispetto alla policy esistente.

```
aws accessanalyzer check-no-new-access \
    --existing-policy-document file://existing-policy.json \
    --new-policy-document file://new-policy.json \
    --policy-type IDENTITY_POLICY
```

Contenuto di `existing-policy.json`:

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
```

```
            "Effect": "Allow",
            "Action": [
                "s3:GetObject",
                "s3:ListBucket"
            ],
            "Resource": [
                "arn:aws:s3:::amzn-s3-demo-bucket",
                "arn:aws:s3:::amzn-s3-demo-bucket/*"
            ]
        }
    ]
}
```

Contenuto di `new-policy.json`:

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "s3:GetObject",
                "s3:GetObjectAcl",
                "s3:ListBucket"
            ],
            "Resource": [
                "arn:aws:s3:::amzn-s3-demo-bucket",
                "arn:aws:s3:::amzn-s3-demo-bucket/*"
            ]
        }
    ]
}
```

Output:

```
{
    "result": "FAIL",
    "message": "The modified permissions grant new access compared to your existing
 policy.",
    "reasons": [
        {
            "description": "New access in the statement with index: 0.",
            "statementIndex": 0
```

```
        }
    ]
}
```

Per ulteriori informazioni, consulta Anteprima dell'accesso con IAM Access Analyzer APIs nella Guida per l'utente AWS IAM.

- Per i dettagli sull'API, consulta AWS CLI Command CheckNoNewAccessReference.

`check-no-public-access`

Il seguente esempio di codice mostra come utilizzare`check-no-public-access`.

AWS CLI

Come verificare se una policy delle risorse può concedere l'accesso pubblico al tipo di risorsa specificato

L'esempio `check-no-public-access` seguente verifica se una policy delle risorse può concedere l'accesso pubblico a un tipo di risorsa specificato.

```
aws accessanalyzer check-no-public-access \
    --policy-document file://check-no-public-access-myfile.json \
    --resource-type AWS::S3::Bucket
```

Contenuto di `myfile.json`:

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "CheckNoPublicAccess",
            "Effect": "Allow",
            "Principal": { "AWS": "arn:aws:iam::111122223333:user/JohnDoe" },
            "Action": [
                "s3:GetObject"
            ]
        }
    ]
}
```

Output:

```
{
    "result": "PASS",
    "message": "The resource policy does not grant public access for the given
 resource type."
}
```

Per ulteriori informazioni, consulta Anteprima dell'accesso con IAM Access Analyzer APIs nella Guida per l'utente AWS IAM.

- Per i dettagli sull'API, consulta AWS CLI Command CheckNoPublicAccessReference.

`create-access-preview`

Il seguente esempio di codice mostra come utilizzare`create-access-preview`.

AWS CLI

Come creare un'anteprima di accesso che consenta di visualizzare in anteprima gli esiti dello strumento di analisi degli accessi IAM relativi alla risorsa corrente prima di distribuire le autorizzazioni delle risorse

L'`create-access-preview`esempio seguente crea un'anteprima di accesso che ti consente di visualizzare in anteprima i risultati di IAM Access Analyzer relativi alla tua risorsa prima di distribuire le autorizzazioni delle risorse nel tuo account. AWS

```
aws accessanalyzer create-access-preview \
    --analyzer-arn arn:aws:access-analyzer:us-west-2:111122223333:analyzer/
ConsoleAnalyzer-account \
    --configurations file://myfile.json
```

Contenuto di `myfile.json`:

```
{
    "arn:aws:s3:::amzn-s3-demo-bucket": {
        "s3Bucket": {
            "bucketPolicy": "{\"Version\":\"2012-10-17\",\"Statement\":[{\"Effect
\":\"Allow\",\"Principal\":{\"AWS\":[\"arn:aws:iam::111122223333:root\"]},\"Action
\":[\"s3:PutObject\",\"s3:PutObjectAcl\"],\"Resource\":\"arn:aws:s3:::amzn-s3-demo-
bucket/*\"}]}",
            "bucketPublicAccessBlock": {
                "ignorePublicAcls": true,
                "restrictPublicBuckets": true
```

```
            },
            "bucketAclGrants": [
                {
                    "grantee": {
                        "id":
 "79a59df900b949e55d96a1e698fbacedfd6e09d98eacf8f8d5218e7cd47ef2be"
                    },
                    "permission": "READ"
                }
            ]
        }
    }
}
```

Output:

```
{
    "id": "3c65eb13-6ef9-4629-8919-a32043619e6b"
}
```

Per ulteriori informazioni, consulta Anteprima dell'accesso con IAM Access Analyzer APIs nella Guida per l'utente IAM.AWS

- Per i dettagli sull'API, consulta AWS CLI Command CreateAccessPreviewReference.

`create-analyzer`

Il seguente esempio di codice mostra come utilizzare`create-analyzer`.

AWS CLI

Come creare un analizzatore

L'`create-analyzer`esempio seguente crea un analizzatore nel tuo AWS account.

```
aws accessanalyzer create-analyzer \
    --analyzer-name example \
    --type ACCOUNT
```

Output:

```
{
```

```
    "arn": "arn:aws:access-analyzer:us-east-2:111122223333:analyzer/example"
}
```

Per ulteriori informazioni, consulta i risultati di Getting started with AWS Identity and Access Management Access Analyzer nella AWS IAM User Guide.

- Per i dettagli sull'API, consulta CreateAnalyzer AWS CLICommand Reference.

`create-archive-rule`

Il seguente esempio di codice mostra come utilizzare`create-archive-rule`.

AWS CLI

Come creare una regola di archiviazione per l'analizzatore specificato

L'`create-archive-rule`esempio seguente crea una regola di archiviazione per l'analizzatore specificato nell' AWS account.

```
aws accessanalyzer create-archive-rule \
    --analyzer-name UnusedAccess-ConsoleAnalyzer-organization \
    --rule-name MyRule \
    --filter '{"resource": {"contains": ["Cognito"]}, "resourceType": {"eq":
 ["AWS::IAM::Role"]}}'
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Regole di archiviazione nella Guida per l'utente di AWS IAM.

- Per i dettagli sull'API, consulta CreateArchiveRule AWS CLICommand Reference.

`delete-analyzer`

Il seguente esempio di codice mostra come utilizzare`delete-analyzer`.

AWS CLI

Come eliminare l'analizzatore specificato

L'`delete-analyzer`esempio seguente elimina l'analizzatore specificato nell'account AWS .

```
aws accessanalyzer delete-analyzer \
    --analyzer-name example
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Regole di archiviazione nella Guida per l'utente di AWS IAM.

- Per i dettagli sull'API, consulta AWS CLI Command DeleteAnalyzerReference.

`delete-archive-rule`

Il seguente esempio di codice mostra come utilizzare`delete-archive-rule`.

AWS CLI

Come eliminare la regola di archiviazione specificata

L'`delete-archive-rule`esempio seguente elimina la regola di archiviazione specificata nell' AWS account.

```
aws accessanalyzer delete-archive-rule \
    --analyzer-name UnusedAccess-ConsoleAnalyzer-organization \
    --rule-name MyRule
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Regole di archiviazione nella Guida per l'utente di AWS IAM.

- Per i dettagli sull'API, consulta DeleteArchiveRule AWS CLICommand Reference.

`get-access-preview`

Il seguente esempio di codice mostra come utilizzare`get-access-preview`.

AWS CLI

Come recuperare informazioni sull'anteprima di accesso per l'analizzatore specificato

L'`get-access-preview`esempio seguente recupera le informazioni su un'anteprima di accesso per l'analizzatore specificato nell'account AWS .

```
aws accessanalyzer get-access-preview \
    --access-preview-id 3c65eb13-6ef9-4629-8919-a32043619e6b \
    --analyzer-arn arn:aws:access-analyzer:us-west-2:111122223333:analyzer/
ConsoleAnalyzer-account
```

Output:

```
{
    "accessPreview": {
        "id": "3c65eb13-6ef9-4629-8919-a32043619e6b",
        "analyzerArn": "arn:aws:access-analyzer:us-west-2:111122223333:analyzer/
ConsoleAnalyzer-account",
        "configurations": {
            "arn:aws:s3:::amzn-s3-demo-bucket": {
                "s3Bucket": {
                    "bucketPolicy": "{\"Version\":\"2012-10-17\",\"Statement\":
[{\"Effect\":\"Allow\",\"Principal\":{\"AWS\":[\"arn:aws:iam::111122223333:root\"]},
\"Action\":[\"s3:PutObject\",\"s3:PutObjectAcl\"],\"Resource\":\"arn:aws:s3:::amzn-
s3-demo-bucket/*\"}]}",
                    "bucketAclGrants": [
                        {
                            "permission": "READ",
                            "grantee": {
                                "id":
 "79a59df900b949e55d96a1e698fbacedfd6e09d98eacf8f8d5218e7cd47ef2be"
                            }
                        }
                    ],
                    "bucketPublicAccessBlock": {
                        "ignorePublicAcls": true,
                        "restrictPublicBuckets": true
                    }
                }
            }
        },
        "createdAt": "2024-02-17T00:18:44+00:00",
        "status": "COMPLETED"
    }
}
```

Per ulteriori informazioni, consulta Anteprima dell'accesso con IAM Access Analyzer APIs nella IAM User Guide.AWS

- Per i dettagli sull'API, consulta AWS CLI Command GetAccessPreviewReference.

`get-analyzed-resource`

Il seguente esempio di codice mostra come utilizzare`get-analyzed-resource`.

AWS CLI

Come recuperare informazioni su una risorsa analizzata

L'`get-analyzed-resource`esempio seguente recupera informazioni su una risorsa che è stata analizzata nel tuo AWS account.

```
aws accessanalyzer get-analyzed-resource \
    --analyzer-arn arn:aws:access-analyzer:us-west-2:111122223333:analyzer/
ConsoleAnalyzer-account \
    --resource-arn arn:aws:s3:::amzn-s3-demo-bucket
```

Output:

```
{
    "resource": {
        "analyzedAt": "2024-02-15T18:01:53.002000+00:00",
        "isPublic": false,
        "resourceArn": "arn:aws:s3:::amzn-s3-demo-bucket",
        "resourceOwnerAccount": "111122223333",
        "resourceType": "AWS::S3::Bucket"
    }
}
```

Per ulteriori informazioni, consulta Using AWS Identity and Access Management Access Analyzer nella AWS IAM User Guide.

- Per i dettagli sull'API, consulta GetAnalyzedResource AWS CLICommand Reference.

`get-analyzer`

Il seguente esempio di codice mostra come utilizzare`get-analyzer`.

AWS CLI

Come recuperare informazioni sull'analizzatore specificato

L'`get-analyzer`esempio seguente recupera le informazioni sull'analizzatore specificato nell'account AWS .

```
aws accessanalyzer get-analyzer \
    --analyzer-name ConsoleAnalyzer-account
```

Output:

```
{
    "analyzer": {
        "arn": "arn:aws:access-analyzer:us-west-2:111122223333:analyzer/
ConsoleAnalyzer-account",
        "createdAt": "2019-12-03T07:28:17+00:00",
        "lastResourceAnalyzed": "arn:aws:sns:us-west-2:111122223333:config-topic",
        "lastResourceAnalyzedAt": "2024-02-15T18:01:53.003000+00:00",
        "name": "ConsoleAnalyzer-account",
        "status": "ACTIVE",
        "tags": {
            "auto-delete": "no"
        },
        "type": "ACCOUNT"
    }
}
```

Per ulteriori informazioni, consulta Using AWS Identity and Access Management Access Analyzer nella AWS IAM User Guide.

- Per i dettagli sull'API, consulta GetAnalyzer AWS CLICommand Reference.

`get-archive-rule`

Il seguente esempio di codice mostra come utilizzare`get-archive-rule`.

AWS CLI

Come recuperare informazioni su una regola di archiviazione

L'`get-archive-rule`esempio seguente recupera informazioni su una regola di archiviazione nel tuo AWS account.

```
aws accessanalyzer get-archive-rule \
    --analyzer-name UnusedAccess-ConsoleAnalyzer-organization \
    --rule-name MyArchiveRule
```

Output:

```
{
```

```
    "archiveRule": {
        "createdAt": "2024-02-15T00:49:27+00:00",
        "filter": {
            "resource": {
                "contains": [
                    "Cognito"
                ]
            },
            "resourceType": {
                "eq": [
                    "AWS::IAM::Role"
                ]
            }
        },
        "ruleName": "MyArchiveRule",
        "updatedAt": "2024-02-15T00:49:27+00:00"
    }
}
```

Per ulteriori informazioni, consulta Regole di archiviazione nella Guida per l'utente di AWS IAM.

- Per i dettagli sull'API, consulta GetArchiveRule AWS CLICommand Reference.

`get-finding-v2`

Il seguente esempio di codice mostra come utilizzare`get-finding-v2`.

AWS CLI

Come recuperare informazioni sull'esito specificato

L'`get-finding-v2`esempio seguente recupera informazioni sul risultato specificato nel tuo AWS account.

```
aws accessanalyzer get-finding-v2 \
    --analyzer-arn arn:aws:access-analyzer:us-west-2:111122223333:analyzer/
ConsoleAnalyzer-organization \
    --id 0910eedb-381e-4e95-adda-0d25c19e6e90
```

Output:

```
{
```

```
    "findingDetails": [
        {
            "externalAccessDetails": {
                "action": [
                    "sts:AssumeRoleWithWebIdentity"
                ],
                "condition": {
                    "cognito-identity.amazonaws.com:aud": "us-
west-2:EXAMPLE0-0000-0000-0000-000000000000"
                },
                "isPublic": false,
                "principal": {
                    "Federated": "cognito-identity.amazonaws.com"
                }
            }
        }
    ],
    "resource": "arn:aws:iam::111122223333:role/Cognito_testpoolAuth_Role",
    "status": "ACTIVE",
    "error": null,
    "createdAt": "2021-02-26T21:17:50.905000+00:00",
    "resourceType": "AWS::IAM::Role",
    "findingType": "ExternalAccess",
    "resourceOwnerAccount": "111122223333",
    "analyzedAt": "2024-02-16T18:17:47.888000+00:00",
    "id": "0910eedb-381e-4e95-adda-0d25c19e6e90",
    "updatedAt": "2021-02-26T21:17:50.905000+00:00"
}
```

Per ulteriori informazioni, consulta Revisione degli esiti nella Guida per l'utente di AWS IAM.

- Per i dettagli sull'API, vedere GetFindingV2 in AWS CLI Command Reference.

`get-finding`

Il seguente esempio di codice mostra come utilizzare`get-finding`.

AWS CLI

Come recuperare informazioni sull'esito specificato

L'`get-finding`esempio seguente recupera informazioni sul risultato specificato nel tuo AWS account.

```
aws accessanalyzer get-finding \
    --analyzer-arn arn:aws:access-analyzer:us-west-2:111122223333:analyzer/
ConsoleAnalyzer-organization \
    --id 0910eedb-381e-4e95-adda-0d25c19e6e90
```

Output:

```
{
    "finding": {
        "id": "0910eedb-381e-4e95-adda-0d25c19e6e90",
        "principal": {
            "Federated": "cognito-identity.amazonaws.com"
        },
        "action": [
            "sts:AssumeRoleWithWebIdentity"
        ],
        "resource": "arn:aws:iam::111122223333:role/Cognito_testpoolAuth_Role",
        "isPublic": false,
        "resourceType": "AWS::IAM::Role",
        "condition": {
            "cognito-identity.amazonaws.com:aud": "us-
west-2:EXAMPLE0-0000-0000-0000-000000000000"
        },
        "createdAt": "2021-02-26T21:17:50.905000+00:00",
        "analyzedAt": "2024-02-16T18:17:47.888000+00:00",
        "updatedAt": "2021-02-26T21:17:50.905000+00:00",
        "status": "ACTIVE",
        "resourceOwnerAccount": "111122223333"
    }
}
```

Per ulteriori informazioni, consulta Revisione degli esiti nella Guida per l'utente di AWS IAM.

- Per i dettagli sull'API, consulta AWS CLI Command GetFindingReference.

`get-generated-policy`

Il seguente esempio di codice mostra come utilizzare`get-generated-policy`.

AWS CLI

Per recuperare la policy generata utilizzando l'API `StartPolicyGeneration`

L'`get-generated-policy`esempio seguente recupera la policy generata utilizzando l' StartPolicyGeneration API del tuo AWS account.

```
aws accessanalyzer get-generated-policy \
    --job-id c557dc4a-0338-4489-95dd-739014860ff9
```

Output:

```
{
    "generatedPolicyResult": {
        "generatedPolicies": [
            {
                "policy": "{\"Version\":\"2012-10-17\",\"Statement\":
[{\"Sid\":\"SupportedServiceSid0\",\"Effect\":\"Allow\",\"Action\":
[\"access-analyzer:GetAnalyzer\",\"access-analyzer:ListAnalyzers\",
\"access-analyzer:ListArchiveRules\",\"access-analyzer:ListFindings
\",\"cloudtrail:DescribeTrails\",\"cloudtrail:GetEventDataStore\",
\"cloudtrail:GetEventSelectors\",\"cloudtrail:GetInsightSelectors
\",\"cloudtrail:GetTrailStatus\",\"cloudtrail:ListChannels\",
\"cloudtrail:ListEventDataStores\",\"cloudtrail:ListQueries\",\"cloudtrail:ListTags
\",\"cloudtrail:LookupEvents\",\"ec2:DescribeRegions\",\"iam:GetAccountSummary
\",\"iam:GetOpenIDConnectProvider\",\"iam:GetRole\",\"iam:ListAccessKeys\",
\"iam:ListAccountAliases\",\"iam:ListOpenIDConnectProviders\",\"iam:ListRoles
\",\"iam:ListSAMLProviders\",\"kms:ListAliases\",\"s3:GetBucketLocation\",
\"s3:ListAllMyBuckets\"],\"Resource\":\"*\"}]}"
            }
        ],
        "properties": {
            "cloudTrailProperties": {
                "endTime": "2024-02-14T22:44:40+00:00",
                "startTime": "2024-02-13T00:30:00+00:00",
                "trailProperties": [
                    {
                        "allRegions": true,
                        "cloudTrailArn": "arn:aws:cloudtrail:us-
west-2:111122223333:trail/my-trail",
                        "regions": []
                    }
                ]
            },
            "isComplete": false,
            "principalArn": "arn:aws:iam::111122223333:role/Admin"
        }
```

```
    },
    "jobDetails": {
        "completedOn": "2024-02-14T22:47:01+00:00",
        "jobId": "c557dc4a-0338-4489-95dd-739014860ff9",
        "startedOn": "2024-02-14T22:44:41+00:00",
        "status": "SUCCEEDED"
    }
}
```

Per ulteriori informazioni, consulta Generazione delle policy per IAM Access Analyzer nella Guida per l'utente di AWS IAM.

- Per i dettagli sull'API, consulta GetGeneratedPolicy AWS CLICommand Reference.

`list-access-preview-findings`

Il seguente esempio di codice mostra come utilizzare`list-access-preview-findings`.

AWS CLI

Come recuperare un elenco degli esiti dell'anteprima di accesso generati dall'anteprima di accesso specificata

L'`list-access-preview-findings`esempio seguente recupera un elenco di risultati dell'anteprima di accesso generati dall'anteprima di accesso specificata nell' AWS account.

```
aws accessanalyzer list-access-preview-findings \
    --access-preview-id 3c65eb13-6ef9-4629-8919-a32043619e6b \
    --analyzer-arn arn:aws:access-analyzer:us-west-2:111122223333:analyzer/
ConsoleAnalyzer-account
```

Output:

```
{
    "findings": [
        {
            "id": "e22fc158-1c87-4c32-9464-e7f405ce8d74",
            "principal": {
                "AWS": "111122223333"
            },
            "action": [
                "s3:PutObject",
```

```
                "s3:PutObjectAcl"
            ],
            "condition": {},
            "resource": "arn:aws:s3:::amzn-s3-demo-bucket",
            "isPublic": false,
            "resourceType": "AWS::S3::Bucket",
            "createdAt": "2024-02-17T00:18:46+00:00",
            "changeType": "NEW",
            "status": "ACTIVE",
            "resourceOwnerAccount": "111122223333",
            "sources": [
                {
                    "type": "POLICY"
                }
            ]
        }
    ]
}
```

Per ulteriori informazioni, consulta Anteprima dell'accesso con IAM Access Analyzer APIs nella AWS IAM User Guide.

- Per i dettagli sull'API, consulta AWS CLI Command ListAccessPreviewFindingsReference.

`list-access-previews`

Il seguente esempio di codice mostra come utilizzare`list-access-previews`.

AWS CLI

Come recuperare un elenco di anteprime di accesso per l'analizzatore specificato

L'`list-access-previews`esempio seguente recupera un elenco di anteprime di accesso per l'analizzatore specificato nell'account. AWS

```
aws accessanalyzer list-access-previews \
    --analyzer-arn arn:aws:access-analyzer:us-west-2:111122223333:analyzer/
ConsoleAnalyzer-account
```

Output:

```
{
    "accessPreviews": [
```

```
        {
            "id": "3c65eb13-6ef9-4629-8919-a32043619e6b",
            "analyzerArn": "arn:aws:access-analyzer:us-west-2:111122223333:analyzer/
ConsoleAnalyzer-account",
            "createdAt": "2024-02-17T00:18:44+00:00",
            "status": "COMPLETED"
        }
    ]
}
```

Per ulteriori informazioni, consulta Anteprima dell'accesso con IAM Access Analyzer APIs nella IAM User Guide.AWS

- Per i dettagli sull'API, consulta AWS CLI Command ListAccessPreviewsReference.

`list-analyzed-resources`

Il seguente esempio di codice mostra come utilizzare`list-analyzed-resources`.

AWS CLI

Come elencare i widget disponibili

L'`list-analyzed-resources`esempio seguente elenca i widget disponibili nel tuo AWS account.

```
aws accessanalyzer list-analyzed-resources \
    --analyzer-arn arn:aws:access-analyzer:us-west-2:111122223333:analyzer/
ConsoleAnalyzer-account \
    --resource-type AWS::IAM::Role
```

Output:

```
{
    "analyzedResources": [
        {
            "resourceArn": "arn:aws:sns:us-west-2:111122223333:Validation-Email",
            "resourceOwnerAccount": "111122223333",
            "resourceType": "AWS::SNS::Topic"
        },
        {
            "resourceArn": "arn:aws:sns:us-west-2:111122223333:admin-alerts",
            "resourceOwnerAccount": "111122223333",
```

```
            "resourceType": "AWS::SNS::Topic"
        },
        {
            "resourceArn": "arn:aws:sns:us-west-2:111122223333:config-topic",
            "resourceOwnerAccount": "111122223333",
            "resourceType": "AWS::SNS::Topic"
        },
        {
            "resourceArn": "arn:aws:sns:us-west-2:111122223333:inspector-topic",
            "resourceOwnerAccount": "111122223333",
            "resourceType": "AWS::SNS::Topic"
        }
    ]
}
```

Per ulteriori informazioni, consulta Using AWS Identity and Access Management Access Analyzer nella AWS IAM User Guide.

- Per i dettagli sull'API, consulta ListAnalyzedResources AWS CLICommand Reference.

`list-analyzers`

Il seguente esempio di codice mostra come utilizzare`list-analyzers`.

AWS CLI

Come recuperare un elenco di analizzatori

L'`list-analyzers`esempio seguente recupera un elenco di analizzatori presenti nell'account. AWS

```
aws accessanalyzer list-analyzers
```

Output:

```
{
    "analyzers": [
        {
            "arn": "arn:aws:access-analyzer:us-west-2:111122223333:analyzer/
UnusedAccess-ConsoleAnalyzer-organization",
            "createdAt": "2024-02-15T00:46:40+00:00",
            "name": "UnusedAccess-ConsoleAnalyzer-organization",
            "status": "ACTIVE",
```

```
            "tags": {
                "auto-delete": "no"
            },
            "type": "ORGANIZATION_UNUSED_ACCESS"
        },
        {
            "arn": "arn:aws:access-analyzer:us-west-2:111122223333:analyzer/
ConsoleAnalyzer-organization",
            "createdAt": "2020-04-25T07:43:28+00:00",
            "lastResourceAnalyzed": "arn:aws:s3:::amzn-s3-demo-bucket",
            "lastResourceAnalyzedAt": "2024-02-15T21:51:56.517000+00:00",
            "name": "ConsoleAnalyzer-organization",
            "status": "ACTIVE",
            "tags": {
                "auto-delete": "no"
            },
            "type": "ORGANIZATION"
        },
        {
            "arn": "arn:aws:access-analyzer:us-west-2:111122223333:analyzer/
ConsoleAnalyzer-account",
            "createdAt": "2019-12-03T07:28:17+00:00",
            "lastResourceAnalyzed": "arn:aws:sns:us-west-2:111122223333:config-
topic",
            "lastResourceAnalyzedAt": "2024-02-15T18:01:53.003000+00:00",
            "name": "ConsoleAnalyzer-account",
            "status": "ACTIVE",
            "tags": {
                "auto-delete": "no"
            },
            "type": "ACCOUNT"
        }
    ]
}
```

Per ulteriori informazioni, consulta Using AWS Identity and Access Management Access Analyzer nella AWS IAM User Guide.

- Per i dettagli sull'API, consulta ListAnalyzers AWS CLICommand Reference.

`list-archive-rules`

Il seguente esempio di codice mostra come utilizzare`list-archive-rules`.

AWS CLI

Come recuperare un elenco di regole di archiviazione create per l'analizzatore specificato

L'`list-archive-rules`esempio seguente recupera un elenco di regole di archiviazione create per l'analizzatore specificato nell'account AWS .

```
aws accessanalyzer list-archive-rules \
    --analyzer-name UnusedAccess-ConsoleAnalyzer-organization
```

Output:

```
{
    "archiveRules": [
        {
            "createdAt": "2024-02-15T00:49:27+00:00",
            "filter": {
                "resource": {
                    "contains": [
                        "Cognito"
                    ]
                },
                "resourceType": {
                    "eq": [
                        "AWS::IAM::Role"
                    ]
                }
            },
            "ruleName": "MyArchiveRule",
            "updatedAt": "2024-02-15T00:49:27+00:00"
        },
        {
            "createdAt": "2024-02-15T23:27:45+00:00",
            "filter": {
                "findingType": {
                    "eq": [
                        "UnusedIAMUserAccessKey"
                    ]
                }
            },
            "ruleName": "ArchiveRule-56125a39-e517-4ff8-afb1-ef06f58db612",
            "updatedAt": "2024-02-15T23:27:45+00:00"
        }
```

```
    ]
}
```

Per ulteriori informazioni, consulta Using AWS Identity and Access Management Access Analyzer nella AWS IAM User Guide.

- Per i dettagli sull'API, consulta ListArchiveRules AWS CLICommand Reference.

`list-findings-v2`

Il seguente esempio di codice mostra come utilizzare`list-findings-v2`.

AWS CLI

Come recuperare un elenco degli esiti generati dall'analizzatore specificato

L'`list-findings-v2`esempio seguente recupera un elenco di risultati generati dall'analizzatore specificato nel tuo AWS account. Questo esempio filtra i risultati per includere solo i ruoli IAM il cui nome contiene `Cognito`.

```
aws accessanalyzer list-findings-v2 \
    --analyzer-arn arn:aws:access-analyzer:us-west-2:111122223333:analyzer/
ConsoleAnalyzer-account \
    --filter '{"resource": {"contains": ["Cognito"]}, "resourceType": {"eq":
 ["AWS::IAM::Role"]}}'
```

Output:

```
{
    "findings": [
        {
            "analyzedAt": "2024-02-16T18:17:47.888000+00:00",
            "createdAt": "2021-02-26T21:17:24.710000+00:00",
            "id": "597f3bc2-3adc-4c18-9879-5c4b23485e46",
            "resource": "arn:aws:iam::111122223333:role/
Cognito_testpoolUnauth_Role",
            "resourceType": "AWS::IAM::Role",
            "resourceOwnerAccount": "111122223333",
            "status": "ACTIVE",
            "updatedAt": "2021-02-26T21:17:24.710000+00:00",
            "findingType": "ExternalAccess"
        },
        {
```

```
            "analyzedAt": "2024-02-16T18:17:47.888000+00:00",
            "createdAt": "2021-02-26T21:17:50.905000+00:00",
            "id": "ce0e221a-85b9-4d52-91ff-d7678075442f",
            "resource": "arn:aws:iam::111122223333:role/Cognito_testpoolAuth_Role",
            "resourceType": "AWS::IAM::Role",
            "resourceOwnerAccount": "111122223333",
            "status": "ACTIVE",
            "updatedAt": "2021-02-26T21:17:50.905000+00:00",
            "findingType": "ExternalAccess"
        }
    ]
}
```

Per ulteriori informazioni, consulta Using AWS Identity and Access Management Access Analyzer nella AWS IAM User Guide.

- Per i dettagli sull'API, consulta ListFindingsV2 in AWS CLI Command Reference.

`list-findings`

Il seguente esempio di codice mostra come utilizzare`list-findings`.

AWS CLI

Come recuperare un elenco degli esiti generati dall'analizzatore specificato

L'`list-findings`esempio seguente recupera un elenco di risultati generati dall'analizzatore specificato nel tuo AWS account. Questo esempio filtra i risultati per includere solo i ruoli IAM il cui nome contiene `Cognito`.

```
aws accessanalyzer list-findings \
    --analyzer-arn arn:aws:access-analyzer:us-west-2:111122223333:analyzer/
ConsoleAnalyzer-account \
    --filter '{"resource": {"contains": ["Cognito"]}, "resourceType": {"eq":
 ["AWS::IAM::Role"]}}'
```

Output:

```
{
    "findings": [
        {
            "id": "597f3bc2-3adc-4c18-9879-5c4b23485e46",
            "principal": {
```

```
                "Federated": "cognito-identity.amazonaws.com"
            },
            "action": [
                "sts:AssumeRoleWithWebIdentity"
            ],
            "resource": "arn:aws:iam::111122223333:role/
Cognito_testpoolUnauth_Role",
            "isPublic": false,
            "resourceType": "AWS::IAM::Role",
            "condition": {
                "cognito-identity.amazonaws.com:aud": "us-
west-2:EXAMPLE0-0000-0000-0000-000000000000"
            },
            "createdAt": "2021-02-26T21:17:24.710000+00:00",
            "analyzedAt": "2024-02-16T18:17:47.888000+00:00",
            "updatedAt": "2021-02-26T21:17:24.710000+00:00",
            "status": "ACTIVE",
            "resourceOwnerAccount": "111122223333"
        },
        {
            "id": "ce0e221a-85b9-4d52-91ff-d7678075442f",
            "principal": {
                "Federated": "cognito-identity.amazonaws.com"
            },
            "action": [
                "sts:AssumeRoleWithWebIdentity"
            ],
            "resource": "arn:aws:iam::111122223333:role/Cognito_testpoolAuth_Role",
            "isPublic": false,
            "resourceType": "AWS::IAM::Role",
            "condition": {
                "cognito-identity.amazonaws.com:aud": "us-
west-2:EXAMPLE0-0000-0000-0000-000000000000"
            },
            "createdAt": "2021-02-26T21:17:50.905000+00:00",
            "analyzedAt": "2024-02-16T18:17:47.888000+00:00",
            "updatedAt": "2021-02-26T21:17:50.905000+00:00",
            "status": "ACTIVE",
            "resourceOwnerAccount": "111122223333"
        }
    ]
}
```

Per ulteriori informazioni, consulta Using AWS Identity and Access Management Access Analyzer nella AWS IAM User Guide.

- Per i dettagli sull'API, consulta ListFindings AWS CLICommand Reference.

`list-policy-generations`

Il seguente esempio di codice mostra come utilizzare`list-policy-generations`.

AWS CLI

Come elencare tutte le generazioni di policy richieste negli ultimi sette giorni

L'`list-policy-generations`esempio seguente elenca tutte le generazioni di policy richieste nel tuo AWS account negli ultimi sette giorni.

```
aws accessanalyzer list-policy-generations
```

Output:

```
{
    "policyGenerations": [
        {
            "completedOn": "2024-02-14T23:43:38+00:00",
            "jobId": "923a56b0-ebb8-4e80-8a3c-a11ccfbcd6f2",
            "principalArn": "arn:aws:iam::111122223333:role/Admin",
            "startedOn": "2024-02-14T23:43:02+00:00",
            "status": "CANCELED"
        },
        {
            "completedOn": "2024-02-14T22:47:01+00:00",
            "jobId": "c557dc4a-0338-4489-95dd-739014860ff9",
            "principalArn": "arn:aws:iam::111122223333:role/Admin",
            "startedOn": "2024-02-14T22:44:41+00:00",
            "status": "SUCCEEDED"
        }
    ]
}
```

Per ulteriori informazioni, consulta Generazione delle policy per IAM Access Analyzer nella Guida per l'utente di AWS IAM.

- Per i dettagli sull'API, consulta ListPolicyGenerations AWS CLICommand Reference.

`list-tags-for-resource`

Il seguente esempio di codice mostra come utilizzare`list-tags-for-resource`.

AWS CLI

Come recuperare un elenco di tag applicati alla risorsa specificata

L'`list-tags-for-resource`esempio seguente recupera un elenco di tag applicati alla risorsa specificata nell' AWS account.

```
aws accessanalyzer list-tags-for-resource \
    --resource-arn arn:aws:access-analyzer:us-west-2:111122223333:analyzer/
ConsoleAnalyzer-account
```

Output:

```
{
    "tags": {
        "Zone-of-trust": "Account",
        "Name": "ConsoleAnalyzer"
    }
}
```

Per ulteriori informazioni, consulta Generazione delle policy per IAM Access Analyzer nella Guida per l'utente di AWS IAM.

- Per i dettagli sull'API, consulta ListTagsForResource AWS CLICommand Reference.

`start-policy-generation`

Il seguente esempio di codice mostra come utilizzare`start-policy-generation`.

AWS CLI

Come avviare una richiesta di generazione di policy

L'`start-policy-generation`esempio seguente avvia una richiesta di generazione di policy nel tuo AWS account.

```
aws accessanalyzer start-policy-generation \
    --policy-generation-details '{"principalArn":"arn:aws:iam::111122223333:role/
Admin"}' \
    --cloud-trail-details file://myfile.json
```

Contenuto di `myfile.json`:

```
{
    "accessRole": "arn:aws:iam::111122223333:role/service-role/
AccessAnalyzerMonitorServiceRole",
    "startTime": "2024-02-13T00:30:00Z",
    "trails": [
        {
            "allRegions": true,
            "cloudTrailArn": "arn:aws:cloudtrail:us-west-2:111122223333:trail/my-
trail"
        }
    ]
}
```

Output:

```
{
    "jobId": "c557dc4a-0338-4489-95dd-739014860ff9"
}
```

Per ulteriori informazioni, consulta Generazione delle policy per IAM Access Analyzer nella Guida per l'utente di AWS IAM.

- Per i dettagli sull'API, consulta StartPolicyGeneration AWS CLICommand Reference.

`start-resource-scan`

Il seguente esempio di codice mostra come utilizzare`start-resource-scan`.

AWS CLI

Come avviare immediatamente una scansione delle policy applicate alla risorsa specificata

L'`start-resource-scan`esempio seguente avvia immediatamente una scansione delle politiche applicate alla risorsa specificata nell'account AWS .

```
aws accessanalyzer start-resource-scan \
    --analyzer-arn arn:aws:access-analyzer:us-west-2:111122223333:analyzer/
ConsoleAnalyzer-account \
    --resource-arn arn:aws:iam::111122223333:role/Cognito_testpoolAuth_Role
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Generazione delle policy per IAM Access Analyzer nella Guida per l'utente di AWS IAM.

- Per i dettagli sull'API, consulta AWS CLI Command StartResourceScanReference.

`tag-resource`

Il seguente esempio di codice mostra come utilizzare`tag-resource`.

AWS CLI

Come aggiungere un tag alla risorsa specificata

L'`tag-resource`esempio seguente aggiunge un tag alla risorsa specificata nell' AWS account.

```
aws accessanalyzer tag-resource \
    --resource-arn arn:aws:access-analyzer:us-west-2:111122223333:analyzer/
ConsoleAnalyzer-account \
    --tags Environment=dev,Purpose=testing
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Using AWS Identity and Access Management Access Analyzer nella AWS IAM User Guide.

- Per i dettagli sull'API, consulta TagResource AWS CLICommand Reference.

`untag-resource`

Il seguente esempio di codice mostra come utilizzare`untag-resource`.

AWS CLI

Come rimuovere i tag dalle risorse specificate

L'`untag-resource`esempio seguente rimuove i tag dalla risorsa specificata nell' AWS account.

```
aws accessanalyzer untag-resource \
    --resource-arn arn:aws:access-analyzer:us-west-2:111122223333:analyzer/
ConsoleAnalyzer-account \
    --tag-keys Environment Purpose
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Using AWS Identity and Access Management Access Analyzer nella AWS IAM User Guide.

- Per i dettagli sull'API, consulta UntagResource AWS CLICommand Reference.

`update-archive-rule`

Il seguente esempio di codice mostra come utilizzare`update-archive-rule`.

AWS CLI

Come aggiornare i criteri e i valori per la regola di archiviazione specificata

L'`update-archive-rule`esempio seguente aggiorna i criteri e i valori per la regola di archiviazione specificata nell' AWS account.

```
aws accessanalyzer update-archive-rule \
    --analyzer-name UnusedAccess-ConsoleAnalyzer-organization \
    --rule-name MyArchiveRule \
    --filter '{"resource": {"contains": ["Cognito"]}, "resourceType": {"eq":
 ["AWS::IAM::Role"]}}'
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Regole di archiviazione nella Guida per l'utente di AWS IAM.

- Per i dettagli sull'API, consulta UpdateArchiveRule AWS CLICommand Reference.

`update-findings`

Il seguente esempio di codice mostra come utilizzare`update-findings`.

AWS CLI

Come aggiornare lo stato degli esiti specificati

L'`update-findings`esempio seguente aggiorna lo stato dei risultati specificati nel tuo AWS account.

```
aws accessanalyzer update-findings \
    --analyzer-arn arn:aws:access-analyzer:us-west-2:111122223333:analyzer/
UnusedAccess-ConsoleAnalyzer-organization \
    --ids 4f319ac3-2e0c-4dc4-bf51-7013a086b6ae 780d586a-2cce-4f72-aff6-359d450e7500
 \
    --status ARCHIVED
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Using AWS Identity and Access Management Access Analyzer nella AWS IAM User Guide.

- Per i dettagli sull'API, consulta UpdateFindings AWS CLICommand Reference.

`validate-policy`

Il seguente esempio di codice mostra come utilizzare`validate-policy`.

AWS CLI

Come richiedere la convalida di una policy e restituire un elenco di esiti

L'esempio `validate-policy` seguente richiede la convalida di una policy e restituisce un elenco di esiti. La policy utilizzata nell'esempio è una policy di attendibilità dei ruoli per un ruolo di Amazon Cognito utilizzato per la federazione delle identità web. Gli esiti generati dalla policy di trust si riferiscono a un valore dell'elemento `Sid` vuoto e a un principale della policy non corrispondente a causa dell'uso errato dell'operazione di assunzione del ruolo, `sts:AssumeRole`. L'operazione di assunzione del ruolo corretta da utilizzare con Cognito è `sts:AssumeRoleWithWebIdentity`.

```
aws accessanalyzer validate-policy \
    --policy-document file://myfile.json \
    --policy-type RESOURCE_POLICY
```

Contenuto di `myfile.json`:

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "",
            "Effect": "Allow",
            "Principal": {
                "Federated": "cognito-identity.amazonaws.com"
            },
            "Action": [
                "sts:AssumeRole",
                "sts:TagSession"
            ],
            "Condition": {
                "StringEquals": {
                    "cognito-identity.amazonaws.com:aud": "us-west-2_EXAMPLE"
                }
            }
        }
    ]
}
```

Output:

```
{
    "findings": [
        {
            "findingDetails": "Add a value to the empty string in the Sid element.",
            "findingType": "SUGGESTION",
            "issueCode": "EMPTY_SID_VALUE",
            "learnMoreLink": "https://docs.aws.amazon.com/IAM/latest/UserGuide/
access-analyzer-reference-policy-checks.html#access-analyzer-reference-policy-
checks-suggestion-empty-sid-value",
            "locations": [
                {
                    "path": [
                        {
                            "value": "Statement"
                        },
                        {
                            "index": 0
                        },
                        {
```

```
                        "value": "Sid"
                    }
                ],
                "span": {
                    "end": {
                        "column": 21,
                        "line": 5,
                        "offset": 81
                    },
                    "start": {
                        "column": 19,
                        "line": 5,
                        "offset": 79
                    }
                }
            }
        ]
    },
    {
        "findingDetails": "The sts:AssumeRole action is invalid with the
 following principal(s): cognito-identity.amazonaws.com. Use a SAML provider
 principal with the sts:AssumeRoleWithSAML action or use an OIDC provider principal
 with the sts:AssumeRoleWithWebIdentity action. Ensure the provider is Federated if
 you use either of the two options.",
        "findingType": "ERROR",
        "issueCode": "MISMATCHED_ACTION_FOR_PRINCIPAL",
        "learnMoreLink": "https://docs.aws.amazon.com/IAM/latest/UserGuide/
access-analyzer-reference-policy-checks.html#access-analyzer-reference-policy-
checks-error-mismatched-action-for-principal",
        "locations": [
            {
                "path": [
                    {
                        "value": "Statement"
                    },
                    {
                        "index": 0
                    },
                    {
                        "value": "Action"
                    },
                    {
                        "index": 0
                    }
```

```
                    ],
                    "span": {
                        "end": {
                            "column": 32,
                            "line": 11,
                            "offset": 274
                        },
                        "start": {
                            "column": 16,
                            "line": 11,
                            "offset": 258
                        }
                    }
                },
                {
                    "path": [
                        {
                            "value": "Statement"
                        },
                        {
                            "index": 0
                        },
                        {
                            "value": "Principal"
                        },
                        {
                            "value": "Federated"
                        }
                    ],
                    "span": {
                        "end": {
                            "column": 61,
                            "line": 8,
                            "offset": 202
                        },
                        "start": {
                            "column": 29,
                            "line": 8,
                            "offset": 170
                        }
                    }
                }
            ]
        },
```

```
        {
            "findingDetails": "The following actions: sts:TagSession are not
 supported by the condition key cognito-identity.amazonaws.com:aud. The condition
 will not be evaluated for these actions. We recommend that you move these actions
 to a different statement without this condition key.",
            "findingType": "ERROR",
            "issueCode": "UNSUPPORTED_ACTION_FOR_CONDITION_KEY",
            "learnMoreLink": "https://docs.aws.amazon.com/IAM/latest/UserGuide/
access-analyzer-reference-policy-checks.html#access-analyzer-reference-policy-
checks-error-unsupported-action-for-condition-key",
            "locations": [
                {
                    "path": [
                        {
                            "value": "Statement"
                        },
                        {
                            "index": 0
                        },
                        {
                            "value": "Action"
                        },
                        {
                            "index": 1
                        }
                    ],
                    "span": {
                        "end": {
                            "column": 32,
                            "line": 12,
                            "offset": 308
                        },
                        "start": {
                            "column": 16,
                            "line": 12,
                            "offset": 292
                        }
                    }
                },
                {
                    "path": [
                        {
                            "value": "Statement"
                        },
```

```
                        {
                            "index": 0
                        },
                        {
                            "value": "Condition"
                        },
                        {
                            "value": "StringEquals"
                        },
                        {
                            "value": "cognito-identity.amazonaws.com:aud"
                        }
                    ],
                    "span": {
                        "end": {
                            "column": 79,
                            "line": 16,
                            "offset": 464
                        },
                        "start": {
                            "column": 58,
                            "line": 16,
                            "offset": 443
                        }
                    }
                }
            ]
        }
    ]
}
```

Per ulteriori informazioni, consulta Verifiche per la validazione delle policy nella Guida per l'utente di AWS IAM.

- Per i dettagli sull'API, consulta ValidatePolicy AWS CLICommand Reference.

Esempi di Image Builder che utilizzano AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with Image Builder.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`create-component`

Il seguente esempio di codice mostra come utilizzare`create-component`.

AWS CLI

Come creare un componente

L'esempio `create-component` seguente crea un componente che utilizza un file di documento JSON e fa riferimento a un documento componente in formato YAML che viene caricato in un bucket Amazon S3.

```
aws imagebuilder create-component \
    --cli-input-json file://create-component.json
```

Contenuto di `create-component.json`:

```
{
    "name": "MyExampleComponent",
    "semanticVersion": "2019.12.02",
    "description": "An example component that builds, validates and tests an image",
    "changeDescription": "Initial version.",
    "platform": "Windows",
    "uri": "s3://s3-bucket-name/s3-bucket-path/component.yaml"
}
```

Output:

```
{
```

```
    "requestId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "clientToken": "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
    "componentBuildVersionArn": "arn:aws:imagebuilder:us-
west-2:123456789012:component/examplecomponent/2019.12.02/1"
}
```

Per ulteriori informazioni, vedere Configurazione e gestione di una pipeline di EC2 immagini di Image Builder utilizzando la AWS CLI nella Guida per l'utente di Image EC2 Builder.

- Per i dettagli sull'API, vedere CreateComponentin Command Reference.AWS CLI

`create-distribution-configuration`

Il seguente esempio di codice mostra come utilizzare`create-distribution-configuration`.

AWS CLI

Come creare una configurazione di distribuzione

L'esempio `create-distribution-configuration` seguente crea una configurazione di distribuzione utilizzando un file JSON.

```
aws imagebuilder create-distribution-configuration \
    --cli-input-json file:/create-distribution-configuration.json
```

Contenuto di `create-distribution-configuration.json`:

```
{
    "name": "MyExampleDistribution",
    "description": "Copies AMI to eu-west-1",
    "distributions": [
        {
            "region": "us-west-2",
            "amiDistributionConfiguration": {
                "name": "Name {{imagebuilder:buildDate}}",
                "description": "An example image name with parameter references",
                "amiTags": {
                    "KeyName": "{{ssm:parameter_name}}"
                },
                "launchPermission": {
                    "userIds": [
                        "123456789012"
```

```
                    ]
                }
            }
        },
        {
            "region": "eu-west-1",
            "amiDistributionConfiguration": {
                "name": "My {{imagebuilder:buildVersion}} image
 {{imagebuilder:buildDate}}",
                "amiTags": {
                    "KeyName": "Value"
                },
                "launchPermission": {
                    "userIds": [
                        "123456789012"
                    ]
                }
            }
        }
    ]
}
```

Output:

```
{
    "requestId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "clientToken": "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
    "distributionConfigurationArn": "arn:aws:imagebuilder:us-
west-2:123456789012:distribution-configuration/myexampledistribution"
}
```

Per ulteriori informazioni, vedere Configurazione e gestione di una pipeline di EC2 immagini di Image Builder utilizzando la AWS CLI nella Guida per l'utente di Image EC2 Builder.

- Per i dettagli sull'API, vedere CreateDistributionConfigurationin Command Reference.AWS CLI

`create-image-pipeline`

Il seguente esempio di codice mostra come utilizzare`create-image-pipeline`.

AWS CLI

Come creare una pipeline di immagini

L'esempio `create-image-pipeline` seguente crea una pipeline di immagini utilizzando un file JSON.

```
aws imagebuilder create-image-pipeline \
    --cli-input-json file://create-image-pipeline.json
```

Contenuto di `create-image-pipeline.json`:

```
{
    "name": "MyWindows2016Pipeline",
    "description": "Builds Windows 2016 Images",
    "imageRecipeArn": "arn:aws:imagebuilder:us-west-2:123456789012:image-recipe/
mybasicrecipe/2019.12.03",
    "infrastructureConfigurationArn": "arn:aws:imagebuilder:us-
west-2:123456789012:infrastructure-configuration/myexampleinfrastructure",
    "distributionConfigurationArn": "arn:aws:imagebuilder:us-
west-2:123456789012:distribution-configuration/myexampledistribution",
    "imageTestsConfiguration": {
        "imageTestsEnabled": true,
        "timeoutMinutes": 60
    },
    "schedule": {
        "scheduleExpression": "cron(0 0 * * SUN)",
        "pipelineExecutionStartCondition":
 "EXPRESSION_MATCH_AND_DEPENDENCY_UPDATES_AVAILABLE"
    },
    "status": "ENABLED"
}
```

Output:

```
{
    "requestId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "clientToken": "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
    "imagePipelineArn": "arn:aws:imagebuilder:us-west-2:123456789012:image-pipeline/
mywindows2016pipeline"
}
```

Per ulteriori informazioni, vedere Configurazione e gestione di una pipeline di EC2 immagini di Image Builder utilizzando la AWS CLI nella Guida per l'utente di Image EC2 Builder.

- Per i dettagli sull'API, vedere CreateImagePipelinein Command Reference.AWS CLI

`create-image-recipe`

Il seguente esempio di codice mostra come utilizzare`create-image-recipe`.

AWS CLI

Come creare una ricetta

L'esempio `create-image-recipe` seguente crea una ricetta di immagine utilizzando un file JSON. I componenti vengono installati nell'ordine in cui sono specificati.

```
aws imagebuilder create-image-recipe \
    --cli-input-json file://create-image-recipe.json
```

Contenuto di `create-image-recipe.json`:

```
{
    "name": "MyBasicRecipe",
    "description": "This example image recipe creates a Windows 2016 image.",
    "semanticVersion": "2019.12.03",
    "components":
    [
        {
            "componentArn": "arn:aws:imagebuilder:us-west-2:123456789012:component/
myexamplecomponent/2019.12.02/1"
        },
        {
            "componentArn": "arn:aws:imagebuilder:us-west-2:123456789012:component/
myimportedcomponent/1.0.0/1"
        }
    ],
    "parentImage": "arn:aws:imagebuilder:us-west-2:aws:image/windows-server-2016-
english-full-base-x86/xxxx.x.x"
}
```

Output:

```
{
    "requestId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "clientToken": "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
    "imageRecipeArn": "arn:aws:imagebuilder:us-west-2:123456789012:image-recipe/
mybasicrecipe/2019.12.03"
```

```
}
```

Per ulteriori informazioni, vedere Configurazione e gestione di una pipeline di EC2 immagini di Image Builder utilizzando la AWS CLI nella Guida per l'utente di Image EC2 Builder.

- Per i dettagli sull'API, vedere CreateImageRecipein Command Reference.AWS CLI

`create-image`

Il seguente esempio di codice mostra come utilizzare`create-image`.

AWS CLI

Come creare un'immagine

L'esempio `create-image` seguente crea un'immagine.

```
aws imagebuilder create-image \
    --image-recipe-arn arn:aws:imagebuilder:us-west-2:123456789012:image-recipe/
mybasicrecipe/2019.12.03 \
    --infrastructure-configuration-arn arn:aws:imagebuilder:us-
west-2:123456789012:infrastructure-configuration/myexampleinfrastructure
```

Output:

```
{
    "requestId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "clientToken": "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
    "imageBuildVersionArn": "arn:aws:imagebuilder:us-west-2:123456789012:image/
mybasicrecipe/2019.12.03/1"
}
```

Per ulteriori informazioni, vedere Configurazione e gestione di una pipeline di EC2 immagini di Image Builder utilizzando la AWS CLI nella Guida per l'utente di Image EC2 Builder.

- Per i dettagli sull'API, vedere CreateImagein Command Reference.AWS CLI

`create-infrastructure-configuration`

Il seguente esempio di codice mostra come utilizzare`create-infrastructure-configuration`.

AWS CLI

Come creare una configurazione dell'infrastruttura

L'esempio `create-infrastructure-configuration` seguente crea una configurazione dell'infrastruttura utilizzando un file JSON.

```
aws imagebuilder create-infrastructure-configuration \
    --cli-input-json file://create-infrastructure-configuration.json
```

Contenuto di `create-infrastructure-configuration.json`:

```
{
    "name": "MyExampleInfrastructure",
    "description": "An example that will retain instances of failed builds",
    "instanceTypes": [
        "m5.large", "m5.xlarge"
    ],
    "instanceProfileName": "EC2InstanceProfileForImageBuilder",
    "securityGroupIds": [
        "sg-a1b2c3d4"
    ],
    "subnetId": "subnet-a1b2c3d4",
    "logging": {
        "s3Logs": {
            "s3BucketName": "bucket-name",
            "s3KeyPrefix": "bucket-path"
        }
    },
    "keyPair": "key-pair-name",
    "terminateInstanceOnFailure": false,
    "snsTopicArn": "arn:aws:sns:us-west-2:123456789012:sns-topic-name"
}
```

Output:

```
{
    "requestId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "clientToken": "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
    "infrastructureConfigurationArn": "arn:aws:imagebuilder:us-
west-2:123456789012:infrastructure-configuration/myexampleinfrastructure"
}
```

Per ulteriori informazioni, vedere Configurazione e gestione di una pipeline di EC2 immagini di Image Builder utilizzando la AWS CLI nella Guida per l'utente di Image EC2 Builder.

- Per i dettagli sull'API, vedere CreateInfrastructureConfigurationin Command Reference.AWS CLI

`delete-component`

Il seguente esempio di codice mostra come utilizzare`delete-component`.

AWS CLI

Come eliminare un componente

L'esempio `delete-component` seguente elimina una versione di build di un componente specificandone l'ARN.

```
aws imagebuilder delete-component \
    --component-build-version-arn arn:aws:imagebuilder:us-
west-2:123456789012:component/myexamplecomponent/2019.12.02/1
```

Output:

```
{
    "requestId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "componentBuildVersionArn": "arn:aws:imagebuilder:us-
west-2:123456789012:component/myexamplecomponent/2019.12.02/1"
}
```

Per ulteriori informazioni, vedere Configurazione e gestione di una pipeline di EC2 immagini di Image Builder utilizzando la AWS CLI nella Guida per l'utente di Image EC2 Builder.

- Per i dettagli sull'API, vedere DeleteComponentin Command Reference.AWS CLI

`delete-image-pipeline`

Il seguente esempio di codice mostra come utilizzare`delete-image-pipeline`.

AWS CLI

Come eliminare una pipeline di immagini

L'esempio `delete-image-pipeline` seguente elimina una configurazione dell'infrastruttura specificandone l'ARN.

```
aws imagebuilder delete-image-pipeline \
    --image-pipeline-arn arn:aws:imagebuilder:us-west-2:123456789012:image-pipeline/my-example-pipeline
```

Output:

```
{
    "requestId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "imagePipelineArn": "arn:aws:imagebuilder:us-west-2:123456789012:image-pipeline/mywindows2016pipeline"
}
```

Per ulteriori informazioni, vedere Configurazione e gestione di una pipeline di EC2 immagini di Image Builder utilizzando la AWS CLI nella Guida per l'utente di Image EC2 Builder.

- Per i dettagli sull'API, vedere DeleteImagePipelinein Command Reference.AWS CLI

`delete-image-recipe`

Il seguente esempio di codice mostra come utilizzare`delete-image-recipe`.

AWS CLI

Come eliminare una ricetta di immagine

L'esempio `delete-image-recipe` seguente elimina una ricetta di immagine specificandone l'ARN.

```
aws imagebuilder delete-image-recipe \
    --image-recipe-arn arn:aws:imagebuilder:us-east-1:123456789012:image-recipe/mybasicrecipe/2019.12.03
```

Output:

```
{
    "requestId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "imageRecipeArn": "arn:aws:imagebuilder:us-west-2:123456789012:image-recipe/mybasicrecipe/2019.12.03"
```

```
}
```

Per ulteriori informazioni, vedere Configurazione e gestione di una pipeline di EC2 immagini di Image Builder utilizzando la AWS CLI nella Guida per l'utente di Image EC2 Builder.

- Per i dettagli sull'API, vedere DeleteImageRecipein Command Reference.AWS CLI

`delete-image`

Il seguente esempio di codice mostra come utilizzare`delete-image`.

AWS CLI

Come eliminare un'immagine

L'esempio `delete-image` seguente elimina una versione di build di un'immagine specificandone l'ARN.

```
aws imagebuilder delete-image \
    --image-build-version-arn arn:aws:imagebuilder:us-west-2:123456789012:image/my-example-image/2019.12.02/1
```

Output:

```
{
    "requestId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "imageBuildVersionArn": "arn:aws:imagebuilder:us-west-2:123456789012:image/mybasicrecipe/2019.12.03/1"
}
```

Per ulteriori informazioni, vedere Configurazione e gestione di una pipeline di EC2 immagini di Image Builder utilizzando la AWS CLI nella Guida per l'utente di Image EC2 Builder.

- Per i dettagli sull'API, vedere DeleteImagein Command Reference.AWS CLI

`delete-infrastructure-configuration`

Il seguente esempio di codice mostra come utilizzare`delete-infrastructure-configuration`.

AWS CLI

Come eliminare una configurazione dell'infrastruttura

L'esempio `delete-infrastructure-configuration` seguente elimina una configurazione dell'infrastruttura specificandone l'ARN.

```
aws imagebuilder delete-infrastructure-configuration \
    --infrastructure-configuration-arn arn:aws:imagebuilder:us-
east-1:123456789012:infrastructure-configuration/myexampleinfrastructure
```

Output:

```
{
    "requestId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "infrastructureConfigurationArn": "arn:aws:imagebuilder:us-
west-2:123456789012:infrastructure-configuration/myexampleinfrastructure"
}
```

Per ulteriori informazioni, vedere Configurazione e gestione di una pipeline di EC2 immagini di Image Builder utilizzando la AWS CLI nella Guida per l'utente di Image EC2 Builder.

- Per i dettagli sull'API, vedere DeleteInfrastructureConfigurationin Command Reference.AWS CLI

`get-component-policy`

Il seguente esempio di codice mostra come utilizzare`get-component-policy`.

AWS CLI

Come ottenere dettagli su una policy sui componenti

L'esempio `get-component-policy` seguente elenca i dettagli di una policy sui componenti specificandone l'ARN.

```
aws imagebuilder get-component-policy \
    --component-arn arn:aws:imagebuilder:us-west-2:123456789012:component/my-
example-component/2019.12.03/1
```

Output:

```
{
    "Policy": "{ "Version":"2012-10-17",        "Statement": [ { "Effect":
 "Allow", "Principal": { "AWS": [ "123456789012" ] }, "Action":
```

```
 [ "imagebuilder:GetComponent", "imagebuilder:ListComponents" ], "Resource":
 [ "arn:aws:imagebuilder:us-west-2:123456789012:component/my-example-
component/2019.12.03/1" ] } ] }"
}
```

Per ulteriori informazioni, vedere Configurazione e gestione di una pipeline di immagini di EC2 Image Builder utilizzando la AWS CLI < https://docs.aws.amazon.com/imagebuilder/ latest/ userguide/managing - image-builder-cli .html>`__ nella Guida per l'utente di Image Builder. EC2

- Per i dettagli sull'API, vedere in Command Reference. GetComponentPolicyAWS CLI

`get-component`

Il seguente esempio di codice mostra come utilizzare`get-component`.

AWS CLI

Come ottenere i dettagli dei componenti

L'esempio `get-component` seguente elenca i dettagli di un componente specificandone l'ARN.

```
aws imagebuilder get-component \
    --component-build-version-arn arn:aws:imagebuilder:us-
west-2:123456789012:component/component-name/1.0.0/1
```

Output:

```
{
    "requestId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "component": {
        "arn": "arn:aws:imagebuilder:us-west-2:123456789012:component/component-
name/1.0.0/1",
        "name": "component-name",
        "version": "1.0.0",
        "type": "TEST",
        "platform": "Linux",
        "owner": "123456789012",
        "data": "name: HelloWorldTestingDocument\ndescription: This is hello world
 testing document.\nschemaVersion: 1.0\n\nphases:\n  - name: test\n    steps:\n
   - name: HelloWorldStep\n        action: ExecuteBash\n        inputs:\n
 commands:\n            - echo \"Hello World! Test.\"\n",
```

```
        "encrypted": true,
        "dateCreated": "2020-01-27T20:43:30.306Z",
        "tags": {}
    }
}
```

Per ulteriori informazioni, vedere Configurazione e gestione di una pipeline di EC2 immagini di Image Builder utilizzando la AWS CLI nella Guida per l'utente di Image EC2 Builder.

- Per i dettagli sull'API, vedere GetComponentin Command Reference.AWS CLI

`get-distribution-configuration`

Il seguente esempio di codice mostra come utilizzare`get-distribution-configuration`.

AWS CLI

Come ottenere i dettagli di una configurazione di distribuzione

L'esempio `get-distribution-configuration` seguente mostra i dettagli di una configurazione di distribuzione specificandone l'ARN.

```
aws imagebuilder get-distribution-configuration \
    --distribution-configuration-arn arn:aws:imagebuilder:us-
west-2:123456789012:distribution-configuration/myexampledistribution
```

Output:

```
{
    "requestId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "distributionConfiguration": {
        "arn": "arn:aws:imagebuilder:us-west-2:123456789012:distribution-
configuration/myexampledistribution",
        "name": "MyExampleDistribution",
        "description": "Copies AMI to eu-west-1 and exports to S3",
        "distributions": [
            {
                "region": "us-west-2",
                "amiDistributionConfiguration": {
                    "name": "Name {{imagebuilder:buildDate}}",
                    "description": "An example image name with parameter
 references",
```

```
                    "amiTags": {
                        "KeyName": "{{ssm:parameter_name}}"
                    },
                    "launchPermission": {
                        "userIds": [
                            "123456789012"
                        ]
                    }
                }
            },
            {
                "region": "eu-west-1",
                "amiDistributionConfiguration": {
                    "name": "My {{imagebuilder:buildVersion}} image
 {{imagebuilder:buildDate}}",
                    "amiTags": {
                        "KeyName": "Value"
                    },
                    "launchPermission": {
                        "userIds": [
                            "123456789012"
                        ]
                    }
                }
            }
        ],
        "dateCreated": "2020-02-19T18:40:10.529Z",
        "tags": {}
    }
}
```

Per ulteriori informazioni, vedere Configurazione e gestione di una pipeline di EC2 immagini di Image Builder utilizzando la AWS CLI nella Guida per l'utente di Image EC2 Builder.

- Per i dettagli sull'API, vedere GetDistributionConfigurationin Command Reference.AWS CLI

`get-image-pipeline`

Il seguente esempio di codice mostra come utilizzare`get-image-pipeline`.

AWS CLI

Come ottenere i dettagli di una pipeline di immagini

L'esempio `get-image-pipeline` seguente elenca i dettagli di una pipeline di immagini specificandone l'ARN.

```
aws imagebuilder get-image-pipeline \
    --image-pipeline-arn arn:aws:imagebuilder:us-west-2:123456789012:image-pipeline/
mywindows2016pipeline
```

Output:

```
{
    "requestId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "imagePipeline": {
        "arn": "arn:aws:imagebuilder:us-west-2:123456789012:image-pipeline/
mywindows2016pipeline",
        "name": "MyWindows2016Pipeline",
        "description": "Builds Windows 2016 Images",
        "platform": "Windows",
        "imageRecipeArn": "arn:aws:imagebuilder:us-west-2:123456789012:image-recipe/
mybasicrecipe/2019.12.03",
        "infrastructureConfigurationArn": "arn:aws:imagebuilder:us-
west-2:123456789012:infrastructure-configuration/myexampleinfrastructure",
        "distributionConfigurationArn": "arn:aws:imagebuilder:us-
west-2:123456789012:distribution-configuration/myexampledistribution",
        "imageTestsConfiguration": {
            "imageTestsEnabled": true,
            "timeoutMinutes": 60
        },
        "schedule": {
            "scheduleExpression": "cron(0 0 * * SUN)",
            "pipelineExecutionStartCondition":
 "EXPRESSION_MATCH_AND_DEPENDENCY_UPDATES_AVAILABLE"
        },
        "status": "ENABLED",
        "dateCreated": "2020-02-19T19:04:01.253Z",
        "dateUpdated": "2020-02-19T19:04:01.253Z",
        "tags": {}
    }
}
```

Per ulteriori informazioni, vedere Configurazione e gestione di una pipeline di EC2 immagini di Image Builder utilizzando la AWS CLI nella Guida per l'utente di Image EC2 Builder.

- Per i dettagli sull'API, vedere GetImagePipelinein Command Reference.AWS CLI

`get-image-policy`

Il seguente esempio di codice mostra come utilizzare`get-image-policy`.

AWS CLI

Come ottenere i dettagli di una policy sulle immagini

L'esempio `get-image-policy` seguente elenca i dettagli di una policy sulle immagini specificandone l'ARN.

```
aws imagebuilder get-image-policy \
    --image-arn arn:aws:imagebuilder:us-west-2:123456789012:image/my-example-image/2019.12.03/1
```

Output:

```
{
    "Policy": "{ "Version":"2012-10-17",        "Statement": [ { "Effect": "Allow",
 "Principal": { "AWS": [ "123456789012" ] }, "Action": [ "imagebuilder:GetImage",
 "imagebuilder:ListImages" ], "Resource": [ "arn:aws:imagebuilder:us-
west-2:123456789012:image/my-example-image/2019.12.03/1" ] } ] }"
}
```

Per ulteriori informazioni, vedere Configurazione e gestione di una pipeline di EC2 immagini di Image Builder utilizzando la AWS CLI nella Guida per l'utente di Image EC2 Builder.

- Per i dettagli sull'API, vedere GetImagePolicyin Command Reference.AWS CLI

`get-image-recipe-policy`

Il seguente esempio di codice mostra come utilizzare`get-image-recipe-policy`.

AWS CLI

Come ottenere i dettagli di una policy sulle ricette

L'esempio `get-image-recipe-policy` seguente elenca i dettagli di una policy sulle ricette di immagini specificandone l'ARN.

```
aws imagebuilder get-image-recipe-policy \
```

```
    --image-recipe-arn arn:aws:imagebuilder:us-west-2:123456789012:image-recipe/my-example-image-recipe/2019.12.03/1
```

Output:

```
{
    "Policy": "{ "Version":"2012-10-17",        "Statement": [ { "Effect":
 "Allow", "Principal": { "AWS": [ "123456789012" ] }, "Action":
 [ "imagebuilder:GetImageRecipe", "imagebuilder:ListImageRecipes" ], "Resource":
 [ "arn:aws:imagebuilder:us-west-2:123456789012:image-recipe/my-example-image-
recipe/2019.12.03/1" ] } ] }"
}
```

Per ulteriori informazioni, vedere Configurazione e gestione di una pipeline di EC2 immagini di Image Builder utilizzando la AWS CLI nella Guida per l'utente di Image EC2 Builder.

- Per i dettagli sull'API, vedere GetImageRecipePolicyin Command Reference.AWS CLI

`get-image`

Il seguente esempio di codice mostra come utilizzare`get-image`.

AWS CLI

Come ottenere i dettagli di un'immagine

L'esempio `get-image` seguente elenca i dettagli di un'immagine specificandone l'ARN.

```
aws imagebuilder get-image \
    --image-build-version-arn arn:aws:imagebuilder:us-west-2:123456789012:image/mybasicrecipe/2019.12.03/1
```

Output:

```
{
    "requestId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "image": {
        "arn": "arn:aws:imagebuilder:us-west-2:123456789012:image/
mybasicrecipe/2019.12.03/1",
        "name": "MyBasicRecipe",
        "version": "2019.12.03/1",
        "platform": "Windows",
```

```
        "state": {
            "status": "BUILDING"
        },
        "imageRecipe": {
            "arn": "arn:aws:imagebuilder:us-west-2:123456789012:image-recipe/
mybasicrecipe/2019.12.03",
            "name": "MyBasicRecipe",
            "description": "This example image recipe creates a Windows 2016
 image.",
            "platform": "Windows",
            "version": "2019.12.03",
            "components": [
                {
                    "componentArn": "arn:aws:imagebuilder:us-
west-2:123456789012:component/myexamplecomponent/2019.12.02/1"
                },
                {
                    "componentArn": "arn:aws:imagebuilder:us-
west-2:123456789012:component/myimportedcomponent/1.0.0/1"
                }
            ],
            "parentImage": "arn:aws:imagebuilder:us-west-2:aws:image/windows-
server-2016-english-full-base-x86/2019.12.17/1",
            "dateCreated": "2020-02-14T19:46:16.904Z",
            "tags": {}
        },
        "infrastructureConfiguration": {
            "arn": "arn:aws:imagebuilder:us-west-2:123456789012:infrastructure-
configuration/myexampleinfrastructure",
            "name": "MyExampleInfrastructure",
            "description": "An example that will retain instances of failed builds",
            "instanceTypes": [
                "m5.large",
                "m5.xlarge"
            ],
            "instanceProfileName": "EC2InstanceProfileForImageFactory",
            "securityGroupIds": [
                "sg-a1b2c3d4"
            ],
            "subnetId": "subnet-a1b2c3d4",
            "logging": {
                "s3Logs": {
                    "s3BucketName": "bucket-name",
                    "s3KeyPrefix": "bucket-path"
```

```
                }
            },
            "keyPair": "Sam",
            "terminateInstanceOnFailure": false,
            "snsTopicArn": "arn:aws:sns:us-west-2:123456789012:sns-name",
            "dateCreated": "2020-02-14T21:21:05.098Z",
            "tags": {}
        },
        "imageTestsConfiguration": {
            "imageTestsEnabled": true,
            "timeoutMinutes": 720
        },
        "dateCreated": "2020-02-14T23:14:13.597Z",
        "outputResources": {
            "amis": []
        },
        "tags": {}
    }
}
```

Per ulteriori informazioni, vedere Configurazione e gestione di una pipeline di EC2 immagini di Image Builder utilizzando la AWS CLI nella Guida per l'utente di Image EC2 Builder.

- Per i dettagli sull'API, vedere GetImagein Command Reference.AWS CLI

`get-infrastructure-configuration`

Il seguente esempio di codice mostra come utilizzare`get-infrastructure-configuration`.

AWS CLI

Come ottenere i dettagli di una configurazione dell'infrastruttura

L'esempio `get-infrastructure-configuration` seguente elenca i dettagli di una configurazione dell'infrastruttura specificandone l'ARN.

```
aws imagebuilder get-infrastructure-configuration \
    --infrastructure-configuration-arn arn:aws:imagebuilder:us-
west-2:123456789012:infrastructure-configuration/myexampleinfrastructure
```

Output:

```
{
```

```
    "requestId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "infrastructureConfiguration": {
        "arn": "arn:aws:imagebuilder:us-west-2:123456789012:infrastructure-
configuration/myexampleinfrastructure",
        "name": "MyExampleInfrastructure",
        "description": "An example that will retain instances of failed builds",
        "instanceTypes": [
            "m5.large",
            "m5.xlarge"
        ],
        "instanceProfileName": "EC2InstanceProfileForImageBuilder",
        "securityGroupIds": [
            "sg-a48c95ef"
        ],
        "subnetId": "subnet-a48c95ef",
        "logging": {
            "s3Logs": {
                "s3BucketName": "bucket-name",
                "s3KeyPrefix": "bucket-path"
            }
        },
        "keyPair": "Name",
        "terminateInstanceOnFailure": false,
        "snsTopicArn": "arn:aws:sns:us-west-2:123456789012:sns-name",
        "dateCreated": "2020-02-19T19:11:51.858Z",
        "tags": {}
    }
}
```

Per ulteriori informazioni, vedere Configurazione e gestione di una pipeline di EC2 immagini di Image Builder utilizzando la AWS CLI nella Guida per l'utente di Image EC2 Builder.

- Per i dettagli sull'API, vedere GetInfrastructureConfigurationin Command Reference.AWS CLI

`import-component`

Il seguente esempio di codice mostra come utilizzare`import-component`.

AWS CLI

Come importare un componente

L'esempio `import-component` seguente importa uno script preesistente utilizzando un file JSON.

```
aws imagebuilder import-component \
    --cli-input-json file://import-component.json
```

Contenuto di `import-component.json`:

```
{
    "name": "MyImportedComponent",
    "semanticVersion": "1.0.0",
    "description": "An example of how to import a component",
    "changeDescription": "First commit message.",
    "format": "SHELL",
    "platform": "Windows",
    "type": "BUILD",
    "uri": "s3://s3-bucket-name/s3-bucket-path/component.yaml"
}
```

Output:

```
{
    "requestId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "clientToken": "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
    "componentBuildVersionArn": "arn:aws:imagebuilder:us-
west-2:123456789012:component/myimportedcomponent/1.0.0/1"
}
```

Per ulteriori informazioni, vedere Configurazione e gestione di una pipeline di EC2 immagini di Image Builder utilizzando la AWS CLI nella Guida per l'utente di Image EC2 Builder.

- Per i dettagli sull'API, vedere ImportComponentin Command Reference.AWS CLI

`list-component-build-versions`

Il seguente esempio di codice mostra come utilizzare`list-component-build-versions`.

AWS CLI

Come elencare le versioni di build dei componenti

L'esempio `list-component-build-versions` seguente elenca le versioni di build dei componenti con una determinata versione semantica.

```
aws imagebuilder list-component-build-versions --component-
version-arn arn:aws:imagebuilder:us-west-2:123456789012:component/
myexamplecomponent/2019.12.02
```

Output:

```
{
    "requestId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "componentSummaryList": [
        {
            "arn": "arn:aws:imagebuilder:us-west-2:123456789012:component/
myexamplecomponent/2019.12.02/1",
            "name": "MyExampleComponent",
            "version": "2019.12.02",
            "platform": "Windows",
            "type": "BUILD",
            "owner": "123456789012",
            "description": "An example component that builds, validates and tests an
 image",
            "changeDescription": "Initial version.",
            "dateCreated": "2020-02-19T18:53:45.940Z",
            "tags": {
                "KeyName": "KeyValue"
            }
        }
    ]
}
```

Per ulteriori informazioni, vedere Configurazione e gestione di una pipeline di EC2 immagini di Image Builder utilizzando la AWS CLI nella Guida per l'utente di Image EC2 Builder.

- Per i dettagli sull'API, vedere ListComponentBuildVersionsin Command Reference.AWS CLI

`list-components`

Il seguente esempio di codice mostra come utilizzare`list-components`.

AWS CLI

Come elencare tutte le versioni semantiche dei componenti

L'esempio `list-components` seguente elenca tutte le versioni semantiche dei componenti a cui puoi accedere. Facoltativamente, puoi applicare un filtro per elencare i componenti di tua proprietà, di proprietà di Amazon o che sono stati condivisi con te da altri account.

```
aws imagebuilder list-components
```

Output:

```
{
    "requestId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "componentVersionList": [
        {
            "arn": "arn:aws:imagebuilder:us-west-2:123456789012:component/component-
name/1.0.0",
            "name": "component-name",
            "version": "1.0.0",
            "platform": "Linux",
            "type": "TEST",
            "owner": "123456789012",
            "dateCreated": "2020-01-27T20:43:30.306Z"
        }
    ]
}
```

Per ulteriori informazioni, vedere Configurazione e gestione di una pipeline di EC2 immagini di Image Builder utilizzando la AWS CLI nella Guida per l'utente di Image EC2 Builder.

- Per i dettagli sull'API, vedere ListComponentsin Command Reference.AWS CLI

`list-distribution-configurations`

Il seguente esempio di codice mostra come utilizzare`list-distribution-configurations`.

AWS CLI

Come elencare le distribuzioni

L'esempio `list-distribution-configurations` seguente elenca tutte le tue distribuzioni.

```
aws imagebuilder list-distribution-configurations
```

Output:

```
{
    "requestId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "distributionConfigurationSummaryList": [
        {
            "arn": "arn:aws:imagebuilder:us-west-2:123456789012:distribution-
configuration/myexampledistribution",
            "name": "MyExampleDistribution",
            "description": "Copies AMI to eu-west-1 and exports to S3",
            "dateCreated": "2020-02-19T18:40:10.529Z",
            "tags": {
                "KeyName": "KeyValue"
            }
        }
    ]
}
```

Per ulteriori informazioni, vedere Configurazione e gestione di una pipeline di EC2 immagini di Image Builder utilizzando la AWS CLI nella Guida per l'utente di Image EC2 Builder.

- Per i dettagli sull'API, vedere ListDistributionConfigurationsin Command Reference.AWS CLI

`list-image-build-versions`

Il seguente esempio di codice mostra come utilizzare`list-image-build-versions`.

AWS CLI

Come elencare le versioni di build di un'immagine

L'esempio `list-image-build-versions` seguente elenca tutte le versioni di build di un'immagine con una versione semantica.

```
aws imagebuilder list-image-build-versions \
    --image-version-arn arn:aws:imagebuilder:us-west-2:123456789012:image/
mybasicrecipe/2019.12.03
```

Output:

```
{
    "requestId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "imageSummaryList": [
        {
            "arn": "arn:aws:imagebuilder:us-west-2:123456789012:image/
mybasicrecipe/2019.12.03/7",
            "name": "MyBasicRecipe",
            "version": "2019.12.03/7",
            "platform": "Windows",
            "state": {
                "status": "FAILED",
                "reason": "Can't start SSM Automation for arn
 arn:aws:imagebuilder:us-west-2:123456789012:image/mybasicrecipe/2019.12.03/7 during
 building. Parameter \"iamInstanceProfileName\" has a null value."
            },
            "owner": "123456789012",
            "dateCreated": "2020-02-19T18:56:11.511Z",
            "outputResources": {
                "amis": []
            },
            "tags": {}
        },
        {
            "arn": "arn:aws:imagebuilder:us-west-2:123456789012:image/
mybasicrecipe/2019.12.03/6",
            "name": "MyBasicRecipe",
            "version": "2019.12.03/6",
            "platform": "Windows",
            "state": {
                "status": "FAILED",
                "reason": "An internal error has occurred."
            },
            "owner": "123456789012",
            "dateCreated": "2020-02-18T22:49:08.142Z",
            "outputResources": {
                "amis": [
                    {
                        "region": "us-west-2",
                        "image": "ami-a1b2c3d4567890ab",
                        "name": "MyBasicRecipe 2020-02-18T22-49-38.704Z",
                        "description": "This example image recipe creates a Windows
 2016 image."
                    },
```

```
                    {
                        "region": "us-west-2",
                        "image": "ami-a1b2c3d4567890ab",
                        "name": "Name 2020-02-18T22-49-08.131Z",
                        "description": "Copies AMI to eu-west-2 and exports to S3"
                    },
                    {
                        "region": "eu-west-2",
                        "image": "ami-a1b2c3d4567890ab",
                        "name": "My 6 image 2020-02-18T22-49-08.131Z",
                        "description": "Copies AMI to eu-west-2 and exports to S3"
                    }
                ]
            },
            "tags": {}
        },
        {
            "arn": "arn:aws:imagebuilder:us-west-2:123456789012:image/
mybasicrecipe/2019.12.03/5",
            "name": "MyBasicRecipe",
            "version": "2019.12.03/5",
            "platform": "Windows",
            "state": {
                "status": "AVAILABLE"
            },
            "owner": "123456789012",
            "dateCreated": "2020-02-18T16:51:48.403Z",
            "outputResources": {
                "amis": [
                    {
                        "region": "us-west-2",
                        "image": "ami-a1b2c3d4567890ab",
                        "name": "MyBasicRecipe 2020-02-18T16-52-18.965Z",
                        "description": "This example image recipe creates a Windows
 2016 image."
                    }
                ]
            },
            "tags": {}
        },
        {
            "arn": "arn:aws:imagebuilder:us-west-2:123456789012:image/
mybasicrecipe/2019.12.03/4",
            "name": "MyBasicRecipe",
```

```
            "version": "2019.12.03/4",
            "platform": "Windows",
            "state": {
                "status": "AVAILABLE"
            },
            "owner": "123456789012",
            "dateCreated": "2020-02-18T16:50:01.827Z",
            "outputResources": {
                "amis": [
                    {
                        "region": "us-west-2",
                        "image": "ami-a1b2c3d4567890ab",
                        "name": "MyBasicRecipe 2020-02-18T16-50-32.280Z",
                        "description": "This example image recipe creates a Windows
 2016 image."
                    }
                ]
            },
            "tags": {}
        },
        {
            "arn": "arn:aws:imagebuilder:us-west-2:123456789012:image/
mybasicrecipe/2019.12.03/3",
            "name": "MyBasicRecipe",
            "version": "2019.12.03/3",
            "platform": "Windows",
            "state": {
                "status": "AVAILABLE"
            },
            "owner": "123456789012",
            "dateCreated": "2020-02-14T23:14:13.597Z",
            "outputResources": {
                "amis": [
                    {
                        "region": "us-west-2",
                        "image": "ami-a1b2c3d4567890ab",
                        "name": "MyBasicRecipe 2020-02-14T23-14-44.243Z",
                        "description": "This example image recipe creates a Windows
 2016 image."
                    }
                ]
            },
            "tags": {}
        },
```

```
        {
            "arn": "arn:aws:imagebuilder:us-west-2:123456789012:image/
mybasicrecipe/2019.12.03/2",
            "name": "MyBasicRecipe",
            "version": "2019.12.03/2",
            "platform": "Windows",
            "state": {
                "status": "FAILED",
                "reason": "SSM execution 'a1b2c3d4-5678-90ab-cdef-EXAMPLE11111'
 failed with status = 'Failed' and failure message = 'Step fails when it is
 verifying the command has completed. Command a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
 returns unexpected invocation result: \n{Status=[Failed], ResponseCode=[1],
 Output=[\n----------ERROR-------\nfailed to run commands: exit status 1],
 OutputPayload=[{\"Status\":\"Failed\",\"ResponseCode\":1,\"Output\":\"\
\n----------ERROR-------\\nfailed to run commands: exit status 1\",\"CommandId\":
\"a1b2c3d4-5678-90ab-cdef-EXAMPLE11111\"}], CommandId=[a1b2c3d4-5678-90ab-cdef-
EXAMPLE11111]}. Please refer to Automation Service Troubleshooting Guide for more
 diagnosis details.'"
            },
            "owner": "123456789012",
            "dateCreated": "2020-02-14T22:57:42.593Z",
            "outputResources": {
                "amis": []
            },
            "tags": {}
        }
    ]
}
```

Per ulteriori informazioni, vedere Configurazione e gestione di una pipeline di EC2 immagini di Image Builder utilizzando la AWS CLI nella Guida per l'utente di Image EC2 Builder.

- Per i dettagli sull'API, vedere ListImageBuildVersionsin Command Reference.AWS CLI

`list-image-pipeline-images`

Il seguente esempio di codice mostra come utilizzare`list-image-pipeline-images`.

AWS CLI

Come elencare le immagini di una pipeline di immagini

L'esempio `list-image-pipeline-images` seguente elenca tutte le immagini create da una determinata pipeline di immagini.

```
aws imagebuilder list-image-pipeline-images \
    --image-pipeline-arn arn:aws:imagebuilder:us-west-2:123456789012:image-pipeline/
mywindows2016pipeline
```

Output:

```
{
    "requestId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "imagePipelineList": [
        {
            "arn": "arn:aws:imagebuilder:us-west-2:123456789012:image-pipeline/
mywindows2016pipeline",
            "name": "MyWindows2016Pipeline",
            "description": "Builds Windows 2016 Images",
            "platform": "Windows",
            "imageRecipeArn": "arn:aws:imagebuilder:us-west-2:123456789012:image-
recipe/mybasicrecipe/2019.12.03",
            "infrastructureConfigurationArn": "arn:aws:imagebuilder:us-
west-2:123456789012:infrastructure-configuration/myexampleinfrastructure",
            "distributionConfigurationArn": "arn:aws:imagebuilder:us-
west-2:123456789012:distribution-configuration/myexampledistribution",
            "imageTestsConfiguration": {
                "imageTestsEnabled": true,
                "timeoutMinutes": 60
            },
            "schedule": {
                "scheduleExpression": "cron(0 0 * * SUN)",
                "pipelineExecutionStartCondition":
 "EXPRESSION_MATCH_AND_DEPENDENCY_UPDATES_AVAILABLE"
            },
            "status": "ENABLED",
            "dateCreated": "2020-02-19T19:04:01.253Z",
            "dateUpdated": "2020-02-19T19:04:01.253Z",
            "tags": {
                "KeyName": "KeyValue"
            }
        },
        {
            "arn": "arn:aws:imagebuilder:us-west-2:123456789012:image-pipeline/sam",
            "name": "PipelineName",
```

```
            "platform": "Linux",
            "imageRecipeArn": "arn:aws:imagebuilder:us-west-2:123456789012:image-
recipe/recipe-name-a1b2c3d45678/1.0.0",
            "infrastructureConfigurationArn": "arn:aws:imagebuilder:us-
west-2:123456789012:infrastructure-configuration/infrastructureconfiguration-name-
a1b2c3d45678",
            "imageTestsConfiguration": {
                "imageTestsEnabled": true,
                "timeoutMinutes": 720
            },
            "status": "ENABLED",
            "dateCreated": "2019-12-16T18:19:02.068Z",
            "dateUpdated": "2019-12-16T18:19:02.068Z",
            "tags": {
                "KeyName": "KeyValue"
            }
        }
    ]
}
```

Per ulteriori informazioni, vedere Configurazione e gestione di una pipeline di EC2 immagini di Image Builder utilizzando la AWS CLI nella Guida per l'utente di Image EC2 Builder.

- Per i dettagli sull'API, vedere ListImagePipelineImagesin Command Reference.AWS CLI

`list-image-recipes`

Il seguente esempio di codice mostra come utilizzare`list-image-recipes`.

AWS CLI

Come elencare le ricette di immagini

L'esempio `list-image-recipes` seguente elenca tutte le tue ricette di immagini.

```
aws imagebuilder list-image-recipes
```

Output:

```
{
    "requestId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "imageRecipeSummaryList": [
```

```
        {
            "arn": "arn:aws:imagebuilder:us-west-2:123456789012:image-recipe/
mybasicrecipe/2019.12.03",
            "name": "MyBasicRecipe",
            "platform": "Windows",
            "owner": "123456789012",
            "parentImage": "arn:aws:imagebuilder:us-west-2:aws:image/windows-
server-2016-english-full-base-x86/2019.x.x",
            "dateCreated": "2020-02-19T18:54:25.975Z",
            "tags": {
                "KeyName": "KeyValue"
            }
        },
        {
            "arn": "arn:aws:imagebuilder:us-west-2:123456789012:image-recipe/recipe-
name-a1b2c3d45678/1.0.0",
            "name": "recipe-name-a1b2c3d45678",
            "platform": "Linux",
            "owner": "123456789012",
            "parentImage": "arn:aws:imagebuilder:us-west-2:aws:image/amazon-linux-2-
x86/2019.11.21",
            "dateCreated": "2019-12-16T18:19:00.120Z",
            "tags": {
                "KeyName": "KeyValue"
            }
        }
    ]
}
```

Per ulteriori informazioni, vedere Configurazione e gestione di una pipeline di EC2 immagini di Image Builder utilizzando la AWS CLI nella Guida per l'utente di Image EC2 Builder.

- Per i dettagli sull'API, vedere ListImageRecipesin Command Reference.AWS CLI

`list-images`

Il seguente esempio di codice mostra come utilizzare`list-images`.

AWS CLI

Come elencare le immagini

L'esempio `list-images` seguente elenca tutte le versioni semantiche a cui hai accesso.

```
aws imagebuilder list-images
```

Output:

```
{
    "requestId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "imageVersionList": [
        {
            "arn": "arn:aws:imagebuilder:us-west-2:123456789012:image/
mybasicrecipe/2019.12.03",
            "name": "MyBasicRecipe",
            "version": "2019.12.03",
            "platform": "Windows",
            "owner": "123456789012",
            "dateCreated": "2020-02-14T21:29:18.810Z"
        }
    ]
}
```

Per ulteriori informazioni, vedere Configurazione e gestione di una pipeline di EC2 immagini di Image Builder utilizzando la AWS CLI nella Guida per l'utente di Image EC2 Builder.

- Per i dettagli sull'API, vedere ListImagesin Command Reference.AWS CLI

`list-infrastructure-configurations`

Il seguente esempio di codice mostra come utilizzare`list-infrastructure-configurations`.

AWS CLI

Come elencare le configurazioni dell'infrastruttura

L'esempio `list-infrastructure-configurations` seguente elenca tutte le configurazioni dell'infrastruttura.

```
aws imagebuilder list-infrastructure-configurations
```

Output:

```
{
```

```
    "requestId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "infrastructureConfigurationSummaryList": [
        {
            "arn": "arn:aws:imagebuilder:us-west-2:123456789012:infrastructure-
configuration/myexampleinfrastructure",
            "name": "MyExampleInfrastructure",
            "description": "An example that will retain instances of failed builds",
            "dateCreated": "2020-02-19T19:11:51.858Z",
            "tags": {}
        },
        {
            "arn": "arn:aws:imagebuilder:us-west-2:123456789012:infrastructure-
configuration/infrastructureconfiguration-name-a1b2c3d45678",
            "name": "infrastructureConfiguration-name-a1b2c3d45678",
            "dateCreated": "2019-12-16T18:19:01.038Z",
            "tags": {
                "KeyName": "KeyValue"
            }
        }
    ]
}
```

Per ulteriori informazioni, vedere Configurazione e gestione di una pipeline di EC2 immagini di Image Builder utilizzando la AWS CLI nella Guida per l'utente di Image EC2 Builder.

- Per i dettagli sull'API, vedere ListInfrastructureConfigurationsin Command Reference.AWS CLI

`list-tags-for-resource`

Il seguente esempio di codice mostra come utilizzare`list-tags-for-resource`.

AWS CLI

Come elencare i tag per una determinata risorsa

L'esempio `list-tags-for-resource` seguente elenca tutti i tag per una determinata risorsa.

```
aws imagebuilder list-tags-for-resource \
    --resource-arn arn:aws:imagebuilder:us-west-2:123456789012:image-pipeline/
mywindows2016pipeline
```

Output:

```
{
    "tags": {
        "KeyName": "KeyValue"
    }
}
```

Per ulteriori informazioni, vedere Configurazione e gestione di una pipeline di EC2 immagini di Image Builder utilizzando la AWS CLI nella Guida per l'utente di Image EC2 Builder.

- Per i dettagli sull'API, vedere ListTagsForResourcein Command Reference.AWS CLI

`put-component-policy`

Il seguente esempio di codice mostra come utilizzare`put-component-policy`.

AWS CLI

Come applicare una policy delle risorse a un componente

Il comando `put-component-policy` seguente applica una policy delle risorse a un componente di compilazione per consentire la condivisione dei componenti di compilazione tra più account. È consigliabile utilizzare il comando `create-resource-share` della CLI di RAM. Se si utilizza il comando CLI EC2 Image Builder`put-component-policy`, è necessario utilizzare anche il comando RAM CLI affinché la risorsa sia visibile a tutti i principali con cui è condivisa la risorsa. `promote-resource-share-create-from-policy`

```
aws imagebuilder put-component-policy \
    --component-arn arn:aws:imagebuilder:us-west-2:123456789012:component/
examplecomponent/2019.12.02/1 \
    --policy '{ "Version":"2012-10-17",        "Statement": [ { "Effect":
 "Allow", "Principal": { "AWS": [ "123456789012" ] }, "Action":
 [ "imagebuilder:GetComponent", "imagebuilder:ListComponents" ],
 "Resource": [ "arn:aws:imagebuilder:us-west-2:123456789012:component/
examplecomponent/2019.12.02/1" ] } ] }'
```

Output:

```
{
    "requestId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "componentArn": "arn:aws:imagebuilder:us-west-2:123456789012:component/
examplecomponent/2019.12.02/1"
```

```
}
```

Per ulteriori informazioni, vedere Configurazione e gestione di una pipeline di EC2 immagini di Image Builder utilizzando la AWS CLI nella Guida per l'utente di Image EC2 Builder.

- Per i dettagli sull'API, vedere PutComponentPolicyin Command Reference.AWS CLI

`put-image-policy`

Il seguente esempio di codice mostra come utilizzare`put-image-policy`.

AWS CLI

Come applicare una policy delle risorse a un'immagine

Il comando `put-image-policy` seguente applica una policy delle risorse a un'immagine per consentire la condivisione delle immagini tra più account. È consigliabile utilizzare il comando create-resource-share della CLI di RAM. Se si utilizza il comando CLI EC2 Image Builder put-image-policy, è necessario utilizzare anche il comando RAM CLI promote-resource-share-create -from-policy affinché la risorsa sia visibile a tutti i principali con cui è condivisa la risorsa.

```
aws imagebuilder put-image-policy \
    --image-arn arn:aws:imagebuilder:us-west-2:123456789012:image/example-image/2019.12.02/1 \
    --policy '{ "Version":"2012-10-17",        "Statement": [ { "Effect": "Allow", "Principal": { "AWS": [ "123456789012" ] }, "Action": [ "imagebuilder:GetImage", "imagebuilder:ListImages" ], "Resource": [ "arn:aws:imagebuilder:us-west-2:123456789012:image/example-image/2019.12.02/1" ] } ] }'
```

Output:

```
{
    "requestId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "imageArn": "arn:aws:imagebuilder:us-west-2:123456789012:image/example-image/2019.12.02/1"
}
```

Per ulteriori informazioni, vedere Configurazione e gestione di una pipeline di EC2 immagini di Image Builder utilizzando la AWS CLI nella Guida per l'utente di Image EC2 Builder.

- Per i dettagli sull'API, vedere PutImagePolicyin Command Reference.AWS CLI

`put-image-recipe-policy`

Il seguente esempio di codice mostra come utilizzare`put-image-recipe-policy`.

AWS CLI

Come applicare una policy delle risorse a una ricetta di immagine

Il comando `put-image-recipe-policy` seguente applica una policy delle risorse a una ricetta di immagine per consentire la condivisione delle ricette di immagini tra più account. È consigliabile utilizzare il comando `create-resource-share` della CLI di RAM. Se si utilizza il comando CLI EC2 Image Builder`put-image-recipe-policy`, è necessario utilizzare anche il comando RAM CLI affinché la risorsa sia visibile a tutti i principali con cui è condivisa la risorsa. `promote-resource-share-create-from-policy`

```
aws imagebuilder put-image-recipe-policy \
    --image-recipe-arn arn:aws:imagebuilder:us-west-2:123456789012:image-recipe/
example-image-recipe/2019.12.02 \
    --policy '{ "Version":"2012-10-17",        "Statement": [ { "Effect":
 "Allow", "Principal": { "AWS": [ "123456789012" ] }, "Action":
 [ "imagebuilder:GetImageRecipe", "imagebuilder:ListImageRecipes" ], "Resource":
 [ "arn:aws:imagebuilder:us-west-2:123456789012:image-recipe/example-image-
recipe/2019.12.02" ] } ] }'
```

Output:

```
{
    "requestId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "imageRecipeArn": "arn:aws:imagebuilder:us-west-2:123456789012:image-recipe/
example-image-recipe/2019.12.02/1"
}
```

Per ulteriori informazioni, vedere Configurazione e gestione di una pipeline di EC2 immagini di Image Builder utilizzando la AWS CLI nella Guida per l'utente di Image EC2 Builder.

- Per i dettagli sull'API, vedere PutImageRecipePolicyin Command Reference.AWS CLI

`start-image-pipeline-execution`

Il seguente esempio di codice mostra come utilizzare`start-image-pipeline-execution`.

AWS CLI

Come avviare manualmente una pipeline di immagini

L'esempio `start-image-pipeline-execution` seguente avvia manualmente una pipeline di immagini.

```
aws imagebuilder start-image-pipeline-execution \
    --image-pipeline-arn arn:aws:imagebuilder:us-west-2:123456789012:image-pipeline/
mywindows2016pipeline
```

Output:

```
{
    "requestId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "clientToken": "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
    "imageBuildVersionArn": "arn:aws:imagebuilder:us-west-2:123456789012:image/
mybasicrecipe/2019.12.03/1"
}
```

Per ulteriori informazioni, vedere Configurazione e gestione di una pipeline di EC2 immagini di Image Builder utilizzando la AWS CLI nella Guida per l'utente di Image EC2 Builder.

- Per i dettagli sull'API, vedere StartImagePipelineExecutionin Command Reference.AWS CLI

`tag-resource`

Il seguente esempio di codice mostra come utilizzare`tag-resource`.

AWS CLI

Come taggare a una risorsa

L'`tag-resource`esempio seguente aggiunge e contrassegna una risorsa a EC2 Image Builder utilizzando un file JSON.

```
aws imagebuilder tag-resource \
    --cli-input-json file://tag-resource.json
```

Contenuto di `tag-resource.json`:

```
{
    "resourceArn": "arn:aws:imagebuilder:us-west-2:123456789012:image-pipeline/
mywindows2016pipeline",
    "tags": {
        "KeyName: "KeyValue"
    }
}
```

Questo comando non produce alcun output.

Per ulteriori informazioni, vedere Configurazione e gestione di una pipeline di EC2 immagini di Image Builder utilizzando la AWS CLI nella Guida per l'utente di Image EC2 Builder.

- Per i dettagli sull'API, vedere TagResourcein Command Reference.AWS CLI

`untag-resource`

Il seguente esempio di codice mostra come utilizzare`untag-resource`.

AWS CLI

Come rimuovere un tag da una risorsa

L'esempio `untag-resource` seguente rimuove un tag da una risorsa utilizzando un file JSON.

```
aws imagebuilder untag-resource \
    --cli-input-json file://tag-resource.json
```

Contenuto di `untag-resource.json`:

```
{
    "resourceArn": "arn:aws:imagebuilder:us-west-2:123456789012:image-pipeline/
mywindows2016pipeline",
    "tagKeys": [
        "KeyName"
    ]
}
```

Questo comando non produce alcun output.

Per ulteriori informazioni, vedere Configurazione e gestione di una pipeline di EC2 immagini di Image Builder utilizzando la AWS CLI nella Guida per l'utente di Image EC2 Builder.

- Per i dettagli sull'API, vedere UntagResourcein Command Reference.AWS CLI

`update-distribution-configuration`

Il seguente esempio di codice mostra come utilizzareu`pdate-distribution-configuration`.

AWS CLI

Come aggiornare una configurazione di distribuzione

L'esempio `update-distribution-configuration` seguente aggiorna una configurazione di distribuzione utilizzando un file JSON.

```
aws imagebuilder update-distribution-configuration \
    --cli-input-json file://update-distribution-configuration.json
```

Contenuto di `update-distribution-configuration.json`:

```
{
    "distributionConfigurationArn": "arn:aws:imagebuilder:us-
west-2:123456789012:distribution-configuration/myexampledistribution",
    "description": "Copies AMI to eu-west-2 and exports to S3",
    "distributions": [
        {
            "region": "us-west-2",
            "amiDistributionConfiguration": {
                "name": "Name {{imagebuilder:buildDate}}",
                "description": "An example image name with parameter references"
            }
        },
        {
            "region": "eu-west-2",
            "amiDistributionConfiguration": {
                "name": "My {{imagebuilder:buildVersion}} image
 {{imagebuilder:buildDate}}"
            }
        }
    ]
}
```

Output:

```
{
    "requestId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
}
```

Per ulteriori informazioni, vedere Configurazione e gestione di una pipeline di EC2 immagini di Image Builder utilizzando la AWS CLI nella Guida per l'utente di Image EC2 Builder.

- Per i dettagli sull'API, vedere UpdateDistributionConfigurationin Command Reference.AWS CLI

`update-image-pipeline`

Il seguente esempio di codice mostra come utilizzareu`pdate-image-pipeline`.

AWS CLI

Come aggiornare una pipeline di immagini

L'esempio `update-image-pipeline` seguente aggiorna una pipeline di immagini utilizzando un file JSON.

```
aws imagebuilder update-image-pipeline \
    --cli-input-json file://update-image-pipeline.json
```

Contenuto di `update-image-pipeline.json`:

```
{
    "imagePipelineArn": "arn:aws:imagebuilder:us-west-2:123456789012:image-pipeline/
mywindows2016pipeline",
    "imageRecipeArn": "arn:aws:imagebuilder:us-west-2:123456789012:image-recipe/
mybasicrecipe/2019.12.03",
    "infrastructureConfigurationArn": "arn:aws:imagebuilder:us-
west-2:123456789012:infrastructure-configuration/myexampleinfrastructure",
    "distributionConfigurationArn": "arn:aws:imagebuilder:us-
west-2:123456789012:distribution-configuration/myexampledistribution",
    "imageTestsConfiguration": {
        "imageTestsEnabled": true,
        "timeoutMinutes": 120
    },
    "schedule": {
        "scheduleExpression": "cron(0 0 * * MON)",
        "pipelineExecutionStartCondition":
 "EXPRESSION_MATCH_AND_DEPENDENCY_UPDATES_AVAILABLE"
```

```
    },
    "status": "DISABLED"
}
```

Output:

```
{
    "requestId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
}
```

Per ulteriori informazioni, vedere Configurazione e gestione di una pipeline di EC2 immagini di Image Builder utilizzando la AWS CLI nella Guida per l'utente di Image EC2 Builder.

- Per i dettagli sull'API, vedere UpdateImagePipelinein Command Reference.AWS CLI

`update-infrastructure-configuration`

Il seguente esempio di codice mostra come utilizzareupdate-infrastructure-configuration.

AWS CLI

Come aggiornare una configurazione dell'infrastruttura

L'esempio update-infrastructure-configuration seguente aggiorna una configurazione dell'infrastruttura utilizzando un file JSON.

```
aws imagebuilder update-infrastructure-configuration \
    --cli-input-json file:/update-infrastructure-configuration.json
```

Contenuto di update-infrastructure-configuration.json:

```
{
    "infrastructureConfigurationArn": "arn:aws:imagebuilder:us-
west-2:123456789012:infrastructure-configuration/myexampleinfrastructure",
    "description": "An example that will terminate instances of failed builds",
    "instanceTypes": [
        "m5.large", "m5.2xlarge"
    ],
    "instanceProfileName": "EC2InstanceProfileForImageFactory",
    "securityGroupIds": [
```

```
        "sg-a48c95ef"
    ],
    "subnetId": "subnet-a48c95ef",
    "logging": {
        "s3Logs": {
            "s3BucketName": "bucket-name",
            "s3KeyPrefix": "bucket-path"
        }
    },
    "terminateInstanceOnFailure": true,
    "snsTopicArn": "arn:aws:sns:us-west-2:123456789012:sns-name"
}
```

Output:

```
{
    "requestId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
}
```

Per ulteriori informazioni, vedere Configurazione e gestione di una pipeline di EC2 immagini di Image Builder utilizzando la AWS CLI nella Guida per l'utente di Image EC2 Builder.

- Per i dettagli sull'API, vedere UpdateInfrastructureConfigurationin Command Reference.AWS CLI

Esempi di Incident Manager che utilizzano AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with Incident Manager.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`create-replication-set`

Il seguente esempio di codice mostra come utilizzare`create-replication-set`.

AWS CLI

Come creare il set di replica

L'esempio `create-replication-set` seguente crea il set di replica utilizzato da Strumento di gestione degli incidenti per replicare e crittografare i dati nell'account Amazon Web Services. In questo esempio vengono utilizzate le Regioni us-east-1 e us-east-2 durante la creazione del set di replica.

```
aws ssm-incidents create-replication-set \
    --regions '{"us-east-1": {"sseKmsKeyId": "arn:aws:kms:us-
east-1:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab"}, "us-east-2":
 {"sseKmsKeyId": "arn:aws:kms:us-
east-1:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab"}}'
```

Output:

```
{
    "replicationSetArns": [
        "arn:aws:ssm-incidents::111122223333:replication-set/c4bcb603-4bf9-
bb3f-413c-08df53673b57"
    ]
}
```

Per ulteriori informazioni, consulta Utilizzo del set di replica di Strumento di gestione degli incidenti nella Guida per l'utente di Strumento di gestione degli incidenti.

- Per i dettagli sull'API, consulta CreateReplicationSet AWS CLICommand Reference.

`create-response-plan`

Il seguente esempio di codice mostra come utilizzare`create-response-plan`.

AWS CLI

Come creare un piano di risposta

L'esempio `create-response-plan` seguente crea un piano di risposta con i dettagli specificati.

```
aws ssm-incidents create-response-plan \
    --chat-channel '{"chatbotSns": ["arn:aws:sns:us-
east-1:111122223333:Standard_User"]}' \
    --display-name "Example response plan" \
    --incident-template '{"impact": 5, "title": "example-incident"}' \
    --name "example-response" \
    --actions '[{"ssmAutomation": {"documentName": "AWSIncidents-
CriticalIncidentRunbookTemplate", "documentVersion": "$DEFAULT",
 "roleArn": "arn:aws:iam::111122223333:role/aws-service-role/ssm-
incidents.amazonaws.com/AWSServiceRoleForIncidentManager", "targetAccount":
 "RESPONSE_PLAN_OWNER_ACCOUNT"}}]' \
    --engagements '["arn:aws:ssm-contacts:us-east-1:111122223333:contact/example"]'
```

Output:

```
{
    "arn": "arn:aws:ssm-incidents::111122223333:response-plan/example-response"
}
```

Per ulteriori informazioni, consulta Preparazione degli incidenti nella Guida per l'utente di Strumento di gestione degli incidenti.

- Per i dettagli sull'API, consulta CreateResponsePlan AWS CLICommand Reference.

`create-timeline-event`

Il seguente esempio di codice mostra come utilizzare`create-timeline-event`.

AWS CLI

Esempio 1: come creare un evento della timeline personalizzato

L'esempio `create-timeline-event` seguente crea un evento della timeline personalizzato all'ora specificata dell'incidente specificato.

```
aws ssm-incidents create-timeline-event \
    --event-data "\"example timeline event\"" \
    --event-time 2022-10-01T20:30:00.000 \
    --event-type "Custom Event" \
```

```
    --incident-record-arn "arn:aws:ssm-incidents::111122223333:incident-record/
Example-Response-Plan/6ebcc812-85f5-b7eb-8b2f-283e4EXAMPLE"
```

Output:

```
{
    "eventId": "c0bcc885-a41d-eb01-b4ab-9d2deEXAMPLE",
    "incidentRecordArn": "arn:aws:ssm-incidents::111122223333:incident-record/
Example-Response-Plan/6ebcc812-85f5-b7eb-8b2f-283e4EXAMPLE"
}
```

Esempio 2: come creare un evento della timeline con una nota evento imprevisto

L'esempio `create-timeline-event` seguente crea un evento della timeline che viene visualizzato nel pannello "Note evento imprevisto".

```
aws ssm-incidents create-timeline-event \
     --event-data "\"New Note\"" \
     --event-type "Note" \
     --incident-record-arn "arn:aws:ssm-incidents::111122223333:incident-record/
Test/6cc46130-ca6c-3b38-68f1-f6abeEXAMPLE" \
     --event-time 2023-06-20T12:06:00.000 \
     --event-references '[{"resource":"arn:aws:ssm-incidents::111122223333:incident-
record/Test/6cc46130-ca6c-3b38-68f1-f6abeEXAMPLE"}]'
```

Output:

```
{
    "eventId": "a41dc885-c0bc-b4ab-eb01-de9d2EXAMPLE",
    "incidentRecordArn": "arn:aws:ssm-incidents::111122223333:incident-record/
Example-Response-Plan/6ebcc812-85f5-b7eb-8b2f-283e4EXAMPLE"
}
```

Per ulteriori informazioni, consulta Dettagli degli incidenti nella Guida per l'utente di Strumento di gestione degli incidenti.

- Per i dettagli sull'API, consulta CreateTimelineEvent AWS CLICommand Reference.

`delete-incident-record`

Il seguente esempio di codice mostra come utilizzare`delete-incident-record`.

AWS CLI

Come eliminare un record di incidente

L'esempio `delete-incident-record` seguente elimina il record di incidente specificato.

```
aws ssm-incidents delete-incident-record \
    --arn "arn:aws:ssm-incidents::111122223333:incident-record/Example-Response-Plan/6ebcc812-85f5-b7eb-8b2f-283e4d844308"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Monitoraggio degli incidenti nella Guida per l'utente di Strumento di gestione degli incidenti.

- Per i dettagli sull'API, consulta DeleteIncidentRecord AWS CLICommand Reference.

`delete-replication-set`

Il seguente esempio di codice mostra come utilizzare`delete-replication-set`.

AWS CLI

Come eliminare il set di repliche

L'esempio `delete-replication-set` seguente elimina il set di repliche dall'account Amazon Web Services. L'eliminazione del set di repliche elimina anche tutti i dati di Strumento di gestione degli incidenti. Questa operazione non può essere annullata.

```
aws ssm-incidents delete-replication-set \
    --arn "arn:aws:ssm-incidents::111122223333:replication-set/c4bcb603-4bf9-bb3f-413c-08df53673b57"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Utilizzo del set di replica di Strumento di gestione degli incidenti nella Guida per l'utente di Strumento di gestione degli incidenti.

- Per i dettagli sull'API, consulta DeleteReplicationSet AWS CLICommand Reference.

`delete-resource-policy`

Il seguente esempio di codice mostra come utilizzare`delete-resource-policy`.

AWS CLI

Come eliminare una policy delle risorse

L'esempio `delete-resource-policy` seguente elimina una policy delle risorse da un piano di risposta. In questo modo verrà revocato l'accesso al principale o all'organizzazione con cui il piano di risposta era stato condiviso.

```
aws ssm-incidents delete-resource-policy \
    --policy-id "be8b57191f0371f1c6827341aa3f0a03" \
    --resource-arn "arn:aws:ssm-incidents::111122223333:response-plan/Example-Response-Plan"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Utilizzo di contatti condivisi e piani di risposta nella Guida per l'utente di Strumento di gestione degli incidenti.

- Per i dettagli sull'API, consulta DeleteResourcePolicy AWS CLICommand Reference.

`delete-response-plan`

Il seguente esempio di codice mostra come utilizzare`delete-response-plan`.

AWS CLI

Come eliminare un piano di risposta

L'esempio `delete-response-plan` seguente elimina il piano di risposta specificato.

```
aws ssm-incidents delete-response-plan \
    --arn "arn:aws:ssm-incidents::111122223333:response-plan/example-response"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Preparazione degli incidenti nella Guida per l'utente di Strumento di gestione degli incidenti.

- Per i dettagli sull'API, consulta DeleteResponsePlan AWS CLICommand Reference.

`delete-timeline-event`

Il seguente esempio di codice mostra come utilizzare`delete-timeline-event`.

AWS CLI

Come eliminare un evento della timeline

L'esempio `delete-timeline-event` seguente elimina un evento della timeline personalizzato dal record dell'incidente specificato.

```
aws ssm-incidents delete-timeline-event \
    --event-id "c0bcc885-a41d-eb01-b4ab-9d2de193643c" \
    --incident-record-arn "arn:aws:ssm-incidents::111122223333:incident-record/
Example-Response-Plan/6ebcc812-85f5-b7eb-8b2f-283e4d844308"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Dettagli degli incidenti nella Guida per l'utente di Strumento di gestione degli incidenti.

- Per i dettagli sull'API, consulta DeleteTimelineEvent AWS CLICommand Reference.

`get-incident-record`

Il seguente esempio di codice mostra come utilizzare`get-incident-record`.

AWS CLI

Come ottenere un record di incidente

L'esempio `get-incident-record` seguente ottiene dettagli sul record dell'incidente specificato.

```
aws ssm-incidents get-incident-record \
    --arn "arn:aws:ssm-incidents::111122223333:incident-record/Example-Response-
Plan/6ebcc812-85f5-b7eb-8b2f-283e4d844308"
```

Output:

```
{
    "incidentRecord": {
        "arn": "arn:aws:ssm-incidents::111122223333:incident-record/Example-
Response-Plan/6ebcc812-85f5-b7eb-8b2f-283e4d844308",
        "automationExecutions": [],
        "creationTime": "2021-05-21T18:16:57.579000+00:00",
```

```
        "dedupeString": "c4bcc812-85e7-938d-2b78-17181176ee1a",
        "impact": 5,
        "incidentRecordSource": {
            "createdBy": "arn:aws:iam::111122223333:user/draliatp",
            "invokedBy": "arn:aws:iam::111122223333:user/draliatp",
            "source": "aws.ssm-incidents.custom"
        },
        "lastModifiedBy": "arn:aws:iam::111122223333:user/draliatp",
        "lastModifiedTime": "2021-05-21T18:16:59.149000+00:00",
        "notificationTargets": [],
        "status": "OPEN",
        "title": "Example-Incident"
    }
}
```

Per ulteriori informazioni, consulta Dettagli degli incidenti nella Guida per l'utente di Strumento di gestione degli incidenti.

- Per i dettagli sull'API, consulta GetIncidentRecord AWS CLICommand Reference.

`get-replication-set`

Il seguente esempio di codice mostra come utilizzare`get-replication-set`.

AWS CLI

Come ottenere il set di replica

L'esempio `get-replication-set` seguente ottiene i dettagli del set di replica utilizzato da Strumento di gestione degli incidenti per replicare e crittografare i dati nel tuo account Amazon Web Services.

```
aws ssm-incidents get-replication-set \
    --arn "arn:aws:ssm-incidents::111122223333:replication-set/c4bcb603-4bf9-bb3f-413c-08df53673b57"
```

Output:

```
{
    "replicationSet": {
        "createdBy": "arn:aws:sts::111122223333:assumed-role/Admin/username",
```

```
        "createdTime": "2021-05-14T17:57:22.010000+00:00",
        "deletionProtected": false,
        "lastModifiedBy": "arn:aws:sts::111122223333:assumed-role/Admin/username",
        "lastModifiedTime": "2021-05-14T17:57:22.010000+00:00",
        "regionMap": {
            "us-east-1": {
                "sseKmsKeyId": "DefaultKey",
                "status": "ACTIVE"
            },
            "us-east-2": {
                "sseKmsKeyId": "DefaultKey",
                "status": "ACTIVE",
                "statusMessage": "Tagging inaccessible"
            }
        },
        "status": "ACTIVE"
    }
}
```

Per ulteriori informazioni, consulta Utilizzo del set di replica di Strumento di gestione degli incidenti nella Guida per l'utente di Strumento di gestione degli incidenti.

- Per i dettagli sull'API, consulta GetReplicationSet AWS CLICommand Reference.

`get-resource-policies`

Il seguente esempio di codice mostra come utilizzare`get-resource-policies`.

AWS CLI

Come elencare le policy delle risorse per un piano di risposta

L'esempio `command-name` seguente elenca le policy delle risorse associate al piano di risposta specificato.

```
aws ssm-incidents get-resource-policies \
--resource-arn "arn:aws:ssm-incidents::111122223333:response-plan/Example-Response-Plan"
```

Output:

```
{
```

```
    "resourcePolicies": [
        {
            "policyDocument": "{\"Version\":\"2012-10-17\",\"Statement\":[{\"Sid
\":\"d901b37a-dbb0-458a-8842-75575c464219-external-principals\",\"Effect\":
\"Allow\",\"Principal\":{\"AWS\":\"arn:aws:iam::222233334444:root\"},\"Action
\":[\"ssm-incidents:GetResponsePlan\",\"ssm-incidents:StartIncident\",\"ssm-
incidents:UpdateIncidentRecord\",\"ssm-incidents:GetIncidentRecord\",\"ssm-
incidents:CreateTimelineEvent\",\"ssm-incidents:UpdateTimelineEvent\",\"ssm-
incidents:GetTimelineEvent\",\"ssm-incidents:ListTimelineEvents\",\"ssm-
incidents:UpdateRelatedItems\",\"ssm-incidents:ListRelatedItems\"],\"Resource\":
[\"arn:aws:ssm-incidents:*:111122223333:response-plan/Example-Response-Plan\",
\"arn:aws:ssm-incidents:*:111122223333:incident-record/Example-Response-Plan/*
\"]}]}",
            "policyId": "be8b57191f0371f1c6827341aa3f0a03",
            "ramResourceShareRegion": "us-east-1"
        }
    ]
}
```

Per ulteriori informazioni, consulta Utilizzo di contatti condivisi e piani di risposta nella Guida per l'utente di Strumento di gestione degli incidenti.

- Per i dettagli sull'API, consulta GetResourcePolicies AWS CLICommand Reference.

`get-response-plan`

Il seguente esempio di codice mostra come utilizzare`get-response-plan`.

AWS CLI

Come ottenere i dettagli di un piano di risposta

L'`command-name`esempio seguente fornisce dettagli su un piano di risposta specifico nel tuo AWS account.

```
aws ssm-incidents get-response-plan \
    --arn "arn:aws:ssm-incidents::111122223333:response-plan/Example-Response-Plan"
```

Output:

```
{
    "actions": [
```

```
        {
            "ssmAutomation": {
                "documentName": "AWSIncidents-CriticalIncidentRunbookTemplate",
                "documentVersion": "$DEFAULT",
                "roleArn": "arn:aws:iam::111122223333:role/aws-service-role/ssm-
incidents.amazonaws.com/AWSServiceRoleForIncidentManager",
                "targetAccount": "RESPONSE_PLAN_OWNER_ACCOUNT"
            }
        }
    ],
    "arn": "arn:aws:ssm-incidents::111122223333:response-plan/Example-Response-
Plan",
    "chatChannel": {
        "chatbotSns": [
            "arn:aws:sns:us-east-1:111122223333:Standard_User"
        ]
    },
    "displayName": "Example response plan",
    "engagements": [
        "arn:aws:ssm-contacts:us-east-1:111122223333:contact/example"
    ],
    "incidentTemplate": {
        "impact": 5,
        "title": "Example-Incident"
    },
    "name": "Example-Response-Plan"
}
```

Per ulteriori informazioni, consulta Preparazione degli incidenti nella Guida per l'utente di Strumento di gestione degli incidenti.

- Per i dettagli sull'API, consulta GetResponsePlan AWS CLICommand Reference.

`get-timeline-event`

Il seguente esempio di codice mostra come utilizzare`get-timeline-event`.

AWS CLI

Come ottenere i dettagli di un evento della timeline

L'esempio `get-timeline-event` seguente restituisce i dettagli dell'evento della timeline specificato.

```
aws ssm-incidents get-timeline-event \
    --event-id 20bcc812-8a94-4cd7-520c-0ff742111424 \
    --incident-record-arn "arn:aws:ssm-incidents::111122223333:incident-record/
Example-Response-Plan/6ebcc812-85f5-b7eb-8b2f-283e4d844308"
```

Output:

```
{
    "event": {
        "eventData": "\"Incident Started\"",
        "eventId": "20bcc812-8a94-4cd7-520c-0ff742111424",
        "eventTime": "2021-05-21T18:16:57+00:00",
        "eventType": "Custom Event",
        "eventUpdatedTime": "2021-05-21T18:16:59.944000+00:00",
        "incidentRecordArn": "arn:aws:ssm-incidents::111122223333:incident-record/
Example-Response-Plan/6ebcc812-85f5-b7eb-8b2f-283e4d844308"
    }
}
```

Per ulteriori informazioni, consulta Dettagli degli incidenti nella Guida per l'utente di Strumento di gestione degli incidenti.

- Per i dettagli sull'API, consulta GetTimelineEvent AWS CLICommand Reference.

`list-incident-records`

Il seguente esempio di codice mostra come utilizzare`list-incident-records`.

AWS CLI

Come elencare i record degli incidenti

L'esempio `command-name` seguente elenca i record degli incidenti nell'account Amazon Web Services.

```
aws ssm-incidents list-incident-records
```

Output:

```
{
    "incidentRecordSummaries": [
```

```
        {
            "arn": "arn:aws:ssm-incidents::111122223333:incident-record/Example-
Response-Plan/6ebcc812-85f5-b7eb-8b2f-283e4d844308",
            "creationTime": "2021-05-21T18:16:57.579000+00:00",
            "impact": 5,
            "incidentRecordSource": {
                "createdBy": "arn:aws:iam::111122223333:user/draliatp",
                "invokedBy": "arn:aws:iam::111122223333:user/draliatp",
                "source": "aws.ssm-incidents.custom"
            },
            "status": "OPEN",
            "title": "Example-Incident"
        }
    ]
}
```

Per ulteriori informazioni, consulta Elenco di incidenti nella Guida per l'utente di Strumento di gestione degli incidenti.

- Per i dettagli sull'API, consulta ListIncidentRecords AWS CLICommand Reference.

`list-related-items`

Il seguente esempio di codice mostra come utilizzare`list-related-items`.

AWS CLI

Come elencare gli elementi correlati

L'esempio `list-related-items` seguente elenca gli elementi correlati dell'incidente specificato.

```
aws ssm-incidents list-related-items \
    --incident-record-arn "arn:aws:ssm-incidents::111122223333:incident-record/
Example-Response-Plan/6ebcc812-85f5-b7eb-8b2f-283e4d844308"
```

Output:

```
{
    "relatedItems": [
        {
            "identifier": {
```

```
                "type": "OTHER",
                "value": {
                    "url": "https://console.aws.amazon.com/systems-manager/opsitems/
oi-8ef82158e190/workbench?region=us-east-1"
                }
            },
            "title": "Example related item"
        },
        {
            "identifier": {
                "type": "PARENT",
                "value": {
                    "arn": "arn:aws:ssm:us-east-1:111122223333:opsitem/
oi-8084126392ac"
                }
            },
            "title": "parentItem"
        }
    ]
}
```

Per ulteriori informazioni, consulta Dettagli degli incidenti nella Guida per l'utente di Strumento di gestione degli incidenti.

- Per i dettagli sull'API, consulta ListRelatedItems AWS CLICommand Reference.

`list-replication-sets`

Il seguente esempio di codice mostra come utilizzare`list-replication-sets`.

AWS CLI

Come elencare il set di replica

L'`list-replication-set`esempio seguente elenca il set di replica utilizzato da Incident Manager per replicare e crittografare i dati nell'account. AWS

```
aws ssm-incidents list-replication-sets
```

Output:

```
{
```

```
    "replicationSetArns": [
        "arn:aws:ssm-incidents::111122223333:replication-set/c4bcb603-4bf9-
bb3f-413c-08df53673b57"
    ]
}
```

Per ulteriori informazioni, consulta Utilizzo del set di replica di Strumento di gestione degli incidenti nella Guida per l'utente di Strumento di gestione degli incidenti.

- Per i dettagli sull'API, consulta Command ListReplicationSetsReference AWS CLI .

`list-response-plans`

Il seguente esempio di codice mostra come utilizzare`list-response-plans`.

AWS CLI

Come elencare i piani di risposta disponibili

L'esempio `list-response-plans` seguente elenca i piani di risposta disponibili nell'account Amazon Web Services.

```
aws ssm-incidents list-response-plans
```

Output:

```
{
    "responsePlanSummaries": [
        {
            "arn": "arn:aws:ssm-incidents::111122223333:response-plan/Example-
Response-Plan",
            "displayName": "Example response plan",
            "name": "Example-Response-Plan"
        }
    ]
}
```

Per ulteriori informazioni, consulta Preparazione degli incidenti nella Guida per l'utente di Strumento di gestione degli incidenti.

- Per i dettagli sull'API, consulta ListResponsePlans AWS CLICommand Reference.

`list-tags-for-resource`

Il seguente esempio di codice mostra come utilizzare`list-tags-for-resource`.

AWS CLI

Come elencare i tag per un piano di risposta

L'esempio `list-tags-for-resource` seguente elenca i tag associati al piano di risposta specificato.

```
aws ssm-incidents list-tags-for-resource \
    --resource-arn "arn:aws:ssm-incidents::111122223333:response-plan/Example-Response-Plan"
```

Output:

```
{
    "tags": {
    "group1": "1"
    }
}
```

Per ulteriori informazioni, consulta Tagging nella Guida per l'utente di Strumento di gestione degli incidenti.

- Per i dettagli sull'API, consulta ListTagsForResource AWS CLICommand Reference.

`list-timeline-events`

Il seguente esempio di codice mostra come utilizzare`list-timeline-events`.

AWS CLI

Come elencare gli eventi della timeline di un incidente

L'esempio `command-name` seguente elenca gli eventi della timeline dell'incidente specificato.

```
aws ssm-incidents list-timeline-events \
    --incident-record-arn "arn:aws:ssm-incidents::111122223333:incident-record/Example-Response-Plan/6ebcc812-85f5-b7eb-8b2f-283e4d844308"
```

Output:


```
{
    "eventSummaries": [
        {
            "eventId": "8cbcc889-35e1-a42d-2429-d6f100799915",
            "eventTime": "2021-05-21T22:36:13.766000+00:00",
            "eventType": "SSM Incident Record Update",
            "eventUpdatedTime": "2021-05-21T22:36:13.766000+00:00",
            "incidentRecordArn": "arn:aws:ssm-incidents::111122223333:incident-
record/Example-Response-Plan/6ebcc812-85f5-b7eb-8b2f-283e4d844308"
        },
        {
            "eventId": "a2bcc825-aab5-1787-c605-f9bb2640d85b",
            "eventTime": "2021-05-21T18:58:46.443000+00:00",
            "eventType": "SSM Incident Record Update",
            "eventUpdatedTime": "2021-05-21T18:58:46.443000+00:00",
            "incidentRecordArn": "arn:aws:ssm-incidents::111122223333:incident-
record/Example-Response-Plan/6ebcc812-85f5-b7eb-8b2f-283e4d844308"
        },
        {
            "eventId": "5abcc812-89c0-b0a8-9437-1c74223d4685",
            "eventTime": "2021-05-21T18:16:59.149000+00:00",
            "eventType": "SSM Incident Record Update",
            "eventUpdatedTime": "2021-05-21T18:16:59.149000+00:00",
            "incidentRecordArn": "arn:aws:ssm-incidents::111122223333:incident-
record/Example-Response-Plan/6ebcc812-85f5-b7eb-8b2f-283e4d844308"
        },
        {
            "eventId": "06bcc812-8820-405e-4065-8d2b14d29b92",
            "eventTime": "2021-05-21T18:16:58+00:00",
            "eventType": "SSM Automation Execution Start Failure for Incident",
            "eventUpdatedTime": "2021-05-21T18:16:58.689000+00:00",
            "incidentRecordArn": "arn:aws:ssm-incidents::111122223333:incident-
record/Example-Response-Plan/6ebcc812-85f5-b7eb-8b2f-283e4d844308"
        },
        {
            "eventId": "20bcc812-8a94-4cd7-520c-0ff742111424",
            "eventTime": "2021-05-21T18:16:57+00:00",
            "eventType": "Custom Event",
            "eventUpdatedTime": "2021-05-21T18:16:59.944000+00:00",
            "incidentRecordArn": "arn:aws:ssm-incidents::111122223333:incident-
record/Example-Response-Plan/6ebcc812-85f5-b7eb-8b2f-283e4d844308"
        },
```

```
        {
            "eventId": "c0bcc885-a41d-eb01-b4ab-9d2de193643c",
            "eventTime": "2020-10-01T20:30:00+00:00",
            "eventType": "Custom Event",
            "eventUpdatedTime": "2021-05-21T22:28:26.299000+00:00",
            "incidentRecordArn": "arn:aws:ssm-incidents::111122223333:incident-
record/Example-Response-Plan/6ebcc812-85f5-b7eb-8b2f-283e4d844308"
        }
    ]
}
```

Per ulteriori informazioni, consulta Dettagli degli incidenti nella Guida per l'utente di Strumento di gestione degli incidenti.

- Per i dettagli sull'API, consulta ListTimelineEvents AWS CLICommand Reference.

`put-resource-policy`

Il seguente esempio di codice mostra come utilizzare`put-resource-policy`.

AWS CLI

Come condividere un piano di risposta e gli incidenti

L'`command-name`esempio seguente aggiunge una politica delle risorse a Example-Response-Plan quella che condivide il piano di risposta e gli incidenti associati con il principale specificato.

```
aws ssm-incidents put-resource-policy \
    --resource-arn "arn:aws:ssm-incidents::111122223333:response-plan/Example-
Response-Plan" \
    --policy "{\"Version\":\"2012-10-17\",\"Statement\":[{\"Sid\":
\"ExampleResourcePolciy\",\"Effect\":\"Allow\",\"Principal\":{\"AWS\":
\"arn:aws:iam::222233334444:root\"},\"Action\":[\"ssm-incidents:GetResponsePlan
\",\"ssm-incidents:StartIncident\",\"ssm-incidents:UpdateIncidentRecord
\",\"ssm-incidents:GetIncidentRecord\",\"ssm-incidents:CreateTimelineEvent
\",\"ssm-incidents:UpdateTimelineEvent\",\"ssm-incidents:GetTimelineEvent
\",\"ssm-incidents:ListTimelineEvents\",\"ssm-incidents:UpdateRelatedItems
\",\"ssm-incidents:ListRelatedItems\"],\"Resource\":[\"arn:aws:ssm-
incidents:*:111122223333:response-plan/Example-Response-Plan\",\"arn:aws:ssm-
incidents:*:111122223333:incident-record/Example-Response-Plan/*\"]}]}"
```

Output:

```
{
    "policyId": "be8b57191f0371f1c6827341aa3f0a03"
}
```

Per ulteriori informazioni, consulta Utilizzo di contatti condivisi e piani di risposta nella Guida per l'utente di Strumento di gestione degli incidenti.

- Per i dettagli sull'API, vedere PutResourcePolicyin AWS CLI Command Reference.

`start-incident`

Il seguente esempio di codice mostra come utilizzare`start-incident`.

AWS CLI

Come avviare un incidente

L'esempio `start-incident` seguente avvia un incidente utilizzando il piano di risposta specificato.

```
aws ssm-incidents start-incident \
    --response-plan-arn "arn:aws:ssm-incidents::111122223333:response-plan/Example-Response-Plan"
```

Output:

```
{
    "incidentRecordArn": "arn:aws:ssm-incidents::682428703967:incident-record/Example-Response-Plan/6ebcc812-85f5-b7eb-8b2f-283e4d844308"
}
```

Per ulteriori informazioni, consulta Creazione di incidenti nella Guida per l'utente di Strumento di gestione degli incidenti.

- Per i dettagli sull'API, consulta StartIncident AWS CLICommand Reference.

`tag-resource`

Il seguente esempio di codice mostra come utilizzare`tag-resource`.

AWS CLI

Come aggiungere tag a un piano di risposta

L'esempio `tag-resource` seguente aggiunge tag a un piano di risposta specificato utilizzando la coppia chiave-valore del tag fornita.

```
aws ssm-incidents tag-resource \
    --resource-arn "arn:aws:ssm-incidents::111122223333:response-plan/Example-Response-Plan" \
    --tags '{"group1":"1"}'
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Tagging nella Guida per l'utente di Strumento di gestione degli incidenti.

- Per i dettagli sull'API, consulta TagResource AWS CLICommand Reference.

`untag-resource`

Il seguente esempio di codice mostra come utilizzare`untag-resource`.

AWS CLI

Come rimuovere tag da un piano di risposta

L'esempio `untag-resource` seguente rimuove i tag specificati dal piano di risposta.

```
aws ssm-incidents untag-resource \
    --resource-arn "arn:aws:ssm-incidents::111122223333:response-plan/Example-Response-Plan" \
    --tag-keys '["group1"]'
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Tagging nella Guida per l'utente di Strumento di gestione degli incidenti.

- Per i dettagli sull'API, consulta UntagResource AWS CLICommand Reference.

`update-deletion-protection`

Il seguente esempio di codice mostra come utilizzare`update-deletion-protection`.

AWS CLI

Come aggiornare la protezione dall'eliminazione del set

L'esempio `update-deletion-protection` seguente aggiorna la protezione dall'eliminazione nell'account per impedirti di eliminare l'ultima Regione nel tuo set di replica.

```
aws ssm-incidents update-deletion-protection \
    --arn "arn:aws:ssm-incidents::111122223333:replication-set/
a2bcc5c9-0f53-8047-7fef-c20749989b40" \
    --deletion-protected
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Utilizzo del set di replica di Strumento di gestione degli incidenti nella Guida per l'utente di Strumento di gestione degli incidenti.

- Per i dettagli sull'API, consulta UpdateDeletionProtection AWS CLICommand Reference.

`update-incident-record`

Il seguente esempio di codice mostra come utilizzare`update-incident-record`.

AWS CLI

Come aggiornare un record di incidente

L'esempio `command-name` seguente risolve l'incidente specificato.

```
aws ssm-incidents update-incident-record \
    --arn "arn:aws:ssm-incidents::111122223333:incident-record/Example-Response-
Plan/6ebcc812-85f5-b7eb-8b2f-283e4d844308" \
    --status "RESOLVED"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Dettagli degli incidenti nella Guida per l'utente di Strumento di gestione degli incidenti.

- Per i dettagli sull'API, consulta UpdateIncidentRecord AWS CLICommand Reference.

`update-related-items`

Il seguente esempio di codice mostra come utilizzare`update-related-items`.

AWS CLI

Come aggiornare un elemento correlato agli incidenti

L'esempio `update-related-item` seguente rimuove un elemento correlato dal record dell'incidente specificato.

```
aws ssm-incidents update-related-items \
    --incident-record-arn "arn:aws:ssm-incidents::111122223333:incident-record/
Example-Response-Plan/6ebcc812-85f5-b7eb-8b2f-283e4d844308" \
    --related-items-update '{"itemToRemove": {"type": "OTHER", "value": {"url":
 "https://console.aws.amazon.com/systems-manager/opsitems/oi-8ef82158e190/workbench?
region=us-east-1"}}}'
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Dettagli degli incidenti nella Guida per l'utente di Strumento di gestione degli incidenti.

- Per i dettagli sull'API, consulta UpdateRelatedItems AWS CLICommand Reference.

`update-replication-set`

Il seguente esempio di codice mostra come utilizzare`update-replication-set`.

AWS CLI

Come aggiornare un set di replica

L'esempio `command-name` seguente elimina la Regione us-east-2 dal set di replica.

```
aws ssm-incidents update-replication-set \
    --arn "arn:aws:ssm-incidents::111122223333:replication-set/
a2bcc5c9-0f53-8047-7fef-c20749989b40" \
    --actions '[{"deleteRegionAction": {"regionName": "us-east-2"}}]'
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Utilizzo del set di replica di Strumento di gestione degli incidenti nella Guida per l'utente di Strumento di gestione degli incidenti.

- Per i dettagli sull'API, consulta UpdateReplicationSet AWS CLICommand Reference.

`update-response-plan`

Il seguente esempio di codice mostra come utilizzare`update-response-plan`.

AWS CLI

Come aggiornare un piano di risposta

L'esempio `update-response-plan` seguente rimuove un canale di chat dal piano di risposta specificato.

```
aws ssm-incidents update-response-plan \
    --arn "arn:aws:ssm-incidents::111122223333:response-plan/Example-Response-Plan"
 \
    --chat-channel '{"empty":{}}'
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Preparazione degli incidenti nella Guida per l'utente di Strumento di gestione degli incidenti.

- Per i dettagli sull'API, consulta UpdateResponsePlan AWS CLICommand Reference.

`update-timeline-event`

Il seguente esempio di codice mostra come utilizzare`update-timeline-event`.

AWS CLI

Come aggiornare un evento della timeline

L'esempio `update-timeline-event` seguente aggiorna l'ora in cui si è verificato l'evento.

```
aws ssm-incidents update-timeline-event \
    --event-id 20bcc812-8a94-4cd7-520c-0ff742111424 \
```

```
    --incident-record-arn "arn:aws:ssm-incidents::111122223333:incident-record/
Example-Response-Plan/6ebcc812-85f5-b7eb-8b2f-283e4d844308" \
    --event-time "2021-05-21T18:10:57+00:00"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Dettagli degli incidenti nella Guida per l'utente di Strumento di gestione degli incidenti.

- Per i dettagli sull'API, consulta UpdateTimelineEvent AWS CLICommand Reference.

Esempi di contatti di Incident Manager utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with Incident Manager Contacts.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`accept-page`

Il seguente esempio di codice mostra come utilizzare`accept-page`.

AWS CLI

Come accettare una pagina durante un coinvolgimento

L'esempio `accept-page` seguente utilizza un codice di accettazione inviato al canale di contatto per accettare una pagina.

```
aws ssm-contacts accept-page \
```

```
    --page-id "arn:aws:ssm-contacts:us-east-2:682428703967:page/
akuam/94ea0c7b-56d9-46c3-b84a-a37c8b067ad3" \
    --accept-type READ \
    --accept-code 425440
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Contatti nella Guida per l'utente di Strumento di gestione degli incidenti.

- Per i dettagli sull'API, consulta AcceptPage AWS CLICommand Reference.

`activate-contact-channel`

Il seguente esempio di codice mostra come utilizzare`activate-contact-channel`.

AWS CLI

Come attivare il canale di contatto di un contatto

L'esempio `activate-contact-channel` seguente attiva un canale di contatto e lo rende utilizzabile come parte di un incidente.

```
aws ssm-contacts activate-contact-channel \
    --contact-channel-id "arn:aws:ssm-contacts:us-east-2:111122223333:contact-
channel/akuam/fc7405c4-46b2-48b7-87b2-93e2f225b90d" \
    --activation-code "466136"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Contatti nella Guida per l'utente di Strumento di gestione degli incidenti.

- Per i dettagli sull'API, consulta ActivateContactChannel AWS CLICommand Reference.

`command-name`

Il seguente esempio di codice mostra come utilizzare`command-name`.

AWS CLI

Come eliminare un contatto

L'esempio `command-name` seguente elimina un contatto. Il contatto non sarà più raggiungibile da nessun piano di escalation che lo riguardi.

```
aws ssm-contacts delete-contact \
    --contact-id "arn:aws:ssm-contacts:us-east-1:682428703967:contact/alejr"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Contatti nella Guida per l'utente di Strumento di gestione degli incidenti.

- Per i dettagli sull'API, consulta CommandName AWS CLICommand Reference.

`create-contact-channel`

Il seguente esempio di codice mostra come utilizzare`create-contact-channel`.

AWS CLI

Come creare un canale di contatto

Crea un canale di contatto di tipo SMS per il contatto Akua Mansa. È possibile creare canali di contatto di tipo SMS, EMAIL o VOICE.

```
aws ssm-contacts create-contact-channel \
    --contact-id "arn:aws:ssm-contacts:us-east-1:111122223333:contact/akuam" \
    --name "akuas sms-test" \
    --type SMS \
    --delivery-address '{"SimpleAddress": "+15005550199"}'
```

Output:

```
{
    "ContactChannelArn": "arn:aws:ssm-contacts:us-east-1:111122223333:contact-
channel/akuam/02f506b9-ea5d-4764-af89-2daa793ff024"
}
```

Per ulteriori informazioni, consulta Contatti nella Guida per l'utente di Strumento di gestione degli incidenti.

- Per i dettagli sull'API, consulta CreateContactChannel AWS CLICommand Reference.

`create-contact`

Il seguente esempio di codice mostra come utilizzare`create-contact`.

AWS CLI

Come creare un contatto

L'esempio `create-contact` seguente crea un contatto nel tuo ambiente con un piano vuoto. Il piano può essere aggiornato dopo aver creato i canali di contatto. Usa il create-contact-channel comando con l'ARN di output di questo comando. Dopo aver creato i canali di contatto per questo contatto, utilizza update-contact per aggiornare il piano.

```
aws ssm-contacts create-contact \
    --alias "akuam" \
    --display-name "Akua Mansa" \
    --type PERSONAL \
    --plan '{"Stages": []}'
```

Output:

```
{
    "ContactArn": "arn:aws:ssm-contacts:us-east-2:111122223333:contact/akuam"
}
```

Per ulteriori informazioni, consulta Contatti nella Guida per l'utente di Strumento di gestione degli incidenti.

- Per i dettagli sull'API, vedere CreateContactin AWS CLI Command Reference.

`deactivate-contact-channel`

Il seguente esempio di codice mostra come utilizzare`deactivate-contact-channel`.

AWS CLI

Come disattivare un canale di contatto

L'esempio `deactivate-contact-channel` seguente disattiva un canale di contatto. Quando si disattiva un canale di contatto, il canale non verrà più paginato durante un incidente. Puoi riattivare un canale di contatto in qualsiasi momento utilizzando il comando `activate-contact-channel`.

```
aws ssm-contacts deactivate-contact-channel \
    --contact-channel-id "arn:aws:ssm-contacts:us-east-2:111122223333:contact-channel/akuam/fc7405c4-46b2-48b7-87b2-93e2f225b90d"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Contatti nella Guida per l'utente di Strumento di gestione degli incidenti.

- Per i dettagli sull'API, consulta DeactivateContactChannel AWS CLICommand Reference.

`delete-contact-channel`

Il seguente esempio di codice mostra come utilizzare`delete-contact-channel`.

AWS CLI

Come eliminare un canale di contatto

L'esempio `delete-contact-channel` seguente elimina un canale di contatto. Eliminare un canale di contatto assicura che il canale non venga paginato durante un incidente.

```
aws ssm-contacts delete-contact-channel \
    --contact-channel-id "arn:aws:ssm-contacts:us-east-1:111122223333:contact-channel/akuam/13149bad-52ee-45ea-ae1e-45857f78f9b2"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Contatti nella Guida per l'utente di Strumento di gestione degli incidenti.

- Per i dettagli sull'API, consulta DeleteContactChannel AWS CLICommand Reference.

`delete-contact`

Il seguente esempio di codice mostra come utilizzare`delete-contact`.

AWS CLI

Come eliminare un contatto

L'esempio `delete-contact` seguente elimina un contatto. Il contatto non sarà più raggiungibile da nessun piano di escalation che lo riguardi.

```
aws ssm-contacts delete-contact \
    --contact-id "arn:aws:ssm-contacts:us-east-1:111122223333:contact/alejr"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Contatti nella Guida per l'utente di Strumento di gestione degli incidenti.

- Per i dettagli sull'API, consulta DeleteContact AWS CLICommand Reference.

`describe-engagement`

Il seguente esempio di codice mostra come utilizzare`describe-engagement`.

AWS CLI

Come descrivere i dettagli di un coinvolgimento

L'esempio `describe-engagement` seguente elenca i dettagli di un coinvolgimento relativo a un contatto o a un piano di escalation. L'oggetto e il contenuto vengono inviati ai canali di contatto.

```
aws ssm-contacts describe-engagement \
    --engagement-id "arn:aws:ssm-contacts:us-east-2:111122223333:engagement/
example_escalation/69e40ce1-8dbb-4d57-8962-5fbe7fc53356"
```

Output:

```
{
    "ContactArn": "arn:aws:ssm-contacts:us-east-2:111122223333:contact/
example_escalation",
    "EngagementArn": "arn:aws:ssm-contacts:us-east-2:111122223333:engagement/
example_escalation/69e40ce1-8dbb-4d57-8962-5fbe7fc53356",
    "Sender": "cli",
    "Subject": "cli-test",
    "Content": "Testing engagements via CLI",
    "PublicSubject": "cli-test",
    "PublicContent": "Testing engagements va CLI",
    "StartTime": "2021-05-18T18:25:41.151000+00:00"
}
```

Per ulteriori informazioni, consulta Contatti nella Guida per l'utente di Strumento di gestione degli incidenti.

- Per i dettagli sull'API, consulta DescribeEngagement AWS CLICommand Reference.

`describe-page`

Il seguente esempio di codice mostra come utilizzare`describe-page`.

AWS CLI

Come inserire i dettagli di una pagina in un canale di contatto

L'esempio `describe-page` seguente elenca i dettagli di una pagina relativa a un canale di contatto. La pagina includerà l'oggetto e il contenuto forniti.

```
aws ssm-contacts describe-page \
    --page-id "arn:aws:ssm-contacts:us-east-2:111122223333:page/akuam/ad0052bd-e606-498a-861b-25726292eb93"
```

Output:

```
{
    "PageArn": "arn:aws:ssm-contacts:us-east-2:111122223333:page/akuam/ad0052bd-e606-498a-861b-25726292eb93",
    "EngagementArn": "arn:aws:ssm-contacts:us-east-2:111122223333:engagement/akuam/78a29753-3674-4ac5-9f83-0468563567f0",
    "ContactArn": "arn:aws:ssm-contacts:us-east-2:111122223333:contact/akuam",
    "Sender": "cli",
    "Subject": "cli-test",
    "Content": "Testing engagements via CLI",
    "PublicSubject": "cli-test",
    "PublicContent": "Testing engagements va CLI",
    "SentTime": "2021-05-18T18:43:29.301000+00:00",
    "ReadTime": "2021-05-18T18:43:55.708000+00:00",
    "DeliveryTime": "2021-05-18T18:43:55.265000+00:00"
}
```

Per ulteriori informazioni, consulta Contatti nella Guida per l'utente di Strumento di gestione degli incidenti.

- Per i dettagli sull'API, consulta DescribePage AWS CLICommand Reference.

`get-contact-channel`

Il seguente esempio di codice mostra come utilizzare`get-contact-channel`.

AWS CLI

Come elencare i dettagli di un canale di contatto

L'esempio `get-contact-channel` seguente elenca i dettagli di un canale di contatto.

```
aws ssm-contacts get-contact-channel \
    --contact-channel-id "arn:aws:ssm-contacts:us-east-2:111122223333:contact-channel/akuam/fc7405c4-46b2-48b7-87b2-93e2f225b90d"
```

Output:

```
{
    "ContactArn": "arn:aws:ssm-contacts:us-east-2:111122223333:contact/akuam",
    "ContactChannelArn": "arn:aws:ssm-contacts:us-east-2:111122223333:contact-channel/akuam/fc7405c4-46b2-48b7-87b2-93e2f225b90d",
    "Name": "akuas sms",
    "Type": "SMS",
    "DeliveryAddress": {
        "SimpleAddress": "+15005550199"
    },
    "ActivationStatus": "ACTIVATED"
}
```

Per ulteriori informazioni, consulta Contatti nella Guida per l'utente di Strumento di gestione degli incidenti.

- Per i dettagli sull'API, consulta GetContactChannel AWS CLICommand Reference.

`get-contact-policy`

Il seguente esempio di codice mostra come utilizzare`get-contact-policy`.

AWS CLI

Come elencare le policy relative alle risorse di un contatto

L'esempio `get-contact-policy` seguente elenca le policy relative alle risorse associate al contatto specificato.

```
aws ssm-contacts get-contact-policy \
    --contact-arn "arn:aws:ssm-contacts:us-east-1:111122223333:contact/akuam"
```

Output:

```
{
    "ContactArn": "arn:aws:ssm-contacts:us-east-1:111122223333:contact/akuam",
    "Policy": "{\"Version\":\"2012-10-17\",\"Statement\":[{\"Sid\":
\"SharePolicyForDocumentationDralia\",\"Effect\":\"Allow\",\"Principal\":
{\"AWS\":\"222233334444\"},\"Action\":[\"ssm-contacts:GetContact\",\"ssm-
contacts:StartEngagement\",\"ssm-contacts:DescribeEngagement\",\"ssm-
contacts:ListPagesByEngagement\",\"ssm-contacts:StopEngagement\"],\"Resource
\":[\"arn:aws:ssm-contacts:*:111122223333:contact/akuam\",\"arn:aws:ssm-
contacts:*:111122223333:engagement/akuam/*\"]}]}"
}
```

Per ulteriori informazioni, consulta Utilizzo di contatti condivisi e piani di risposta nella Guida per l'utente di Strumento di gestione degli incidenti.

- Per i dettagli sull'API, consulta GetContactPolicy AWS CLICommand Reference.

`get-contact`

Il seguente esempio di codice mostra come utilizzare`get-contact`.

AWS CLI

Esempio 1: come descrivere un piano di contatto

L'esempio `get-contact` seguente descrive un contatto.

```
aws ssm-contacts get-contact \
    --contact-id "arn:aws:ssm-contacts:us-east-2:111122223333:contact/akuam"
```

Output:

```
{
    "ContactArn": "arn:aws:ssm-contacts:us-east-2:111122223333:contact/akuam",
    "Alias": "akuam",
    "DisplayName": "Akua Mansa",
    "Type": "PERSONAL",
    "Plan": {
```

```
        "Stages": [
            {
                "DurationInMinutes": 5,
                "Targets": [
                    {
                        "ChannelTargetInfo": {
                            "ContactChannelId": "arn:aws:ssm-contacts:us-
east-2:111122223333:contact-channel/akuam/beb25840-5ac8-4644-95cc-7a8de390fa65",
                            "RetryIntervalInMinutes": 1
                        }
                    }
                ]
            },
            {
                "DurationInMinutes": 5,
                "Targets": [
                    {
                        "ChannelTargetInfo": {
                            "ContactChannelId": "arn:aws:ssm-contacts:us-
east-2:111122223333:contact-channel/akuam/49f3c24d-5f9f-4638-ae25-3f49e04229ad",
                            "RetryIntervalInMinutes": 1
                        }
                    }
                ]
            },
            {
                "DurationInMinutes": 5,
                "Targets": [
                    {
                        "ChannelTargetInfo": {
                            "ContactChannelId": "arn:aws:ssm-contacts:us-
east-2:111122223333:contact-channel/akuam/77d4f447-f619-4954-afff-85551e369c2a",
                            "RetryIntervalInMinutes": 1
                        }
                    }
                ]
            }
        ]
    }
}
```

Esempio 2: come descrivere un piano di escalation

L'esempio `get-contact` seguente descrive un piano di escalation.

```
aws ssm-contacts get-contact \
--contact-id "arn:aws:ssm-contacts:us-east-2:111122223333:contact/
example_escalation"
```

Output:

```
{
    "ContactArn": "arn:aws:ssm-contacts:us-east-2:111122223333:contact/
example_escalation",
    "Alias": "example_escalation",
    "DisplayName": "Example Escalation",
    "Type": "ESCALATION",
    "Plan": {
        "Stages": [
            {
                "DurationInMinutes": 5,
                "Targets": [
                    {
                        "ContactTargetInfo": {
                            "ContactId": "arn:aws:ssm-contacts:us-
east-2:111122223333:contact/akuam",
                            "IsEssential": true
                        }
                    }
                ]
            },
            {
                "DurationInMinutes": 5,
                "Targets": [
                    {
                        "ContactTargetInfo": {
                            "ContactId": "arn:aws:ssm-contacts:us-
east-2:111122223333:contact/alejr",
                            "IsEssential": false
                        }
                    }
                ]
            },
            {
                "DurationInMinutes": 0,
                "Targets": [
                    {
                        "ContactTargetInfo": {
```

```
                            "ContactId": "arn:aws:ssm-contacts:us-
east-2:111122223333:contact/anasi",
                            "IsEssential": false
                        }
                    }
                ]
            }
        ]
    }
}
```

Per ulteriori informazioni, consulta Contatti nella Guida per l'utente di Strumento di gestione degli incidenti.

- Per i dettagli sull'API, consulta GetContact AWS CLICommand Reference.

`list-contact-channels`

Il seguente esempio di codice mostra come utilizzare`list-contact-channels`.

AWS CLI

Come elencare i canali di contatto di un contatto

L'esempio `list-contact-channels` seguente elenca i canali di contatto disponibili del contatto specificato.

```
aws ssm-contacts list-contact-channels \
    --contact-id "arn:aws:ssm-contacts:us-east-2:111122223333:contact/akuam"
```

Output:

```
{
    [
        {
            "ContactArn": "arn:aws:ssm-contacts:us-east-2:111122223333:contact/
akuam",
            "Name": "akuas email",
            "Type": "EMAIL",
            "DeliveryAddress": {
                "SimpleAddress": "akuam@example.com"
            },
            "ActivationStatus": "NOT_ACTIVATED"
```

```
        },
        {
            "ContactChannelArn": "arn:aws:ssm-contacts:us-
east-2:111122223333:contact-channel/akuam/fc7405c4-46b2-48b7-87b2-93e2f225b90d",
            "ContactArn": "arn:aws:ssm-contacts:us-east-2:111122223333:contact/
akuam",
            "Name": "akuas sms",
            "Type": "SMS",
            "DeliveryAddress": {
                "SimpleAddress": "+15005550100"
            },
            "ActivationStatus": "ACTIVATED"
        }
    ]
}
```

Per ulteriori informazioni, consulta Contatti nella Guida per l'utente di Strumento di gestione degli incidenti.

- Per i dettagli sull'API, consulta ListContactChannels AWS CLICommand Reference.

`list-contacts`

Il seguente esempio di codice mostra come utilizzare`list-contacts`.

AWS CLI

Come elencare tutti i piani di escalation e i contatti

L'esempio `list-contacts` seguente elenca i contatti e i piani di escalation presenti nel tuo account.

```
aws ssm-contacts list-contacts
```

Output:

```
{
    "Contacts": [
        {
            "ContactArn": "arn:aws:ssm-contacts:us-east-2:111122223333:contact/
akuam",
            "Alias": "akuam",
            "DisplayName": "Akua Mansa",
```

```
            "Type": "PERSONAL"
        },
        {
            "ContactArn": "arn:aws:ssm-contacts:us-east-2:111122223333:contact/
alejr",
            "Alias": "alejr",
            "DisplayName": "Alejandro Rosalez",
            "Type": "PERSONAL"
        },
        {
            "ContactArn": "arn:aws:ssm-contacts:us-east-2:111122223333:contact/
anasi",
            "Alias": "anasi",
            "DisplayName": "Ana Carolina Silva",
            "Type": "PERSONAL"
        },
        {
            "ContactArn": "arn:aws:ssm-contacts:us-east-2:111122223333:contact/
example_escalation",
            "Alias": "example_escalation",
            "DisplayName": "Example Escalation",
            "Type": "ESCALATION"
        }
    ]
}
```

Per ulteriori informazioni, consulta Contatti nella Guida per l'utente di Strumento di gestione degli incidenti.

- Per i dettagli sull'API, consulta ListContacts AWS CLICommand Reference.

`list-engagements`

Il seguente esempio di codice mostra come utilizzare`list-engagements`.

AWS CLI

Come elencare tutti i coinvolgimenti

L'esempio `list-engagements` seguente elenca i coinvolgimenti relativi ai piani di escalation e ai contatti. Puoi anche elencare i coinvolgimenti per un singolo incidente.

```
aws ssm-contacts list-engagements
```

Output:

```
{
    "Engagements": [
        {
            "EngagementArn": "arn:aws:ssm-contacts:us-
east-2:111122223333:engagement/akuam/91792571-0b53-4821-9f73-d25d13d9e529",
            "ContactArn": "arn:aws:ssm-contacts:us-east-2:111122223333:contact/
akuam",
            "Sender": "cli",
            "StartTime": "2021-05-18T20:37:50.300000+00:00"
        },
        {
            "EngagementArn": "arn:aws:ssm-contacts:us-
east-2:111122223333:engagement/akuam/78a29753-3674-4ac5-9f83-0468563567f0",
            "ContactArn": "arn:aws:ssm-contacts:us-east-2:111122223333:contact/
akuam",
            "Sender": "cli",
            "StartTime": "2021-05-18T18:40:26.666000+00:00"
        },
        {
            "EngagementArn": "arn:aws:ssm-contacts:us-
east-2:111122223333:engagement/
example_escalation/69e40ce1-8dbb-4d57-8962-5fbe7fc53356",
            "ContactArn": "arn:aws:ssm-contacts:us-east-2:111122223333:contact/
example_escalation",
            "Sender": "cli",
            "StartTime": "2021-05-18T18:25:41.151000+00:00"
        },
        {
            "EngagementArn": "arn:aws:ssm-contacts:us-
east-2:111122223333:engagement/akuam/607ced0e-e8fa-4ea7-8958-a237b8803f8f",
            "ContactArn": "arn:aws:ssm-contacts:us-east-2:111122223333:contact/
akuam",
            "Sender": "cli",
            "StartTime": "2021-05-18T18:20:58.093000+00:00"
        }
    ]
}
```

Per ulteriori informazioni, consulta Contatti nella Guida per l'utente di Strumento di gestione degli incidenti.

- Per i dettagli sull'API, consulta ListEngagements AWS CLICommand Reference.

`list-page-receipts`

Il seguente esempio di codice mostra come utilizzare`list-page-receipts`.

AWS CLI

Come elencare le ricevute della pagina

L'esempio `command-name` seguente indica se una pagina è stata ricevuta o meno da un contatto.

```
aws ssm-contacts list-page-receipts \
    --page-id "arn:aws:ssm-contacts:us-east-2:111122223333:page/
akuam/94ea0c7b-56d9-46c3-b84a-a37c8b067ad3"
```

Output:

```
{
    "Receipts": [
        {
            "ContactChannelArn": "arn:aws:ssm-contacts:us-
east-2:111122223333:contact-channel/akuam/fc7405c4-46b2-48b7-87b2-93e2f225b90d",
            "ReceiptType": "DELIVERED",
            "ReceiptInfo": "425440",
            "ReceiptTime": "2021-05-18T20:42:57.485000+00:00"
        },
        {
            "ContactChannelArn": "arn:aws:ssm-contacts:us-
east-2:111122223333:contact-channel/akuam/fc7405c4-46b2-48b7-87b2-93e2f225b90d",
            "ReceiptType": "READ",
            "ReceiptInfo": "425440",
            "ReceiptTime": "2021-05-18T20:42:57.907000+00:00"
        },
        {
            "ContactChannelArn": "arn:aws:ssm-contacts:us-
east-2:111122223333:contact-channel/akuam/fc7405c4-46b2-48b7-87b2-93e2f225b90d",
            "ReceiptType": "SENT",
            "ReceiptInfo": "SM6656c19132f1465f9c9c1123a5dde7c9",
            "ReceiptTime": "2021-05-18T20:40:52.962000+00:00"
        }
    ]
}
```

Per ulteriori informazioni, consulta Contatti nella Guida per l'utente di Strumento di gestione degli incidenti.

- Per i dettagli sull'API, consulta ListPageReceipts AWS CLICommand Reference.

`list-pages-by-contact`

Il seguente esempio di codice mostra come utilizzare`list-pages-by-contact`.

AWS CLI

Come elencare le pagine per contatto

L'esempio `list-pages-by-contact` seguente elenca tutte le pagine relative al contatto specificato.

```
aws ssm-contacts list-pages-by-contact \
    --contact-id "arn:aws:ssm-contacts:us-east-2:111122223333:contact/akuam"
```

Output:

```
{
    "Pages": [
        {
            "PageArn": "arn:aws:ssm-contacts:us-east-2:111122223333:page/akuam/
ad0052bd-e606-498a-861b-25726292eb93",
            "EngagementArn": "arn:aws:ssm-contacts:us-
east-2:111122223333:engagement/akuam/78a29753-3674-4ac5-9f83-0468563567f0",
            "ContactArn": "arn:aws:ssm-contacts:us-east-2:111122223333:contact/
akuam",
            "Sender": "cli",
            "SentTime": "2021-05-18T18:43:29.301000+00:00",
            "DeliveryTime": "2021-05-18T18:43:55.265000+00:00",
            "ReadTime": "2021-05-18T18:43:55.708000+00:00"
        }
    ]
}
```

Per ulteriori informazioni, consulta Contatti nella Guida per l'utente di Strumento di gestione degli incidenti.

- Per i dettagli sull'API, consulta ListPagesByContact AWS CLICommand Reference.

`list-pages-by-engagement`

Il seguente esempio di codice mostra come utilizzare`list-pages-by-engagement`.

AWS CLI

Come elencare le pagine dei canali di contatto avviate da un coinvolgimento.

L'esempio `list-pages-by-engagement` seguente elenca le pagine che si sono verificate durante l'interazione con il piano di coinvolgimento definito.

```
aws ssm-contacts list-pages-by-engagement \
    --engagement-id "arn:aws:ssm-contacts:us-east-2:111122223333:engagement/
akuam/78a29753-3674-4ac5-9f83-0468563567f0"
```

Output:

```
{
    "Pages": [
        {
            "PageArn": "arn:aws:ssm-contacts:us-east-2:111122223333:page/akuam/
ad0052bd-e606-498a-861b-25726292eb93",
            "EngagementArn": "arn:aws:ssm-contacts:us-
east-2:111122223333:engagement/akuam/78a29753-3674-4ac5-9f83-0468563567f0",
            "ContactArn": "arn:aws:ssm-contacts:us-east-2:111122223333:contact/
akuam",
            "Sender": "cli",
            "SentTime": "2021-05-18T18:40:27.245000+00:00"
        }
    ]
}
```

Per ulteriori informazioni, consulta Contatti nella Guida per l'utente di Strumento di gestione degli incidenti.

- Per i dettagli sull'API, consulta ListPagesByEngagement AWS CLICommand Reference.

`list-tags-for-resource`

Il seguente esempio di codice mostra come utilizzare`list-tags-for-resource`.

AWS CLI

Come elencare i tag di un contatto

L'esempio `list-tags-for-resource` seguente elenca i tag del contatto specificato.

```
aws ssm-contacts list-tags-for-resource \
    --resource-arn "arn:aws:ssm-contacts:us-east-1:111122223333:contact/akuam"
```

Output:

```
{
    "Tags": [
        {
            "Key": "group1",
            "Value": "1"
        }
    ]
}
```

Per ulteriori informazioni, consulta Tagging nella Guida per l'utente di Strumento di gestione degli incidenti.

- Per i dettagli sull'API, consulta ListTagsForResource AWS CLICommand Reference.

`put-contact-policy`

Il seguente esempio di codice mostra come utilizzare`put-contact-policy`.

AWS CLI

Come condividere un contatto e i coinvolgimenti

L'esempio `put-contact-policy` seguente aggiunge una policy sulle risorse al contatto Akua che condivide il contatto e i coinvolgimenti correlati con il principale.

```
aws ssm-contacts put-contact-policy \
    --contact-arn "arn:aws:ssm-contacts:us-east-1:111122223333:contact/akuam" \
    --policy "{\"Version\":\"2012-10-17\",\"Statement\":[{\"Sid\":
\"ExampleResourcePolicy\",\"Action\":[\"ssm-contacts:GetContact\",\"ssm-
contacts:StartEngagement\",\"ssm-contacts:DescribeEngagement\",\"ssm-
contacts:ListPagesByEngagement\",\"ssm-contacts:StopEngagement\"],
```

```
\"Principal\":{\"AWS\":\"222233334444\"},\"Effect\":\"Allow\",\"Resource
\":[\"arn:aws:ssm-contacts:*:111122223333:contact\/akuam\",\"arn:aws:ssm-
contacts:*:111122223333:engagement\/akuam\/*\"]}]}"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Utilizzo di contatti condivisi e piani di risposta nella Guida per l'utente di Strumento di gestione degli incidenti.

- Per i dettagli sull'API, consulta PutContactPolicy AWS CLICommand Reference.

`send-activation-code`

Il seguente esempio di codice mostra come utilizzare`send-activation-code`.

AWS CLI

Come inviare un codice di attivazione

L'esempio `send-activation-code` seguente invia un codice di attivazione e un messaggio al canale di contatto specificato.

```
aws ssm-contacts send-activation-code \
    --contact-channel-id "arn:aws:ssm-contacts:us-east-1:111122223333:contact-
channel/akuam/8ddae2d1-12c8-4e45-b852-c8587266c400"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Contatti nella Guida per l'utente di Strumento di gestione degli incidenti.

- Per i dettagli sull'API, consulta SendActivationCode AWS CLICommand Reference.

`start-engagement`

Il seguente esempio di codice mostra come utilizzare`start-engagement`.

AWS CLI

Esempio 1: come eseguire la paginazione dei canali di contatto di un contatto

Il comando `start-engagement` seguente esegue la paginazione dei canali di contatto del contatto. Sender, subject, public-subject e public-content sono tutti campi privi di valori. Incident

Manager invia l'oggetto e il contenuto ai canali di contatto VOICE o EMAIL forniti. Incident Manager invia public-subject e public-content ai canali di contatto SMS forniti. Il mittente viene utilizzato per tenere traccia di chi ha avviato il coinvolgimento.

```
aws ssm-contacts start-engagement \
    --contact-id  "arn:aws:ssm-contacts:us-east-2:111122223333:contact/akuam" \
    --sender "cli" \
    --subject "cli-test" \
    --content "Testing engagements via CLI" \
    --public-subject "cli-test" \
    --public-content "Testing engagements va CLI"
```

Output:

```
{
    "EngagementArn": "arn:aws:ssm-contacts:us-east-2:111122223333:engagement/
akuam/607ced0e-e8fa-4ea7-8958-a237b8803f8f"
}
```

Per ulteriori informazioni, consulta Contatti nella Guida per l'utente di Strumento di gestione degli incidenti.

Esempio 2: come inserire un contatto nel piano di escalation fornito.

Il comando `start-engagement` seguente coinvolge i contatti attraverso un piano di escalation. Ogni contatto viene paginato in base al proprio piano di coinvolgimento.

```
aws ssm-contacts start-engagement \
    --contact-id  "arn:aws:ssm-contacts:us-east-2:111122223333:contact/
example_escalation" \
    --sender "cli" \
    --subject "cli-test" \
    --content "Testing engagements via CLI" \
    --public-subject "cli-test" \
    --public-content "Testing engagements va CLI"
```

Output:

```
{
    "EngagementArn": "arn:aws:ssm-contacts:us-east-2:111122223333:engagement/
example_escalation/69e40ce1-8dbb-4d57-8962-5fbe7fc53356"
```

```
}
```

Per ulteriori informazioni, consulta Contatti nella Guida per l'utente di Strumento di gestione degli incidenti.

- Per i dettagli sull'API, consulta StartEngagement AWS CLICommand Reference.

`stop-engagement`

Il seguente esempio di codice mostra come utilizzare`stop-engagement`.

AWS CLI

Come arrestare un coinvolgimento

L'esempio `stop-engagement` seguente impedisce a un coinvolgimento di cercare ulteriori contatti e canali di contatto.

```
aws ssm-contacts stop-engagement \
    --engagement-id "arn:aws:ssm-contacts:us-east-2:111122223333:engagement/
example_escalation/69e40ce1-8dbb-4d57-8962-5fbe7fc53356"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Contatti nella Guida per l'utente di Strumento di gestione degli incidenti.

- Per i dettagli sull'API, consulta StopEngagement AWS CLICommand Reference.

`tag-resource`

Il seguente esempio di codice mostra come utilizzare`tag-resource`.

AWS CLI

Come applicare tag a un contatto

L'esempio `tag-resource` seguente contrassegna un contatto specificato con la coppia chiave di tag-valore fornita.

```
aws ssm-contacts tag-resource \
    --resource-arn "arn:aws:ssm-contacts:us-east-1:111122223333:contact/akuam" \
```

```
    --tags '[{"Key":"group1","Value":"1"}]'
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Tagging nella Guida per l'utente di Strumento di gestione degli incidenti.

- Per i dettagli sull'API, consulta TagResource AWS CLICommand Reference.

`untag-resource`

Il seguente esempio di codice mostra come utilizzare`untag-resource`.

AWS CLI

Come rimuovere i tag da un contatto

L'esempio `untag-resource` seguente rimuove il tag group1 dal contatto specificato.

```
aws ssm-contacts untag-resource \
    --resource-arn "arn:aws:ssm-contacts:us-east-1:111122223333:contact/akuam" \
    --tag-keys "group1"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Tagging nella Guida per l'utente di Strumento di gestione degli incidenti.

- Per i dettagli sull'API, consulta UntagResource AWS CLICommand Reference.

`update-contact-channel`

Il seguente esempio di codice mostra come utilizzare`update-contact-channel`.

AWS CLI

Come aggiornare un canale di contatto

L'esempio `update-contact-channel` seguente aggiorna il nome e l'indirizzo di consegna di un canale di contatto.

```
aws ssm-contacts update-contact-channel \
```

```
    --contact-channel-id "arn:aws:ssm-contacts:us-east-2:111122223333:contact-
channel/akuam/49f3c24d-5f9f-4638-ae25-3f49e04229ad" \
    --name "akuas voice channel" \
    --delivery-address '{"SimpleAddress": "+15005550198"}'
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Contatti nella Guida per l'utente di Strumento di gestione degli incidenti.

- Per i dettagli sull'API, consulta UpdateContactChannel AWS CLICommand Reference.

`update-contact`

Il seguente esempio di codice mostra come utilizzare`update-contact`.

AWS CLI

Come aggiornare il piano di coinvolgimento dei contatti

L'esempio `update-contact` seguente aggiorna il piano di coinvolgimento del contatto Akua per includere i tre tipi di canali di contatto. Questa operazione viene eseguita dopo la creazione dei canali di contatto per Akua.

```
aws ssm-contacts update-contact \
    --contact-id "arn:aws:ssm-contacts:us-east-2:111122223333:contact/akuam" \
    --plan '{"Stages": [{"DurationInMinutes": 5, "Targets": [{"ChannelTargetInfo":
 {"ContactChannelId": "arn:aws:ssm-contacts:us-east-2:111122223333:contact-
channel/akuam/beb25840-5ac8-4644-95cc-7a8de390fa65","RetryIntervalInMinutes":
 1 }}]}, {"DurationInMinutes": 5, "Targets": [{"ChannelTargetInfo":
{"ContactChannelId": "arn:aws:ssm-contacts:us-east-2:111122223333:contact-channel/
akuam/49f3c24d-5f9f-4638-ae25-3f49e04229ad", "RetryIntervalInMinutes": 1}}]},
 {"DurationInMinutes": 5, "Targets": [{"ChannelTargetInfo": {"ContactChannelId":
 "arn:aws:ssm-contacts:us-east-2:111122223333:contact-channel/akuam/77d4f447-
f619-4954-afff-85551e369c2a","RetryIntervalInMinutes": 1 }}]}]}'
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Contatti nella Guida per l'utente di Strumento di gestione degli incidenti.

- Per i dettagli sull'API, consulta UpdateContact AWS CLICommand Reference.

Esempi di utilizzo di Amazon Inspector AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando Amazon Inspector. AWS Command Line Interface

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`add-attributes-to-findings`

Il seguente esempio di codice mostra come usare`add-attributes-to-findings`.

AWS CLI

Come aggiungere attributi agli esiti

Il comando `add-attribute-to-finding` seguente assegna un attributo con la chiave `Example` e il valore `example` all'esito con l'ARN `arn:aws:inspector:us-west-2:123456789012:target/0-0kFIPusq/template/0-8l1VIE0D/run/0-Z02cjjug/finding/0-T8yM9mEU`:

```
aws inspector add-attributes-to-findings --finding-arns arn:aws:inspector:us-west-2:123456789012:target/0-0kFIPusq/template/0-8l1VIE0D/run/0-Z02cjjug/finding/0-T8yM9mEU --attributes key=Example,value=example
```

Output:

```
{
    "failedItems": {}
}
```

Per ulteriori informazioni consulta Esiti di Amazon Inspector nella Guida per l'utente di Amazon Inspector.

- Per i dettagli sull'API, consulta AddAttributesToFindings AWS CLICommand Reference.

`associate-member`

Il seguente esempio di codice mostra come utilizzare`associate-member`.

AWS CLI

Esempio: associare un AWS account a un amministratore delegato di Amazon Inspector

L'`associate-member`esempio seguente associa un AWS account a un amministratore delegato di Amazon Inspector.

```
aws inspector2 associate-member \
    --account-id 123456789012
```

Output:

```
{
    "accountId": "123456789012"
}
```

Per ulteriori informazioni, consulta la sezione Gestione di più account in Amazon Inspector with AWS Organizations nella Amazon Inspector User Guide.

- Per i dettagli sull'API, consulta AWS CLI Command AssociateMemberReference.

`create-assessment-target`

Il seguente esempio di codice mostra come utilizzare`create-assessment-target`.

AWS CLI

Come creare una destinazione di valutazione

Il comando `create-assessment-target` seguente crea una destinazione di valutazione denominata `ExampleAssessmentTarget` utilizzando il gruppo di risorse con l'ARN `arn:aws:inspector:us-west-2:123456789012:resourcegroup/0-AB6DMKnv`:

```
aws inspector create-assessment-target --assessment-target-
name ExampleAssessmentTarget --resource-group-arn arn:aws:inspector:us-
west-2:123456789012:resourcegroup/0-AB6DMKnv
```

Output:

```
{
    "assessmentTargetArn": "arn:aws:inspector:us-west-2:123456789012:target/0-
nvgVhaxX"
}
```

Per ulteriori informazioni, consulta Destinazioni delle valutazioni di Amazon Inspector nella Guida per l'utente di Amazon Inspector.

- Per i dettagli sull'API, consulta CreateAssessmentTarget AWS CLICommand Reference.

`create-assessment-template`

Il seguente esempio di codice mostra come utilizzare`create-assessment-template`.

AWS CLI

Come creare un modello di valutazione

Il comando `create-assessment-template` seguente crea un modello di valutazione denominato `ExampleAssessmentTemplate` per la destinazione di valutazione con l'ARN `arn:aws:inspector:us-west-2:123456789012:target/0-nvgVhaxX`:

```
aws inspector create-assessment-template --assessment-target-
arn arn:aws:inspector:us-west-2:123456789012:target/0-nvgVhaxX --assessment-
template-name ExampleAssessmentTemplate --duration-in-seconds 180 --rules-package-
arns arn:aws:inspector:us-west-2:758058086616:rulespackage/0-9hgA516p --user-
attributes-for-findings key=ExampleTag,value=examplevalue
```

Output:

```
{
    "assessmentTemplateArn": "arn:aws:inspector:us-west-2:123456789012:target/0-
nvgVhaxX/template/0-it5r2S4T"
}
```

Per ulteriori informazioni, consulta Modelli di valutazione ed esecuzioni delle valutazioni di Amazon Inspector nella Guida per l'utente di Amazon Inspector.

- Per i dettagli sull'API, consulta CreateAssessmentTemplate AWS CLICommand Reference.

`create-filter`

Il seguente esempio di codice mostra come utilizzare`create-filter`.

AWS CLI

Come creare un filtro

L'esempio `create-filter` seguente crea una regola di soppressione che omette gli esiti del tipo di istanza ECR.

```
aws inspector2 create-filter \
    --name "ExampleSuppressionRuleECR" \
    --description "This suppression rule omits ECR instance type findings" \
    --action SUPPRESS \
    --filter-criteria 'resourceType=[{comparison="EQUALS",
 value="AWS_ECR_INSTANCE"}]'
```

Output:

```
{
    "arn": "arn:aws:inspector2:us-west-2:123456789012:owner/o-EXAMPLE222/filter/
EXAMPLE444444444"
}
```

Per ulteriori informazioni, consulta Filtro degli esiti di Amazon Inspector nella Guida per l'utente di Amazon Inspector.

- Per i dettagli sull'API, consulta CreateFilter AWS CLICommand Reference.

`create-findings-report`

Il seguente esempio di codice mostra come utilizzare`create-findings-report`.

AWS CLI

Come creare un report degli esiti

L'esempio `create-findings-report` seguente crea un report degli esiti.

```
aws inspector2 create-findings-report \
    --report-format CSV \
    --s3-destination bucketName=inspector-sbom-123456789012,keyPrefix=sbom-
key,kmsKeyArn=arn:aws:kms:us-west-2:123456789012:key/a1b2c3d4-5678-90ab-cdef-
EXAMPLE33333 \
    --filter-criteria '{"ecrImageRepositoryName":
[{"comparison":"EQUALS","value":"debian"}]}'
```

Output:

```
{
    "reportId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE33333"
}
```

Per ulteriori informazioni, consulta Gestione degli esiti in Amazon Inspector nella Guida per l'utente di Amazon Inspector.

- Per i dettagli sull'API, consulta CreateFindingsReport AWS CLICommand Reference.

`create-resource-group`

Il seguente esempio di codice mostra come utilizzare`create-resource-group`.

AWS CLI

Come creare un gruppo di risorse

Il comando `create-resource-group` seguente crea un gruppo di risorse utilizzando la chiave di tag `Name` e il valore `example`:

```
aws inspector create-resource-group --resource-group-tags key=Name,value=example
```

Output:

```
{
   "resourceGroupArn": "arn:aws:inspector:us-west-2:123456789012:resourcegroup/0-
AB6DMKnv"
}
```

Per ulteriori informazioni, consulta Destinazioni delle valutazioni di Amazon Inspector nella Guida per l'utente di Amazon Inspector.

- Per i dettagli sull'API, consulta CreateResourceGroup AWS CLICommand Reference.

`create-sbom-export`

Il seguente esempio di codice mostra come utilizzare`create-sbom-export`.

AWS CLI

Come creare un report SBOM (Software Bill of Materials)

L'esempio `create-sbom-export` seguente crea un report SBOM (Software Bill of Materials).

```
aws inspector2 create-sbom-export \
    --report-format SPDX_2_3 \
    --resource-filter-criteria
 'ecrRepositoryName=[{comparison="EQUALS",value="debian"}]' \
    --s3-destination bucketName=inspector-sbom-123456789012,keyPrefix=sbom-
key,kmsKeyArn=arn:aws:kms:us-west-2:123456789012:key/a1b2c3d4-5678-90ab-cdef-
EXAMPLE33333
```

Output:

```
{
    "reportId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE33333"
}
```

Per ulteriori informazioni, consulta Exporting SBOMs with Amazon Inspector nella Amazon Inspector User Guide.

- Per i dettagli sull'API, consulta Command CreateSbomExportReference AWS CLI .

`delete-assessment-run`

Il seguente esempio di codice mostra come utilizzare`delete-assessment-run`.

AWS CLI

Come eliminare un ciclo di valutazione

Il comando `delete-assessment-run` seguente elimina il ciclo di valutazione con l'ARN `arn:aws:inspector:us-west-2:123456789012:target/0-nvgVhaxX/template/0-it5r2S4T/run/0-11LMTAVe`:

```
aws inspector delete-assessment-run --assessment-run-arn arn:aws:inspector:us-west-2:123456789012:target/0-nvgVhaxX/template/0-it5r2S4T/run/0-11LMTAVe
```

Per ulteriori informazioni, consulta Modelli di valutazione ed esecuzioni delle valutazioni di Amazon Inspector nella Guida per l'utente di Amazon Inspector.

- Per i dettagli sull'API, consulta DeleteAssessmentRun AWS CLICommand Reference.

`delete-assessment-target`

Il seguente esempio di codice mostra come utilizzare`delete-assessment-target`.

AWS CLI

Come eliminare un target di valutazione

Il comando `delete-assessment-target` seguente elimina la destinazione di valutazione con l'ARN `arn:aws:inspector:us-west-2:123456789012:target/0-0kFIPusq`:

```
aws inspector delete-assessment-target --assessment-target-arn arn:aws:inspector:us-west-2:123456789012:target/0-0kFIPusq
```

Per ulteriori informazioni, consulta Destinazioni delle valutazioni di Amazon Inspector nella Guida per l'utente di Amazon Inspector.

- Per i dettagli sull'API, consulta DeleteAssessmentTarget AWS CLICommand Reference.

`delete-assessment-template`

Il seguente esempio di codice mostra come utilizzare`delete-assessment-template`.

AWS CLI

Come eliminare un modello di valutazione

Il comando `delete-assessment-template` seguente elimina il modello di valutazione con l'ARN `arn:aws:inspector:us-west-2:123456789012:target/0-nvgVhaxX/template/0-it5r2S4T`:

```
aws inspector delete-assessment-template --assessment-template-
arn arn:aws:inspector:us-west-2:123456789012:target/0-nvgVhaxX/template/0-it5r2S4T
```

Per ulteriori informazioni, consulta Modelli di valutazione ed esecuzioni delle valutazioni di Amazon Inspector nella Guida per l'utente di Amazon Inspector.

- Per i dettagli sull'API, consulta DeleteAssessmentTemplate AWS CLICommand Reference.

`delete-filter`

Il seguente esempio di codice mostra come utilizzare`delete-filter`.

AWS CLI

Come eliminare un filtro

L'esempio `delete-filter` seguente elimina un filtro.

```
aws inspector2 delete-filter \
    --arn "arn:aws:inspector2:us-west-2:123456789012:owner/o-EXAMPLE222/filter/
EXAMPLE444444444"
```

Output:

```
{
    "arn": "arn:aws:inspector2:us-west-2:123456789012:owner/o-EXAMPLE222/filter/
EXAMPLE444444444"
}
```

Per ulteriori informazioni, consulta Filtro degli esiti di Amazon Inspector nella Guida per l'utente di Amazon Inspector.

- Per i dettagli sull'API, consulta DeleteFilter AWS CLICommand Reference.

`describe-assessment-runs`

Il seguente esempio di codice mostra come utilizzare`describe-assessment-runs`.

AWS CLI

Come descrivere i cicli di valutazione

Il comando `describe-assessment-run` seguente descrive un ciclo di valutazione con l'ARN `arn:aws:inspector:us-west-2:123456789012:target/0-0kFIPusq/template/0-4r1V2mAw/run/0-MKkpXXPE`:

```
aws inspector describe-assessment-runs --assessment-run-arns arn:aws:inspector:us-west-2:123456789012:target/0-0kFIPusq/template/0-4r1V2mAw/run/0-MKkpXXPE
```

Output:

```
{
        "assessmentRuns": [
          {
                "arn": "arn:aws:inspector:us-west-2:123456789012:target/0-0kFIPusq/
template/0-4r1V2mAw/run/0-MKkpXXPE",
                "assessmentTemplateArn": "arn:aws:inspector:us-
west-2:123456789012:target/0-0kFIPusq/template/0-4r1V2mAw",
                "completedAt": 1458680301.4,
                "createdAt": 1458680170.035,
                "dataCollected": true,
                "durationInSeconds": 3600,
                "name": "Run 1 for ExampleAssessmentTemplate",
                "notifications": [],
                "rulesPackageArns": [
                  "arn:aws:inspector:us-west-2:758058086616:rulespackage/0-X1KXtawP"
                ],
                "startedAt": 1458680170.161,
                "state": "COMPLETED",
                "stateChangedAt": 1458680301.4,
                "stateChanges": [
                  {
                        "state": "CREATED",
                        "stateChangedAt": 1458680170.035
                  },
                  {
                        "state": "START_DATA_COLLECTION_PENDING",
                        "stateChangedAt": 1458680170.065
                  },
                  {
                        "state": "START_DATA_COLLECTION_IN_PROGRESS",
                        "stateChangedAt": 1458680170.096
                  },
                  {
                        "state": "COLLECTING_DATA",
```

```
                        "stateChangedAt": 1458680170.161
                    },
                    {
                        "state": "STOP_DATA_COLLECTION_PENDING",
                        "stateChangedAt": 1458680239.883
                    },
                    {
                        "state": "DATA_COLLECTED",
                        "stateChangedAt": 1458680299.847
                    },
                    {
                        "state": "EVALUATING_RULES",
                        "stateChangedAt": 1458680300.099
                    },
                    {
                        "state": "COMPLETED",
                        "stateChangedAt": 1458680301.4
                    }
                ],
                "userAttributesForFindings": []
            }
        ],
        "failedItems": {}
}
```

Per ulteriori informazioni, consulta Modelli di valutazione ed esecuzioni delle valutazioni di Amazon Inspector nella Guida per l'utente di Amazon Inspector.

- Per i dettagli sull'API, consulta DescribeAssessmentRuns AWS CLICommand Reference.

`describe-assessment-targets`

Il seguente esempio di codice mostra come utilizzare`describe-assessment-targets`.

AWS CLI

Come descrivere le destinazioni di valutazione

Il comando `describe-assessment-targets` seguente descrive la destinazione di valutazione con l'ARN `arn:aws:inspector:us-west-2:123456789012:target/0-0kFIPusq`:

```
aws inspector describe-assessment-targets --assessment-target-
arns arn:aws:inspector:us-west-2:123456789012:target/0-0kFIPusq
```

Output:

```
{
      "assessmentTargets": [
        {
              "arn": "arn:aws:inspector:us-west-2:123456789012:target/0-0kFIPusq",
              "createdAt": 1458074191.459,
              "name": "ExampleAssessmentTarget",
              "resourceGroupArn": "arn:aws:inspector:us-
west-2:123456789012:resourcegroup/0-PyGXopAI",
              "updatedAt": 1458074191.459
        }
      ],
      "failedItems": {}
}
```

Per ulteriori informazioni, consulta Destinazioni delle valutazioni di Amazon Inspector nella Guida per l'utente di Amazon Inspector.

- Per i dettagli sull'API, consulta DescribeAssessmentTargets AWS CLICommand Reference.

`describe-assessment-templates`

Il seguente esempio di codice mostra come utilizzare`describe-assessment-templates`.

AWS CLI

Come descrivere i modelli di valutazione

Il comando `describe-assessment-templates` seguente descrive il modello di valutazione con l'ARN `arn:aws:inspector:us-west-2:123456789012:target/0-0kFIPusq/template/0-4r1V2mAw`:

```
aws inspector describe-assessment-templates --assessment-template-
arns arn:aws:inspector:us-west-2:123456789012:target/0-0kFIPusq/template/0-4r1V2mAw
```

Output:

```
{
      "assessmentTemplates": [
        {
```

```
            "arn": "arn:aws:inspector:us-west-2:123456789012:target/0-0kFIPusq/
template/0-4r1V2mAw",
            "assessmentTargetArn": "arn:aws:inspector:us-
west-2:123456789012:target/0-0kFIPusq",
            "createdAt": 1458074191.844,
            "durationInSeconds": 3600,
            "name": "ExampleAssessmentTemplate",
            "rulesPackageArns": [
              "arn:aws:inspector:us-west-2:758058086616:rulespackage/0-X1KXtawP"
            ],
            "userAttributesForFindings": []
        }
      ],
      "failedItems": {}
}
```

Per ulteriori informazioni, consulta Modelli di valutazione ed esecuzioni delle valutazioni di Amazon Inspector nella Guida per l'utente di Amazon Inspector.

- Per i dettagli sull'API, consulta DescribeAssessmentTemplates AWS CLICommand Reference.

`describe-cross-account-access-role`

Il seguente esempio di codice mostra come utilizzare`describe-cross-account-access-role`.

AWS CLI

Come descrivere il ruolo di accesso multi-account

Il `describe-cross-account-access-role` comando seguente descrive il ruolo IAM che consente ad Amazon Inspector di accedere al tuo AWS account:

```
aws inspector describe-cross-account-access-role
```

Output:

```
{
        "registeredAt": 1458069182.826,
        "roleArn": "arn:aws:iam::123456789012:role/inspector",
        "valid": true
}
```

Per ulteriori informazioni, consulta Impostazione di Amazon Inspector nella Guida per l'utente di Amazon Inspector.

- Per i dettagli sull'API, consulta DescribeCrossAccountAccessRole AWS CLICommand Reference.

`describe-findings`

Il seguente esempio di codice mostra come utilizzare`describe-findings`.

AWS CLI

Come descrivere gli esiti

Il comando `describe-findings` seguente descrive l'esito con l'ARN `arn:aws:inspector:us-west-2:123456789012:target/0-0kFIPusq/template/0-4r1V2mAw/run/0-MKkpXXPE/finding/0-HwPnsDm4`:

```
aws inspector describe-findings --finding-arns arn:aws:inspector:us-west-2:123456789012:target/0-0kFIPusq/template/0-4r1V2mAw/run/0-MKkpXXPE/finding/0-HwPnsDm4
```

Output:

```
{
      "failedItems": {},
      "findings": [
        {
              "arn": "arn:aws:inspector:us-west-2:123456789012:target/0-0kFIPusq/
template/0-4r1V2mAw/run/0-MKkpXXPE/finding/0-HwPnsDm4",
              "assetAttributes": {
                "ipv4Addresses": [],
                "schemaVersion": 1
              },
              "assetType": "ec2-instance",
              "attributes": [],
              "confidence": 10,
              "createdAt": 1458680301.37,
              "description": "Amazon Inspector did not find any potential security
 issues during this assessment.",
              "indicatorOfCompromise": false,
              "numericSeverity": 0,
```

```
            "recommendation": "No remediation needed.",
            "schemaVersion": 1,
            "service": "Inspector",
            "serviceAttributes": {
              "assessmentRunArn": "arn:aws:inspector:us-
west-2:123456789012:target/0-0kFIPusq/template/0-4r1V2mAw/run/0-MKkpXXPE",
              "rulesPackageArn": "arn:aws:inspector:us-
west-2:758058086616:rulespackage/0-X1KXtawP",
              "schemaVersion": 1
            },
            "severity": "Informational",
            "title": "No potential security issues found",
            "updatedAt": 1458680301.37,
            "userAttributes": []
        }
      ]
}
```

Per ulteriori informazioni consulta Esiti di Amazon Inspector nella Guida per l'utente di Amazon Inspector.

- Per i dettagli sull'API, consulta DescribeFindings AWS CLICommand Reference.

`describe-resource-groups`

Il seguente esempio di codice mostra come utilizzare`describe-resource-groups`.

AWS CLI

Come descrivere i gruppi di risorse

Il comando `describe-resource-groups` seguente descrive il gruppo di risorse con l'ARN `arn:aws:inspector:us-west-2:123456789012:resourcegroup/0-PyGXopAI`:

```
aws inspector describe-resource-groups --resource-group-arns arn:aws:inspector:us-
west-2:123456789012:resourcegroup/0-PyGXopAI
```

Output:

```
{
      "failedItems": {},
      "resourceGroups": [
        {
```

```
            "arn": "arn:aws:inspector:us-west-2:123456789012:resourcegroup/0-
PyGXopAI",
            "createdAt": 1458074191.098,
            "tags": [
              {
                    "key": "Name",
                    "value": "example"
              }
            ]
        }
      ]
}
```

Per ulteriori informazioni, consulta Destinazioni delle valutazioni di Amazon Inspector nella Guida per l'utente di Amazon Inspector.

- Per i dettagli sull'API, consulta DescribeResourceGroups AWS CLICommand Reference.

`describe-rules-packages`

Il seguente esempio di codice mostra come utilizzare`describe-rules-packages`.

AWS CLI

Come descrivere i pacchetti di regole

Il comando `describe-rules-packages` seguente descrive il pacchetto di regole con l'ARN `arn:aws:inspector:us-west-2:758058086616:rulespackage/0-9hgA516p`:

```
aws inspector describe-rules-packages --rules-package-arns arn:aws:inspector:us-
west-2:758058086616:rulespackage/0-9hgA516p
```

Output:

```
{
      "failedItems": {},
      "rulesPackages": [
        {
            "arn": "arn:aws:inspector:us-
west-2:758058086616:rulespackage/0-9hgA516p",
            "description": "The rules in this package help verify whether the EC2
 instances in your application are exposed to Common Vulnerabilities and
```

```
                Exposures (CVEs). Attacks can exploit unpatched vulnerabilities to
 compromise the confidentiality, integrity, or availability of your service
                or data. The CVE system provides a reference for publicly known
 information security vulnerabilities and exposures. For more information, see
                [https://cve.mitre.org/](https://cve.mitre.org/). If a particular CVE
 appears in one of the produced Findings at the end of a completed
                Inspector assessment, you can search [https://cve.mitre.org/](https://
cve.mitre.org/) using the CVE's ID (for example, \"CVE-2009-0021\") to
                find detailed information about this CVE, its severity, and how to
 mitigate it. ",
                "name": "Common Vulnerabilities and Exposures",
                "provider": "Amazon Web Services, Inc.",
                "version": "1.1"
            }
        ]
}
```

Per ulteriori informazioni, consulta Pacchetti di regole e regole di Amazon Inspector nella Guida per l'utente di Amazon Inspector.

- Per i dettagli sull'API, consulta DescribeRulesPackages AWS CLICommand Reference.

`disassociate-member`

Il seguente esempio di codice mostra come utilizzare`disassociate-member`.

AWS CLI

Esempio: come rimuovere l'associazione di un account membro da un amministratore delegato di Amazon Inspector

L'`disassociate-member`esempio seguente dissocia un AWS account da un amministratore delegato di Amazon Inspector.

```
aws inspector2 disassociate-member \
    --account-id 123456789012
```

Output:

```
{
    "accountId": "123456789012"
```

```
}
```

Per ulteriori informazioni, consulta la sezione Gestione di più account in Amazon Inspector with AWS Organizations nella Amazon Inspector User Guide.

- Per i dettagli sull'API, consulta AWS CLI Command DisassociateMemberReference.

`get-configuration`

Il seguente esempio di codice mostra come utilizzare`get-configuration`.

AWS CLI

Come ottenere la configurazione delle impostazioni delle scansioni di Inspector

L'esempio `get-configuration` seguente ottiene la configurazione delle impostazioni per le scansioni di Inspector.

```
aws inspector2 get-configuration
```

Output:

```
{
    "ec2Configuration": {
        "scanModeState": {
            "scanMode": "EC2_HYBRID",
            "scanModeStatus": "SUCCESS"
        }
    },
    "ecrConfiguration": {
        "rescanDurationState": {
            "pullDateRescanDuration": "DAYS_90",
            "rescanDuration": "DAYS_30",
            "status": "SUCCESS",
            "updatedAt": "2024-05-14T21:16:20.237000+00:00"
        }
     }
}
```

Per ulteriori informazioni, consulta Scansione automatizzata delle risorse con Amazon Inspector nella Guida per l'utente di Amazon Inspector.

- Per i dettagli sull'API, consulta GetConfiguration AWS CLICommand Reference.

`get-member`

Il seguente esempio di codice mostra come utilizzare`get-member`.

AWS CLI

Esempio: come ottenere le informazioni sui membri per l'organizzazione

aws inspector2 get-member --account-id 123456789012

Output:

```
{
        "member": {
        "accountId": "123456789012",
        "delegatedAdminAccountId": "123456789012",
        "relationshipStatus": "ENABLED",
        "updatedAt": "2023-09-11T09:57:20.520000-07:00"
    }
}
```

Per ulteriori informazioni, consulta la sezione Gestione di più account in Amazon Inspector with AWS Organizations nella Amazon Inspector User Guide.

- Per i dettagli sull'API, consulta AWS CLI Command GetMemberReference.

`get-telemetry-metadata`

Il seguente esempio di codice mostra come utilizzare`get-telemetry-metadata`.

AWS CLI

Come ottenere i metadati di telemetria

Il comando `get-telemetry-metadata` seguente genera informazioni sui dati raccolti per il ciclo di valutazione con l'ARN `arn:aws:inspector:us-west-2:123456789012:target/0-0kFIPusq/template/0-4r1V2mAw/run/0-MKkpXXPE`:

```
aws inspector get-telemetry-metadata --assessment-run-arn arn:aws:inspector:us-west-2:123456789012:target/0-0kFIPusq/template/0-4r1V2mAw/run/0-MKkpXXPE
```

Output:


```
{
      "telemetryMetadata": [
        {
              "count": 2,
              "dataSize": 345,
              "messageType": "InspectorDuplicateProcess"
        },
        {
              "count": 3,
              "dataSize": 255,
              "messageType": "InspectorTimeEventMsg"
        },
        {
              "count": 4,
              "dataSize": 1082,
              "messageType": "InspectorNetworkInterface"
        },
        {
              "count": 2,
              "dataSize": 349,
              "messageType": "InspectorDnsEntry"
        },
        {
              "count": 11,
              "dataSize": 2514,
              "messageType": "InspectorDirectoryInfoMsg"
        },
        {
              "count": 1,
              "dataSize": 179,
              "messageType": "InspectorTcpV6ListeningPort"
        },
        {
              "count": 101,
              "dataSize": 10949,
              "messageType": "InspectorTerminal"
        },
        {
              "count": 26,
              "dataSize": 5916,
              "messageType": "InspectorUser"
        },
```

```
        {
              "count": 282,
              "dataSize": 32148,
              "messageType": "InspectorDynamicallyLoadedCodeModule"
        },
        {
              "count": 18,
              "dataSize": 10172,
              "messageType": "InspectorCreateProcess"
        },
        {
              "count": 3,
              "dataSize": 8001,
              "messageType": "InspectorProcessPerformance"
        },
        {
              "count": 1,
              "dataSize": 360,
              "messageType": "InspectorOperatingSystem"
        },
        {
              "count": 6,
              "dataSize": 546,
              "messageType": "InspectorStopProcess"
        },
        {
              "count": 1,
              "dataSize": 1553,
              "messageType": "InspectorInstanceMetaData"
        },
        {
              "count": 2,
              "dataSize": 434,
              "messageType": "InspectorTcpV4Connection"
        },
        {
              "count": 474,
              "dataSize": 2960322,
              "messageType": "InspectorPackageInfo"
        },
        {
              "count": 3,
              "dataSize": 2235,
              "messageType": "InspectorSystemPerformance"
```

```
        },
        {
              "count": 105,
              "dataSize": 46048,
              "messageType": "InspectorCodeModule"
        },
        {
              "count": 1,
              "dataSize": 182,
              "messageType": "InspectorUdpV6ListeningPort"
        },
        {
              "count": 2,
              "dataSize": 371,
              "messageType": "InspectorUdpV4ListeningPort"
        },
        {
              "count": 18,
              "dataSize": 8362,
              "messageType": "InspectorKernelModule"
        },
        {
              "count": 29,
              "dataSize": 48788,
              "messageType": "InspectorConfigurationInfo"
        },
        {
              "count": 1,
              "dataSize": 79,
              "messageType": "InspectorMonitoringStart"
        },
        {
              "count": 5,
              "dataSize": 0,
              "messageType": "InspectorSplitMsgBegin"
        },
        {
              "count": 51,
              "dataSize": 4593,
              "messageType": "InspectorGroup"
        },
        {
              "count": 1,
              "dataSize": 184,
```

```
                "messageType": "InspectorTcpV4ListeningPort"
        },
        {
                "count": 1159,
                "dataSize": 3146579,
                "messageType": "Total"
        },
        {
                "count": 5,
                "dataSize": 0,
                "messageType": "InspectorSplitMsgEnd"
        },
        {
                "count": 1,
                "dataSize": 612,
                "messageType": "InspectorLoadImageInProcess"
        }
      ]
}
```

- Per i dettagli sull'API, consulta GetTelemetryMetadata AWS CLICommand Reference.

`list-account-permissions`

Il seguente esempio di codice mostra come utilizzare`list-account-permissions`.

AWS CLI

Come elencare le autorizzazioni dell'account

L'esempio `list-account-permissions` seguente elenca le autorizzazioni dell'account.

```
aws inspector2 list-account-permissions
```

Output:

```
{
    "permissions": [
        {
            "operation": "ENABLE_SCANNING",
            "service": "ECR"
        },
```

```
        {
            "operation": "DISABLE_SCANNING",
            "service": "ECR"
        },
        {
            "operation": "ENABLE_REPOSITORY",
            "service": "ECR"
        },
        {
            "operation": "DISABLE_REPOSITORY",
            "service": "ECR"
        },
        {
            "operation": "ENABLE_SCANNING",
            "service": "EC2"
        },
        {
            "operation": "DISABLE_SCANNING",
            "service": "EC2"
        },
        {
            "operation": "ENABLE_SCANNING",
            "service": "LAMBDA"
        },
        {
            "operation": "DISABLE_SCANNING",
            "service": "LAMBDA"
        }
    ]
}
```

Per ulteriori informazioni, consulta Identity and Access Management per Amazon Inspector nella Guida per l'utente di Amazon Inspector.

- Per i dettagli sull'API, consulta ListAccountPermissions AWS CLICommand Reference.

`list-assessment-run-agents`

Il seguente esempio di codice mostra come utilizzare`list-assessment-run-agents`.

AWS CLI

Come elencare gli agenti del ciclo di valutazione

Il comando `list-assessment-run-agents` seguente elenca gli agenti del ciclo di valutazione con l'ARN specificato.

```
aws inspector list-assessment-run-agents \
    --assessment-run-arn arn:aws:inspector:us-west-2:123456789012:target/0-0kFIPusq/template/0-4r1V2mAw/run/0-MKkpXXPE
```

Output:

```
{
    "assessmentRunAgents": [
        {
            "agentHealth": "HEALTHY",
            "agentHealthCode": "HEALTHY",
            "agentId": "i-49113b93",
            "assessmentRunArn": "arn:aws:inspector:us-
west-2:123456789012:target/0-0kFIPusq/template/0-4r1V2mAw/run/0-MKkpXXPE",
            "telemetryMetadata": [
                {
                    "count": 2,
                    "dataSize": 345,
                    "messageType": "InspectorDuplicateProcess"
                },
                {
                    "count": 3,
                    "dataSize": 255,
                    "messageType": "InspectorTimeEventMsg"
                },
                {
                    "count": 4,
                    "dataSize": 1082,
                    "messageType": "InspectorNetworkInterface"
                },
                {
                    "count": 2,
                    "dataSize": 349,
                    "messageType": "InspectorDnsEntry"
                },
                {
                    "count": 11,
                    "dataSize": 2514,
                    "messageType": "InspectorDirectoryInfoMsg"
                },
```

```
                {
                    "count": 1,
                    "dataSize": 179,
                    "messageType": "InspectorTcpV6ListeningPort"
                },
                {
                    "count": 101,
                    "dataSize": 10949,
                    "messageType": "InspectorTerminal"
                },
                {
                    "count": 26,
                    "dataSize": 5916,
                    "messageType": "InspectorUser"
                },
                {
                    "count": 282,
                    "dataSize": 32148,
                    "messageType": "InspectorDynamicallyLoadedCodeModule"
                },
                {
                    "count": 18,
                    "dataSize": 10172,
                    "messageType": "InspectorCreateProcess"
                },
                {
                    "count": 3,
                    "dataSize": 8001,
                    "messageType": "InspectorProcessPerformance"
                },
                {
                    "count": 1,
                    "dataSize": 360,
                    "messageType": "InspectorOperatingSystem"
                },
                {
                    "count": 6,
                    "dataSize": 546,
                    "messageType": "InspectorStopProcess"
                },
                {
                    "count": 1,
                    "dataSize": 1553,
                    "messageType": "InspectorInstanceMetaData"
```

```
                },
                {
                    "count": 2,
                    "dataSize": 434,
                    "messageType": "InspectorTcpV4Connection"
                },
                {
                    "count": 474,
                    "dataSize": 2960322,
                    "messageType": "InspectorPackageInfo"
                },
                {
                    "count": 3,
                    "dataSize": 2235,
                    "messageType": "InspectorSystemPerformance"
                },
                {
                    "count": 105,
                    "dataSize": 46048,
                    "messageType": "InspectorCodeModule"
                },
                {
                    "count": 1,
                    "dataSize": 182,
                    "messageType": "InspectorUdpV6ListeningPort"
                },
                {
                    "count": 2,
                    "dataSize": 371,
                    "messageType": "InspectorUdpV4ListeningPort"
                },
                {
                    "count": 18,
                    "dataSize": 8362,
                    "messageType": "InspectorKernelModule"
                },
                {
                    "count": 29,
                    "dataSize": 48788,
                    "messageType": "InspectorConfigurationInfo"
                },
                {
                    "count": 1,
                    "dataSize": 79,
```

```
                    "messageType": "InspectorMonitoringStart"
                },
                {
                    "count": 5,
                    "dataSize": 0,
                    "messageType": "InspectorSplitMsgBegin"
                },
                {
                    "count": 51,
                    "dataSize": 4593,
                    "messageType": "InspectorGroup"
                },
                {
                    "count": 1,
                    "dataSize": 184,
                    "messageType": "InspectorTcpV4ListeningPort"
                },
                {
                    "count": 1159,
                    "dataSize": 3146579,
                    "messageType": "Total"
                },
                {
                    "count": 5,
                    "dataSize": 0,
                    "messageType": "InspectorSplitMsgEnd"
                },
                {
                    "count": 1,
                    "dataSize": 612,
                    "messageType": "InspectorLoadImageInProcess"
                }
            ]
        }
    ]
}
```

Per ulteriori informazioni, consulta Agenti AWS nella Guida per l'utente di Amazon Inspector.

- Per i dettagli sull'API, consulta ListAssessmentRunAgents AWS CLICommand Reference.

`list-assessment-runs`

Il seguente esempio di codice mostra come utilizzare`list-assessment-runs`.

AWS CLI

Come elencare i cicli di valutazione

Il comando `list-assessment-runs` seguente elenca tutti i cicli di valutazione esistenti.

```
aws inspector list-assessment-runs
```

Output:

```
{
    "assessmentRunArns": [
        "arn:aws:inspector:us-west-2:123456789012:target/0-0kFIPusq/
template/0-4r1V2mAw/run/0-MKkpXXPE",
        "arn:aws:inspector:us-west-2:123456789012:target/0-0kFIPusq/
template/0-4r1V2mAw/run/0-v5D6fI3v"
    ]
}
```

Per ulteriori informazioni, consulta Modelli di valutazione Amazon Inspector ed esecuzioni di valutazioni nella Guida per l'utente di Amazon Inspector.

- Per i dettagli sull'API, consulta ListAssessmentRuns AWS CLICommand Reference.

`list-assessment-targets`

Il seguente esempio di codice mostra come utilizzare`list-assessment-targets`.

AWS CLI

Come elencare le destinazioni di valutazione

Il comando `list-assessment-targets` seguente elenca tutte le destinazioni di valutazione esistenti:

```
aws inspector list-assessment-targets
```

Output:

```
{
       "assessmentTargetArns": [
       "arn:aws:inspector:us-west-2:123456789012:target/0-0kFIPusq"
       ]
}
```

Per ulteriori informazioni, consulta Destinazioni delle valutazioni di Amazon Inspector nella Guida per l'utente di Amazon Inspector.

- Per i dettagli sull'API, consulta ListAssessmentTargets AWS CLICommand Reference.

`list-assessment-templates`

Il seguente esempio di codice mostra come utilizzare`list-assessment-templates`.

AWS CLI

Come elencare i modelli di valutazione

Il comando `list-assessment-templates` seguente elenca tutti i modelli di valutazione esistenti:

```
aws inspector list-assessment-templates
```

Output:

```
{
       "assessmentTemplateArns": [
       "arn:aws:inspector:us-west-2:123456789012:target/0-0kFIPusq/
template/0-4r1V2mAw",
       "arn:aws:inspector:us-west-2:123456789012:target/0-0kFIPusq/template/0-
Uza6ihLh"
       ]
}
```

Per ulteriori informazioni, consulta Modelli di valutazione ed esecuzioni delle valutazioni di Amazon Inspector nella Guida per l'utente di Amazon Inspector.

- Per i dettagli sull'API, consulta ListAssessmentTemplates AWS CLICommand Reference.

`list-coverage-statistics`

Il seguente esempio di codice mostra come utilizzare`list-coverage-statistics`.

AWS CLI

Esempio 1: come elencare le statistiche di copertura in base ai gruppi

L'`list-coverage-statistics`esempio seguente elenca le statistiche di copertura dell' AWS ambiente in uso per gruppi.

```
aws inspector2 list-coverage-statistics \
   --group-by RESOURCE_TYPE
```

Output:

```
{
    "countsByGroup": [
        {
            "count": 56,
            "groupKey": "AWS_LAMBDA_FUNCTION"
        },
        {
            "count": 27,
            "groupKey": "AWS_ECR_REPOSITORY"
        },
        {
            "count": 18,
            "groupKey": "AWS_EC2_INSTANCE"
        },
        {
            "count": 3,
            "groupKey": "AWS_ECR_CONTAINER_IMAGE"
        },
        {
            "count": 1,
            "groupKey": "AWS_ACCOUNT"
        }
    ],
    "totalCounts": 105
}
```

Per ulteriori informazioni, consulta la sezione Valutazione della copertura di Amazon Inspector del AWS tuo ambiente nella Amazon Inspector User Guide.

Esempio 2: come elencare le statistiche di copertura in base al tipo di risorse

L'`list-coverage-statistics`esempio seguente elenca le statistiche di copertura del tuo AWS ambiente per tipo di risorsa.

```
aws inspector2 list-coverage-statistics
    --filter-criteria '{"resourceType":
[{"comparison":"EQUALS","value":"AWS_ECR_REPOSITORY"}]}'
    --group-by SCAN_STATUS_REASON
```

Output:

```
{
    "countsByGroup": [
        {
            "count": 27,
            "groupKey": "SUCCESSFUL"
        }
    ],
    "totalCounts": 27
}
```

Per ulteriori informazioni, consulta la sezione Valutazione della copertura di Amazon Inspector del AWS tuo ambiente nella Amazon Inspector User Guide.

Esempio 3: come elencare le statistiche di copertura in base al nome del repository ECR

L'`list-coverage-statistics`esempio seguente elenca le statistiche di copertura del tuo AWS ambiente in base al nome del repository ECR.

```
aws inspector2 list-coverage-statistics
   --filter-criteria '{"ecrRepositoryName":
[{"comparison":"EQUALS","value":"debian"}]}'
   --group-by SCAN_STATUS_REASON
```

Output:

```
{
```

```
    "countsByGroup": [
        {
            "count": 3,
            "groupKey": "SUCCESSFUL"
        }
    ],
    "totalCounts": 3
}
```

Per ulteriori informazioni, consulta la sezione Valutazione della copertura di Amazon Inspector del AWS tuo ambiente nella Amazon Inspector User Guide.

- Per i dettagli sull'API, consulta Command ListCoverageStatisticsReference AWS CLI .

`list-coverage`

Il seguente esempio di codice mostra come utilizzare`list-coverage`.

AWS CLI

Esempio 1: come elencare i dettagli sulla copertura dell'ambiente

L'esempio `list-coverage` seguente elenca i dettagli relativi alla copertura dell'ambiente.

```
aws inspector2 list-coverage
```

Output:

```
{
    "coveredResources": [
        {
            "accountId": "123456789012",
            "lastScannedAt": "2024-05-20T16:23:20-07:00",
            "resourceId": "i-EXAMPLE55555555555",
            "resourceMetadata": {
                "ec2": {
                    "amiId": "ami-EXAMPLE6666666666",
                    "platform": "LINUX"
                }
            },
            "resourceType": "AWS_EC2_INSTANCE",
            "scanStatus": {
                "reason": "SUCCESSFUL",
```

```
                "statusCode": "ACTIVE"
            },
            "scanType": "PACKAGE"
        }
    ]
}
```

Esempio 2: come elencare i dettagli sulla copertura in relazione al tipo di risorsa della funzione Lambda

L'esempio `list-coverage` seguente elenca i dettagli relativi al tipo di risorsa della funzione Lamda.

```
aws inspector2 list-coverage
    --filter-criteria '{"resourceType":
[{"comparison":"EQUALS","value":"AWS_LAMBDA_FUNCTION"}]}'
```

Output:

```
{
    "coveredResources": [
        {
            "accountId": "123456789012",
            "resourceId": "arn:aws:lambda:us-west-2:123456789012:function:Eval-
container-scan-results:$LATEST",
            "resourceMetadata": {
                "lambdaFunction": {
                    "functionName": "Eval-container-scan-results",
                    "functionTags": {},
                    "layers": [],
                    "runtime": "PYTHON_3_7"
                }
            },
            "resourceType": "AWS_LAMBDA_FUNCTION",
            "scanStatus": {
                "reason": "SUCCESSFUL",
                "statusCode": "ACTIVE"
            },
            "scanType": "CODE"
        }
    ]
}
```

- Per i dettagli sull'API, consulta ListCoverage AWS CLICommand Reference.

`list-delegated-admin-accounts`

Il seguente esempio di codice mostra come utilizzare`list-delegated-admin-accounts`.

AWS CLI

Come elencare le informazioni sull'account di amministratore delegato dell'organizzazione

L'esempio `list-delegated-admin-accounts` seguente elenca le informazioni sull'account di amministratore delegato dell'organizzazione.

```
aws inspector2 list-delegated-admin-accounts
```

Output:

```
{
    "delegatedAdminAccounts": [
        {
            "accountId": "123456789012",
            "status": "ENABLED"
        }
    ]
}
```

Per ulteriori informazioni, consulta Designazione di un amministratore delegato per Amazon Inspector nella Guida per l'utente di Amazon Inspector.

- Per i dettagli sull'API, consulta ListDelegatedAdminAccounts AWS CLICommand Reference.

`list-event-subscriptions`

Il seguente esempio di codice mostra come utilizzare`list-event-subscriptions`.

AWS CLI

Come elencare le sottoscrizioni agli eventi

Il comando `list-event-subscriptions` seguente elenca tutte le sottoscrizioni agli eventi per il modello di valutazione con l'ARN `arn:aws:inspector:us-west-2:123456789012:target/0-nvgVhaxX/template/0-7sbz2Kz0`:

```
aws inspector list-event-subscriptions --resource-arn arn:aws:inspector:us-
west-2:123456789012:target/0-nvgVhaxX/template/0-7sbz2Kz0
```

Output:

```
{
      "subscriptions": [
        {
              "eventSubscriptions": [
                {
                      "event": "ASSESSMENT_RUN_COMPLETED",
                      "subscribedAt": 1459455440.867
                }
              ],
              "resourceArn": "arn:aws:inspector:us-west-2:123456789012:target/0-
nvgVhaxX/template/0-7sbz2Kz0",
              "topicArn": "arn:aws:sns:us-west-2:123456789012:exampletopic"
        }
      ]
}
```

Per ulteriori informazioni, consulta Modelli di valutazione ed esecuzioni delle valutazioni di Amazon Inspector nella Guida per l'utente di Amazon Inspector.

- Per i dettagli sull'API, consulta ListEventSubscriptions AWS CLICommand Reference.

`list-filters`

Il seguente esempio di codice mostra come utilizzare`list-filters`.

AWS CLI

Come elencare i filtri associati all'account utilizzato per attivare Amazon Inspector

Gli esempi `list-filters` seguenti elencano i filtri associati all'account utilizzato per attivare Amazon Inspector.

```
aws inspector2 list-filters
```

Output:

```
{
```

```
    "filters": [
        {
            "action": "SUPPRESS",
            "arn": "arn:aws:inspector2:us-west-2:123456789012:owner/o-EXAMPLE222/
filter/EXAMPLE444444444",
            "createdAt": "2024-05-15T21:11:08.602000+00:00",
            "criteria": {
                "resourceType": [
                    {
                        "comparison": "EQUALS",
                        "value": "AWS_EC2_INSTANCE"
                    },
                ]
            },
            "description": "This suppression rule omits EC2 instance type findings",
            "name": "ExampleSuppressionRuleEC2",
            "ownerId": "o-EXAMPLE222",
            "tags": {},
            "updatedAt": "2024-05-15T21:11:08.602000+00:00"
        },
        {
            "action": "SUPPRESS",
            "arn": "arn:aws:inspector2:us-east-1:813737243517:owner/o-EXAMPLE222/
filter/EXAMPLE444444444",
            "createdAt": "2024-05-15T21:28:27.054000+00:00",
            "criteria": {
                "resourceType": [
                    {
                        "comparison": "EQUALS",
                        "value": "AWS_ECR_INSTANCE"
                    }
                ]
            },
            "description": "This suppression rule omits ECR instance type findings",
            "name": "ExampleSuppressionRuleECR",
            "ownerId": "o-EXAMPLE222",
            "tags": {},
            "updatedAt": "2024-05-15T21:28:27.054000+00:00"
        }
    ]
}
```

Per ulteriori informazioni, consulta Filtro degli esiti di Amazon Inspector nella Guida per l'utente di Amazon Inspector.

- Per i dettagli sull'API, consulta ListFilters AWS CLICommand Reference.

`list-findings`

Il seguente esempio di codice mostra come utilizzare`list-findings`.

AWS CLI

Come elencare gli esiti

Il comando `list-findings` seguente elenca tutti gli esiti generati:

```
aws inspector list-findings
```

Output:

```
{
        "findingArns": [
        "arn:aws:inspector:us-west-2:123456789012:target/0-0kFIPusq/
template/0-4r1V2mAw/run/0-MKkpXXPE/finding/0-HwPnsDm4",
        "arn:aws:inspector:us-west-2:123456789012:target/0-0kFIPusq/
template/0-4r1V2mAw/run/0-v5D6fI3v/finding/0-tyvmqBLy"
      ]
}
```

Per ulteriori informazioni consulta Esiti di Amazon Inspector nella Guida per l'utente di Amazon Inspector.

- Per i dettagli sull'API, consulta ListFindings AWS CLICommand Reference.

`list-members`

Il seguente esempio di codice mostra come utilizzare`list-members`.

AWS CLI

Esempio 1: come elencare tutti gli account membro associati all'amministratore delegato di Amazon Inspector per l'organizzazione

aws inspector2 list-members --only-associated

Output:

```
{
        {
             "members": [
        {
             "accountId": "123456789012",
             "delegatedAdminAccountId": "123456789012",
             "relationshipStatus": "ENABLED",
             "updatedAt": "2023-09-11T09:57:20.520000-07:00"
        },
        {
             "accountId": "123456789012",
             "delegatedAdminAccountId": "123456789012",
             "relationshipStatus": "ENABLED",
             "updatedAt": "2024-08-12T10:13:01.472000-07:00"
        },
        {
             "accountId": "625032911453",
             "delegatedAdminAccountId": "123456789012",
             "relationshipStatus": "ENABLED",
             "updatedAt": "2023-09-11T09:57:20.438000-07:00"
        },
        {
            "accountId": "715411239211",
            "delegatedAdminAccountId": "123456789012",
            "relationshipStatus": "ENABLED",
            "updatedAt": "2024-04-24T09:14:57.471000-07:00"
        }
    ]
}
```

Per ulteriori informazioni, consulta la sezione Gestione di più account in Amazon Inspector with AWS Organizations nella Amazon Inspector User Guide.

Esempio 2: come elencare tutti gli account membro associati e dissociati dall'amministratore delegato di Amazon Inspector per l'organizzazione

membri della lista aws inspector2 - no-only-associated

Output:

```
{
```

```
        {
            "members": [
        {
            "accountId": "123456789012",
            "delegatedAdminAccountId": "123456789012",
            "relationshipStatus": "REMOVED",
            "updatedAt": "2024-05-15T11:34:53.326000-07:00"
        },
        {
            "accountId": "123456789012",
            "delegatedAdminAccountId": "123456789012",
            "relationshipStatus": "ENABLED",
            "updatedAt": "2023-09-11T09:57:20.520000-07:00"
        },
        {
            "accountId": "123456789012",
            "delegatedAdminAccountId": "123456789012",
            "relationshipStatus": "ENABLED",
            "updatedAt": "2024-08-12T10:13:01.472000-07:00"
        },
        {
            "accountId": "123456789012",
            "delegatedAdminAccountId": "123456789012",
            "relationshipStatus": "ENABLED",
            "updatedAt": "2023-09-11T09:57:20.438000-07:00"
        },
        {
            "accountId": "123456789012",
            "delegatedAdminAccountId": "123456789012",
            "relationshipStatus": "ENABLED",
            "updatedAt": "2024-04-24T09:14:57.471000-07:00"
        }
    ]
}
```

Per ulteriori informazioni, consulta la sezione Gestione di più account in Amazon Inspector with AWS Organizations nella Amazon Inspector User Guide.

- Per i dettagli sull'API, consulta AWS CLI Command ListMembersReference.

`list-rules-packages`

Il seguente esempio di codice mostra come utilizzare`list-rules-packages`.

AWS CLI

Come elencare i pacchetti di regole

Il comando `list-rules-packages` seguente elenca tutti i pacchetti di regole di Amazon Inspector disponibili:

```
aws inspector list-rules-packages
```

Output:

```
{
       "rulesPackageArns": [
         "arn:aws:inspector:us-west-2:758058086616:rulespackage/0-9hgA516p",
         "arn:aws:inspector:us-west-2:758058086616:rulespackage/0-H5hpSawc",
         "arn:aws:inspector:us-west-2:758058086616:rulespackage/0-JJOtZiqQ",
         "arn:aws:inspector:us-west-2:758058086616:rulespackage/0-vg5GGHSD"
       ]
 }
```

Per ulteriori informazioni, consulta Pacchetti di regole e regole di Amazon Inspector nella Guida per l'utente di Amazon Inspector.

- Per i dettagli sull'API, consulta ListRulesPackages AWS CLICommand Reference.

`list-tags-for-resource`

Il seguente esempio di codice mostra come utilizzare`list-tags-for-resource`.

AWS CLI

Come elencare i tag per le risorse

Il comando `list-tags-for-resource` seguente elenca tutti i tag associati al modello di valutazione con l'ARN `arn:aws:inspector:us-west-2:123456789012:target/0-0kFIPusq/template/0-gcwFliYu`:

```
aws inspector list-tags-for-resource --resource-arn arn:aws:inspector:us-
west-2:123456789012:target/0-0kFIPusq/template/0-gcwFliYu
```

Output:

```
{
      "tags": [
        {
              "key": "Name",
              "value": "Example"
        }
      ]
}
```

Per ulteriori informazioni, consulta Modelli di valutazione ed esecuzioni delle valutazioni di Amazon Inspector nella Guida per l'utente di Amazon Inspector.

- Per i dettagli sull'API, consulta ListTagsForResource AWS CLICommand Reference.

`list-usage-totals`

Il seguente esempio di codice mostra come utilizzare`list-usage-totals`.

AWS CLI

Come elencare i totali di utilizzo negli ultimi 30 giorni

Gli esempi `list-usage-totals` seguenti elencano i totali di utilizzo negli ultimi 30 giorni.

```
aws inspector2 list-usage-totals
```

Output:

```
{
    "totals": [
        {
            "accountId": "123456789012",
            "usage": [
                {
                    "currency": "USD",
                    "estimatedMonthlyCost": 4.6022044647,
                    "total": 1893.4784083333334,
                    "type": "EC2_AGENTLESS_INSTANCE_HOURS"
                },
                {
                    "currency": "USD",
```

```
                    "estimatedMonthlyCost": 18.892449279,
                    "total": 10882.050784722222,
                    "type": "EC2_INSTANCE_HOURS"
                },
                {
                    "currency": "USD",
                    "estimatedMonthlyCost": 5.4525363736,
                    "total": 6543.043648333333,
                    "type": "LAMBDA_FUNCTION_CODE_HOURS"
                },
                {
                    "currency": "USD",
                    "estimatedMonthlyCost": 3.9064080309,
                    "total": 9375.379274166668,
                    "type": "LAMBDA_FUNCTION_HOURS"
                },
                {
                    "currency": "USD",
                    "estimatedMonthlyCost": 0.06,
                    "total": 6.0,
                    "type": "ECR_RESCAN"
                },
                {
                    "currency": "USD",
                    "estimatedMonthlyCost": 0.09,
                    "total": 1.0,
                    "type": "ECR_INITIAL_SCAN"
                }
            ]
        }
    ]
}
```

Per ulteriori informazioni, consulta Monitoraggio di utilizzo e costi in Amazon Inspector nella Guida per l'utente di Amazon Inspector.

- Per i dettagli sull'API, consulta ListUsageTotals AWS CLICommand Reference.

`preview-agents`

Il seguente esempio di codice mostra come utilizzare`preview-agents`.

AWS CLI

Come visualizzare in anteprima gli agenti

Il `preview-agents` comando seguente visualizza in anteprima gli agenti installati sulle EC2 istanze che fanno parte dell'obiettivo di valutazione con l'ARN di: `arn:aws:inspector:us-west-2:123456789012:target/0-0kFIPusq`

```
aws inspector preview-agents --preview-agents-arn arn:aws:inspector:us-west-2:123456789012:target/0-0kFIPusq
```

Output:

```
{
      "agentPreviews": [
        {
              "agentId": "i-49113b93"
        }
      ]
}
```

Per ulteriori informazioni, consulta Destinazioni delle valutazioni di Amazon Inspector nella Guida per l'utente di Amazon Inspector.

- Per i dettagli sull'API, vedere PreviewAgentsin AWS CLI Command Reference.

`register-cross-account-access-role`

Il seguente esempio di codice mostra come utilizzare`register-cross-account-access-role`.

AWS CLI

Come registrare il ruolo di accesso multi-account

Il `register-cross-account-access-role` comando seguente registra il ruolo IAM con l'ARN `arn:aws:iam::123456789012:role/inspector` utilizzato da Amazon Inspector per EC2 elencare le istanze all'inizio dell'esecuzione della valutazione o quando si chiama il comando preview-agents:

```
aws inspector register-cross-account-access-role --role-arn arn:aws:iam::123456789012:role/inspector
```

Per ulteriori informazioni, consulta Impostazione di Amazon Inspector nella Guida per l'utente di Amazon Inspector.

- Per i dettagli sull'API, consulta Command Reference. RegisterCrossAccountAccessRoleAWS CLI

`remove-attributes-from-findings`

Il seguente esempio di codice mostra come utilizzare`remove-attributes-from-findings`.

AWS CLI

Come rimuovere gli attributi dagli esiti

Il comando `remove-attributes-from-finding` seguente rimuove l'attributo con la chiave `Example` e il valore `example` dall'esito con l'ARN `arn:aws:inspector:us-west-2:123456789012:target/0-0kFIPusq/template/0-8l1VIE0D/run/0-Z02cjjug/finding/0-T8yM9mEU`:

```
aws inspector remove-attributes-from-findings --finding-arns arn:aws:inspector:us-west-2:123456789012:target/0-0kFIPusq/template/0-8l1VIE0D/run/0-Z02cjjug/finding/0-T8yM9mEU --attribute-keys key=Example,value=example
```

Output:

```
{
      "failedItems": {}
}
```

Per ulteriori informazioni consulta Esiti di Amazon Inspector nella Guida per l'utente di Amazon Inspector.

- Per i dettagli sull'API, consulta RemoveAttributesFromFindings AWS CLICommand Reference.

`set-tags-for-resource`

Il seguente esempio di codice mostra come utilizzare`set-tags-for-resource`.

AWS CLI

Come impostare i tag per una risorsa

Il comando `set-tags-for-resource` seguente imposta il tag con la chiave `Example` e il valore `example` sul modello di valutazione con l'ARN `arn:aws:inspector:us-west-2:123456789012:target/0-nvgVhaxX/template/0-7sbz2Kz0`:

```
aws inspector set-tags-for-resource --resource-arn arn:aws:inspector:us-
west-2:123456789012:target/0-nvgVhaxX/template/0-7sbz2Kz0 --
tags key=Example,value=example
```

Per ulteriori informazioni, consulta Modelli di valutazione ed esecuzioni delle valutazioni di Amazon Inspector nella Guida per l'utente di Amazon Inspector.

- Per i dettagli sull'API, consulta SetTagsForResource AWS CLICommand Reference.

`start-assessment-run`

Il seguente esempio di codice mostra come utilizzare`start-assessment-run`.

AWS CLI

Come avviare un ciclo di valutazione

Il comando `start-assessment-run` seguente avvia il ciclo di valutazione denominato `examplerun` utilizzando il modello di valutazione con l'ARN `arn:aws:inspector:us-west-2:123456789012:target/0-nvgVhaxX/template/0-it5r2S4T`:

```
aws inspector start-assessment-run --assessment-run-name examplerun --assessment-
template-arn arn:aws:inspector:us-west-2:123456789012:target/0-nvgVhaxX/template/0-
it5r2S4T
```

Output:

```
{
      "assessmentRunArn": "arn:aws:inspector:us-west-2:123456789012:target/0-
nvgVhaxX/template/0-it5r2S4T/run/0-jOoroxyY"
}
```

Per ulteriori informazioni, consulta Modelli di valutazione ed esecuzioni delle valutazioni di Amazon Inspector nella Guida per l'utente di Amazon Inspector.

- Per i dettagli sull'API, consulta StartAssessmentRun AWS CLICommand Reference.

`stop-assessment-run`

Il seguente esempio di codice mostra come utilizzare`stop-assessment-run`.

AWS CLI

Come arrestare un ciclo di valutazione

Il comando `stop-assessment-run` seguente arresta il ciclo di valutazione con l'ARN `arn:aws:inspector:us-west-2:123456789012:target/0-nvgVhaxX/template/0-it5r2S4T/run/0-jOoroxyY`:

```
aws inspector stop-assessment-run --assessment-run-arn arn:aws:inspector:us-west-2:123456789012:target/0-nvgVhaxX/template/0-it5r2S4T/run/0-jOoroxyY
```

Per ulteriori informazioni, consulta Modelli di valutazione ed esecuzioni delle valutazioni di Amazon Inspector nella Guida per l'utente di Amazon Inspector.

- Per i dettagli sull'API, consulta StopAssessmentRun AWS CLICommand Reference.

`subscribe-to-event`

Il seguente esempio di codice mostra come utilizzare`subscribe-to-event`.

AWS CLI

Come iscriversi a un evento

L'esempio seguente abilita il processo di invio delle notifiche di Amazon SNS sull'evento `ASSESSMENT_RUN_COMPLETED` in relazione all'argomento con l'ARN di `arn:aws:sns:us-west-2:123456789012:exampletopic`.

```
aws inspector subscribe-to-event \
    --event ASSESSMENT_RUN_COMPLETED \
    --resource-arn arn:aws:inspector:us-west-2:123456789012:target/0-nvgVhaxX/template/0-7sbz2Kz0 \
    --topic-arn arn:aws:sns:us-west-2:123456789012:exampletopic
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Modelli di valutazione ed esecuzioni delle valutazioni di Amazon Inspector nella Guida per l'utente di Amazon Inspector.

- Per i dettagli sull'API, consulta SubscribeToEvent AWS CLICommand Reference.

`unsubscribe-from-event`

Il seguente esempio di codice mostra come utilizzare`unsubscribe-from-event`.

AWS CLI

Come annullare l'iscrizione a un evento

Il comando `unsubscribe-from-event` seguente disabilita il processo di invio delle notifiche di Amazon SNS sull'evento `ASSESSMENT_RUN_COMPLETED` in relazione all'argomento con l'ARN `arn:aws:sns:us-west-2:123456789012:exampletopic`:

```
aws inspector unsubscribe-from-event --event ASSESSMENT_RUN_COMPLETED --resource-
arn arn:aws:inspector:us-west-2:123456789012:target/0-nvgVhaxX/template/0-7sbz2Kz0
 --topic arn:aws:sns:us-west-2:123456789012:exampletopic
```

Per ulteriori informazioni, consulta Modelli di valutazione ed esecuzioni delle valutazioni di Amazon Inspector nella Guida per l'utente di Amazon Inspector.

- Per i dettagli sull'API, consulta UnsubscribeFromEvent AWS CLICommand Reference.

`update-assessment-target`

Il seguente esempio di codice mostra come utilizzare`update-assessment-target`.

AWS CLI

Come aggiornare una destinazione di valutazione

Il comando `update-assessment-target` seguente aggiorna la destinazione di valutazione con l'ARN `arn:aws:inspector:us-west-2:123456789012:target/0-nvgVhaxX`, il nome `Example` e il gruppo di risorse con l'ARN `arn:aws:inspector:us-west-2:123456789012:resourcegroup/0-yNbgL5Pt`:

```
aws inspector update-assessment-target --assessment-target-arn arn:aws:inspector:us-
west-2:123456789012:target/0-nvgVhaxX --assessment-target-name Example --resource-
group-arn arn:aws:inspector:us-west-2:123456789012:resourcegroup/0-yNbgL5Pt
```

Per ulteriori informazioni, consulta Destinazioni delle valutazioni di Amazon Inspector nella Guida per l'utente di Amazon Inspector.

- Per i dettagli sull'API, consulta UpdateAssessmentTarget AWS CLICommand Reference.

`update-filter`

Il seguente esempio di codice mostra come utilizzare`update-filter`.

AWS CLI

Come aggiornare un filtro

L'esempio `update-filter` seguente aggiorna un filtro in modo da omettere gli esiti Lambda anziché gli esiti dell'istanza ECR.

```
aws inspector2 update-filter \
   --filter-arn "arn:aws:inspector2:us-west-2:123456789012:owner/o-EXAMPLE222/
filter/EXAMPLE444444444" \
   --name "ExampleSuppressionRuleLambda" \
   --description "This suppression rule omits Lambda instance findings" \
   --reason "Updating filter to omit Lambda instance findings instead of ECR
 instance findings"
```

Output:

```
{
    "filters": [
        {
            "action": "SUPPRESS",
            "arn": "arn:aws:inspector2:us-west-2:123456789012:owner/o-EXAMPLE222/
filter/EXAMPLE444444444",
            "createdAt": "2024-05-15T21:28:27.054000+00:00",
            "criteria": {
                "resourceType": [
                    {
                        "comparison": "EQUALS",
                        "value": "AWS_ECR_INSTANCE"
                    }
                ]
            },
            "description": "This suppression rule omits Lambda instance findings",
            "name": "ExampleSuppressionRuleLambda",
```

```
            "ownerId": "o-EXAMPLE222",
            "reason": "Updating filter to omit Lambda instance findings instead of
 ECR instance findings",
            "tags": {},
            "updatedAt": "2024-05-15T22:23:13.665000+00:00"
        }
    ]
}
```

Per ulteriori informazioni, consulta Gestione degli esiti in Amazon Inspector nella Guida per l'utente di Amazon Inspector.

- Per i dettagli sull'API, consulta UpdateFilter AWS CLICommand Reference.

AWS IoT esempi utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with AWS IoT.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`accept-certificate-transfer`

Il seguente esempio di codice mostra come utilizzare`accept-certificate-transfer`.

AWS CLI

Accettare il certificato di un dispositivo trasferito da un altro AWS account

L'`accept-certificate-transfer`esempio seguente accetta il certificato di un dispositivo trasferito da un altro AWS account. Il certificato viene identificato mediante il rispettivo ID.

```
aws iot accept-certificate-transfer \
    --certificate-
id 488b6a7f2acdeb00a77384e63c4e40b18bEXAMPLEe57b7272ba44c45e3448142
```

Il comando non produce output.

Per ulteriori informazioni, consulta Trasferire un certificato a un altro account nella Guida per gli sviluppatori di AWS IoT Core.

- Per i dettagli sull'API, vedere AcceptCertificateTransferin AWS CLI Command Reference.

`add-thing-to-billing-group`

Il seguente esempio di codice mostra come utilizzare`add-thing-to-billing-group`.

AWS CLI

Esempio 1: come aggiungere un oggetto in base al nome a un gruppo di fatturazione

L'esempio `add-thing-to-billing-group` seguente aggiunge l'oggetto denominato `MyLightBulb` al gruppo di fatturazione denominato `GroupOne`.

```
aws iot add-thing-to-billing-group \
    --billing-group-name GroupOne \
    --thing-name MyLightBulb
```

Questo comando non produce alcun output.

Esempio 2: come aggiungere un oggetto in base all'ARN a un gruppo di fatturazione

L'esempio `add-thing-to-billing-group` seguente aggiunge un oggetto con un ARN specificato a un gruppo di fatturazione con l'ARN specificato. Specificare un ARN è utile se si lavora con AWS più regioni o account. Può aiutarti a verificare che tu stia aggiungendo risorse nella Regione e nell'account corretti.

```
aws iot add-thing-to-thing-group \
    --billing-group-arn "arn:aws:iot:us-west-2:123456789012:billinggroup/GroupOne" \
    --thing-arn "arn:aws:iot:us-west-2:123456789012:thing/MyOtherLightBulb"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Gruppi di fatturazione nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta AWS CLI Command AddThingToBillingGroupReference.

`add-thing-to-thing-group`

Il seguente esempio di codice mostra come utilizzare`add-thing-to-thing-group`.

AWS CLI

Come aggiungere un oggetto a un gruppo

L'esempio `add-thing-to-thing-group` seguente aggiunge l'oggetto specificato al gruppo di oggetti specificato.

```
aws iot add-thing-to-thing-group \
    --thing-name MyLightBulb \
    --thing-group-name LightBulbs
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Gruppi di oggetti nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta AddThingToThingGroup AWS CLICommand Reference.

`associate-targets-with-job`

Il seguente esempio di codice mostra come utilizzare`associate-targets-with-job`.

AWS CLI

Come associare un gruppo di oggetti a un processo continuo

L'esempio `associate-targets-with-job` seguente associa il gruppo di oggetti specificato al processo continuo specificato.

```
aws iot associate-targets-with-job \
    --targets "arn:aws:iot:us-west-2:123456789012:thinggroup/LightBulbs" \
    --job-id "example-job-04"
```

Output:

```
{
    "jobArn": "arn:aws:iot:us-west-2:123456789012:job/example-job-04",
    "jobId": "example-job-04",
    "description": "example continuous job"
}
```

Per ulteriori informazioni, consulta Creazione e gestione di processi (CLI) nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta AssociateTargetsWithJob AWS CLICommand Reference.

`attach-policy`

Il seguente esempio di codice mostra come utilizzare`attach-policy`.

AWS CLI

Esempio 1: come collegare una policy a un gruppo di oggetti

L'esempio `attach-policy` seguente associa la policy specificata a un gruppo di oggetti identificato dal relativo ARN.

```
aws iot attach-policy \
    --target "arn:aws:iot:us-west-2:123456789012:thinggroup/LightBulbs" \
    --policy-name "UpdateDeviceCertPolicy"
```

Il comando non produce output.

Per ulteriori informazioni, consulta Gruppi di oggetti nella Guida per gli sviluppatori di AWS IoT.

Esempio 2: come collegare una policy a un certificato

L'esempio `attach-policy` seguente associa la policy `UpdateDeviceCertPolicy` al principale specificato da un certificato.

```
aws iot attach-policy \
    --policy-name UpdateDeviceCertPolicy \
    --target "arn:aws:iot:us-
west-2:123456789012:cert/4f0ba725787aa94d67d2fca420eca022242532e8b3c58e7465c7778b443fd65e"
```

Il comando non produce output.

Per ulteriori informazioni, consulta Allegare una policy AWS IoT a un certificato di dispositivo nella AWS IoT Developers Guide.

- Per i dettagli sull'API, consulta AttachPolicy AWS CLICommand Reference.

`attach-security-profile`

Il seguente esempio di codice mostra come utilizzare`attach-security-profile`.

AWS CLI

Come associare un profilo di sicurezza a tutti i dispositivi non registrati

L'`attach-security-profile`esempio seguente associa il profilo di sicurezza AWS IoT Device Defender `Testprofile` denominato a tutti i dispositivi non registrati `us-west-2` nella regione di questo account. AWS

```
aws iot attach-security-profile \
    --security-profile-name Testprofile \
    --security-profile-target-arn "arn:aws:iot:us-west-2:123456789012:all/
unregistered-things"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Comandi di rilevamento nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta Command AttachSecurityProfileReference AWS CLI .

`attach-thing-principal`

Il seguente esempio di codice mostra come utilizzare`attach-thing-principal`.

AWS CLI

Come associare un certificato a un oggetto

L'`attach-thing-principal`esempio seguente allega un certificato all' MyTemperatureSensor oggetto. Il certificato è identificato da un ARN. Puoi trovare l'ARN per un certificato nella console AWS IoT.

```
aws iot attach-thing-principal \
```

```
    --thing-name MyTemperatureSensor \
    --principal arn:aws:iot:us-
west-2:123456789012:cert/2e1eb273792174ec2b9bf4e9b37e6c6c692345499506002a351597670552 78e8
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Come gestire gli oggetti con il registro nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta AttachThingPrincipal AWS CLICommand Reference.

`cancel-audit-mitigation-actions-task`

Il seguente esempio di codice mostra come utilizzare`cancel-audit-mitigation-actions-task`.

AWS CLI

Come annullare un'attività relativa alle azioni di mitigazione degli audit

L'esempio `cancel-audit-mitigations-action-task` seguente annulla l'applicazione delle azioni di mitigazione per l'attività specificata. Non è possibile annullare le attività già completate.

```
aws iot cancel-audit-mitigation-actions-task
    --task-id "myActionsTaskId"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta CancelAuditMitigationActionsTask (Mitigation Action Commands) nella AWS IoT Developer Guide.

- Per i dettagli sull'API, consulta CancelAuditMitigationActionsTask AWS CLICommand Reference.

`cancel-audit-task`

Il seguente esempio di codice mostra come utilizzare`cancel-audit-task`.

AWS CLI

Come annullare un'attività di audit

L'esempio `cancel-audit-task` seguente annulla un'attività di audit con l'ID di attività specificato. Non è possibile annullare un'attività completata.

```
aws iot cancel-audit-task \
    --task-id a3aea009955e501a31b764abe1bebd3d
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Comandi di auditing nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta CancelAuditTask AWS CLICommand Reference.

`cancel-certificate-transfer`

Il seguente esempio di codice mostra come utilizzare`cancel-certificate-transfer`.

AWS CLI

Per annullare il trasferimento di un certificato su un altro AWS account

L'esempio `cancel-certificate-transfer` seguente annulla il trasferimento del certificato specificato. Il certificato viene identificato mediante il rispettivo ID. Puoi trovare l'ID di un certificato nella console AWS IoT.

```
aws iot cancel-certificate-transfer \
    --certificate-
id f0f33678c7c9a046e5cc87b2b1a58dfa0beec26db78addd5e605d630e05c7fc8
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Trasferire un certificato a un altro account nella Guida per gli sviluppatori di AWS IoT Core.

- Per i dettagli sull'API, consulta CancelCertificateTransfer AWS CLICommand Reference.

`cancel-job-execution`

Il seguente esempio di codice mostra come utilizzare`cancel-job-execution`.

AWS CLI

Come annullare l'esecuzione di un processo su un dispositivo

L'esempio `cancel-job-execution` seguente annulla l'esecuzione del processo specificato su un dispositivo. Se il processo non è nello stato `QUEUED`, è necessario aggiungere il parametro `--force`.

```
aws iot cancel-job-execution \
    --job-id "example-job-03" \
    --thing-name "MyRPi"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Creazione e gestione di processi (CLI) nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta CancelJobExecution AWS CLICommand Reference.

`cancel-job`

Il seguente esempio di codice mostra come utilizzare`cancel-job`.

AWS CLI

Come annullare un processo

L'esempio `cancel-job` seguente annulla il processo specificato.

```
aws iot cancel-job \
    --job-job "example-job-03"
```

Output:

```
{
    "jobArn": "arn:aws:iot:us-west-2:123456789012:job/example-job-03",
    "jobId": "example-job-03",
    "description": "example job test"
}
```

Per ulteriori informazioni, consulta Creazione e gestione di processi (CLI) nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta CancelJob AWS CLICommand Reference.

`clear-default-authorizer`

Il seguente esempio di codice mostra come utilizzare`clear-default-authorizer`.

AWS CLI

Come annullare il sistema di autorizzazione predefinito

L'esempio `clear-default-authorizer` seguente annulla il sistema di autorizzazione personalizzato predefinito attualmente configurato. Dopo aver eseguito questo comando, non è presente alcun sistema di autorizzazione predefinito. In caso di utilizzo di un sistema di autorizzazione personalizzato, è necessario specificarlo in base al nome nelle intestazioni della richiesta HTTP.

```
aws iot clear-default-authorizer
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta la ClearDefaultAuthorizersezione AWS IoT API Reference.

- Per i dettagli sull'API, consulta ClearDefaultAuthorizer AWS CLICommand Reference.

`confirm-topic-rule-destination`

Il seguente esempio di codice mostra come utilizzare`confirm-topic-rule-destination`.

AWS CLI

Come confermare la destinazione di una regola dell'argomento

L'esempio `confirm-topic-rule-destination` seguente conferma la destinazione di una regola dell'argomento con un token di conferma ricevuto su un endpoint HTTP.

```
aws iot confirm-topic-rule-destination \
    --confirmation-token "AYADeIcmtq-
ZkxfpiWIQqHWM5ucAXwABABVhd3MtY3J5cHRvLXB1YmxpYy1rZXkAREFxY1E0UmlGeDg0V21BZWZ1VjZtZWFRVUJJUkt
aywpPqg8YEsa1lD4B40aJ2s1wEHKMybiF1RoOZzYisI0IvslzQY5UmCkqq3tV-3f7-
nKfosgIAAAADAAAEAAAAAAAAAAAAAAAAAAi9RMgy-
V19V9m6Iw2xfbw_____wAAAAEAAAAAAAAAAAAAAEAAAB1hw4SokgUcxiJ3gTO6n50NLJVpzyQR1UmPIj5sShqXEQGcO
iufgrzTePl8RZYOWr0O6Aj9DiVzJZx-1iD6Pu-
G6PUw1kaO7Knzs2B4AD0qfrHUF4pYRTvyUgBnMGUCMQC8ZRmhKqntd_c6Kgrow3bMUDBvNqo2qZr8Z8Jm2rzgseROlAn
PIetJ803Z4IlIlF8xXlcdPGP-PV1dOXFemyL8g"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Conferma della destinazione di una regola dell'argomento nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta ConfirmTopicRuleDestination AWS CLICommand Reference.

`create-audit-suppression`

Il seguente esempio di codice mostra come utilizzare`create-audit-suppression`.

AWS CLI

Come creare una soppressione degli esiti di audit

L'`create-audit-suppression`esempio seguente crea una soppressione dei risultati di controllo per una politica denominata "virtualMachinePolicy" che è stata contrassegnata come eccessivamente permissiva.

```
aws iot create-audit-suppression \
    --check-name IOT_POLICY_OVERLY_PERMISSIVE_CHECK \
    --resource-identifier
 policyVersionIdentifier={"policyName"="virtualMachinePolicy","policyVersionId"="1"}
 \
    --no-suppress-indefinitely \
    --expiration-date 2020-10-20
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Soppressioni degli esiti degli audit nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta Command Reference. CreateAuditSuppressionAWS CLI

`create-authorizer`

Il seguente esempio di codice mostra come utilizzare`create-authorizer`.

AWS CLI

Come creare un sistema di autorizzazione personalizzato

L'esempio `create-authorizer` seguente crea un sistema di autorizzazione personalizzato che utilizza la funzione Lambda specificata come parte di un servizio di autenticazione personalizzato.

```
   aws iot create-authorizer \
       --authorizer-name "CustomAuthorizer" \
       --authorizer-function-arn "arn:aws:lambda:us-
west-2:123456789012:function:CustomAuthorizerFunction" \
       --token-key-name "MyAuthToken" \
       --status ACTIVE \
       --token-signing-public-keys FIRST_KEY="-----BEGIN PUBLIC KEY-----
MIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEA1uJOB4lQPgG/lM6ZfIwo
Z+7ENxAio9q6QD4FFqjGZsvjtYwjoe1RKK0U8Eq9xb5O3kRSmyIwTzwzm/f4Gf0Y
ZUloJ+t3PUUwHrmbYTAgTrCUgRFygjfgVwGCPs5ZAX4Eyqt5cr+AIHIiUDbxSa7p
zwOBKPeic0asNJpqT8PkBbRaKyleJh5oo81NDHHmVtbBm5A5YiJjqYXLaVAowKzZ
+GqsNvAQ9Jy1wI2VrEa1OfL8flDB/BJLm7zjpfPOHDJQgID0XnZwAlNnZcOhCwIx
50g2LW2Oy9R/dmqtDmJiVP97Z4GykxPvwlYHrUXY0iW1R3AR/Ac1NhCTGZMwVDB1
lQIDAQAB
-----END PUBLIC KEY-----"
```

Output:

```
{
    "authorizerName": "CustomAuthorizer",
    "authorizerArn": "arn:aws:iot:us-west-2:123456789012:authorizer/
CustomAuthorizer2"
}
```

Per ulteriori informazioni, consulta la CreateAuthorizersezione AWS IoT API Reference.

- Per i dettagli sull'API, consulta CreateAuthorizer AWS CLICommand Reference.

`create-billing-group`

Il seguente esempio di codice mostra come utilizzare`create-billing-group`.

AWS CLI

Come creare un gruppo di fatturazione

L'esempio `create-billing-group` seguente crea un gruppo di fatturazione semplice denominato `GroupOne`.

```
aws iot create-billing-group \
    --billing-group-name GroupOne
```

Output:

```
{
    "billingGroupName": "GroupOne",
    "billingGroupArn": "arn:aws:iot:us-west-2:123456789012:billinggroup/GroupOne",
    "billingGroupId": "103de383-114b-4f51-8266-18f209ef5562"
}
```

Per ulteriori informazioni, consulta Gruppi di fatturazione nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta CreateBillingGroup AWS CLICommand Reference.

`create-certificate-from-csr`

Il seguente esempio di codice mostra come utilizzare`create-certificate-from-csr`.

AWS CLI

Come creare un certificato da una richiesta di firma del certificato (CSR)

L'esempio `create-certificate-from-csr` seguente crea un certificato di dispositivo da una CSR. È possibile utilizzare il comando `openssl` per creare una CSR.

```
aws iot create-certificate-from-csr \
    --certificate-signing-request=file://certificate.csr
```

Output:

```
{
    "certificateArn": "arn:aws:iot:us-west-2:123456789012:cert/
c0c57bbc8baaf4631a9a0345c957657f5e710473e3ddbee1428d216d54d53ac9",
        "certificateId":
 "c0c57bbc8baaf4631a9a0345c957657f5e710473e3ddbee1428d216d54d53ac9",
        "certificatePem": "<certificate-text>"
}
```

Per ulteriori informazioni, consulta CreateCertificateFromCSR nell'AWS IoT API Reference.

- Per i dettagli sull'API, consulta CreateCertificateFromCsr AWS CLICommand Reference.

`create-custom-metric`

Il seguente esempio di codice mostra come utilizzare`create-custom-metric`.

AWS CLI

Come creare una metrica personalizzata pubblicata dai dispositivi su Device Defender

L'esempio `create-custom-metric` seguente crea una metrica personalizzata che misura la percentuale della batteria.

```
aws iot create-custom-metric \
    --metric-name "batteryPercentage" \
    --metric-type "number" \
    --display-name "Remaining battery percentage." \
    --region us-east-1 \
    --client-request-token "02ccb92b-33e8-4dfa-a0c1-35b181ed26b0"
```

Output:

```
{
    "metricName": "batteryPercentage",
    "metricArn": "arn:aws:iot:us-east-1:1234564789012:custommetric/
batteryPercentage"
}
```

Per ulteriori informazioni, consulta Parametri personalizzati nella Guida per gli sviluppatori di AWS IoT Core.

- Per i dettagli sull'API, consulta CreateCustomMetric AWS CLICommand Reference.

`create-dimension`

Il seguente esempio di codice mostra come utilizzare`create-dimension`.

AWS CLI

Come creare una dimensione

L'esempio `create-dimension` seguente crea una dimensione con un singolo filtro di argomento denominato `TopicFilterForAuthMessages`.

```
aws iot create-dimension \
    --name TopicFilterForAuthMessages \
    --type TOPIC_FILTER \
    --string-values device/+/auth
```

Output:

```
{
    "name": "TopicFilterForAuthMessages",
    "arn": "arn:aws:iot:eu-west-2:123456789012:dimension/TopicFilterForAuthMessages"
}
```

Per ulteriori informazioni, consulta Comandi di rilevamento nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta CreateDimension AWS CLICommand Reference.

`create-domain-configuration`

Il seguente esempio di codice mostra come utilizzare`create-domain-configuration`.

AWS CLI

Come creare una configurazione di dominio

L'`create-domain-configuration`esempio seguente crea una configurazione di dominio AWS-managed con un tipo di servizio diDATA.

```
aws iot create-domain-configuration \
    --domain-configuration-name "additionalDataDomain" \
    --service-type "DATA"
```

Output:

```
{
    "domainConfigurationName": "additionalDataDomain",
    "domainConfigurationArn": "arn:aws:iot:us-
west-2:123456789012:domainconfiguration/additionalDataDomain/dikMh"
```

```
}
```

Per ulteriori informazioni, consulta Endpoint configurabili nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, vedere CreateDomainConfigurationin AWS CLI Command Reference.

`create-dynamic-thing-group`

Il seguente esempio di codice mostra come utilizzare`create-dynamic-thing-group`.

AWS CLI

Come creare un gruppo di oggetti dinamico

L'esempio `create-dynamic-thing-group` seguente crea un gruppo di oggetti dinamico che contiene qualsiasi oggetto con un attributo di temperatura maggiore di 60 gradi. È necessario abilitare l'indicizzazione della flotta AWS IoT prima di poter utilizzare i gruppi di oggetti dinamici.

```
aws iot create-dynamic-thing-group \
    --thing-group-name "RoomTooWarm" \
    --query-string "attributes.temperature>60"
```

Output:

```
{
    "thingGroupName": "RoomTooWarm",
    "thingGroupArn": "arn:aws:iot:us-west-2:123456789012:thinggroup/RoomTooWarm",
    "thingGroupId": "9d52492a-fc87-43f4-b6e2-e571d2ffcad1",
    "indexName": "AWS_Things",
    "queryString": "attributes.temperature>60",
    "queryVersion": "2017-09-30"
}
```

Per ulteriori informazioni, consulta Gruppi di oggetti dinamici nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta AWS CLI Command CreateDynamicThingGroupReference.

`create-job`

Il seguente esempio di codice mostra come utilizzare`create-job`.

AWS CLI

Esempio 1: come creare un processo

L'`create-job`esempio seguente crea un semplice processo AWS IoT che invia un documento JSON al `MyRaspberryPi` dispositivo.

```
aws iot create-job \
    --job-id "example-job-01" \
    --targets "arn:aws:iot:us-west-2:123456789012:thing/MyRaspberryPi" \
    --document file://example-job.json \
    --description "example job test" \
    --target-selection SNAPSHOT
```

Output:

```
{
    "jobArn": "arn:aws:iot:us-west-2:123456789012:job/example-job-01",
    "jobId": "example-job-01",
    "description": "example job test"
}
```

Esempio 2: come creare un processo continuo

L'esempio `create-job` seguente crea un processo che continua a essere eseguito dopo che gli oggetti specificati come destinazioni hanno completato il processo. In questo esempio, la destinazione è un gruppo di oggetti e quindi, quando vengono aggiunti nuovi dispositivi al gruppo, il processo continuo viene eseguito su questi nuovi oggetti.

aws iot create-job --job-id «example-job-04" --targets «arn:aws:iot:us-west- 2:123456789012: -job.json --description «esempio di lavoro continuo» --target-selection CONTINUOUS thinggroup/DeadBulbs" --document file://example

Output:

```
{
    "jobArn": "arn:aws:iot:us-west-2:123456789012:job/example-job-04",
    "jobId": "example-job-04",
    "description": "example continuous job"
}
```

Per ulteriori informazioni, consulta Creazione e gestione di processi (CLI) nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'CreateJobAPI, consulta AWS CLI Command Reference.

`create-keys-and-certificate`

Il seguente esempio di codice mostra come utilizzare`create-keys-and-certificate`.

AWS CLI

Come creare una coppia di chiavi RSA e generare un certificato X.509

L'esempio `create-keys-and-certificate` seguente crea una coppia di chiavi RSA a 2.048 bit e genera un certificato X.509 utilizzando la chiave pubblica emessa. Poiché è l'unica volta in cui l' AWS IoT fornisce la chiave privata per questo certificato, assicurati di conservarlo in un luogo sicuro.

```
aws iot create-keys-and-certificate \
    --certificate-pem-outfile "myTest.cert.pem" \
    --public-key-outfile "myTest.public.key" \
    --private-key-outfile "myTest.private.key"
```

Output:

```
{
    "certificateArn": "arn:aws:iot:us-
west-2:123456789012:cert/9894ba17925e663f1d29c23af4582b8e3b7619c31f3fbd93adcb51ae54b83dc2",
    "certificateId":
 "9894ba17925e663f1d29c23af4582b8e3b7619c31f3fbd93adcb51ae54b83dc2",
    "certificatePem": "
-----BEGIN CERTIFICATE-----
MIICiTCCEXAMPLE6m7oRw0uXOjANBgkqhkiG9w0BAQUFADCBiDELMAkGA1UEBhMC
VVMxCzAJBgNVBAgEXAMPLEAwDgYDVQQHEwdTZWF0dGxlMQ8wDQYDVQQKEwZBbWF6
b24xFDASBgNVBAsTC0lBTSEXAMPLE2xlMRIwEAYDVQQDEwlUZXN0Q2lsYWMxHzAd
BgkqhkiG9w0BCQEWEG5vb25lQGFtYEXAMPLEb20wHhcNMTEwNDI1MjA0NTIxWhcN
MTIwNDI0MjA0NTIxWjCBiDELMAkGA1UEBhMCEXAMPLEJBgNVBAgTAldBMRAwDgYD
VQQHEwdTZWF0dGxlMQ8wDQYDVQQKEwZBbWF6b24xFDAEXAMPLEsTC0lBTSBDb25z
b2xlMRIwEAYDVQQDEwlUZXN0Q2lsYWMxHzAdBgkqhkiG9w0BCQEXAMPLE25lQGFt
YXpvbi5jb20wgZ8wDQYJKoZIhvcNAQEBBQADgY0AMIGJAoGBAMaK0dn+aEXAMPLE
EXAMPLEfEvySWtC2XADZ4nB+BLYgVIk60CpiwsZ3G93vUEIO3IyNoH/f0wYK8m9T
rDHudUZEXAMPLELG5M43q7Wgc/MbQITxOUSQv7c7ugFFDzQGBzZswY6786m86gpE
```

```
Ibb3OhjZnzcvQAEXAMPLEWIMm2nrAgMBAAEwDQYJKoZIhvcNAQEFBQADgYEAtCu4
nUhVVxYUntneD9+h8Mg9qEXAMPLEyExzyLwaxlAoo7TJHidbtS4J5iNmZgXL0Fkb
FFBjvSfpJIlJ00zbhNYS5f6GuoEDEXAMPLEBHjJnyp378OD8uTs7fLvjx79LjSTb
NYiytVbZPQUQ5Yaxu2jXnimvw3rrszlaEXAMPLE=
-----END CERTIFICATE-----\n",
    "keyPair": {
        "PublicKey": "-----BEGIN PUBLIC KEY-----
\nMIIBIjANBgkqhkEXAMPLEQEFAAOCAQ8AMIIBCgKCAQEAEXAMPLE1nnyJwKSMHw4h\nMMEXAMPLEuuN/
dMAS3fyce8DW/4+EXAMPLEyjmoF/YVF/gHr99VEEXAMPLE5VF13\n59VK7cEXAMPLE67GK+y+jikqXOgHh/
xJTwo
+sGpWEXAMPLEDz18xOd2ka4tCzuWEXAMPLEahJbYkCPUBSU8opVkR7qkEXAMPLE1DR6sx2HocliOOLtu6Fkw91swQWEX
\GB3ZPrNh0PzQYvjUStZeccyNCx2EXAMPLEvp9mQOUXP6plfgxwKRX2fEXAMPLEDa
\nhJLXkX3rHU2xbxJSq7D+XEXAMPLEcw+LyFhI5mgFRl88eGdsAEXAMPLElnI9EesG\nFQIDAQAB\n-----
END PUBLIC KEY-----\n",
        "PrivateKey": "-----BEGIN RSA PRIVATE KEY-----\nkey omittted for security
 reasons\n-----END RSA PRIVATE KEY-----\n"
    }
}
```

Per ulteriori informazioni, consulta Creare e registrare un certificato per dispositivi AWS IoT nella Guida per sviluppatori AWS IoT.

- Per i dettagli sull'API, consulta Command CreateKeysAndCertificateReference AWS CLI .

`create-mitigation-action`

Il seguente esempio di codice mostra come utilizzare`create-mitigation-action`.

AWS CLI

Come creare un'azione di mitigazione

L'esempio `create-mitigation-action` seguente definisce un'azione di mitigazione denominata `AddThingsToQuarantineGroup1Action` che, una volta applicata, sposta gli oggetti nel gruppo di oggetti denominato `QuarantineGroup1`. Questa azione sostituisce i gruppi di oggetti dinamici.

```
aws iot create-mitigation-action --cli-input-json file::params.json
```

Contenuto di `params.json`:

```
{
```

```
    "actionName": "AddThingsToQuarantineGroup1Action",
    "actionParams": {
        "addThingsToThingGroupParams": {
            "thingGroupNames": [
                "QuarantineGroup1"
            ],
            "overrideDynamicGroups": true
        }
    },
    "roleArn": "arn:aws:iam::123456789012:role/service-role/
MoveThingsToQuarantineGroupRole"
}
```

Output:

```
{
    "actionArn": "arn:aws:iot:us-west-2:123456789012:mitigationaction/
AddThingsToQuarantineGroup1Action",
    "actionId": "992e9a63-a899-439a-aa50-4e20c52367e1"
}
```

Per ulteriori informazioni, consulta CreateMitigationAction (Mitigation Action Commands) nella AWS IoT Developer Guide.

- Per i dettagli sull'API, consulta CreateMitigationAction AWS CLICommand Reference.

`create-ota-update`

Il seguente esempio di codice mostra come utilizzare`create-ota-update`.

AWS CLI

Come creare un aggiornamento OTA da utilizzare con Amazon FreeRTOS

L'`create-ota-update`esempio seguente crea un AWS IoT OTAUpdate su un gruppo target di cose o gruppi. Questo fa parte di un aggiornamento di Amazon over-the-air FreerTOS che consente di distribuire nuove immagini del firmware su un singolo dispositivo o un gruppo di dispositivi.

```
aws iot create-ota-update \
    --cli-input-json file://create-ota-update.json
```

Contenuto di `create-ota-update.json`:

```
{
    "otaUpdateId": "ota12345",
    "description": "A critical update needed right away.",
    "targets": [
        "device1",
        "device2",
        "device3",
        "device4"
    ],
    "targetSelection": "SNAPSHOT",
    "awsJobExecutionsRolloutConfig": {
        "maximumPerMinute": 10
    },
    "files": [
        {
          "fileName": "firmware.bin",
          "fileLocation": {
            "stream": {
              "streamId": "004",
              "fileId":123
            }
          },
          "codeSigning": {
            "awsSignerJobId": "48c67f3c-63bb-4f92-a98a-4ee0fbc2bef6"
          }
        }
    ]
    "roleArn": "arn:aws:iam:123456789012:role/service-role/my_ota_role"
}
```

Output:

```
{
     "otaUpdateId": "ota12345",
     "awsIotJobId": "job54321",
     "otaUpdateArn": "arn:aws:iot:us-west-2:123456789012:otaupdate/itsaupdate",
     "awsIotJobArn": "arn:aws:iot:us-west-2:123456789012:job/itsajob",
     "otaUpdateStatus": "CREATE_IN_PROGRESS"
}
```

Per ulteriori informazioni, consulta Create OTAUpdate in the AWS IoT API Reference.

- Per i dettagli sull'API, consulta CreateOtaUpdate AWS CLICommand Reference.

`create-policy-version`

Il seguente esempio di codice mostra come utilizzare`create-policy-version`.

AWS CLI

Come aggiornare una policy con una nuova versione

L'esempio `create-policy-version` seguente aggiorna una definizione di policy, creando una nuova versione di policy. Questo esempio imposta inoltre la nuova versione come predefinita.

```
aws iot create-policy-version \
    --policy-name UpdateDeviceCertPolicy \
    --policy-document file://policy.json \
    --set-as-default
```

Contenuto di `policy.json`:

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action":  "iot:UpdateCertificate",
            "Resource": "*"
        }
    ]
}
```

Output:

```
{
    "policyArn": "arn:aws:iot:us-west-2:123456789012:policy/UpdateDeviceCertPolicy",
    "policyDocument": "{ \"Version\": \"2012-10-17\", \"Statement\": [ { \"Effect\":
 \"Allow\", \"Action\":  \"iot:UpdateCertificate\", \"Resource\": \"*\" } ] }",
    "policyVersionId": "2",
    "isDefaultVersion": true
}
```

Per ulteriori informazioni, consulta Policy AWS IoT nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta CreatePolicyVersion AWS CLICommand Reference.

`create-policy`

Il seguente esempio di codice mostra come utilizzare`create-policy`.

AWS CLI

Per creare una policy AWS IoT

L'`create-policy`esempio seguente crea una policy AWS IoT denominata TemperatureSensorPolicy. Il `policy.json` file contiene istruzioni che consentono azioni politiche AWS IoT.

```
aws iot create-policy \
    --policy-name TemperatureSensorPolicy \
    --policy-document file://policy.json
```

Contenuto di `policy.json`:

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "iot:Publish",
                "iot:Receive"
            ],
            "Resource": [
                "arn:aws:iot:us-west-2:123456789012:topic/topic_1",
                "arn:aws:iot:us-west-2:123456789012:topic/topic_2"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "iot:Subscribe"
            ],
            "Resource": [
```

```
                "arn:aws:iot:us-west-2:123456789012:topicfilter/topic_1",
                "arn:aws:iot:us-west-2:123456789012:topicfilter/topic_2"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "iot:Connect"
            ],
            "Resource": [
                "arn:aws:iot:us-west-2:123456789012:client/basicPubSub"
            ]
        }
    ]
}
```

Output:

```
{
    "policyName": "TemperatureSensorPolicy",
    "policyArn": "arn:aws:iot:us-west-2:123456789012:policy/
TemperatureSensorPolicy",
    "policyDocument": "{
        \"Version\": \"2012-10-17\",
        \"Statement\": [
            {
                \"Effect\": \"Allow\",
                \"Action\": [
                    \"iot:Publish\",
                    \"iot:Receive\"
                ],
                \"Resource\": [
                    \"arn:aws:iot:us-west-2:123456789012:topic/topic_1\",
                    \"arn:aws:iot:us-west-2:123456789012:topic/topic_2\"
                ]
            },
            {
                \"Effect\": \"Allow\",
                \"Action\": [
                    \"iot:Subscribe\"
                ],
                \"Resource\": [
                    \"arn:aws:iot:us-west-2:123456789012:topicfilter/topic_1\",
```

```
                    \"arn:aws:iot:us-west-2:123456789012:topicfilter/topic_2\"
                ]
            },
            {
                \"Effect\": \"Allow\",
                \"Action\": [
                    \"iot:Connect\"
                ],
                \"Resource\": [
                    \"arn:aws:iot:us-west-2:123456789012:client/basicPubSub\"
                ]
            }
        ]
    }",
    "policyVersionId": "1"
}
```

Per ulteriori informazioni, consulta Policy AWS IoT nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta CreatePolicy AWS CLICommand Reference.

`create-provisioning-claim`

Il seguente esempio di codice mostra come utilizzare`create-provisioning-claim`.

AWS CLI

Come creare un'attestazione di provisioning

L'esempio `create-provisioning-claim` seguente crea un'attestazione di provisioning da un modello di provisioning.

```
aws iot create-provisioning-claim \
    --template-name MyTestProvisioningTemplate
```

Output:

```
{
    "certificateId":
 "78de02184b2ce80cf8fb709bda59e62b19fb83513590483eb0434589476ab09f",
    "certificatePem": "-----BEGIN CERTIFICATE-----\nMIIDdzCCAl
+gAwIBAgIUXSZhEBLztMLZ2fHG
```

```
14gV0NymYY0wDQYJKoZIhvcNAQEL
\nBQAwfjELMAkGA1UEBhMCVVMxEzARBgNVBAgMCldhc2hpbmd0b24xEDAOBg
VBAcM\nB1NlYXR0bGUxGDAWBgNVBAoMD0FtYXpvbi5jb20gSW5jLjEgMB4GA1UECwwXQW1h
\nem9uIElvVCBQcm9
2aXNpb25pbmcxDDAKBgNVBAUTAzEuMDAeFw0yMDA3MjgxNjQ0\nMDZaFw0yMDA3MjgxNjUxMDZaMEsxSTBHBgNVB
AMMQDFhNDEyM2VkNmIxYjU3MzE3\nZTgzMTJmY2MzN2FiNTdhY2MzYTZkZGVjOGQ5OGY3NzUwMWRlMjc0YjhmYTQ
xN2Iw\nggEiMA0GCSqGSIb3EXAMPLEAA4IBDwAwggEKAoIBAQDBhKI94ktKLqTwnj+ayOq1\nTAJt/
N6s6IJDZvl
rYjkC0E7wzaeY3TprWk03S29vUzVuEOXHXQXZbihgpg2m6fza\nkWm9/
wpjzE9ny5+xkPGVH4Wnwz7yK5m8S0agL
T96cRBSWnWmonOWdY0GKVzni0CA\n+iyGudgrFKm7Eae/
v18oXrf82KtOAGO4xG0KE2WKYHsT1fx3c9xZhlXP/eX
Lhv00\n+lGp0WVw9PbhKfrxliKJ5q6sL5nVUaUHq6hlQPYwsATeOvAp3u0ak5zgTyL0fg7Y
\nPyKk6VYwLW62r+V
YBSForEMOAhkq3LsP/rjxpEKmi2W4lPVS6oFZRKcD+H1Kyil5\nAgMBAAGjIDAeMAwGA1UdEwEB/
wQCMAAwDgYDV
R0PAQH/BAQDAgeAMA0GCSqGSIb3\nDQEBCwUAA4IBAQAGgix2k6nVqbZFKq97/fZBzLGS0dyz5rT/
E41cDIRX+1j
EPW41\nw0D+2sXheCZLZZnSkvIiP74IToNeXDrjdcaodeGFVHIElRjhMIq+4ZebPbRLtidF
\nRc2hfcTAlqq9Z6v
5Vk6BeM1tu0RqH1wPoVUccLPya8EjNCbnJZUmGdOfrN/Y9pho\n5ikV+HPeZhG/k6dhE2GsQJyKFVHL/
uBgKSily
1bRyWU1r6qcpWBNBHjUoD7HgOwD
\nnzMh4XRb2FQDsqFalkCSYmeL8IVC49sgPD9Otyp5uteGMTy62usAAUQdq/f
ZvrWg\nOkFpwMVnGKVKT7Kg0kKOLzKWOBB2Jm4/gmrJ\n-----END CERTIFICATE-----\n",
    "keyPair": {
        "PublicKey": "-----BEGIN PUBLIC KEY-----
\nMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCg
KCAQEAwYSiPeJLSi6k8J4/msjq
\ntUwCbfzerOiCQ2b5a2I5AtBO8M2nmN06a1pNN0tvb1M1bhDlx10F2W4oYKYN
pun8\n2pFpvf8KY8xPZ8ufsZDxlR+Fp8M+8iuZvEtGoC0/enEQUlp1pqJzlnWNBilc54tA
\ngPoshrnYKxSpuxGn
v79fKF63/NirTgBjuMRtChNlimEXAMPLE3PcWYZVz/3ly4b9\nNPpRqdFlcPT24Sn68ZYiieaurC
+Z1VGlB6uoZU
D2MLAE3jrwKd7tGpOc4E8i9H4O\n2D8ipOlWMC1utq/
lWAUhaKxDDgIZKty7D/648aRCpotluJT1UuqBWUSnA/h9
Ssop\neQIDAQAB\n-----END PUBLIC KEY-----\n",
        "PrivateKey": "-----BEGIN RSA PRIVATE KEY-----
\nMIIEowIBAAKCAQEAwYSiPeJLSi6k8J4/
msjqtUwCbfzerOiCQ2b5a2I5AtBO8M2n
\nmN06a1pNN0tvb1M1bhDlx10F2W4oYKYNpun82pFpvf8KY8xPZ8ufsZ
DxlR+Fp8M+\n8iuZvEtGoC0/enEQUlp1pqJzlnWNBilc54tAgPoshrnYKxSpuxGnv79fKF63/Nir
\nTgBjuMRtCh
```

```
NlimB7E9X8d3PcWYZVz/3ly4b9NPpRqdFlcPT24Sn68ZYiieaurC+Z
\n1VGlB6uoZUD2MLAE3jrwKd7tGpOc4E8i
9H4O2D8ipOlWMC1utq/lWAUhaKxDDgIZ\nKty7D/648aRCpotluJT1UuqBWUSnA/
h9SsopeQIDAQABAoIBAEAybN
QUtx9T2/nK\ntZT2pA4iugecxI4dz+DmT0XVXs5VJmrx/
nBSq6ejXExEpSIMO4RY7LE3ZdJcnd56\nF7tQkkY7yR
VzfxHeXFU1krOIPuxWebNOrRoPZr+1RSer+wv2aBC525+88pVuR6tM
\nm3pgkrR2ycCj9FdOUoQxdjHBHaM5PDmJ
9aSxCKdg3nReepeGwsR2TQA+m2vVxWk7\nou0+91eTOP+/QfP7P8ZjOIkO2XivlRcVDyN/
E4QXPKuIkM/8vS8VK+
E9pATQ0MtB\n2lw8R/YU5AJd6jlEXAMPLEGU2UzRzInNWiLtkPPPqgqXXhxOf+mxByjcMalVJk0L
\nhOG2ROUCgY
EA+ROcHNHy/XbsP7FihOhEh+6Q2QxQ2ncBUPYbBazrR8Hn+7SCICQK
\nVyYfd8Ajfq3e7RsKVL5SlMBp7Slidxak
bIn28fKfPn62DaemGCIoyDgLpF+eUxBx
\ngzbCiBZga8brfurza43UZjKZLpg3hq721+FeAiXi1Nma4Yr9YWEHEN
8CgYEAxuWt\npzdWWmsiFzfsAw0sy9ySDA/xr5WRWzJyAqUsjsks6rxNzWebpufnYHcmtW7pLdqM
\nkboHwN2pXa
kmZvrk2nKkEMq5brBYGDXuxDe+V369Bianx8aZFyIsckA7OwXW1w1h
\ngRC5rQ4XOgp3+Jmw7eAO8LRYDjaN846+
QbtO2KcCgYAWS0UL51bijQR0ZwI0dz27\nFQVuCAYsp748aurcRTACCj8jbnK/
QbqTNlxWsaH7ssBjZKo2D5sAqY
BRtASWODab\naHXsDhVm2Jye+ESLoHMaCLoyCkT3ll8yqXIcEDStMO7fO1Ryag164EiJvSIrMfny\nNL/
fXVjCSH
/udCxdzPt+7QKBgQC+LAD7rxdr4J9538hTqpc4XK9vxRbrMXEH55XH
\nHbMa2xONZXpmeTgEQBukyohCVceyRhK9
i0e6irZTjVXghOeoTpC8VXkzcnzouTiQ
\neFQQSGfnp7Ioe6UIz23715pKduzSNkMSKrG924ktv7CyDBF1gBQI5g
aDoHnddJBJ\nPRTIZQKBgA8MASXtTxQntRwXXzR92U0vAighiuRkB/mx9jQpUcK1qiqHbkAMqgNF
\nPFCBYIUbFT
iYKKKeJNbyJQvjfsJCkAnaFJ+RnTxk0Q6Wjm20peJ/ii4QiDdnigoE\nvdlc5cFQewWb4/
zqAtPdinkPlN94ileI
79XQdc7RlJ0jpgTimL+V\n-----END RSA PRIVATE KEY-----\n"
    },
    "expiration": 1595955066.0
}
```

Per ulteriori informazioni, consulta Provisioning per utente attendibile nella Guida per gli sviluppatori di AWS IoT Core.

- Per i dettagli sull'API, consulta CreateProvisioningClaim AWS CLICommand Reference.

`create-provisioning-template-version`

Il seguente esempio di codice mostra come utilizzare`create-provisioning-template-version`.

AWS CLI

Come creare una versione per un modello di provisioning

L'esempio seguente crea una versione per il modello di provisioning specificato. Il corpo della nuova versione viene fornito nel file `template.json`.

```
aws iot create-provisioning-template-version \
    --template-name widget-template \
    --template-body file://template.json
```

Contenuto di `template.json`:

```
{
    "Parameters" : {
        "DeviceLocation": {
            "Type": "String"
        }
    },
    "Mappings": {
        "LocationTable": {
            "Seattle": {
                "LocationUrl": "https://example.aws"
            }
        }
    },
    "Resources" : {
        "thing" : {
            "Type" : "AWS::IoT::Thing",
            "Properties" : {
                "AttributePayload" : {
                    "version" : "v1",
                    "serialNumber" : "serialNumber"
                },
                "ThingName" : {"Fn::Join":["",["ThingPrefix_",
{"Ref":"SerialNumber"}]]},
                "ThingTypeName" : {"Fn::Join":["",["ThingTypePrefix_",
{"Ref":"SerialNumber"}]]},
```

```
                "ThingGroups" : ["widgets", "WA"],
                "BillingGroup": "BillingGroup"
            },
            "OverrideSettings" : {
                "AttributePayload" : "MERGE",
                "ThingTypeName" : "REPLACE",
                "ThingGroups" : "DO_NOTHING"
            }
        },
        "certificate" : {
            "Type" : "AWS::IoT::Certificate",
            "Properties" : {
                "CertificateId": {"Ref": "AWS::IoT::Certificate::Id"},
                "Status" : "Active"
            }
        },
        "policy" : {
            "Type" : "AWS::IoT::Policy",
            "Properties" : {
                "PolicyDocument" : {
                    "Version":"2012-10-17",
                    "Statement": [{
                        "Effect": "Allow",
                        "Action":["iot:Publish"],
                        "Resource": ["arn:aws:iot:us-east-1:123456789012:topic/foo/
bar"]
                    }]
                }
            }
        }
    },
    "DeviceConfiguration": {
        "FallbackUrl": "https://www.example.com/test-site",
        "LocationUrl": {
            "Fn::FindInMap": ["LocationTable",{"Ref": "DeviceLocation"},
 "LocationUrl"]}
        }
    }
}
```

Output:

```
{
```

```
    "templateArn": "arn:aws:iot:us-east-1:123456789012:provisioningtemplate/widget-
template",
    "templateName": "widget-template",
    "versionId": 2,
    "isDefaultVersion": false
}
```

Per ulteriori informazioni, consulta Tunnel sicuro di AWS IoT nella Guida per gli sviluppatori di AWS IoT Core.

- Per i dettagli sull'API, consulta CreateProvisioningTemplateVersion AWS CLICommand Reference.

`create-provisioning-template`

Il seguente esempio di codice mostra come utilizzare`create-provisioning-template`.

AWS CLI

Come creare un modello di provisioning

L'esempio `create-provisioning-template` seguente crea un modello di provisioning come definito dal file `template.json`.

```
aws iot create-provisioning-template \
    --template-name widget-template \
    --description "A provisioning template for widgets" \
    --provisioning-role-arn arn:aws:iam::123456789012:role/Provision_role \
    --template-body file://template.json
```

Contenuto di `template.json`:

```
{
    "Parameters" : {
        "DeviceLocation": {
            "Type": "String"
        }
    },
    "Mappings": {
        "LocationTable": {
            "Seattle": {
                "LocationUrl": "https://example.aws"
            }
```

```
        }
    },
    "Resources" : {
        "thing" : {
            "Type" : "AWS::IoT::Thing",
            "Properties" : {
                "AttributePayload" : {
                    "version" : "v1",
                    "serialNumber" : "serialNumber"
                },
                "ThingName" : {"Fn::Join":["",["ThingPrefix_",
{"Ref":"SerialNumber"}]]},
                "ThingTypeName" : {"Fn::Join":["",["ThingTypePrefix_",
{"Ref":"SerialNumber"}]]},
                "ThingGroups" : ["widgets", "WA"],
                "BillingGroup": "BillingGroup"
            },
            "OverrideSettings" : {
                "AttributePayload" : "MERGE",
                "ThingTypeName" : "REPLACE",
                "ThingGroups" : "DO_NOTHING"
            }
        },
        "certificate" : {
            "Type" : "AWS::IoT::Certificate",
            "Properties" : {
                "CertificateId": {"Ref": "AWS::IoT::Certificate::Id"},
                "Status" : "Active"
            }
        },
        "policy" : {
            "Type" : "AWS::IoT::Policy",
            "Properties" : {
                "PolicyDocument" : {
                    "Version":"2012-10-17",
                    "Statement": [{
                        "Effect": "Allow",
                        "Action":["iot:Publish"],
                        "Resource": ["arn:aws:iot:us-east-1:504350838278:topic/foo/
bar"]
                    }]
                }
            }
        }
```

```
    },
    "DeviceConfiguration": {
        "FallbackUrl": "https://www.example.com/test-site",
        "LocationUrl": {
            "Fn::FindInMap": ["LocationTable",{"Ref": "DeviceLocation"},
 "LocationUrl"]}
        }
    }
}
```

Output:

```
{
    "templateArn": "arn:aws:iot:us-east-1:123456789012:provisioningtemplate/widget-
template",
    "templateName": "widget-template",
    "defaultVersionId": 1
}
```

Per ulteriori informazioni, consulta Tunnel sicuro di AWS IoT nella Guida per gli sviluppatori di AWS IoT Core.

- Per i dettagli sull'API, consulta CreateProvisioningTemplate AWS CLICommand Reference.

`create-role-alias`

Il seguente esempio di codice mostra come utilizzare`create-role-alias`.

AWS CLI

Come creare un alias di ruolo

L'esempio `create-role-alias` seguente crea un alias di ruolo denominato `LightBulbRole` per il ruolo specificato.

```
aws iot create-role-alias \
    --role-alias LightBulbRole \
    --role-arn arn:aws:iam::123456789012:role/lightbulbrole-001
```

Output:

```
{
```

```
    "roleAlias": "LightBulbRole",
    "roleAliasArn": "arn:aws:iot:us-west-2:123456789012:rolealias/LightBulbRole"
}
```

Per ulteriori informazioni, consulta la CreateRoleAliassezione AWS IoT API Reference.

- Per i dettagli sull'API, consulta CreateRoleAlias AWS CLICommand Reference.

`create-scheduled-audit`

Il seguente esempio di codice mostra come utilizzare`create-scheduled-audit`.

AWS CLI

Come creare un audit pianificato

L'esempio `create-scheduled-audit` seguente crea un audit pianificato eseguito settimanalmente, di mercoledì, per verificare se i certificati CA o di dispositivo sono in scadenza.

```
aws iot create-scheduled-audit \
    --scheduled-audit-name WednesdayCertCheck \
    --frequency WEEKLY \
    --day-of-week WED \
    --target-check-
names CA_CERTIFICATE_EXPIRING_CHECK DEVICE_CERTIFICATE_EXPIRING_CHECK
```

Output:

```
{
    "scheduledAuditArn": "arn:aws:iot:us-west-2:123456789012:scheduledaudit/
WednesdayCertCheck"
}
```

Per ulteriori informazioni, consulta Comandi di auditing nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta CreateScheduledAudit AWS CLICommand Reference.

`create-security-profile`

Il seguente esempio di codice mostra come utilizzare`create-security-profile`.

AWS CLI

Come creare un profilo di sicurezza

L'esempio `create-security-profile` seguente crea un profilo di sicurezza che verifica se la larghezza di banda cellulare supera una soglia o se si verificano più di 10 errori di autorizzazione in un periodo di 5 minuti.

```
aws iot create-security-profile \
    --security-profile-name PossibleIssue \
    --security-profile-description "Check to see if authorization fails 10 times in
 5 minutes or if cellular bandwidth exceeds 128"  \
    --behaviors "[{\"name\":\"CellularBandwidth\",\"metric\":\"aws:message-byte-size
\",\"criteria\":{\"comparisonOperator\":\"greater-than\",\"value\":{\"count\":128},
\"consecutiveDatapointsToAlarm\":1,\"consecutiveDatapointsToClear\":1}},{\"name
\":\"Authorization\",\"metric\":\"aws:num-authorization-failures\",\"criteria\":
{\"comparisonOperator\":\"less-than\",\"value\":{\"count\":10},\"durationSeconds
\":300,\"consecutiveDatapointsToAlarm\":1,\"consecutiveDatapointsToClear\":1}}]"
```

Output:

```
{
    "securityProfileName": "PossibleIssue",
    "securityProfileArn": "arn:aws:iot:us-west-2:123456789012:securityprofile/
PossibleIssue"
}
```

Per ulteriori informazioni, consulta Comandi di rilevamento nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta CreateSecurityProfile AWS CLICommand Reference.

`create-stream`

Il seguente esempio di codice mostra come utilizzare`create-stream`.

AWS CLI

Come creare un flusso per la distribuzione di uno o più file di grandi dimensioni in blocchi su MQTT

L'esempio `create-stream` seguente crea un flusso per la distribuzione di uno o più file di grandi dimensioni in blocchi su MQTT. Un flusso trasporta byte di dati in blocchi o blocchi in pacchetti come messaggi MQTT da un'origine come S3. A un flusso possono essere associati uno o più file.

```
aws iot create-stream \
    --cli-input-json file://create-stream.json
```

Contenuto di `create-stream.json`:

```
{
    "streamId": "stream12345",
    "description": "This stream is used for Amazon FreeRTOS OTA Update 12345.",
    "files": [
        {
            "fileId": 123,
            "s3Location": {
                "bucket":"codesign-ota-bucket",
                "key":"48c67f3c-63bb-4f92-a98a-4ee0fbc2bef6"
            }
        }
    ],
    "roleArn": "arn:aws:iam:123456789012:role/service-role/my_ota_stream_role"
}
```

Output:

```
{
     "streamId": "stream12345",
     "streamArn": "arn:aws:iot:us-west-2:123456789012:stream/stream12345",
     "description": "This stream is used for Amazon FreeRTOS OTA Update 12345.",
     "streamVersion": "1"
}
```

Per ulteriori informazioni, consulta la CreateStreamsezione AWS IoT API Reference.

- Per i dettagli sull'API, consulta CreateStream AWS CLICommand Reference.

`create-thing-group`

Il seguente esempio di codice mostra come utilizzare`create-thing-group`.

AWS CLI

Esempio 1: come creare un gruppo di oggetti

L'esempio `create-thing-group` seguente crea un gruppo di oggetti denominato `LightBulbs` con una descrizione e due attributi.

```
aws iot create-thing-group \
    --thing-group-name LightBulbs \
    --thing-group-properties "thingGroupDescription=\"Generic bulb group\",
 attributePayload={attributes={Manufacturer=AnyCompany,wattage=60}}"
```

Output:

```
{
    "thingGroupName": "LightBulbs",
    "thingGroupArn": "arn:aws:iot:us-west-2:123456789012:thinggroup/LightBulbs",
    "thingGroupId": "9198bf9f-1e76-4a88-8e8c-e7140142c331"
}
```

Esempio 2: come creare un gruppo di oggetti che fa parte di un gruppo principale

L'esempio `create-thing-group` seguente crea un gruppo di oggetti denominato `HalogenBulbs` con un gruppo di oggetti principale denominato `LightBulbs`.

```
aws iot create-thing-group \
    --thing-group-name HalogenBulbs \
    --parent-group-name LightBulbs
```

Output:

```
{
    "thingGroupName": "HalogenBulbs",
    "thingGroupArn": "arn:aws:iot:us-west-2:123456789012:thinggroup/HalogenBulbs",
    "thingGroupId": "f4ec6b84-b42b-499d-9ce1-4dbd4d4f6f6e"
}
```

Per ulteriori informazioni, consulta Gruppi di oggetti nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta CreateThingGroup AWS CLICommand Reference.

`create-thing-type`

Il seguente esempio di codice mostra come utilizzare`create-thing-type`.

AWS CLI

Come definire un tipo di oggetto

L'esempio `create-thing-type` seguente definisce un tipo di oggetto e gli attributi associati.

```
aws iot create-thing-type \
    --thing-type-name "LightBulb" \
    --thing-type-properties "thingTypeDescription=light bulb type,
 searchableAttributes=wattage,model"
```

Output:

```
{
    "thingTypeName": "LightBulb",
    "thingTypeArn": "arn:aws:iot:us-west-2:123456789012:thingtype/LightBulb",
    "thingTypeId": "ce3573b0-0a3c-45a7-ac93-4e0ce14cd190"
}
```

Per ulteriori informazioni, consulta Tipi di oggetti nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta CreateThingType AWS CLICommand Reference.

`create-thing`

Il seguente esempio di codice mostra come utilizzare`create-thing`.

AWS CLI

Esempio 1: come creare un record oggetto nel registro

L'`create-thing`esempio seguente crea una voce per un dispositivo nel registro degli AWS oggetti IoT.

```
aws iot create-thing \
    --thing-name SampleIoTThing
```

Output:

```
{
    "thingName": "SampleIoTThing",
    "thingArn": "arn:aws:iot:us-west-2: 123456789012:thing/SampleIoTThing",
    "thingId": " EXAMPLE1-90ab-cdef-fedc-ba987EXAMPLE "
}
```

Esempio 2: come definire un oggetto associato a un tipo di oggetto

L'esempio `create-thing` seguente crea un oggetto con il tipo specificato e i relativi attributi.

```
aws iot create-thing \
    --thing-name "MyLightBulb" \
    --thing-type-name "LightBulb" \
    --attribute-payload "{"attributes": {"wattage":"75", "model":"123"}}"
```

Output:

```
{
    "thingName": "MyLightBulb",
    "thingArn": "arn:aws:iot:us-west-2:123456789012:thing/MyLightBulb",
    "thingId": "40da2e73-c6af-406e-b415-15acae538797"
}
```

Per ulteriori informazioni, consulta Come gestire gli oggetti con il registro e Tipi di oggetti nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, vedere CreateThingin AWS CLI Command Reference.

`create-topic-rule-destination`

Il seguente esempio di codice mostra come utilizzare`create-topic-rule-destination`.

AWS CLI

Come creare la destinazione di una regola dell'argomento

L'esempio `create-topic-rule-destination` seguente crea una destinazione di regole dell'argomento per un endpoint HTTP.

```
aws iot create-topic-rule-destination \
```

```
    --destination-configuration httpUrlConfiguration={confirmationUrl=https://
example.com}
```

Output:

```
{
    "topicRuleDestination": {
        "arn": "arn:aws:iot:us-west-2:123456789012:ruledestination/http/
a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
        "status": "IN_PROGRESS",
        "statusReason": "Awaiting confirmation. Confirmation message sent on
 2020-07-09T22:47:54.154Z; no response received from the endpoint.",
        "httpUrlProperties": {
            "confirmationUrl": "https://example.com"
        }
    }
}
```

Per ulteriori informazioni, consulta Creazione della destinazione di una regola dell'argomento nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta CreateTopicRuleDestination AWS CLICommand Reference.

`create-topic-rule`

Il seguente esempio di codice mostra come utilizzare`create-topic-rule`.

AWS CLI

Come creare una regola che invii un avviso Amazon SNS

L'esempio `create-topic-rule` seguente crea una regola che invia un messaggio Amazon SNS quando le letture del livello di umidità del terreno, rilevate in un device shadow, sono basse.

```
aws iot create-topic-rule \
    --rule-name "LowMoistureRule" \
    --topic-rule-payload file://plant-rule.json
```

L'esempio richiede il salvataggio del seguente codice JSON in un file denominato `plant-rule.json`:

```
{
```

```
    "sql": "SELECT * FROM '$aws/things/MyRPi/shadow/update/accepted' WHERE
 state.reported.moisture = 'low'\n",
    "description": "Sends an alert whenever soil moisture level readings are too
 low.",
    "ruleDisabled": false,
    "awsIotSqlVersion": "2016-03-23",
    "actions": [{
            "sns": {
                "targetArn": "arn:aws:sns:us-
west-2:123456789012:MyRPiLowMoistureTopic",
                "roleArn": "arn:aws:iam::123456789012:role/service-role/
MyRPiLowMoistureTopicRole",
                "messageFormat": "RAW"
            }
    }]
}
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Creating an AWS IoT Rule nella AWS IoT Developers Guide.

- Per i dettagli sull'API, consulta CreateTopicRule AWS CLICommand Reference.

`delete-account-audit-configuration`

Il seguente esempio di codice mostra come utilizzare`delete-account-audit-configuration`.

AWS CLI

Per disabilitare tutti i controlli di controllo per il tuo AWS account

L'`delete-account-audit-configuration`esempio seguente ripristina le impostazioni predefinite per AWS IoT Device Defender per questo account, disabilitando tutti i controlli di controllo e cancellando i dati di configurazione. Elimina inoltre tutti gli audit pianificati per questo account. Utilizza questo comando con cautela.

```
aws iot delete-account-audit-configuration \
    --delete-scheduled-audits
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Comandi di auditing nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta DeleteAccountAuditConfigurationCommand Reference.AWS CLI

`delete-audit-suppression`

Il seguente esempio di codice mostra come utilizzare`delete-audit-suppression`.

AWS CLI

Come eliminare la soppressione dell'esito di un audit

L'esempio `delete-audit-suppression` seguente elimina una soppressione degli esiti di un audit per DEVICE_CERTIFICATE_EXPIRING_CHECK.

```
aws iot delete-audit-suppression \
    --check-name DEVICE_CERTIFICATE_EXPIRING_CHECK \
    --resource-identifier deviceCertificateId="c7691e<shortened>"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Soppressioni degli esiti degli audit nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta DeleteAuditSuppression AWS CLICommand Reference.

`delete-authorizer`

Il seguente esempio di codice mostra come utilizzare`delete-authorizer`.

AWS CLI

Come eliminare un sistema di autorizzazione personalizzato

L'esempio `delete-authorizer` seguente elimina il sistema di autorizzazione denominato `CustomAuthorizer`. Un sistema di autorizzazione personalizzato deve trovarsi nello stato `INACTIVE` prima di poterlo eliminare.

```
aws iot delete-authorizer \
    --authorizer-name CustomAuthorizer
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta DeleteAuthorizerla AWS IoT Developer Guide.

- Per i dettagli sull'API, consulta DeleteAuthorizer AWS CLICommand Reference.

`delete-billing-group`

Il seguente esempio di codice mostra come utilizzare`delete-billing-group`.

AWS CLI

Come eliminare un gruppo di fatturazione

L'esempio `delete-billing-group` seguente elimina il gruppo di fatturazione specificato. È possibile eliminare un gruppo di fatturazione anche se contiene uno o più oggetti.

```
aws iot delete-billing-group \
    --billing-group-name BillingGroupTwo
```

Il comando non produce output.

Per ulteriori informazioni, consulta Gruppi di fatturazione nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta DeleteBillingGroup AWS CLICommand Reference.

`delete-ca-certificate`

Il seguente esempio di codice mostra come utilizzare`delete-ca-certificate`.

AWS CLI

Come eliminare un certificato CA

L'esempio `delete-ca-certificate` seguente elimina il certificato CA con l'ID specificato.

```
aws iot delete-ca-certificate \
    --certificate-
id f4efed62c0142f16af278166f61962501165c4f053629520742646005 8cd1467
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Delete CACertificate nel riferimento dell'API AWS IoT.

- Per i dettagli sull'API, consulta DeleteCaCertificate AWS CLICommand Reference.

`delete-certificate`

Il seguente esempio di codice mostra come utilizzare`delete-certificate`.

AWS CLI

Come eliminare un certificato del dispositivo

L'esempio `delete-certificate` seguente elimina il certificato di dispositivo con l'ID specificato.

```
aws iot delete-certificate \
    --certificate-
id c0c57bbc8baaf4631a9a0345c957657f5e710473e3ddbee1428d216d54d53ac9
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta la DeleteCertificatesezione AWS IoT API Reference.

- Per i dettagli sull'API, consulta DeleteCertificate AWS CLICommand Reference.

`delete-custom-metric`

Il seguente esempio di codice mostra come utilizzare`delete-custom-metric`.

AWS CLI

Come eliminare una metrica personalizzata

L'esempio `delete-custom-metric` seguente elimina una metrica personalizzata.

```
aws iot delete-custom-metric \
    --metric-name batteryPercentage \
    --region us-east-1
```

Output:

```
HTTP 200
```

Per ulteriori informazioni, consulta Parametri personalizzati nella Guida per gli sviluppatori di AWS IoT Core.

- Per i dettagli sull'API, consulta DeleteCustomMetric AWS CLICommand Reference.

`delete-dimension`

Il seguente esempio di codice mostra come utilizzare`delete-dimension`.

AWS CLI

Come eliminare una dimensione

L'esempio `delete-dimension` seguente elimina una dimensione denominata `TopicFilterForAuthMessages`.

```
aws iot delete-dimension \
    --name TopicFilterForAuthMessages
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Comandi di rilevamento nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta DeleteDimension AWS CLICommand Reference.

`delete-domain-configuration`

Il seguente esempio di codice mostra come utilizzare`delete-domain-configuration`.

AWS CLI

Come eliminare una configurazione di dominio

L'`delete-domain-configuration`esempio seguente elimina una configurazione di dominio denominata `additionalDataDomain` dall' AWS account.

```
aws iot delete-domain-configuration \
    --domain-configuration-name "additionalDataDomain" \
    --domain-configuration-status "OK"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Endpoint configurabili nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta DeleteDomainConfiguration AWS CLICommand Reference.

`delete-dynamic-thing-group`

Il seguente esempio di codice mostra come utilizzare`delete-dynamic-thing-group`.

AWS CLI

Come eliminare un gruppo di oggetti dinamico

L'esempio `delete-dynamic-thing-group` seguente elimina il gruppo di oggetti dinamici specificato.

```
aws iot delete-dynamic-thing-group \
    --thing-group-name "RoomTooWarm"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Gruppi di oggetti dinamici nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta DeleteDynamicThingGroup AWS CLICommand Reference.

`delete-job-execution`

Il seguente esempio di codice mostra come utilizzare`delete-job-execution`.

AWS CLI

Come eliminare l'esecuzione di un processo

L'esempio `delete-job-execution` seguente elimina l'esecuzione del processo specificato su un dispositivo. Utilizza `describe-job-execution` per ottenere il numero di esecuzione.

```
aws iot delete-job-execution
    --job-id "example-job-02"
    --thing-name "MyRaspberryPi"
    --execution-number 1
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Creazione e gestione di processi (CLI) nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta DeleteJobExecution AWS CLICommand Reference.

`delete-job`

Il seguente esempio di codice mostra come utilizzare`delete-job`.

AWS CLI

Come eliminare un processo

L'esempio `delete-job` seguente elimina il processo specificato. Specificando l'opzione `--force`, il processo viene eliminato anche se lo stato è `IN_PROGRESS`.

```
aws iot delete-job \
    --job-id "example-job-04" \
    --force
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Creazione e gestione di processi (CLI) nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta DeleteJob AWS CLICommand Reference.

`delete-mitigation-action`

Il seguente esempio di codice mostra come utilizzare`delete-mitigation-action`.

AWS CLI

Come eliminare un'azione di mitigazione

L'esempio `delete-mitigation-action` seguente elimina l'azione di mitigazione specificata.

```
aws iot delete-mitigation-action \
    --action-name AddThingsToQuarantineGroup1Action
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta DeleteMitigationAction (Mitigation Action Commands) nella AWS IoT Developer Guide.

- Per i dettagli sull'API, consulta DeleteMitigationAction AWS CLICommand Reference.

`delete-ota-update`

Il seguente esempio di codice mostra come utilizzare`delete-ota-update`.

AWS CLI

Come eliminare un aggiornamento OTA

L'esempio `delete-ota-update` seguente elimina l'aggiornamento OTA specificato.

```
aws iot delete-ota-update \
    --ota-update-id ota12345 \
    --delete-stream \
    --force-delete-aws-job
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Delete OTAUpdate nel riferimento dell'API AWS IoT.

- Per i dettagli sull'API, consulta DeleteOtaUpdate AWS CLICommand Reference.

`delete-policy-version`

Il seguente esempio di codice mostra come utilizzare`delete-policy-version`.

AWS CLI

Come eliminare una versione di policy

L'`delete-policy-version`esempio seguente elimina la versione 2 della politica specificata dal tuo AWS account.

```
aws iot delete-policy-version \
    --policy-name UpdateDeviceCertPolicy \
    --policy-version-id 2
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Policy IoT AWS nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta DeletePolicyVersion AWS CLICommand Reference.

`delete-policy`

Il seguente esempio di codice mostra come utilizzare`delete-policy`.

AWS CLI

Come eliminare una policy

L'`delete-policy`esempio seguente elimina la politica specificata dal tuo AWS account.

```
aws iot delete-policy --policy-name UpdateDeviceCertPolicy
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Policy AWS IoT nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta DeletePolicy AWS CLICommand Reference.

`delete-provisioning-template-version`

Il seguente esempio di codice mostra come utilizzare`delete-provisioning-template-version`.

AWS CLI

Come eliminare una versione del modello di provisioning

L'esempio `delete-provisioning-template-version` seguente elimina la versione 2 del modello di provisioning specificato.

```
aws iot delete-provisioning-template-version \
    --version-id 2 \
    --template-name "widget-template"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Tunnel sicuro di AWS IoT nella Guida per gli sviluppatori di AWS IoT Core.

- Per i dettagli sull'API, consulta DeleteProvisioningTemplateVersion AWS CLICommand Reference.

`delete-provisioning-template`

Il seguente esempio di codice mostra come utilizzare`delete-provisioning-template`.

AWS CLI

Come eliminare un modello di provisioning

L'esempio `delete-provisioning-template` seguente elimina il modello di provisioning specificato.

```
aws iot delete-provisioning-template \
    --template-name widget-template
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Tunnel sicuro di AWS IoT nella Guida per gli sviluppatori di AWS IoT Core.

- Per i dettagli sull'API, consulta DeleteProvisioningTemplate AWS CLICommand Reference.

`delete-registration-code`

Il seguente esempio di codice mostra come utilizzare`delete-registration-code`.

AWS CLI

Come eliminare il codice di registrazione

L'`delete-registration-code`esempio seguente elimina un codice di registrazione specifico AWS dell'account IoT.

```
aws iot delete-registration-code
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Uso di un certificato personale nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta Command DeleteRegistrationCodeReference AWS CLI .

`delete-role-alias`

Il seguente esempio di codice mostra come utilizzare`delete-role-alias`.

AWS CLI

Per eliminare un alias di ruolo AWS IoT

L'`delete-role-alias`esempio seguente elimina un alias di ruolo AWS IoT denominato. `LightBulbRole`

```
aws iot delete-role-alias \
    --role-alias LightBulbRole
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Authorizing Direct Calls to AWS Services nella AWS IoT Developer Guide.

- Per i dettagli sull'API, consulta DeleteRoleAlias AWS CLICommand Reference.

`delete-scheduled-audit`

Il seguente esempio di codice mostra come utilizzare`delete-scheduled-audit`.

AWS CLI

Come eliminare un audit pianificato

L'`delete-scheduled-audit`esempio seguente elimina l'audit pianificato di AWS IoT Device Defender denominato. `AWSIoTDeviceDefenderDailyAudit`

```
aws iot delete-scheduled-audit \
    --scheduled-audit-name AWSIoTDeviceDefenderDailyAudit
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Comandi di auditing nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta AWS CLI Command DeleteScheduledAuditReference.

`delete-security-profile`

Il seguente esempio di codice mostra come utilizzare`delete-security-profile`.

AWS CLI

Come eliminare un profilo di sicurezza

L'esempio `delete-security-profile` seguente elimina un profilo di sicurezza denominato `PossibleIssue`.

```
aws iot delete-security-profile \
    --security-profile-name PossibleIssue
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Comandi di rilevamento nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta DeleteSecurityProfile AWS CLICommand Reference.

`delete-stream`

Il seguente esempio di codice mostra come utilizzare`delete-stream`.

AWS CLI

Come eliminare un flusso

L'esempio `delete-stream` seguente elimina il flusso specificato.

```
aws iot delete-stream \
    --stream-id stream12345
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta la DeleteStreamsezione AWS IoT API Reference.

- Per i dettagli sull'API, consulta DeleteStream AWS CLICommand Reference.

`delete-thing-group`

Il seguente esempio di codice mostra come utilizzare`delete-thing-group`.

AWS CLI

Come eliminare un gruppo di oggetti

L'esempio `delete-thing-group` seguente elimina il gruppo di oggetti specificato. Non è possibile eliminare un gruppo di oggetti se contiene gruppi di oggetti secondari.

```
aws iot delete-thing-group \
    --thing-group-name DefectiveBulbs
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Gruppi di oggetti nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta DeleteThingGroup AWS CLICommand Reference.

`delete-thing-type`

Il seguente esempio di codice mostra come utilizzare`delete-thing-type`.

AWS CLI

Esempio 1: come eliminare un tipo di oggetto

L'esempio `delete-thing-type` seguente elimina un tipo di oggetto obsoleto.

```
aws iot delete-thing-type \
    --thing-type-name "obsoleteThingType"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Tipi di oggetti nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta DeleteThingType AWS CLICommand Reference.

`delete-thing`

Il seguente esempio di codice mostra come utilizzare`delete-thing`.

AWS CLI

Come visualizzare informazioni dettagliate su un oggetto

L'`delete-thing`esempio seguente elimina un elemento dal registro AWS IoT per il tuo AWS account.

as iot delete-thing --thing-name» FourthBulb

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Come gestire gli oggetti con il registro nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta Command Reference. DeleteThingAWS CLI

`delete-topic-rule-destination`

Il seguente esempio di codice mostra come utilizzare`delete-topic-rule-destination`.

AWS CLI

Come eliminare la destinazione di una regola di argomento

L'esempio `delete-topic-rule-destination` seguente elimina la destinazione di una regola di argomento specificata.

```
aws iot delete-topic-rule-destination \
    --arn "arn:aws:iot:us-west-2:123456789012:ruledestination/http/
a1b2c3d4-5678-90ab-cdef-11111EXAMPLE"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Eliminazione di una destinazione della regola dell'argomento nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta DeleteTopicRuleDestination AWS CLICommand Reference.

`delete-topic-rule`

Il seguente esempio di codice mostra come utilizzare`delete-topic-rule`.

AWS CLI

Come eliminare una regola

L'esempio `delete-topic-rule` seguente elimina la regola specificata.

```
aws iot delete-topic-rule \
    --rule-name "LowMoistureRule"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Eliminazione di una regola nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta DeleteTopicRule AWS CLICommand Reference.

`delete-v2-logging-level`

Il seguente esempio di codice mostra come utilizzare`delete-v2-logging-level`.

AWS CLI

Come eliminare il livello di registrazione di log per un gruppo di oggetti

L'esempio `delete-v2-logging-level` seguente elimina il livello di registrazione di log per il gruppo di oggetti specificato.

```
aws iot delete-v2-logging-level \
    --target-type THING_GROUP \
    --target-name LightBulbs
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta DeleteV2 LoggingLevel in AWS CLI Command Reference.

`deprecate-thing-type`

Il seguente esempio di codice mostra come utilizzare. `deprecate-thing-type`

AWS CLI

Esempio 1: come rendere obsoleto un tipo di oggetto

L'esempio `deprecate-thing-type` seguente rende obsoleto un tipo di oggetto in modo che gli utenti non possano associarvi nuovi oggetti.

```
aws iot deprecate-thing-type \
    --thing-type-name "obsoleteThingType"
```

Questo comando non produce alcun output.

Esempio 2: come invertire l'obsolescenza di un tipo di oggetto

L'esempio `deprecate-thing-type` seguente inverte l'obsolescenza di un tipo di oggetto, in modo che gli utenti possano riassociarvi nuovi oggetti.

```
aws iot deprecate-thing-type \
    --thing-type-name "obsoleteThingType" \
    --undo-deprecate
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Tipi di oggetti nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta DeprecateThingType AWS CLICommand Reference.

`describe-account-audit-configuration`

Il seguente esempio di codice mostra come utilizzare`describe-account-audit-configuration`.

AWS CLI

Come visualizzare le impostazioni per la configurazione di audit correnti

L'`describe-account-audit-configuration`esempio seguente elenca le impostazioni correnti per la configurazione di audit di AWS IoT Device Defender.

```
aws iot describe-account-audit-configuration
```

Output:

```
{
    "roleArn": "arn:aws:iam::123456789012:role/service-role/
AWSIoTDeviceDefenderAudit_1551201085996",
    "auditNotificationTargetConfigurations": {
        "SNS": {
            "targetArn": "arn:aws:sns:us-west-2:123456789012:ddaudits",
            "roleArn": "arn:aws:iam::123456789012:role/service-role/
AWSIoTDeviceDefenderAudit",
            "enabled": true
        }
```

```
    },
    "auditCheckConfigurations": {
        "AUTHENTICATED_COGNITO_ROLE_OVERLY_PERMISSIVE_CHECK": {
            "enabled": true
        },
        "CA_CERTIFICATE_EXPIRING_CHECK": {
            "enabled": true
        },
        "CONFLICTING_CLIENT_IDS_CHECK": {
            "enabled": true
        },
        "DEVICE_CERTIFICATE_EXPIRING_CHECK": {
            "enabled": true
        },
        "DEVICE_CERTIFICATE_SHARED_CHECK": {
            "enabled": true
        },
        "IOT_POLICY_OVERLY_PERMISSIVE_CHECK": {
            "enabled": true
        },
        "LOGGING_DISABLED_CHECK": {
            "enabled": true
        },
        "REVOKED_CA_CERTIFICATE_STILL_ACTIVE_CHECK": {
            "enabled": true
        },
        "REVOKED_DEVICE_CERTIFICATE_STILL_ACTIVE_CHECK": {
            "enabled": true
        },
        "UNAUTHENTICATED_COGNITO_ROLE_OVERLY_PERMISSIVE_CHECK": {
            "enabled": true
        }
    }
}
```

Per ulteriori informazioni, consulta Comandi di auditing nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta DescribeAccountAuditConfiguration AWS CLICommand Reference.

`describe-audit-finding`

Il seguente esempio di codice mostra come utilizzare`describe-audit-finding`.

AWS CLI

Come elencare i dettagli relativi agli esiti di un audit

L'`describe-audit-finding`esempio seguente elenca i dettagli del risultato di audit di AWS IoT Device Defender specificato. Un audit può produrre vari esiti. Utilizza il comando `list-audit-findings` per ricavare un elenco degli esiti di un audit per ottenere `findingId`.

```
aws iot describe-audit-finding \
    --finding-id "ef4826b8-e55a-44b9-b460-5c485355371b"
```

Output:

```
{
    "finding": {
        "findingId": "ef4826b8-e55a-44b9-b460-5c485355371b",
        "taskId": "873ed69c74a9ec8fa9b8e88e9abc4661",
        "checkName": "IOT_POLICY_OVERLY_PERMISSIVE_CHECK",
        "taskStartTime": 1576012045.745,
        "findingTime": 1576012046.168,
        "severity": "CRITICAL",
        "nonCompliantResource": {
            "resourceType": "IOT_POLICY",
            "resourceIdentifier": {
                "policyVersionIdentifier": {
                    "policyName": "smp-ggrass-group_Core-policy",
                    "policyVersionId": "1"
                }
            }
         },
        "reasonForNonCompliance": "Policy allows broad access to IoT data plane
 actions: [iot:Subscribe, iot:Connect, iot:GetThingShadow, iot:DeleteThingShadow,
 iot:UpdateThingShadow, iot:Publish].",
        "reasonForNonComplianceCode":
 "ALLOWS_BROAD_ACCESS_TO_IOT_DATA_PLANE_ACTIONS"
    }
}
```

Per ulteriori informazioni, consulta Controllo dei risultati dell'audit (Comandi di auditing) nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, vedere DescribeAuditFindingin AWS CLI Command Reference.

`describe-audit-mitigation-actions-task`

Il seguente esempio di codice mostra come utilizzare`describe-audit-mitigation-actions-task`.

AWS CLI

Come mostrare i dettagli di un'attività relativa alle azioni di mitigazione degli audit

L'esempio `describe-audit-mitigation-actions-task` seguente mostra i dettagli dell'attività specificata, in cui l'azione `ResetPolicyVersionAction` è stata applicata a un esito. I risultati includono l'ora di inizio e l'ora di fine dell'attività, il numero di esiti presi individuati (e il risultato) e la definizione dell'azione applicata nell'ambito di questa attività.

```
aws iot describe-audit-mitigation-actions-task \
    --task-id ResetPolicyTask01
```

Output:

```
{
    "taskStatus": "COMPLETED",
    "startTime": "2019-12-10T15:13:19.457000-08:00",
    "endTime": "2019-12-10T15:13:19.947000-08:00",
    "taskStatistics": {
        "IOT_POLICY_OVERLY_PERMISSIVE_CHECK": {
            "totalFindingsCount": 1,
            "failedFindingsCount": 0,
            "succeededFindingsCount": 1,
            "skippedFindingsCount": 0,
            "canceledFindingsCount": 0
        }
    },
    "target": {
        "findingIds": [
            "ef4826b8-e55a-44b9-b460-5c485355371b"
        ]
    },
    "auditCheckToActionsMapping": {
        "IOT_POLICY_OVERLY_PERMISSIVE_CHECK": [
            "ResetPolicyVersionAction"
        ]
    },
    "actionsDefinition": [
```

```
        {
            "name": "ResetPolicyVersionAction",
            "id": "1ea0b415-bef1-4a01-bd13-72fb63c59afb",
            "roleArn": "arn:aws:iam::123456789012:role/service-role/
ReplacePolicyVersionRole",
            "actionParams": {
                "replaceDefaultPolicyVersionParams": {
                    "templateName": "BLANK_POLICY"
                }
            }
        }
    ]
}
```

Per ulteriori informazioni, consulta DescribeAuditMitigationActionsTask (Mitigation Action Commands) nella AWS IoT Developer Guide.

- Per i dettagli sull'API, consulta DescribeAuditMitigationActionsTask AWS CLICommand Reference.

`describe-audit-suppression`

Il seguente esempio di codice mostra come utilizzare`describe-audit-suppression`.

AWS CLI

Come ottenere i dettagli relativi alla soppressione degli esiti di audit

L'esempio `describe-audit-suppression` seguente elenca i dettagli relativi alla soppressione degli esiti di un audit.

```
aws iot describe-audit-task \
    --task-id "787ed873b69cb4d6cdbae6ddd06996c5"
```

Output:

```
{
    "taskStatus": "COMPLETED",
    "taskType": "SCHEDULED_AUDIT_TASK",
    "taskStartTime": 1596168096.157,
    "taskStatistics": {
        "totalChecks": 1,
```

```
        "inProgressChecks": 0,
        "waitingForDataCollectionChecks": 0,
        "compliantChecks": 0,
        "nonCompliantChecks": 1,
        "failedChecks": 0,
        "canceledChecks": 0
    },
    "scheduledAuditName": "AWSIoTDeviceDefenderDailyAudit",
    "auditDetails": {
        "DEVICE_CERTIFICATE_EXPIRING_CHECK": {
            "checkRunStatus": "COMPLETED_NON_COMPLIANT",
            "checkCompliant": false,
            "totalResourcesCount": 195,
            "nonCompliantResourcesCount": 2
        }
    }
}
```

Per ulteriori informazioni, consulta Soppressioni degli esiti degli audit nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta DescribeAuditSuppression AWS CLICommand Reference.

`describe-audit-task`

Il seguente esempio di codice mostra come utilizzare`describe-audit-task`.

AWS CLI

Come ottenere informazioni su un'istanza di audit

L'`describe-audit-task`esempio seguente ottiene informazioni su un'istanza di un audit AWS IoT Device Defender. Se l'audit è completo, nei risultati vengono incluse le statistiche riassuntive relative all'esecuzione.

```
aws iot describe-audit-task \
    --task-id a3aea009955e501a31b764abe1bebd3d
```

Output:

```
 {
    "taskStatus": "COMPLETED",
```

```
    "taskType": "ON_DEMAND_AUDIT_TASK",
    "taskStartTime": 1560356923.434,
    "taskStatistics": {
        "totalChecks": 3,
        "inProgressChecks": 0,
        "waitingForDataCollectionChecks": 0,
        "compliantChecks": 3,
        "nonCompliantChecks": 0,
        "failedChecks": 0,
        "canceledChecks": 0
    },
    "auditDetails": {
        "CA_CERTIFICATE_EXPIRING_CHECK": {
            "checkRunStatus": "COMPLETED_COMPLIANT",
            "checkCompliant": true,
            "totalResourcesCount": 0,
            "nonCompliantResourcesCount": 0
        },
        "DEVICE_CERTIFICATE_EXPIRING_CHECK": {
            "checkRunStatus": "COMPLETED_COMPLIANT",
            "checkCompliant": true,
            "totalResourcesCount": 6,
            "nonCompliantResourcesCount": 0
        },
        "REVOKED_CA_CERTIFICATE_STILL_ACTIVE_CHECK": {
            "checkRunStatus": "COMPLETED_COMPLIANT",
            "checkCompliant": true,
            "totalResourcesCount": 0,
            "nonCompliantResourcesCount": 0
        }
    }
}
```

Per ulteriori informazioni, consulta Comandi di auditing nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta DescribeAuditTask AWS CLICommand Reference.

`describe-authorizer`

Il seguente esempio di codice mostra come utilizzare`describe-authorizer`.

AWS CLI

Come ottenere informazioni su un sistema di autorizzazione personalizzato

L'esempio `describe-authorizer` seguente visualizza i dettagli del sistema di autorizzazione personalizzato specificato.

```
aws iot describe-authorizer \
    --authorizer-name CustomAuthorizer
```

Output:

```
{
    "authorizerDescription": {
        "authorizerName": "CustomAuthorizer",
        "authorizerArn": "arn:aws:iot:us-west-2:123456789012:authorizer/
CustomAuthorizer",
        "authorizerFunctionArn": "arn:aws:lambda:us-
west-2:123456789012:function:CustomAuthorizerFunction",
        "tokenKeyName": "MyAuthToken",
        "tokenSigningPublicKeys": {
            "FIRST_KEY": "-----BEGIN PUBLIC KEY-----
\nMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEA1uJOB4lQPgG/lM6ZfIwo
\nZ+7ENxAio9q6QD4FFqjGZsvjtYwjoe1RKK0U8Eq9xb5O3kRSmyIwTzwzm/f4Gf0Y
\nZUloJ+t3PUUwHrmbYTAgTrCUgRFygjfgVwGCPs5ZAX4Eyqt5cr+AIHIiUDbxSa7p
\nzwOBKPeic0asNJpqT8PkBbRaKyleJh5oo81NDHHmVtbBm5A5YiJjqYXLaVAowKzZ\n
+GqsNvAQ9Jy1wI2VrEa1OfL8flDB/BJLm7zjpfPOHDJQgID0XnZwAlNnZcOhCwIx\n50g2LW2Oy9R/
dmqtDmJiVP97Z4GykxPvwlYHrUXY0iW1R3AR/Ac1NhCTGZMwVDB1\nlQIDAQAB\n-----END PUBLIC
 KEY-----"
        },
        "status": "ACTIVE",
        "creationDate": 1571245658.069,
        "lastModifiedDate": 1571245658.069
    }
}
```

Per ulteriori informazioni, consulta la DescribeAuthorizersezione AWS IoT API Reference.

- Per i dettagli sull'API, consulta DescribeAuthorizer AWS CLICommand Reference.

`describe-billing-group`

Il seguente esempio di codice mostra come utilizzare`describe-billing-group`.

AWS CLI

Come ottenere informazioni su un gruppo di fatturazione

L'esempio `describe-billing-group` seguente ottiene informazioni per il gruppo di fatturazione specificato.

```
aws iot describe-billing-group --billing-group-name GroupOne
```

Output:

```
{
    "billingGroupName": "GroupOne",
    "billingGroupId": "103de383-114b-4f51-8266-18f209ef5562",
    "billingGroupArn": "arn:aws:iot:us-west-2:123456789012:billinggroup/GroupOne",
    "version": 1,
    "billingGroupProperties": {},
    "billingGroupMetadata": {
        "creationDate": 1560199355.378
    }
}
```

Per ulteriori informazioni, consulta Gruppi di fatturazione nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta DescribeBillingGroup AWS CLICommand Reference.

`describe-ca-certificate`

Il seguente esempio di codice mostra come utilizzare`describe-ca-certificate`.

AWS CLI

Come ottenere dettagli su un certificato CA

L'esempio `describe-ca-certificate` seguente visualizza i dettagli del certificato CA specificato.

```
aws iot describe-ca-certificate \
    --certificate-
id f4efed62c0142f16af278166f61962501165c4f0536295207426460058cd1467
```

Output:

```
{
```

```
    "certificateDescription": {
        "certificateArn": "arn:aws:iot:us-west-2:123456789012:cacert/
f4efed62c0142f16af278166f61962501165c4f053629520742646005 8cd1467",
        "certificateId":
 "f4efed62c0142f16af278166f61962501165c4f053629520742646005 8cd1467",
        "status": "INACTIVE",
        "certificatePem": "-----BEGIN CERTIFICATE-----
\nMIICzzCCAbegEXAMPLEJANVEPWXl8taPMA0GCSqGSIb3DQEBBQUAMB4xCzAJBgNV
\nBAYTAlVTMQ8wDQYDVQQKDAZBbWF6b24wHhcNMTkwOTI0MjEzMTE1WhcNMjkwOTIx
\nMjEzMTE1WjAeMQswCQYDVQQGEwJVUzEPMA0GA1UECgwGQW1hem9uMIIBIjANBgkq
\nhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAzd3R3ioalCS0MhFWfBrVGR036EK07UAf
\nVdz9EXAMPLE1VczICbADnATK522kEIB51/18VzlFtAhQL5V5eybXKnB7QebNer5m
\n4Yibx7shR5oqNzFsrXWxuugN5+w5gEfqNMawOjhF4LsculKG49yuqjcDU19/13ua
\n3B2gxs1Pe7TiWWvUskzxnbO1F2WCshbEJvqY8fIWtGYCjTeJAgQ9hvZx/69XhKen
\nwV9LJwOQxrsUS0Ty8IHwbB8fRy72VM3u7fJoaU+nO4jD5cqaoEPtzoeFUEXAMPLE
\nyVAJpqHwgbYbcUfn7V+AB6yh1+0Fa1rEQGuZDPGyJslxwr5vh8nRewIDAQABoxAw
\nDjAMBgNVHRMEBTADAQH/MA0GCSqGSIb3DQEBBQUAA4IBAQA+3a5CV3IJgOnd0AgI
\nBgVMtmYzTvqAngx26aG9/spvCjXckh2SBF+EcBlCFwH1yakwjJL1dR4yarnrfxgI
\nEqP4AOYVimAVoQ5FBwnloHe16+3qtDiblU9DeXBUCtS55EcfrEXAMPLEYtXdqU5C
\nU9ia4KAjV0dxW1+EFYMwX5eGeb0gDTNHBylV6B/fOSZiQAwDYp4x3B+gAP+a/bWB
\nu1umOqtBdWe6L6/83L+JhaTByqV25iVJ4c/UZUnG8926wUlDM9zQvEXuEVvzZ7+m\n4PSNqst/
nVOvnLpoG4e0WgcJgANuB33CSWtjWSuYsbhmqQRknGhREXAMPLEZT4fm\nfo0e\n-----END
 CERTIFICATE-----\n",
        "ownedBy": "123456789012",
        "creationDate": 1569365372.053,
        "autoRegistrationStatus": "DISABLE",
        "lastModifiedDate": 1569365372.053,
        "customerVersion": 1,
        "generationId": "c5c2eb95-140b-4f49-9393-6aaac85b2a90",
        "validity": {
            "notBefore": 1569360675.0,
            "notAfter": 1884720675.0
        }
    }
}
```

Per ulteriori informazioni, consulta Descrivi CACertificate nel riferimento dell'API AWS IoT.

- Per i dettagli sull'API, consulta DescribeCaCertificate AWS CLICommand Reference.

`describe-certificate`

Il seguente esempio di codice mostra come utilizzare`describe-certificate`.

AWS CLI

Come ottenere informazioni su un certificato

L'esempio `describe-certificate` seguente visualizza i dettagli del certificato specificato.

```
aws iot describe-certificate \
    --certificate-
id "4f0ba725787aa94d67d2fca420eca022242532e8b3c58e7465c7778b443fd65e"
```

Output:

```
{
    "certificateDescription": {
        "certificateArn": "arn:aws:iot:us-
west-2:123456789012:cert/4f0ba725787aa94d67d2fca420eca022242532e8b3c58e7465c7778b443fd65e",
        "certificateId":
 "4f0ba725787aa94d67d2fca420eca022242532e8b3c58e7465c7778b443fd65e",
        "status": "ACTIVE",
        "certificatePem": "-----BEGIN CERTIFICATE-----
MIICiTEXAMPLEQD6m7oRw0uXOjANBgkqhkiG9w0BAQUFADCBiDELMAkGA1UEBhMC
VVMxCzAJBgNVBEXAMPLEMRAwDgYDVQQHEwdTZWF0dGxlMQ8wDQYDVQQKEwZBbWF6
b24xFDASBgNVBAsTC0lBTSBDEXAMPLElMRIwEAYDVQQDEwlUZXN0Q2lsYWMxHzAd
BgkqhkiG9w0BCQEWEG5vb25lQGFtYXpvbi5EXAMPLEcNMTEwNDI1MjA0NTIxWhcN
MTIwNDI0MjA0NTIxWjCBiDELMAkGA1UEBhMCVVMxCzAJBgNEXAMPLEdBMRAwDgYD
VQQHEwdTZWF0dGxlMQ8wDQYDVQQKEwZBbWF6b24xFDASBgNVBAsTC0lBEXAMPLEz
b2xEXAMPLEYDVQQDEwlUZXN0Q2lsYWMxHzAdBgkqhkiG9w0BCQEWEG5vb25lQGFt
YXpvbi5jb20wgZ8EXAMPLEZIhvcNAQEBBQADgY0AMIGJAoGBAMaK0dn+a4GmWIWJ
21uUSfwfEvySWtC2XADZ4nB+BLYEXAMPLEpiwsZ3G93vUEIO3IyNoH/f0wYK8m9T
rDHudUZg3qX4waLG5M43q7Wgc/MbQITxOUSQv7c7EXAMPLEGBzZswY6786m86gpE
Ibb3OhjZnzcvQAaRHhdlQWIMm2nrAgMBAAEwDQYJKoZIhvcNAQEFEXAMPLEAtCu4
nUhVVxYUnEXAMPLE8Mg9q6q+auNKyExzyLwaxlAoo7TJHidbtS4J5iNmZgXL0Fkb
FFBjvSfpJIlJ00zbhNYS5f6GEXAMPLEl0ZxBHjJnyp378OD8uTs7fLvjx79LjSTb
NYiytVbZPQUQ5Yaxu2jXnimvw3rrszlaEXAMPLE=
-----END CERTIFICATE-----",
        "ownedBy": "123456789012",
        "creationDate": 1541022751.983,
        "lastModifiedDate": 1541022751.983,
        "customerVersion": 1,
        "transferData": {},
        "generationId": "6974fbed-2e61-4114-bc5e-4204cc79b045",
        "validity": {
            "notBefore": 1541022631.0,
```

```
            "notAfter": 2524607999.0
        }
    }
}
```

Per ulteriori informazioni, consulta la DescribeCertificatesezione AWS IoT API Reference.

- Per i dettagli sull'API, consulta DescribeCertificate AWS CLICommand Reference.

`describe-custom-metric`

Il seguente esempio di codice mostra come utilizzare`describe-custom-metric`.

AWS CLI

Come ottenere informazioni su una metrica personalizzata di Device Defender

L'esempio `describe-custom-metric` seguente ottiene informazioni su una metrica personalizzata denominata `myCustomMetric`.

```
aws iot describe-custom-metric \
    --metric-name myCustomMetric
```

Output:

```
{
    "metricName": "myCustomMetric",
    "metricArn": "arn:aws:iot:us-east-1:1234564789012:custommetric/myCustomMetric",
    "metricType": "number",
    "displayName": "My custom metric",
    "creationDate": 2020-11-17T23:02:12.879000-09:00,
    "lastModifiedDate": 2020-11-17T23:02:12.879000-09:00
}
```

Per ulteriori informazioni, consulta Parametri personalizzati nella Guida per gli sviluppatori di AWS IoT Core.

- Per i dettagli sull'API, consulta DescribeCustomMetric AWS CLICommand Reference.

`describe-default-authorizer`

Il seguente esempio di codice mostra come utilizzare`describe-default-authorizer`.

AWS CLI

Come ottenere informazioni sul sistema di autorizzazione personalizzato predefinito

L'esempio `describe-default-authorizer` seguente visualizza i dettagli del sistema di autorizzazione personalizzato predefinito.

```
aws iot describe-default-authorizer
```

Output:

```
{
    "authorizerName": "CustomAuthorizer",
    "authorizerArn": "arn:aws:iot:us-west-2:123456789012:authorizer/
CustomAuthorizer"
}
```

Per ulteriori informazioni, consulta la DescribeDefaultAuthorizersezione AWS IoT API Reference.

- Per i dettagli sull'API, consulta DescribeDefaultAuthorizer AWS CLICommand Reference.

`describe-dimension`

Il seguente esempio di codice mostra come utilizzare`describe-dimension`.

AWS CLI

Come ottenere informazioni su una dimensione

L'esempio `describe-dimension` seguente ottiene informazioni su una dimensione denominata `TopicFilterForAuthMessages`.

```
aws iot describe-dimension \
    --name TopicFilterForAuthMessages
```

Output:

```
{
    "name": "TopicFilterForAuthMessages",
```

```
    "arn": "arn:aws:iot:eu-west-2:123456789012:dimension/
TopicFilterForAuthMessages",
    "type": "TOPIC_FILTER",
    "stringValues": [
        "device/+/auth"
    ],
    "creationDate": 1578620223.255,
    "lastModifiedDate": 1578620223.255
}
```

Per ulteriori informazioni, consulta Comandi di rilevamento nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta DescribeDimension AWS CLICommand Reference.

`describe-domain-configuration`

Il seguente esempio di codice mostra come utilizzare`describe-domain-configuration`.

AWS CLI

Come descrivere una configurazione di dominio

L'esempio `describe-domain-configuration` seguente visualizza i dettagli sulla configurazione di dominio specificata.

```
aws iot describe-domain-configuration \
    --domain-configuration-name "additionalDataDomain"
```

Output:

```
{
    "domainConfigurationName": "additionalDataDomain",
    "domainConfigurationArn": "arn:aws:iot:us-
east-1:758EXAMPLE143:domainconfiguration/additionalDataDomain/norpw",
    "domainName": "d055exampleed74y71zfd-ats.beta.us-east-1.iot.amazonaws.com",
    "serverCertificates": [],
    "domainConfigurationStatus": "ENABLED",
    "serviceType": "DATA",
    "domainType": "AWS_MANAGED",
    "lastStatusChangeDate": 1601923783.774
```

```
}
```

Per ulteriori informazioni, consulta Endpoint configurabili nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta DescribeDomainConfiguration AWS CLICommand Reference.

`describe-endpoint`

Il seguente esempio di codice mostra come utilizzare`describe-endpoint`.

AWS CLI

Esempio 1: Per ottenere l' AWS endpoint corrente

L'`describe-endpoint`esempio seguente recupera l' AWS endpoint predefinito a cui vengono applicati tutti i comandi.

```
aws iot describe-endpoint
```

Output:

```
{
    "endpointAddress": "abc123defghijk.iot.us-west-2.amazonaws.com"
}
```

Per ulteriori informazioni, consulta DescribeEndpointla AWS IoT Developer Guide.

Esempio 2: come ottenere l'endpoint ATS

L'esempio `describe-endpoint` seguente recupera l'endpoint Amazon Trust Services (ATS).

```
aws iot describe-endpoint \
    --endpoint-type iot:Data-ATS
```

Output:

```
{
    "endpointAddress": "abc123defghijk-ats.iot.us-west-2.amazonaws.com"
}
```

Per ulteriori informazioni, consulta Certificati X.509 e IoT nella AWSAWS IoT Developer Guide.

- Per i dettagli sull'API, consulta AWS CLI Command DescribeEndpointReference.

`describe-event-configurations`

Il seguente esempio di codice mostra come utilizzare`describe-event-configurations`.

AWS CLI

Come mostrare quali tipi di eventi vengono pubblicati

L'esempio `describe-event-configurations` seguente elenca la configurazione che controlla quali eventi vengono generati quando qualcosa viene aggiunto, aggiornato o eliminato.

```
aws iot describe-event-configurations
```

Output:

```
{
    "eventConfigurations": {
        "CA_CERTIFICATE": {
            "Enabled": false
        },
        "CERTIFICATE": {
            "Enabled": false
        },
        "JOB": {
            "Enabled": false
        },
        "JOB_EXECUTION": {
            "Enabled": false
        },
        "POLICY": {
            "Enabled": false
        },
        "THING": {
            "Enabled": false
        },
        "THING_GROUP": {
            "Enabled": false
        },
        "THING_GROUP_HIERARCHY": {
```

```
            "Enabled": false
        },
        "THING_GROUP_MEMBERSHIP": {
            "Enabled": false
        },
        "THING_TYPE": {
            "Enabled": false
        },
        "THING_TYPE_ASSOCIATION": {
            "Enabled": false
        }
    }
}
```

Per ulteriori informazioni, consulta Messaggi di eventi nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta DescribeEventConfigurations AWS CLICommand Reference.

`describe-index`

Il seguente esempio di codice mostra come utilizzare`describe-index`.

AWS CLI

Come recuperare lo stato corrente dell'indice degli oggetti

L'esempio `describe-index` seguente recupera lo stato corrente dell'indice degli oggetti.

```
aws iot describe-index \
    --index-name "AWS_Things"
```

Output:

```
{
    "indexName": "AWS_Things",
    "indexStatus": "ACTIVE",
    "schema": "REGISTRY_AND_SHADOW_AND_CONNECTIVITY_STATUS"
}
```

Per ulteriori informazioni, consulta Gestione dell'indicizzazione degli oggetti nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta DescribeIndex AWS CLICommand Reference.

`describe-job-execution`

Il seguente esempio di codice mostra come utilizzare`describe-job-execution`.

AWS CLI

Come ottenere i dettagli di esecuzione di un processo su un dispositivo

L'esempio `describe-job-execution` seguente ottiene i dettagli di esecuzione del processo specificato.

```
aws iot describe-job-execution \
    --job-id "example-job-01" \
    --thing-name "MyRaspberryPi"
```

Output:

```
{
    "execution": {
        "jobId": "example-job-01",
        "status": "QUEUED",
        "statusDetails": {},
        "thingArn": "arn:aws:iot:us-west-2:123456789012:thing/MyRaspberryPi",
        "queuedAt": 1560787023.636,
        "lastUpdatedAt": 1560787023.636,
        "executionNumber": 1,
        "versionNumber": 1
    }
}
```

Per ulteriori informazioni, consulta Creazione e gestione di processi (CLI) nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta DescribeJobExecution AWS CLICommand Reference.

`describe-job`

Il seguente esempio di codice mostra come utilizzare`describe-job`.

AWS CLI

Come ottenere lo stato dettagliato di un processo

L'esempio `describe-job` seguente ottiene lo stato dettagliato del progetto con ID `example-job-01`.

```
aws iot describe-job \
    --job-id "example-job-01"
```

Output:

```
{
    "job": {
        "jobArn": "arn:aws:iot:us-west-2:123456789012:job/example-job-01",
        "jobId": "example-job-01",
        "targetSelection": "SNAPSHOT",
        "status": "IN_PROGRESS",
        "targets": [
            "arn:aws:iot:us-west-2:123456789012:thing/MyRaspberryPi"
        ],
        "description": "example job test",
        "presignedUrlConfig": {},
        "jobExecutionsRolloutConfig": {},
        "createdAt": 1560787022.733,
        "lastUpdatedAt": 1560787026.294,
        "jobProcessDetails": {
            "numberOfCanceledThings": 0,
            "numberOfSucceededThings": 0,
            "numberOfFailedThings": 0,
            "numberOfRejectedThings": 0,
            "numberOfQueuedThings": 1,
            "numberOfInProgressThings": 0,
            "numberOfRemovedThings": 0,
            "numberOfTimedOutThings": 0
        },
        "timeoutConfig": {}
    }
}
```

Per ulteriori informazioni, consulta Creazione e gestione di processi (CLI) nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta DescribeJob AWS CLICommand Reference.

`describe-mitigation-action`

Il seguente esempio di codice mostra come utilizzare`describe-mitigation-action`.

AWS CLI

Come visualizzare i dettagli di un'azione di mitigazione definita

L'esempio `describe-mitigation-action` seguente visualizza i dettagli dell'azione di mitigazione specificata.

```
aws iot describe-mitigation-action \
    --action-name AddThingsToQuarantineGroupAction
```

Output:

```
{
    "actionName": "AddThingsToQuarantineGroupAction",
    "actionType": "ADD_THINGS_TO_THING_GROUP",
    "actionArn": "arn:aws:iot:us-west-2:123456789012:mitigationaction/
AddThingsToQuarantineGroupAction",
    "actionId": "2fd2726d-98e1-4abf-b10f-09465ccd6bfa",
    "roleArn": "arn:aws:iam::123456789012:role/service-role/
MoveThingsToQuarantineGroupRole",
    "actionParams": {
        "addThingsToThingGroupParams": {
            "thingGroupNames": [
                "QuarantineGroup1"
            ],
            "overrideDynamicGroups": true
        }
    },
    "creationDate": "2019-12-10T11:09:35.999000-08:00",
    "lastModifiedDate": "2019-12-10T11:09:35.999000-08:00"
}
```

Per ulteriori informazioni, consulta DescribeMitigationAction (Mitigation Action Commands) nella AWS IoT Developer Guide.

- Per i dettagli sull'API, consulta DescribeMitigationAction AWS CLICommand Reference.

`describe-provisioning-template-version`

Il seguente esempio di codice mostra come utilizzare`describe-provisioning-template-version`.

AWS CLI

Come descrivere una versione del modello di provisioning

L'esempio `describe-provisioning-template-version` seguente descrive una versione del modello di provisioning.

```
aws iot describe-provisioning-template-version \
    --template-name MyTestProvisioningTemplate \
    --version-id 1
```

Output:

```
{
    "versionId": 1,
    "creationDate": 1589308310.574,
    "templateBody": "{
        \"Parameters\":{
            \"SerialNumber\":{
                \"Type\":\"String\"
            },
            \"AWS::IoT::Certificate::Id\":{
                \"Type\":\"String\"
            }
        },
        \"Resources\":{
            \"certificate\":{
                \"Properties\":{
                    \"CertificateId\":{
                        \"Ref\":\"AWS::IoT::Certificate::Id\"
                    },
                    \"Status\":\"Active\"
                },
                \"Type\":\"AWS::IoT::Certificate\"
            },
            \"policy\":{
                \"Properties\":{
```

```
                    \"PolicyName\":\"MyIotPolicy\"
                },
                \"Type\":\"AWS::IoT::Policy\"
            },
            \"thing\":{
                \"OverrideSettings\":{
                    \"AttributePayload\":\"MERGE\",
                    \"ThingGroups\":\"DO_NOTHING\",
                    \"ThingTypeName\":\"REPLACE\"
                },
                \"Properties\":{
                    \"AttributePayload\":{},
                    \"ThingGroups\":[],
                    \"ThingName\":{
                        \"Fn::Join\":[
                            \"\",
                            [
                                \"DemoGroup_\",
                                {\"Ref\":\"SerialNumber\"}
                            ]
                        ]
                    },
                    \"ThingTypeName\":\"VirtualThings\"
                },
                \"Type\":\"AWS::IoT::Thing\"
            }
        }
    }",
    "isDefaultVersion": true
}
```

Per ulteriori informazioni, consulta Provisioning di dispositivi che non dispongono di certificati dispositivo mediante il provisioning del parco istanze dispositivi nella Guida per gli sviluppatori di AWS IoT Core.

- Per i dettagli sull'API, consulta DescribeProvisioningTemplateVersion AWS CLICommand Reference.

`describe-provisioning-template`

Il seguente esempio di codice mostra come utilizzare`describe-provisioning-template`.

AWS CLI

Come descrivere un modello di provisioning

L'esempio `describe-provisioning-template` seguente descrive un modello di provisioning.

```
aws iot describe-provisioning-template \
    --template-name MyTestProvisioningTemplate
```

Output:

```
{
    "templateArn": "arn:aws:iot:us-west-2:57EXAMPLE833:provisioningtemplate/
MyTestProvisioningTemplate",
    "templateName": "MyTestProvisioningTemplate",
    "creationDate": 1589308310.574,
    "lastModifiedDate": 1589308345.539,
    "defaultVersionId": 1,
    "templateBody": "{
        \"Parameters\":{
            \"SerialNumber\":{
                \"Type\":\"String\"
            },
            \"AWS::IoT::Certificate::Id\":{
                \"Type\":\"String\"
            }
        },
        \"Resources\":{
            \"certificate\":{
                \"Properties\":{
                    \"CertificateId\":{
                        \"Ref\":\"AWS::IoT::Certificate::Id\"
                    },
                    \"Status\":\"Active\"
                },
                \"Type\":\"AWS::IoT::Certificate\"
            },
            \"policy\":{
                \"Properties\":{
                    \"PolicyName\":\"MyIotPolicy\"
                },
                \"Type\":\"AWS::IoT::Policy\"
            },
```

```
            \"thing\":{
                \"OverrideSettings\":{
                    \"AttributePayload\":\"MERGE\",
                    \"ThingGroups\":\"DO_NOTHING\",
                    \"ThingTypeName\":\"REPLACE\"
                },
                \"Properties\":{
                    \"AttributePayload\":{},
                    \"ThingGroups\":[],
                    \"ThingName\":{
                        \"Fn::Join\":[
                            \"\",
                            [
                                \"DemoGroup_\",
                                {\"Ref\":\"SerialNumber\"}
                            ]
                        ]
                    },
                    \"ThingTypeName\":\"VirtualThings\"
                },
                \"Type\":\"AWS::IoT::Thing\"
            }
        }
    }",
    "enabled": true,
    "provisioningRoleArn": "arn:aws:iam::571032923833:role/service-role/IoT_access"
}
```

Per ulteriori informazioni, consulta Provisioning di dispositivi che non dispongono di certificati dispositivo mediante il provisioning del parco istanze dispositivi nella Guida per gli sviluppatori di AWS IoT Core.

- Per i dettagli sull'API, consulta DescribeProvisioningTemplate AWS CLICommand Reference.

`describe-role-alias`

Il seguente esempio di codice mostra come utilizzare`describe-role-alias`.

AWS CLI

Per ottenere informazioni su un alias di ruolo AWS IoT

L'esempio `describe-role-alias` seguente visualizza i dettagli dell'alias di ruolo specificato.

```
aws iot describe-role-alias \
    --role-alias LightBulbRole
```

Output:

```
{
    "roleAliasDescription": {
        "roleAlias": "LightBulbRole",
        "roleAliasArn": "arn:aws:iot:us-west-2:123456789012:rolealias/
LightBulbRole",
        "roleArn": "arn:aws:iam::123456789012:role/light_bulb_role_001",
        "owner": "123456789012",
        "credentialDurationSeconds": 3600,
        "creationDate": 1570558643.221,
        "lastModifiedDate": 1570558643.221
    }
}
```

Per ulteriori informazioni, consulta la DescribeRoleAliassezione AWS IoT API Reference.

- Per i dettagli sull'API, consulta DescribeRoleAlias AWS CLICommand Reference.

`describe-scheduled-audit`

Il seguente esempio di codice mostra come utilizzare`describe-scheduled-audit`.

AWS CLI

Come ottenere informazioni su un audit pianificato

L'`describe-scheduled-audit`esempio seguente ottiene informazioni dettagliate su un audit pianificato di AWS IOT Device Defender denominato`AWSIoTDeviceDefenderDailyAudit`.

```
aws iot describe-scheduled-audit \
    --scheduled-audit-name AWSIoTDeviceDefenderDailyAudit
```

Output:

```
{
    "frequency": "DAILY",
```

```
    "targetCheckNames": [
        "AUTHENTICATED_COGNITO_ROLE_OVERLY_PERMISSIVE_CHECK",
        "CONFLICTING_CLIENT_IDS_CHECK",
        "DEVICE_CERTIFICATE_SHARED_CHECK",
        "IOT_POLICY_OVERLY_PERMISSIVE_CHECK",
        "REVOKED_CA_CERTIFICATE_STILL_ACTIVE_CHECK",
        "UNAUTHENTICATED_COGNITO_ROLE_OVERLY_PERMISSIVE_CHECK"
    ],
    "scheduledAuditName": "AWSIoTDeviceDefenderDailyAudit",
    "scheduledAuditArn": "arn:aws:iot:us-west-2:123456789012:scheduledaudit/
AWSIoTDeviceDefenderDailyAudit"
}
```

Per ulteriori informazioni, consulta Comandi di auditing nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta DescribeScheduledAudit AWS CLICommand Reference.

`describe-security-profile`

Il seguente esempio di codice mostra come utilizzare`describe-security-profile`.

AWS CLI

Come ottenere informazioni su un profilo di sicurezza

L'`describe-security-profile`esempio seguente ottiene informazioni sul profilo di sicurezza AWS IoT Device Defender denominato `PossibleIssue`.

```
aws iot describe-security-profile \
    --security-profile-name PossibleIssue
```

Output:

```
{
    "securityProfileName": "PossibleIssue",
    "securityProfileArn": "arn:aws:iot:us-west-2:123456789012:securityprofile/
PossibleIssue",
    "securityProfileDescription": "check to see if authorization fails 10 times in 5
 minutes or if cellular bandwidth exceeds 128",
    "behaviors": [
        {
            "name": "CellularBandwidth",
```

```
            "metric": "aws:message-byte-size",
            "criteria": {
                "comparisonOperator": "greater-than",
                "value": {
                    "count": 128
                },
                "consecutiveDatapointsToAlarm": 1,
                "consecutiveDatapointsToClear": 1
            }
        },
        {
            "name": "Authorization",
            "metric": "aws:num-authorization-failures",
            "criteria": {
                "comparisonOperator": "greater-than",
                "value": {
                    "count": 10
                },
                "durationSeconds": 300,
                "consecutiveDatapointsToAlarm": 1,
                "consecutiveDatapointsToClear": 1
            }
        }
    ],
    "version": 1,
    "creationDate": 1560278102.528,
    "lastModifiedDate": 1560278102.528
}
```

Per ulteriori informazioni, consulta Comandi di rilevamento nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta DescribeSecurityProfile AWS CLICommand Reference.

`describe-stream`

Il seguente esempio di codice mostra come utilizzare`describe-stream`.

AWS CLI

Come ottenere informazioni su un flusso

L'esempio `describe-stream` seguente mostra i dettagli sul flusso specificato.

```
aws iot describe-stream \
    --stream-id stream12345
```

Output:

```
{
    "streamInfo": {
        "streamId": "stream12345",
        "streamArn": "arn:aws:iot:us-west-2:123456789012:stream/stream12345",
        "streamVersion": 1,
        "description": "This stream is used for Amazon FreeRTOS OTA Update 12345.",
        "files": [
            {
                "fileId": "123",
                "s3Location": {
                    "bucket":"codesign-ota-bucket",
                    "key":"48c67f3c-63bb-4f92-a98a-4ee0fbc2bef6"
                }
            }
        ],
        "createdAt": 1557863215.995,
        "lastUpdatedAt": 1557863215.995,
        "roleArn": "arn:aws:iam:123456789012:role/service-role/my_ota_stream_role"
    }
}
```

Per ulteriori informazioni, consulta la DescribeStreamsezione AWS IoT API Reference.

- Per i dettagli sull'API, consulta DescribeStream AWS CLICommand Reference.

`describe-thing-group`

Il seguente esempio di codice mostra come utilizzare`describe-thing-group`.

AWS CLI

Come ottenere informazioni su un gruppo di oggetti

L'esempio `describe-thing-group` seguente ottiene informazioni sul gruppo di oggetti denominato `HalogenBulbs`.

```
aws iot describe-thing-group \
```

```
    --thing-group-name HalogenBulbs
```

Output:

```
{
    "thingGroupName": "HalogenBulbs",
    "thingGroupId": "f4ec6b84-b42b-499d-9ce1-4dbd4d4f6f6e",
    "thingGroupArn": "arn:aws:iot:us-west-2:123456789012:thinggroup/HalogenBulbs",
    "version": 1,
    "thingGroupProperties": {},
    "thingGroupMetadata": {
        "parentGroupName": "LightBulbs",
        "rootToParentThingGroups": [
            {
                "groupName": "LightBulbs",
                "groupArn": "arn:aws:iot:us-west-2:123456789012:thinggroup/
LightBulbs"
            }
        ],
        "creationDate": 1559927609.897
    }
}
```

Per ulteriori informazioni, consulta Gruppi di oggetti nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta DescribeThingGroup AWS CLICommand Reference.

`describe-thing-type`

Il seguente esempio di codice mostra come utilizzare`describe-thing-type`.

AWS CLI

Come ottenere informazioni su un tipo di oggetto

L'`describe-thing-type`esempio seguente mostra informazioni sul tipo di oggetto specificato definito nell' AWS account.

```
aws iot describe-thing-type \
    --thing-type-name "LightBulb"
```

Output:

```
{
    "thingTypeName": "LightBulb",
    "thingTypeId": "ce3573b0-0a3c-45a7-ac93-4e0ce14cd190",
    "thingTypeArn": "arn:aws:iot:us-west-2:123456789012:thingtype/LightBulb",
    "thingTypeProperties": {
        "thingTypeDescription": "light bulb type",
        "searchableAttributes": [
            "model",
            "wattage"
        ]
    },
    "thingTypeMetadata": {
        "deprecated": false,
        "creationDate": 1559772562.498
    }
}
```

Per ulteriori informazioni, consulta Tipi di oggetti nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta DescribeThingType AWS CLICommand Reference.

`describe-thing`

Il seguente esempio di codice mostra come utilizzare`describe-thing`.

AWS CLI

Come visualizzare informazioni dettagliate su un oggetto

L'`describe-thing`esempio seguente mostra informazioni su un oggetto (dispositivo) definito nel registro AWS IoT per l' AWS account.

aws iot describe-thing --thing-name "» MyLightBulb

Output:

```
{
    "defaultClientId": "MyLightBulb",
    "thingName": "MyLightBulb",
    "thingId": "40da2e73-c6af-406e-b415-15acae538797",
    "thingArn": "arn:aws:iot:us-west-2:123456789012:thing/MyLightBulb",
    "thingTypeName": "LightBulb",
```

```
    "attributes": {
        "model": "123",
        "wattage": "75"
    },
    "version": 1
}
```

Per ulteriori informazioni, consulta Come gestire gli oggetti con il registro nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta Command Reference. DescribeThingAWS CLI

`detach-policy`

Il seguente esempio di codice mostra come utilizzare`detach-policy`.

AWS CLI

Esempio 1: scollegare una policy AWS IoT da un gruppo di oggetti

L'esempio `detach-policy` seguente scollega la policy specificata da un gruppo di oggetti e, per estensione, da tutti gli oggetti presenti nel gruppo e in eventuali gruppi secondari.

```
aws iot detach-policy \
    --target "arn:aws:iot:us-west-2:123456789012:thinggroup/LightBulbs" \
    --policy-name "MyFirstGroup_Core-policy"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Gruppi di oggetti nella Guida per gli sviluppatori di AWS IoT.

Esempio 2: scollegare una policy AWS IoT da un certificato del dispositivo

L'`detach-policy`esempio seguente separa la TemperatureSensorPolicy policy da un certificato di dispositivo identificato da ARN.

```
aws iot detach-policy \
    --policy-name TemperatureSensorPolicy \
    --target arn:aws:iot:us-
west-2:123456789012:cert/488b6a7f2acdeb00a77384e63c4e40b18b1b3caaae57b7272ba44c45e3448142
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, vedere DetachPolicyin AWS CLI Command Reference.

`detach-security-profile`

Il seguente esempio di codice mostra come utilizzare`detach-security-profile`.

AWS CLI

Come rimuovere l'associazione di un profilo di sicurezza da una destinazione

L'`detach-security-profile`esempio seguente rimuove l'associazione tra il profilo di sicurezza AWS IoT Device Defender denominato `Testprofile` e l'oggetto All Registered Things.

```
aws iot detach-security-profile \
    --security-profile-name Testprofile \
    --security-profile-target-arn "arn:aws:iot:us-west-2:123456789012:all/
registered-things"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Comandi di rilevamento nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta DetachSecurityProfile AWS CLICommand Reference.

`detach-thing-principal`

Il seguente esempio di codice mostra come utilizzare`detach-thing-principal`.

AWS CLI

Staccare un oggetto certificate/principal da una cosa

L'esempio `detach-thing-principal` seguente rimuove un certificato che rappresenta un principale dall'oggetto specificato.

```
aws iot detach-thing-principal \
    --thing-name "MyLightBulb" \
```

```
    --principal "arn:aws:iot:us-
west-2:123456789012:cert/604c48437a57b7d5fc5d137c5be75011c6ee67c9a6943683a1acb4b1626bac36"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Come gestire gli oggetti con il registro nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta DetachThingPrincipal AWS CLICommand Reference.

`disable-topic-rule`

Il seguente esempio di codice mostra come utilizzare`disable-topic-rule`.

AWS CLI

Come disabilitare una regola dell'argomento

L'esempio `disable-topic-rule` seguente disabilita la regola dell'argomento specificata.

```
aws iot disable-topic-rule \
    --rule-name "MyPlantPiMoistureAlertRule"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Visualizzazione delle regole nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta DisableTopicRule AWS CLICommand Reference.

`enable-topic-rule`

Il seguente esempio di codice mostra come utilizzare`enable-topic-rule`.

AWS CLI

Come abilitare una regola dell'argomento

L'esempio `enable-topic-rule` seguente abilita (o riabilita) la regola dell'argomento specificata.

```
aws iot enable-topic-rule \
```

```
    --rule-name "MyPlantPiMoistureAlertRule"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Visualizzazione delle regole nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta EnableTopicRule AWS CLICommand Reference.

`get-behavior-model-training-summaries`

Il seguente esempio di codice mostra come utilizzare`get-behavior-model-training-summaries`.

AWS CLI

Come elencare lo stato di addestramento dei modelli di un profilo Device Defender ML Detect

L'esempio `get-behavior-model-training-summaries` seguente elenca lo stato di addestramento dei modelli per i comportamenti configurati nel profilo di sicurezza scelto. Per ogni comportamento vengono elencati il nome, lo stato del modello e la percentuale di punti dati raccolti.

```
aws iot get-behavior-model-training-summaries \
    --security-profile-name MySecuirtyProfileName
```

Output:

```
{
    "summaries": [
        {
            "securityProfileName": "MySecuirtyProfileName",
            "behaviorName": "Messages_sent_ML_behavior",
            "modelStatus": "PENDING_BUILD",
            "datapointsCollectionPercentage": 0.0
        },
        {
            "securityProfileName": "MySecuirtyProfileName",
            "behaviorName": "Messages_received_ML_behavior",
            "modelStatus": "PENDING_BUILD",
            "datapointsCollectionPercentage": 0.0
        },
```

```
        {
            "securityProfileName": "MySecuirtyProfileName",
            "behaviorName": "Authorization_failures_ML_behavior",
            "modelStatus": "PENDING_BUILD",
            "datapointsCollectionPercentage": 0.0
        },
        {
            "securityProfileName": "MySecuirtyProfileName",
            "behaviorName": "Message_size_ML_behavior",
            "modelStatus": "PENDING_BUILD",
            "datapointsCollectionPercentage": 0.0
        },
        {
            "securityProfileName": "MySecuirtyProfileName",
            "behaviorName": "Connection_attempts_ML_behavior",
            "modelStatus": "PENDING_BUILD",
            "datapointsCollectionPercentage": 0.0
        },
        {
            "securityProfileName": "MySPNoALerts",
            "behaviorName": "Disconnects_ML_behavior",
            "modelStatus": "PENDING_BUILD",
            "datapointsCollectionPercentage": 0.0
        }
    ]
}
```

Per ulteriori informazioni, consulta GetBehaviorModelTrainingSummaries (Detect Commands) nella AWS IoT Developer Guide.

- Per i dettagli sull'API, consulta GetBehaviorModelTrainingSummaries AWS CLICommand Reference.

`get-cardinality`

Il seguente esempio di codice mostra come utilizzare`get-cardinality`.

AWS CLI

Come restituire il conteggio approssimativo dei valori univoci che corrispondono alla query

È possibile utilizzare il seguente script di configurazione per creare 10 oggetti che rappresentano 10 sensori di temperatura. Ogni nuovo oggetto contiene 3 attributi.

```
# Bash script. If in other shells, type `bash` before running
Temperatures=(70 71 72 73 74 75 47 97 98 99)
Racks=(Rack1 Rack1 Rack2 Rack2 Rack3 Rack4 Rack5 Rack6 Rack6 Rack6)
IsNormal=(true true true true true true false false false false)
for ((i=0; i<10 ; i++))
do
  thing=$(aws iot create-thing --thing-name "TempSensor$i" --attribute-payload
 attributes="{temperature=${Temperatures[i]},rackId=${Racks[i]},stateNormal=
${IsNormal[i]}}")
  aws iot describe-thing --thing-name "TempSensor$i"
done
```

Output di esempio dello script di configurazione:

```
{
    "version": 1,
    "thingName": "TempSensor0",
    "defaultClientId": "TempSensor0",
    "attributes": {
        "rackId": "Rack1",
        "stateNormal": "true",
        "temperature": "70"
    },
    "thingArn": "arn:aws:iot:us-east-1:123456789012:thing/TempSensor0",
    "thingId": "example1-90ab-cdef-fedc-ba987example"
}
```

L'esempio `get-cardinality` seguente esegue una query sui 10 sensori creati dallo script di configurazione e restituisce il numero di rack con sensori di temperatura che segnalano valori di temperatura anomali. Se il valore della temperatura è inferiore a 60 o superiore a 80, il sensore di temperatura si trova in uno stato anomalo.

```
aws iot get-cardinality \
    --aggregation-field "attributes.rackId" \
    --query-string "thingName:TempSensor* AND attributes.stateNormal:false"
```

Output:

```
{
    "cardinality": 2
}
```

Per ulteriori informazioni, consulta Querying for Aggregate Data< https://docs.aws.amazon.com/iot/ latest/developerguide/index -aggregate.html> nella IoT Developer Guide.AWS

- Per i dettagli sull'API, consulta GetCardinalityCommand Reference.AWS CLI

`get-effective-policies`

Il seguente esempio di codice mostra come utilizzare`get-effective-policies`.

AWS CLI

Come elencare le policy che influiscono su un oggetto

L'esempio `get-effective-policies` seguente elenca le policy che influiscono sull'oggetto specificato, incluse le policy associate a tutti i gruppi a cui appartiene.

```
aws iot get-effective-policies \
    --thing-name TemperatureSensor-001 \
    --principal arn:aws:iot:us-
west-2:123456789012:cert/488b6a7f2acdeb00a77384e63c4e40b18b1b3caaae57b7272ba44c45e3448142
```

Output:

```
{
    "effectivePolicies": [
        {
            "policyName": "TemperatureSensorPolicy",
            "policyArn": "arn:aws:iot:us-west-2:123456789012:policy/
TemperatureSensorPolicy",
            "policyDocument": "{
                \"Version\": \"2012-10-17\",
                \"Statement\": [
                    {
                        \"Effect\": \"Allow\",
                        \"Action\": [
                            \"iot:Publish\",
                            \"iot:Receive\"
                        ],
                        \"Resource\": [
                            \"arn:aws:iot:us-west-2:123456789012:topic/topic_1\",
                            \"arn:aws:iot:us-west-2:123456789012:topic/topic_2\"
                        ]
                    },
```

```
                    {
                        \"Effect\": \"Allow\",
                        \"Action\": [
                            \"iot:Subscribe\"
                        ],
                        \"Resource\": [
                            \"arn:aws:iot:us-west-2:123456789012:topicfilter/
topic_1\",
                            \"arn:aws:iot:us-west-2:123456789012:topicfilter/
topic_2\"
                        ]
                    },
                    {
                        \"Effect\": \"Allow\",
                        \"Action\": [
                            \"iot:Connect\"
                        ],
                        \"Resource\": [
                            \"arn:aws:iot:us-west-2:123456789012:client/basicPubSub
\"
                        ]
                    }
                ]
            }"
        }
    ]
}
```

Per ulteriori informazioni, consulta Recupero delle policy valide per un oggetto nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta GetEffectivePolicies AWS CLICommand Reference.

`get-indexing-configuration`

Il seguente esempio di codice mostra come utilizzare`get-indexing-configuration`.

AWS CLI

Come ottenere la configurazione di indicizzazione degli oggetti

L'`get-indexing-configuration`esempio seguente ottiene i dati di configurazione correnti per l'indicizzazione della flotta AWS IoT.

```
aws iot get-indexing-configuration
```

Output:

```
{
    "thingIndexingConfiguration": {
        "thingIndexingMode": "OFF",
        "thingConnectivityIndexingMode": "OFF"
    },
    "thingGroupIndexingConfiguration": {
        "thingGroupIndexingMode": "OFF"
    }
}
```

Per ulteriori informazioni, consulta Gestione dell'indicizzazione degli oggetti nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta AWS CLI Command GetIndexingConfigurationReference.

`get-job-document`

Il seguente esempio di codice mostra come utilizzare`get-job-document`.

AWS CLI

Come recuperare il documento per un processo

L'esempio `get-job-document` seguente visualizza i dettagli sul documento relativo al processo con ID `example-job-01`.

```
aws iot get-job-document \
    --job-id "example-job-01"
```

Output:

```
{
    "document": "\n{\n    \"operation\":\"customJob\",\n    \"otherInfo\":
\"someValue\"\n}\n"
}
```

Per ulteriori informazioni, consulta Creazione e gestione di processi (CLI) nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta GetJobDocument AWS CLICommand Reference.

`get-logging-options`

Il seguente esempio di codice mostra come utilizzare`get-logging-options`.

AWS CLI

Come ottenere le opzioni di registrazione di log

L'`get-logging-options`esempio seguente ottiene le opzioni di registrazione correnti per l' AWS account.

```
aws iot get-logging-options
```

Output:

```
{
    "roleArn": "arn:aws:iam::123456789012:role/service-role/iotLoggingRole",
    "logLevel": "ERROR"
}
```

Per ulteriori informazioni, consulta titolo nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta GetLoggingOptions AWS CLICommand Reference.

`get-ota-update`

Il seguente esempio di codice mostra come utilizzare`get-ota-update`.

AWS CLI

Come recuperare informazioni su un aggiornamento OTA

L'esempio `get-ota-update` seguente mostra i dettagli dell'aggiornamento OTA specificato.

```
aws iot get-ota-update \
```

```
    --ota-update-id ota12345
```

Output:

```
{
    "otaUpdateInfo": {
        "otaUpdateId": "ota12345",
        "otaUpdateArn": "arn:aws:iot:us-west-2:123456789012:otaupdate/itsaupdate",
        "creationDate": 1557863215.995,
        "lastModifiedDate": 1557863215.995,
        "description": "A critical update needed right away.",
        "targets": [
           "device1",
           "device2",
           "device3",
           "device4"
        ],
        "targetSelection": "SNAPSHOT",
        "protocols": ["HTTP"],
        "awsJobExecutionsRolloutConfig": {
           "maximumPerMinute": 10
        },
        "otaUpdateFiles": [
            {
                "fileName": "firmware.bin",
                "fileLocation": {
                    "stream": {
                        "streamId": "004",
                        "fileId":123
                    }
                },
                "codeSigning": {
                    "awsSignerJobId": "48c67f3c-63bb-4f92-a98a-4ee0fbc2bef6"
                }
            }
        ],
        "roleArn": "arn:aws:iam:123456789012:role/service-role/my_ota_role"
        "otaUpdateStatus": "CREATE_COMPLETE",
        "awsIotJobId": "job54321",
        "awsIotJobArn": "arn:aws:iot:us-west-2:123456789012:job/job54321",
        "errorInfo": {
        }
    }
```

```
}
```

Per ulteriori informazioni, consulta Get OTAUpdate in the AWS IoT API Reference.

- Per i dettagli sull'API, consulta GetOtaUpdate AWS CLICommand Reference.

`get-percentiles`

Il seguente esempio di codice mostra come utilizzare`get-percentiles`.

AWS CLI

Come raggruppare i valori aggregati che corrispondono alla query in raggruppamenti percentili

È possibile utilizzare il seguente script di configurazione per creare 10 oggetti che rappresentano 10 sensori di temperatura. Ogni nuovo oggetto contiene 1 attributo.

```
# Bash script. If in other shells, type `bash` before running
Temperatures=(70 71 72 73 74 75 47 97 98 99)
for ((i=0; i<10 ; i++))
do
    thing=$(aws iot create-thing --thing-name "TempSensor$i" --attribute-payload
 attributes="{temperature=${Temperatures[i]}}")
    aws iot describe-thing --thing-name "TempSensor$i"
done
```

Output di esempio dello script di configurazione:

```
{
    "version": 1,
    "thingName": "TempSensor0",
    "defaultClientId": "TempSensor0",
    "attributes": {
        "temperature": "70"
    },
    "thingArn": "arn:aws:iot:us-east-1:123456789012:thing/TempSensor0",
    "thingId": "example1-90ab-cdef-fedc-ba987example"
}
```

L'esempio `get-percentiles` seguente esegue una query sui 10 sensori creati dallo script di configurazione e restituisce un valore per ogni gruppo percentile specificato. Il gruppo percentile

"10" contiene il valore del campo aggregato che si verifica in circa il 10% dei valori corrispondenti alla query. Nell'output seguente, {"percent": 10.0, "value": 67.7} significa che circa il 10,0% dei valori di temperatura è inferiore a 67,7.

```
aws iot get-percentiles \
    --aggregation-field "attributes.temperature" \
    --query-string "thingName:TempSensor*" \
    --percents 10 25 50 75 90
```

Output:

```
{
    "percentiles": [
        {
            "percent": 10.0,
            "value": 67.7
        },
        {
            "percent": 25.0,
            "value": 71.25
        },
        {
            "percent": 50.0,
            "value": 73.5
        },
        {
            "percent": 75.0,
            "value": 91.5
        },
        {
            "percent": 90.0,
            "value": 98.1
        }
    ]
}
```

Per ulteriori informazioni, consulta Interrogazione di dati aggregati nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta GetPercentiles AWS CLICommand Reference.

`get-policy-version`

Il seguente esempio di codice mostra come utilizzare`get-policy-version`.

AWS CLI

Come ottenere informazioni su una versione specifica di una policy

L'esempio `get-policy-version` seguente ottiene informazioni sulla prima versione della policy specificata.

```
aws iot get-policy \
    --policy-name UpdateDeviceCertPolicy
    --policy-version-id "1"
```

Output:

```
{
    "policyArn": "arn:aws:iot:us-west-2:123456789012:policy/UpdateDeviceCertPolicy",
    "policyName": "UpdateDeviceCertPolicy",
    "policyDocument": "{ \"Version\": \"2012-10-17\", \"Statement\": [ { \"Effect\":
 \"Allow\", \"Action\":  \"iot:UpdateCertificate\", \"Resource\": \"*\" } ] }",
    "policyVersionId": "1",
    "isDefaultVersion": false,
    "creationDate": 1559925941.924,
    "lastModifiedDate": 1559926175.458,
    "generationId":
 "5066f1b6712ce9d2a1e56399771649a272d6a921762fead080e24fe52f24e042"
}
```

Per ulteriori informazioni, consulta Policy AWS IoT nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta GetPolicyVersion AWS CLICommand Reference.

`get-policy`

Il seguente esempio di codice mostra come utilizzare`get-policy`.

AWS CLI

Come ottenere informazioni sulla versione predefinita di una policy

L'esempio `get-policy` seguente recupera informazioni sulla versione predefinita della policy specificata.

```
aws iot get-policy \
    --policy-name UpdateDeviceCertPolicy
```

Output:

```
{
    "policyName": "UpdateDeviceCertPolicy",
    "policyArn": "arn:aws:iot:us-west-2:123456789012:policy/UpdateDeviceCertPolicy",
    "policyDocument": "{ \"Version\": \"2012-10-17\", \"Statement\": [ { \"Effect\":
 \"Allow\", \"Action\":  \"iot:UpdateCertificate\", \"Resource\": \"*\" } ] }",
    "defaultVersionId": "2",
    "creationDate": 1559925941.924,
    "lastModifiedDate": 1559925941.924,
    "generationId":
 "5066f1b6712ce9d2a1e56399771649a272d6a921762fead080e24fe52f24e042"
}
```

Per ulteriori informazioni, consulta Policy AWS IoT nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta GetPolicy AWS CLICommand Reference.

`get-registration-code`

Il seguente esempio di codice mostra come utilizzare`get-registration-code`.

AWS CLI

Per ottenere il codice di registrazione AWS specifico per l'account

L'`get-registration-code`esempio seguente recupera il codice di registrazione specifico dell' AWS account.

```
aws iot get-registration-code
```

Output:

```
{
```

```
    "registrationCode":
 "15c51ae5e36ba59ba77042df1115862076bea4bd15841c838fcb68d5010a614c"
}
```

Per ulteriori informazioni, consulta Uso di un certificato personale nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta Command GetRegistrationCodeReference AWS CLI .

`get-statistics`

Il seguente esempio di codice mostra come utilizzare`get-statistics`.

AWS CLI

Come cercare dati aggregati nell'indice del dispositivo

L'esempio `get-statistics` seguente restituisce il numero di elementi con una proprietà denominata `connectivity.connected` impostata su `false` (ovvero il numero di dispositivi non connessi) nel device shadow.

```
aws iot get-statistics \
    --index-name AWS_Things \
    --query-string "connectivity.connected:false"
```

Output:

```
{
    "statistics": {
        "count": 6
    }
}
```

Per ulteriori informazioni, consulta Ottenimento di statistiche sul parco istanze del dispositivo nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta GetStatistics AWS CLICommand Reference.

`get-topic-rule-destination`

Il seguente esempio di codice mostra come utilizzare`get-topic-rule-destination`.

AWS CLI

Come ottenere la destinazione di una regola dell'argomento

L'esempio `get-topic-rule-destination` seguente ottiene informazioni sulla destinazione di una regola dell'argomento.

```
aws iot get-topic-rule-destination \
    --arn "arn:aws:iot:us-west-2:123456789012:ruledestination/http/
a1b2c3d4-5678-90ab-cdef-11111EXAMPLE"
```

Output:

```
{
    "topicRuleDestination": {
        "arn": "arn:aws:iot:us-west-2:123456789012:ruledestination/http/
a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
        "status": "DISABLED",
        "httpUrlProperties": {
            "confirmationUrl": "https://example.com"
        }
    }
}
```

Per ulteriori informazioni, consulta Utilizzo delle destinazioni delle regole degli argomenti nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta GetTopicRuleDestination AWS CLICommand Reference.

`get-topic-rule`

Il seguente esempio di codice mostra come utilizzare`get-topic-rule`.

AWS CLI

Come ottenere informazioni su una regola

L'esempio `get-topic-rule` seguente ottiene informazioni sulla regola specificata.

```
aws iot get-topic-rule \
    --rule-name MyRPiLowMoistureAlertRule
```

Output:

```
{
    "ruleArn": "arn:aws:iot:us-west-2:123456789012:rule/MyRPiLowMoistureAlertRule",
    "rule": {
        "ruleName": "MyRPiLowMoistureAlertRule",
        "sql": "SELECT * FROM '$aws/things/MyRPi/shadow/update/accepted' WHERE
 state.reported.moisture = 'low'\n                    ",
        "description": "Sends an alert whenever soil moisture level readings are too
 low.",
        "createdAt": 1558624363.0,
        "actions": [
            {
                "sns": {
                    "targetArn": "arn:aws:sns:us-
west-2:123456789012:MyRPiLowMoistureTopic",
                    "roleArn": "arn:aws:iam::123456789012:role/service-role/
MyRPiLowMoistureTopicRole",
                    "messageFormat": "RAW"
                }
            }
        ],
        "ruleDisabled": false,
        "awsIotSqlVersion": "2016-03-23"
    }
}
```

Per ulteriori informazioni, consulta Visualizzazione delle regole nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta GetTopicRule AWS CLICommand Reference.

`get-v2-logging-options`

Il seguente esempio di codice mostra come utilizzare`get-v2-logging-options`.

AWS CLI

Come elencare le opzioni di registrazione di log correnti

L'`get-v2-logging-options`esempio seguente elenca le opzioni di registrazione correnti per AWS IoT.

```
aws iot get-v2-logging-options
```

Output:

```
{
    "roleArn": "arn:aws:iam::094249569039:role/service-role/iotLoggingRole",
    "defaultLogLevel": "WARN",
    "disableAllLogs": false
}
```

Per ulteriori informazioni, consulta titolo nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta GetV2 LoggingOptions in AWS CLI Command Reference.

`list-active-violations`

Il seguente esempio di codice mostra come utilizzare. `list-active-violations`

AWS CLI

Come elencare le violazioni attive

L'esempio `list-active-violations` seguente elenca tutte le violazioni per il profilo di sicurezza specificato.

```
aws iot list-active-violations \
    --security-profile-name Testprofile
```

Output:

```
{
    "activeViolations": [
        {
            "violationId": "174db59167fa474c80a652ad1583fd44",
            "thingName": "iotconsole-1560269126751-1",
            "securityProfileName": "Testprofile",
            "behavior": {
                "name": "Authorization",
                "metric": "aws:num-authorization-failures",
                "criteria": {
                    "comparisonOperator": "greater-than",
                    "value": {
```

```
                    "count": 10
                },
                "durationSeconds": 300,
                "consecutiveDatapointsToAlarm": 1,
                "consecutiveDatapointsToClear": 1
            }
        },
        "lastViolationValue": {
            "count": 0
        },
        "lastViolationTime": 1560293700.0,
        "violationStartTime": 1560279000.0
    },
    {
        "violationId": "c8a9466a093d3b7b35cd44ca58bdbeab",
        "thingName": "TvnQoEoU",
        "securityProfileName": "Testprofile",
        "behavior": {
            "name": "CellularBandwidth",
            "metric": "aws:message-byte-size",
            "criteria": {
                "comparisonOperator": "greater-than",
                "value": {
                    "count": 128
                },
                "consecutiveDatapointsToAlarm": 1,
                "consecutiveDatapointsToClear": 1
            }
        },
        "lastViolationValue": {
            "count": 110
        },
        "lastViolationTime": 1560369000.0,
        "violationStartTime": 1560276600.0
    },
    {
        "violationId": "74aa393adea02e6648f3ac362beed55e",
        "thingName": "iotconsole-1560269232412-2",
        "securityProfileName": "Testprofile",
        "behavior": {
            "name": "Authorization",
            "metric": "aws:num-authorization-failures",
            "criteria": {
                "comparisonOperator": "greater-than",
```

```
                    "value": {
                        "count": 10
                    },
                    "durationSeconds": 300,
                    "consecutiveDatapointsToAlarm": 1,
                    "consecutiveDatapointsToClear": 1
                }
            },
            "lastViolationValue": {
                "count": 0
            },
            "lastViolationTime": 1560276600.0,
            "violationStartTime": 1560276600.0
        },
        {
            "violationId": "1e6ab5f7cf39a1466fcd154e1377e406",
            "thingName": "TvnQoEoU",
            "securityProfileName": "Testprofile",
            "behavior": {
                "name": "Authorization",
                "metric": "aws:num-authorization-failures",
                "criteria": {
                    "comparisonOperator": "greater-than",
                    "value": {
                        "count": 10
                    },
                    "durationSeconds": 300,
                    "consecutiveDatapointsToAlarm": 1,
                    "consecutiveDatapointsToClear": 1
                }
            },
            "lastViolationValue": {
                "count": 0
            },
            "lastViolationTime": 1560369000.0,
            "violationStartTime": 1560276600.0
        }
    ]
}
```

- Per i dettagli sull'API, consulta ListActiveViolations AWS CLICommand Reference.

`list-attached-policies`

Il seguente esempio di codice mostra come utilizzare`list-attached-policies`.

AWS CLI

Esempio 1: come elencare le policy collegate a un gruppo

L'esempio `list-attached-policies` seguente elenca le policy collegate al gruppo specificato.

```
aws iot list-attached-policies \
    --target "arn:aws:iot:us-west-2:123456789012:thinggroup/LightBulbs"
```

Output:

```
{
    "policies": [
        {
            "policyName": "UpdateDeviceCertPolicy",
            "policyArn": "arn:aws:iot:us-west-2:123456789012:policy/
UpdateDeviceCertPolicy"
        }
    ]
}
```

Per ulteriori informazioni, consulta Gruppi di oggetti nella Guida per gli sviluppatori di AWS IoT.

Esempio 2: come elencare le policy collegate a un certificato di dispositivo

L'`list-attached-policies`esempio seguente elenca le policy AWS IoT allegate al certificato del dispositivo. Il certificato viene identificato mediante il rispettivo ARN.

```
aws iot list-attached-policies \
    --target arn:aws:iot:us-
west-2:123456789012:cert/488b6a7f2acdeb00a77384e63c4e40b18b1b3caaae57b7272ba44c45e3448142
```

Output:

```
{
    "policies": [
        {
            "policyName": "TemperatureSensorPolicy",
```

```
            "policyArn": "arn:aws:iot:us-west-2:123456789012:policy/
TemperatureSensorPolicy"
        }
    ]
}
```

Per ulteriori informazioni, consulta Gruppi di oggetti nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta ListAttachedPolicies AWS CLICommand Reference.

`list-audit-findings`

Il seguente esempio di codice mostra come utilizzare`list-audit-findings`.

AWS CLI

Esempio 1: come elencare tutti gli esiti di un audit

L'`list-audit-findings`esempio seguente elenca tutti i risultati di un audit AWS IoT Device Defender con un ID attività specificato.

```
aws iot list-audit-findings \
    --task-id a3aea009955e501a31b764abe1bebd3d
```

Output:

```
{
    "findings": []
}
```

Esempio 2: come elencare gli esiti per un tipo di controllo di audit

L'`list-audit-findings`esempio seguente mostra i risultati degli audit di AWS IoT Device Defender eseguiti tra il 5 giugno 2019 e il 19 giugno 2019 in cui i dispositivi condividono un certificato del dispositivo. Quando specifichi un nome per il controllo, devi fornire un'ora di inizio e un'ora di fine.

```
aws iot list-audit-findings \
    --check-name DEVICE_CERTIFICATE_SHARED_CHECK \
    --start-time 1559747125 \
    --end-time 1560962028
```

Output:


```
{
    "findings": [
        {
            "taskId": "eeef61068b0eb03c456d746c5a26ee04",
            "checkName": "DEVICE_CERTIFICATE_SHARED_CHECK",
            "taskStartTime": 1560161017.172,
            "findingTime": 1560161017.592,
            "severity": "CRITICAL",
            "nonCompliantResource": {
                "resourceType": "DEVICE_CERTIFICATE",
                "resourceIdentifier": {
                    "deviceCertificateId":
 "b193ab7162c0fadca83246d24fa090300a1236fe58137e121b011804d8ac1d6b"
                }
            },
            "relatedResources": [
                {
                    "resourceType": "CLIENT_ID",
                    "resourceIdentifier": {
                        "clientId": "ZipxgAIl"
                    },
                    "additionalInfo": {
                        "CONNECTION_TIME": "1560086374068"
                    }
                },
                {
                    "resourceType": "CLIENT_ID",
                    "resourceIdentifier": {
                        "clientId": "ZipxgAIl"
                    },
                    "additionalInfo": {
                        "CONNECTION_TIME": "1560081552187",
                        "DISCONNECTION_TIME": "1560086371552"
                    }
                },
                {
                    "resourceType": "CLIENT_ID",
                    "resourceIdentifier": {
                        "clientId": "ZipxgAIl"
                    },
                    "additionalInfo": {
                        "CONNECTION_TIME": "1559289863631",
```

```
                        "DISCONNECTION_TIME": "1560081532716"
                    }
                }
            ],
            "reasonForNonCompliance": "Certificate shared by one or more devices.",
            "reasonForNonComplianceCode": "CERTIFICATE_SHARED_BY_MULTIPLE_DEVICES"
        },
        {
            "taskId": "bade6b5efd2e1b1569822f6021b39cf5",
            "checkName": "DEVICE_CERTIFICATE_SHARED_CHECK",
            "taskStartTime": 1559988217.27,
            "findingTime": 1559988217.655,
            "severity": "CRITICAL",
            "nonCompliantResource": {
                "resourceType": "DEVICE_CERTIFICATE",
                "resourceIdentifier": {
                    "deviceCertificateId":
"b193ab7162c0fadca83246d24fa090300a1236fe58137e121b011804d8ac1d6b"
                }
            },
            "relatedResources": [
                {
                    "resourceType": "CLIENT_ID",
                    "resourceIdentifier": {
                        "clientId": "xShGENLW"
                    },
                    "additionalInfo": {
                        "CONNECTION_TIME": "1559972350825"
                    }
                },
                {
                    "resourceType": "CLIENT_ID",
                    "resourceIdentifier": {
                        "clientId": "xShGENLW"
                    },
                    "additionalInfo": {
                        "CONNECTION_TIME": "1559255062002",
                        "DISCONNECTION_TIME": "1559972350616"
                    }
                }
            ],
            "reasonForNonCompliance": "Certificate shared by one or more devices.",
            "reasonForNonComplianceCode": "CERTIFICATE_SHARED_BY_MULTIPLE_DEVICES"
        },
```

```
        {
            "taskId": "c23f6233ba2d35879c4bb2810fb5ffd6",
            "checkName": "DEVICE_CERTIFICATE_SHARED_CHECK",
            "taskStartTime": 1559901817.31,
            "findingTime": 1559901817.767,
            "severity": "CRITICAL",
            "nonCompliantResource": {
                "resourceType": "DEVICE_CERTIFICATE",
                "resourceIdentifier": {
                    "deviceCertificateId":
 "b193ab7162c0fadca83246d24fa090300a1236fe58137e121b011804d8ac1d6b"
                }
            },
            "relatedResources": [
                {
                    "resourceType": "CLIENT_ID",
                    "resourceIdentifier": {
                        "clientId": "TvnQoEoU"
                    },
                    "additionalInfo": {
                        "CONNECTION_TIME": "1559826729768"
                    }
                },
                {
                    "resourceType": "CLIENT_ID",
                    "resourceIdentifier": {
                        "clientId": "TvnQoEoU"
                    },
                    "additionalInfo": {
                        "CONNECTION_TIME": "1559345920964",
                        "DISCONNECTION_TIME": "1559826728402"
                    }
                }
            ],
            "reasonForNonCompliance": "Certificate shared by one or more devices.",
            "reasonForNonComplianceCode": "CERTIFICATE_SHARED_BY_MULTIPLE_DEVICES"
        }
    ]
}
```

Per ulteriori informazioni, consulta Comandi di auditing nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta AWS CLI Command ListAuditFindingsReference.

`list-audit-mitigation-actions-executions`

Il seguente esempio di codice mostra come utilizzare`list-audit-mitigation-actions-executions`.

AWS CLI

Come elencare i dettagli dell'esecuzione di un'azione di mitigazione degli audit

Un'attività di mitigazione degli audit applica un'azione di mitigazione a uno o più risultati di un audit di AWS IoT Device Defender. L'esempio `list-audit-mitigation-actions-executions` seguente elenca i dettagli dell'attività di mitigazione con l'azione `taskId` specificata e per gli esiti indicati.

```
aws iot list-audit-mitigation-actions-executions \
    --task-id myActionsTaskId \
    --finding-id 0edbaaec-2fe1-4cf5-abc9-d4c3e51f7464
```

Output:

```
{
    "actionsExecutions": [
        {
            "taskId": "myActionsTaskId",
            "findingId": "0edbaaec-2fe1-4cf5-abc9-d4c3e51f7464",
            "actionName": "ResetPolicyVersionAction",
            "actionId": "1ea0b415-bef1-4a01-bd13-72fb63c59afb",
            "status": "COMPLETED",
            "startTime": "2019-12-10T15:19:13.279000-08:00",
            "endTime": "2019-12-10T15:19:13.337000-08:00"
        }
    ]
}
```

Per ulteriori informazioni, consulta ListAuditMitigationActionsExecutions (Mitigation Action Commands) nella AWS IoT Developer Guide.

- Per i dettagli sull'API, consulta ListAuditMitigationActionsExecutions AWS CLICommand Reference.

`list-audit-mitigation-actions-tasks`

Il seguente esempio di codice mostra come utilizzare`list-audit-mitigation-actions-tasks`.

AWS CLI

Come elencare le attività relative alle azioni di mitigazione degli audit

L'esempio `list-audit-mitigation-actions-tasks` seguente elenca le azioni di mitigazione applicate agli esiti entro il periodo di tempo specificato.

```
aws iot list-audit-mitigation-actions-tasks \
    --start-time 1594157400 \
    --end-time 1594157430
```

Output:

```
{
    "tasks": [
        {
                "taskId": "0062f2d6-3999-488f-88c7-bef005414103",
                "startTime": "2020-07-07T14:30:15.172000-07:00",
            "taskStatus": "COMPLETED"
        }
    ]
}
```

Per ulteriori informazioni, consulta ListAuditMitigationActionsTasks (Mitigation Action Commands) nella AWS IoT Developer Guide.

- Per i dettagli sull'API, consulta ListAuditMitigationActionsTasks AWS CLICommand Reference.

`list-audit-suppressions`

Il seguente esempio di codice mostra come utilizzare`list-audit-suppressions`.

AWS CLI

Come elencare tutte le soppressioni degli esiti degli audit

L'esempio `list-audit-suppressions` seguente elenca tutte le soppressioni degli esiti di audit attive.

```
aws iot list-audit-suppressions
```

Output:

```
{
    "suppressions": [
        {
        "checkName": "DEVICE_CERTIFICATE_EXPIRING_CHECK",
            "resourceIdentifier": {
                "deviceCertificateId": "c7691e<shortened>"
            },
        "expirationDate": 1597881600.0,
        "suppressIndefinitely": false
        }
    ]
}
```

Per ulteriori informazioni, consulta Soppressioni degli esiti degli audit nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta ListAuditSuppressions AWS CLICommand Reference.

`list-audit-tasks`

Il seguente esempio di codice mostra come utilizzare`list-audit-tasks`.

AWS CLI

Come elencare tutti gli esiti di un audit

L'esempio `list-audit-tasks` seguente elenca le attività di audit eseguite tra il 5 giugno 2019 e il 12 giugno 2019.

```
aws iot list-audit-tasks \
    --start-time 1559747125 \
    --end-time 1560357228
```

Output:

```
{
    "tasks": [
        {
```

```
            "taskId": "a3aea009955e501a31b764abe1bebd3d",
            "taskStatus": "COMPLETED",
            "taskType": "ON_DEMAND_AUDIT_TASK"
        },
        {
            "taskId": "f76b4b5102b632cd9ae38a279c266da1",
            "taskStatus": "COMPLETED",
            "taskType": "SCHEDULED_AUDIT_TASK"
        },
        {
            "taskId": "51d9967d9f9ff4d26529505f6d2c444a",
            "taskStatus": "COMPLETED",
            "taskType": "SCHEDULED_AUDIT_TASK"
        },
        {
            "taskId": "eeef61068b0eb03c456d746c5a26ee04",
            "taskStatus": "COMPLETED",
            "taskType": "SCHEDULED_AUDIT_TASK"
        },
        {
            "taskId": "041c49557b7c7b04c079a49514b55589",
            "taskStatus": "COMPLETED",
            "taskType": "SCHEDULED_AUDIT_TASK"
        },
        {
            "taskId": "82c7f2afac1562d18a4560be73998acc",
            "taskStatus": "COMPLETED",
            "taskType": "SCHEDULED_AUDIT_TASK"
        },
        {
            "taskId": "bade6b5efd2e1b1569822f6021b39cf5",
            "taskStatus": "COMPLETED",
            "taskType": "SCHEDULED_AUDIT_TASK"
        },
        {
            "taskId": "c23f6233ba2d35879c4bb2810fb5ffd6",
            "taskStatus": "COMPLETED",
            "taskType": "SCHEDULED_AUDIT_TASK"
        },
        {
            "taskId": "ac9086b7222a2f5e2e17bb6fd30b3aeb",
            "taskStatus": "COMPLETED",
            "taskType": "SCHEDULED_AUDIT_TASK"
        }
```

```
    ]
}
```

Per ulteriori informazioni, consulta Comandi di auditing nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta ListAuditTasks AWS CLICommand Reference.

`list-authorizers`

Il seguente esempio di codice mostra come utilizzare`list-authorizers`.

AWS CLI

Come elencare un sistema di autorizzazione personalizzato

L'`list-authorizers`esempio seguente elenca gli autorizzatori personalizzati del tuo AWS account.

```
aws iot list-authorizers
```

Output:

```
{
    "authorizers": [
        {
            "authorizerName": "CustomAuthorizer",
            "authorizerArn": "arn:aws:iot:us-west-2:123456789012:authorizer/
CustomAuthorizer"
        },
        {
            "authorizerName": "CustomAuthorizer2",
            "authorizerArn": "arn:aws:iot:us-west-2:123456789012:authorizer/
CustomAuthorizer2"
        },
        {
            "authorizerName": "CustomAuthorizer3",
            "authorizerArn": "arn:aws:iot:us-west-2:123456789012:authorizer/
CustomAuthorizer3"
        }
    ]
}
```

Per ulteriori informazioni, consulta la ListAuthorizerssezione AWS IoT API Reference.

- Per i dettagli sull'API, consulta ListAuthorizers AWS CLICommand Reference.

`list-billing-groups`

Il seguente esempio di codice mostra come utilizzare`list-billing-groups`.

AWS CLI

Per elencare i gruppi di fatturazione per il tuo AWS account e la tua regione

L'`list-billing-groups`esempio seguente elenca tutti i gruppi di fatturazione definiti per l' AWS account e AWS la regione.

```
aws iot list-billing-groups
```

Output:

```
{
    "billingGroups": [
        {
            "groupName": "GroupOne",
            "groupArn": "arn:aws:iot:us-west-2:123456789012:billinggroup/GroupOne"
        }
    ]
}
```

Per ulteriori informazioni, consulta Gruppi di fatturazione nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta ListBillingGroups AWS CLICommand Reference.

`list-ca-certificates`

Il seguente esempio di codice mostra come utilizzare`list-ca-certificates`.

AWS CLI

Per elencare i certificati CA registrati nel tuo AWS account

L'`list-ca-certificates`esempio seguente elenca i certificati CA registrati nel tuo AWS account.

```
aws iot list-ca-certificates
```

Output:

```
{
    "certificates": [
        {
            "certificateArn": "arn:aws:iot:us-west-2:123456789012:cacert/
f4efed62c0142f16af278166f61962501165c4f053629520742646005 8cd1467",
            "certificateId":
 "f4efed62c0142f16af278166f61962501165c4f053629520742646005 8cd1467",
            "status": "INACTIVE",
            "creationDate": 1569365372.053
        }
    ]
}
```

Per ulteriori informazioni, consulta Uso di un certificato personale nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta ListCaCertificates AWS CLICommand Reference.

`list-certificates-by-ca`

Il seguente esempio di codice mostra come utilizzare`list-certificates-by-ca`.

AWS CLI

Come elencare tutti i certificati di dispositivo firmati con un certificato CA

L'`list-certificates-by-ca`esempio seguente elenca tutti i certificati dei dispositivi presenti nell' AWS account che sono firmati con il certificato CA specificato.

```
aws iot list-certificates-by-ca \
    --ca-certificate-
id f4efed62c0142f16af278166f61962501165c4f053629520742646005 8cd1467
```

Output:

```
{
    "certificates": [
```

```
        {
            "certificateArn": "arn:aws:iot:us-
west-2:123456789012:cert/488b6a7f2acdeb00a77384e63c4e40b18b1b3caaae57b7272ba44c45e3448142",
            "certificateId":
 "488b6a7f2acdeb00a77384e63c4e40b18b1b3caaae57b7272ba44c45e3448142",
            "status": "ACTIVE",
            "creationDate": 1569363250.557
        }
    ]
}
```

Per ulteriori informazioni, consulta ListCertificatesByCA nell'AWS IoT API Reference.

- Per i dettagli sull'API, consulta ListCertificatesByCa AWS CLICommand Reference.

`list-certificates`

Il seguente esempio di codice mostra come utilizzare`list-certificates`.

AWS CLI

Esempio 1: Per elencare i certificati registrati nel tuo AWS account

L'esempio `list-certificates` seguente elenca tutti i certificati registrati nel tuo account. Se disponi di un limite di paging superiore rispetto al valore predefinito di 25, puoi utilizzare il valore di risposta `nextMarker` di questo comando e fornirlo al comando successivo per ottenere il successivo batch di risultati. Ripeti l'operazione finché `nextMarker` non restituisce alcun valore.

```
aws iot list-certificates
```

Output:

```
{
    "certificates": [
        {
            "certificateArn": "arn:aws:iot:us-
west-2:123456789012:cert/604c48437a57b7d5fc5d137c5be75011c6ee67c9a6943683a1acb4b1626bac36",
            "certificateId":
 "604c48437a57b7d5fc5d137c5be75011c6ee67c9a6943683a1acb4b1626bac36",
            "status": "ACTIVE",
            "creationDate": 1556810537.617
        },
```

```
        {
            "certificateArn": "arn:aws:iot:us-
west-2:123456789012:cert/262a1ac8a7d8aa72f6e96e365480f7313aa9db74b8339ec65d34dc3074e1c31e",
            "certificateId":
 "262a1ac8a7d8aa72f6e96e365480f7313aa9db74b8339ec65d34dc3074e1c31e",
            "status": "ACTIVE",
            "creationDate": 1546447050.885
        },
        {
            "certificateArn": "arn:aws:iot:us-west-2:123456789012:cert/
b193ab7162c0fadca83246d24fa090300a1236fe58137e121b011804d8ac1d6b",
            "certificateId":
 "b193ab7162c0fadca83246d24fa090300a1236fe58137e121b011804d8ac1d6b",
            "status": "ACTIVE",
            "creationDate": 1546292258.322
        },
        {
            "certificateArn": "arn:aws:iot:us-
west-2:123456789012:cert/7aebeea3845d14a44ec80b06b8b78a89f3f8a706974b8b34d18f5adf0741db42",
            "certificateId":
 "7aebeea3845d14a44ec80b06b8b78a89f3f8a706974b8b34d18f5adf0741db42",
            "status": "ACTIVE",
            "creationDate": 1541457693.453
        },
        {
            "certificateArn": "arn:aws:iot:us-
west-2:123456789012:cert/54458aa39ebb3eb39c91ffbbdcc3a6ca1c7c094d1644b889f735a6fc2cd9a7e3",
            "certificateId":
 "54458aa39ebb3eb39c91ffbbdcc3a6ca1c7c094d1644b889f735a6fc2cd9a7e3",
            "status": "ACTIVE",
            "creationDate": 1541113568.611
        },
        {
            "certificateArn": "arn:aws:iot:us-
west-2:123456789012:cert/4f0ba725787aa94d67d2fca420eca022242532e8b3c58e7465c7778b443fd65e",
            "certificateId":
 "4f0ba725787aa94d67d2fca420eca022242532e8b3c58e7465c7778b443fd65e",
            "status": "ACTIVE",
            "creationDate": 1541022751.983
        }
    ]
}
```

- Per i dettagli sull'API, consulta ListCertificates AWS CLICommand Reference.

`list-custom-metrics`

Il seguente esempio di codice mostra come utilizzare`list-custom-metrics`.

AWS CLI

Come elencare le metriche personalizzate

L'esempio `list-custom-metrics` seguente elenca tutte le metriche personalizzate.

```
aws iot list-custom-metrics \
    --region us-east-1
```

Output:

```
{
    "metricNames": [
        "batteryPercentage"
    ]
}
```

Per ulteriori informazioni, consulta Parametri personalizzati nella Guida per gli sviluppatori di AWS IoT Core.

- Per i dettagli sull'API, consulta ListCustomMetrics AWS CLICommand Reference.

`list-dimensions`

Il seguente esempio di codice mostra come utilizzare`list-dimensions`.

AWS CLI

Per elencare le dimensioni del tuo AWS account

L'`list-dimensions`esempio seguente elenca tutte le dimensioni di AWS IoT Device Defender definite nell'account AWS .

```
aws iot list-dimensions
```

Output:

```
{
    "dimensionNames": [
        "TopicFilterForAuthMessages",
        "TopicFilterForActivityMessages"
    ]
}
```

Per ulteriori informazioni, consulta Comandi di rilevamento nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta ListDimensions AWS CLICommand Reference.

`list-domain-configurations`

Il seguente esempio di codice mostra come utilizzare`list-domain-configurations`.

AWS CLI

Come elencare le configurazioni di dominio

L'`list-domain-configurations`esempio seguente elenca le configurazioni di dominio AWS dell'account con il tipo di servizio specificato.

```
aws iot list-domain-configurations \
    --service-type "DATA"
```

Output:

```
{
    "domainConfigurations":
    [
        {
            "domainConfigurationName": "additionalDataDomain",
            "domainConfigurationArn": "arn:aws:iot:us-
west-2:123456789012:domainconfiguration/additionalDataDomain/dikMh",
            "serviceType": "DATA"
        },

        {
            "domainConfigurationName": "iot:Jobs",
            "domainConfigurationArn": "arn:aws:iot:us-
west-2:123456789012:domainconfiguration/iot:Jobs",
```

```
            "serviceType": "JOBS"
        },
        {
            "domainConfigurationName": "iot:Data-ATS",
            "domainConfigurationArn": "arn:aws:iot:us-
west-2:123456789012:domainconfiguration/iot:Data-ATS",
            "serviceType": "DATA"
        },
        {
            "domainConfigurationName": "iot:CredentialProvider",
            "domainConfigurationArn": "arn:aws:iot:us-
west-2:123456789012:domainconfiguration/iot:CredentialProvider",
            "serviceType": "CREDENTIAL_PROVIDER"
        }
    ]
}
```

Per ulteriori informazioni, consulta Endpoint configurabili nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta ListDomainConfigurations AWS CLICommand Reference.

`list-indices`

Il seguente esempio di codice mostra come utilizzare`list-indices`.

AWS CLI

Come elencare gli indici di ricerca configurati

L'`list-indices`esempio seguente elenca tutti gli indici di ricerca configurati nel tuo AWS account. Se non hai abilitato l'indicizzazione degli oggetti, potresti non avere alcun indice.

```
aws iot list-indices
```

Output:

```
{
    "indexNames": [
        "AWS_Things"
    ]
```

```
}
```

Per ulteriori informazioni, consulta Gestione dell'indicizzazione degli oggetti nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta ListIndices AWS CLICommand Reference.

`list-job-executions-for-job`

Il seguente esempio di codice mostra come utilizzare`list-job-executions-for-job`.

AWS CLI

Per elencare le offerte di lavoro presenti nel tuo AWS account

L'`list-job-executions-for-job`esempio seguente elenca tutte le esecuzioni di job per un job nel tuo AWS account, specificato da JobID.

```
aws iot list-job-executions-for-job \
    --job-id my-ota-job
```

Output:

```
{
     "executionSummaries": [
         {
             "thingArn": "arn:aws:iot:us-east-1:123456789012:thing/my_thing",
             "jobExecutionSummary": {
                 "status": "QUEUED",
                 "queuedAt": "2022-03-07T15:58:42.195000-08:00",
                 "lastUpdatedAt": "2022-03-07T15:58:42.195000-08:00",
                 "executionNumber": 1,
                 "retryAttempt": 0
             }
         }
     ]
 }
```

Per ulteriori informazioni, consulta Creazione e gestione di processi (CLI) nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, vedere ListJobExecutionsForJobin AWS CLI Command Reference.

`list-job-executions-for-thing`

Il seguente esempio di codice mostra come utilizzare`list-job-executions-for-thing`.

AWS CLI

Come elencare i processi eseguiti per un oggetto

L'esempio `list-job-executions-for-thing` seguente elenca tutti i processi eseguiti per l'oggetto denominato `MyRaspberryPi`.

```
aws iot list-job-executions-for-thing \
    --thing-name "MyRaspberryPi"
```

Output:

```
{
    "executionSummaries": [
        {
            "jobId": "example-job-01",
            "jobExecutionSummary": {
                "status": "QUEUED",
                "queuedAt": 1560787023.636,
                "lastUpdatedAt": 1560787023.636,
                "executionNumber": 1
            }
        }
    ]
}
```

Per ulteriori informazioni, consulta Creazione e gestione di processi (CLI) nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta ListJobExecutionsForThing AWS CLICommand Reference.

`list-jobs`

Il seguente esempio di codice mostra come utilizzare`list-jobs`.

AWS CLI

Per elencare le offerte di lavoro presenti nel tuo AWS account

L'`list-jobs`esempio seguente elenca tutti i lavori del tuo AWS account, ordinati in base allo stato del lavoro.

```
aws iot list-jobs
```

Output:

```
{
    "jobs": [
        {
            "jobArn": "arn:aws:iot:us-west-2:123456789012:job/example-job-01",
            "jobId": "example-job-01",
            "targetSelection": "SNAPSHOT",
            "status": "IN_PROGRESS",
            "createdAt": 1560787022.733,
            "lastUpdatedAt": 1560787026.294
        }
    ]
}
```

Per ulteriori informazioni, consulta Creazione e gestione di processi (CLI) nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta ListJobs AWS CLICommand Reference.

`list-mitigation-actions`

Il seguente esempio di codice mostra come utilizzare`list-mitigation-actions`.

AWS CLI

Come elencare tutte le azioni di mitigazione definite

L'`list-mitigation-actions`esempio seguente elenca tutte le azioni di mitigazione definite per l' AWS account e la regione. Per ogni azione vengono elencati il nome, l'ARN e la data di creazione.

```
aws iot list-mitigation-actions
```

Output:

```
{
    "actionIdentifiers": [
        {
            "actionName": "DeactivateCACertAction",
            "actionArn": "arn:aws:iot:us-west-2:123456789012:mitigationaction/
DeactivateCACertAction",
            "creationDate": "2019-12-10T11:12:47.574000-08:00"
        },
        {
            "actionName": "ResetPolicyVersionAction",
            "actionArn": "arn:aws:iot:us-west-2:123456789012:mitigationaction/
ResetPolicyVersionAction",
            "creationDate": "2019-12-10T11:11:48.920000-08:00"
        },
        {
            "actionName": "PublishFindingToSNSAction",
            "actionArn": "arn:aws:iot:us-west-2:123456789012:mitigationaction/
PublishFindingToSNSAction",
            "creationDate": "2019-12-10T11:10:49.546000-08:00"
        },
        {
            "actionName": "AddThingsToQuarantineGroupAction",
            "actionArn": "arn:aws:iot:us-west-2:123456789012:mitigationaction/
AddThingsToQuarantineGroupAction",
            "creationDate": "2019-12-10T11:09:35.999000-08:00"
        },
        {
            "actionName": "UpdateDeviceCertAction",
            "actionArn": "arn:aws:iot:us-west-2:123456789012:mitigationaction/
UpdateDeviceCertAction",
            "creationDate": "2019-12-10T11:08:44.263000-08:00"
        },
        {
            "actionName": "SampleMitigationAction",
            "actionArn": "arn:aws:iot:us-west-2:123456789012:mitigationaction/
SampleMitigationAction",
            "creationDate": "2019-12-10T11:03:41.840000-08:00"
        }
    ]
}
```

Per ulteriori informazioni, consulta ListMitigationActions (Mitigation Action Commands) nella AWS IoT Developer Guide.

- Per i dettagli sull'API, consulta ListMitigationActions AWS CLICommand Reference.

`list-mitigations-actions`

Il seguente esempio di codice mostra come utilizzare`list-mitigations-actions`.

AWS CLI

Come elencare tutte le azioni di mitigazione definite

L'`list-mitigations-actions`esempio seguente elenca tutte le azioni di mitigazione definite per l' AWS account e la regione. Per ogni azione vengono elencati il nome, l'ARN e la data di creazione.

```
aws iot list-mitigation-actions
```

Output:

```
{
    "actionIdentifiers": [
        {
            "actionName": "DeactivateCACertAction",
            "actionArn": "arn:aws:iot:us-west-2:123456789012:mitigationaction/
DeactivateCACertAction",
            "creationDate": "2019-12-10T11:12:47.574000-08:00"
        },
        {
            "actionName": "ResetPolicyVersionAction",
            "actionArn": "arn:aws:iot:us-west-2:123456789012:mitigationaction/
ResetPolicyVersionAction",
            "creationDate": "2019-12-10T11:11:48.920000-08:00"
        },
        {
            "actionName": "PublishFindingToSNSAction",
            "actionArn": "arn:aws:iot:us-west-2:123456789012:mitigationaction/
PublishFindingToSNSAction",
            "creationDate": "2019-12-10T11:10:49.546000-08:00"
        },
        {
            "actionName": "AddThingsToQuarantineGroupAction",
            "actionArn": "arn:aws:iot:us-west-2:123456789012:mitigationaction/
AddThingsToQuarantineGroupAction",
```

```
            "creationDate": "2019-12-10T11:09:35.999000-08:00"
        },
        {
            "actionName": "UpdateDeviceCertAction",
            "actionArn": "arn:aws:iot:us-west-2:123456789012:mitigationaction/
UpdateDeviceCertAction",
            "creationDate": "2019-12-10T11:08:44.263000-08:00"
        },
        {
            "actionName": "SampleMitigationAction",
            "actionArn": "arn:aws:iot:us-west-2:123456789012:mitigationaction/
SampleMitigationAction",
            "creationDate": "2019-12-10T11:03:41.840000-08:00"
        }
    ]
}
```

Per ulteriori informazioni, consulta ListMitigationActions (Mitigation Action Commands) nella AWS IoT Developer Guide.

- Per i dettagli sull'API, consulta ListMitigationsActions AWS CLICommand Reference.

`list-ota-updates`

Il seguente esempio di codice mostra come utilizzare`list-ota-updates`.

AWS CLI

Come elencare gli aggiornamenti OTA per l'account

L'esempio `list-ota-updates` seguente elenca gli aggiornamenti OTA disponibili.

```
aws iot list-ota-updates
```

Output:

```
{
    "otaUpdates": [
        {
            "otaUpdateId": "itsaupdate",
            "otaUpdateArn": "arn:aws:iot:us-west-2:123456789012:otaupdate/
itsaupdate",
```

```
            "creationDate": 1557863215.995
        }
    ]
}
```

Per ulteriori informazioni, consulta List OTAUpdates in AWS IoT API Reference.

- Per i dettagli sull'API, consulta ListOtaUpdates AWS CLICommand Reference.

`list-outgoing-certificates`

Il seguente esempio di codice mostra come utilizzare`list-outgoing-certificates`.

AWS CLI

Per elencare i certificati trasferiti su un altro AWS account

L'`list-outgoing-certificates`esempio seguente elenca tutti i certificati dei dispositivi che stanno per essere trasferiti su un altro AWS account utilizzando il `transfer-certificate` comando.

```
aws iot list-outgoing-certificates
```

Output:

```
{
    "outgoingCertificates": [
        {
            "certificateArn": "arn:aws:iot:us-
west-2:030714055129:cert/488b6a7f2acdeb00a77384e63c4e40b18b1b3caaae57b7272ba44c45e3448142",
            "certificateId":
 "488b6a7f2acdeb00a77384e63c4e40b18b1b3caaae57b7272ba44c45e3448142",
            "transferredTo": "030714055129",
            "transferDate": 1569427780.441,
            "creationDate": 1569363250.557
        }
    ]
}
```

Per ulteriori informazioni, consulta la ListOutgoingCertificatessezione AWS IoT API Reference.

- Per i dettagli sull'API, consulta ListOutgoingCertificates AWS CLICommand Reference.

`list-policies`

Il seguente esempio di codice mostra come utilizzare`list-policies`.

AWS CLI

Per elencare le politiche definite nel tuo AWS account

L'`list-policies`esempio seguente elenca tutte le politiche definite nel tuo AWS account.

```
aws iot list-policies
```

Output:

```
{
    "policies": [
        {
            "policyName": "UpdateDeviceCertPolicy",
            "policyArn": "arn:aws:iot:us-west-2:123456789012:policy/
UpdateDeviceCertPolicy"
        },
        {
            "policyName": "PlantIoTPolicy",
            "policyArn": "arn:aws:iot:us-west-2:123456789012:policy/PlantIoTPolicy"
        },
        {
            "policyName": "MyPiGroup_Core-policy",
            "policyArn": "arn:aws:iot:us-west-2:123456789012:policy/MyPiGroup_Core-
policy"
        }
    ]
}
```

Per ulteriori informazioni, consulta Policy AWS IoT nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta ListPolicies AWS CLICommand Reference.

`list-policy-versions`

Il seguente esempio di codice mostra come utilizzare`list-policy-versions`.

AWS CLI

Esempio 1: come visualizzare tutte le versioni di una policy

L'esempio `list-policy-versions` seguente elenca tutte le versioni della policy specificata e le relative date di creazione.

```
aws iot list-policy-versions \
    --policy-name LightBulbPolicy
```

Output:

```
{
    "policyVersions": [
        {
            "versionId": "2",
            "isDefaultVersion": true,
            "createDate": 1559925941.924
        },
        {
            "versionId": "1",
            "isDefaultVersion": false,
            "createDate": 1559925941.924
        }
    ]
}
```

Per ulteriori informazioni, consulta Policy AWS IoT nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta ListPolicyVersions AWS CLICommand Reference.

`list-principal-things`

Il seguente esempio di codice mostra come utilizzare`list-principal-things`.

AWS CLI

Come elencare gli oggetti collegati a un principale

L'esempio `list-principal-things` seguente elenca gli oggetti collegati al principale specificato da un ARN.

```
aws iot list-principal-things \
    --principal arn:aws:iot:us-
west-2:123456789012:cert/2e1eb273792174ec2b9bf4e9b37e6c6c692345499506002a351597670552 78e8
```

Output:

```
{
    "things": [
        "DeskLamp",
        "TableLamp"
    ]
}
```

Per ulteriori informazioni, consulta la ListPrincipalThingssezione AWS IoT API Reference.

- Per i dettagli sull'API, consulta ListPrincipalThings AWS CLICommand Reference.

`list-provisioning-template-versions`

Il seguente esempio di codice mostra come utilizzare`list-provisioning-template-versions`.

AWS CLI

Come elencare le versioni di un modello di provisioning

L'esempio `list-provisioning-template-versions` seguente elenca le versioni disponibili del modello di provisioning specificato.

```
aws iot list-provisioning-template-versions \
    --template-name "widget-template"
```

Output:

```
{
    "versions": [
        {
            "versionId": 1,
            "creationDate": 1574800471.339,
            "isDefaultVersion": true
        },
```

```
        {
            "versionId": 2,
            "creationDate": 1574801192.317,
            "isDefaultVersion": false
        }
    ]
}
```

Per ulteriori informazioni, consulta Tunnel sicuro di AWS IoT nella Guida per gli sviluppatori di AWS IoT Core.

- Per i dettagli sull'API, consulta ListProvisioningTemplateVersions AWS CLICommand Reference.

`list-provisioning-templates`

Il seguente esempio di codice mostra come utilizzare`list-provisioning-templates`.

AWS CLI

Come elencare i modelli di provisioning

L'`list-provisioning-templates`esempio seguente elenca tutti i modelli di provisioning presenti nell' AWS account.

```
aws iot list-provisioning-templates
```

Output:

```
{
    "templates": [
        {
            "templateArn": "arn:aws:iot:us-east-1:123456789012:provisioningtemplate/
widget-template",
            "templateName": "widget-template",
            "description": "A provisioning template for widgets",
            "creationDate": 1574800471.367,
            "lastModifiedDate": 1574801192.324,
            "enabled": false
        }
    ]
```

```
}
```

Per ulteriori informazioni, consulta Tunnel sicuro di AWS IoT nella Guida per gli sviluppatori di AWS IoT Core.

- Per i dettagli sull'API, consulta ListProvisioningTemplates AWS CLICommand Reference.

`list-role-aliases`

Il seguente esempio di codice mostra come utilizzare`list-role-aliases`.

AWS CLI

Per elencare gli alias dei ruoli AWS IoT nel tuo account AWS

L'`list-role-aliases`esempio seguente elenca gli alias dei ruoli AWS IoT nel tuo AWS account.

```
aws iot list-role-aliases
```

Output:

```
{
    "roleAliases": [
        "ResidentAlias",
        "ElectricianAlias"
    ]
}
```

Per ulteriori informazioni, consulta la ListRoleAliasessezione AWS IoT API Reference.

- Per i dettagli sull'API, consulta ListRoleAliases AWS CLICommand Reference.

`list-scheduled-audits`

Il seguente esempio di codice mostra come utilizzare`list-scheduled-audits`.

AWS CLI

Per elencare gli audit programmati per il tuo account AWS

L'`list-scheduled-audits`esempio seguente elenca tutti gli audit programmati per il tuo AWS account.

```
aws iot list-scheduled-audits
```

Output:

```
{
    "scheduledAudits": [
        {
            "scheduledAuditName": "AWSIoTDeviceDefenderDailyAudit",
            "scheduledAuditArn": "arn:aws:iot:us-west-2:123456789012:scheduledaudit/
AWSIoTDeviceDefenderDailyAudit",
            "frequency": "DAILY"
        },
        {
            "scheduledAuditName": "AWSDeviceDefenderWeeklyAudit",
            "scheduledAuditArn": "arn:aws:iot:us-west-2:123456789012:scheduledaudit/
AWSDeviceDefenderWeeklyAudit",
            "frequency": "WEEKLY",
            "dayOfWeek": "SUN"
        }
    ]
}
```

Per ulteriori informazioni, consulta Comandi di auditing nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta ListScheduledAudits AWS CLICommand Reference.

`list-security-profiles-for-target`

Il seguente esempio di codice mostra come utilizzare`list-security-profiles-for-target`.

AWS CLI

Come elencare i profili di sicurezza collegati a una destinazione

L'`list-security-profiles-for-target`esempio seguente elenca i profili di sicurezza AWS IoT Device Defender collegati a dispositivi non registrati.

```
aws iot list-security-profiles-for-target  \
```

```
    --security-profile-target-arn "arn:aws:iot:us-west-2:123456789012:all/
unregistered-things"
```

Output:

```
{
    "securityProfileTargetMappings": [
        {
            "securityProfileIdentifier": {
                "name": "Testprofile",
                "arn": "arn:aws:iot:us-west-2:123456789012:securityprofile/
Testprofile"
            },
            "target": {
                "arn": "arn:aws:iot:us-west-2:123456789012:all/unregistered-things"
            }
        }
    ]
}
```

Per ulteriori informazioni, consulta Comandi di rilevamento nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta AWS CLI Command ListSecurityProfilesForTargetReference.

`list-security-profiles`

Il seguente esempio di codice mostra come utilizzare`list-security-profiles`.

AWS CLI

Per elencare i profili di sicurezza del tuo AWS account

L'`list-security-profiles`esempio seguente elenca tutti i profili di sicurezza AWS IoT Device Defender definiti nell'account AWS .

```
aws iot list-security-profiles
```

Output:

```
{
```

```
    "securityProfileIdentifiers": [
        {
            "name": "Testprofile",
            "arn": "arn:aws:iot:us-west-2:123456789012:securityprofile/Testprofile"
        }
    ]
}
```

Per ulteriori informazioni, consulta Comandi di rilevamento nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta ListSecurityProfiles AWS CLICommand Reference.

`list-streams`

Il seguente esempio di codice mostra come utilizzare`list-streams`.

AWS CLI

Come elencare i flussi nell'account

L'`list-streams`esempio seguente elenca tutti gli stream del tuo AWS account.

```
aws iot list-streams
```

Output:

```
{
    "streams": [
        {
            "streamId": "stream12345",
            "streamArn": "arn:aws:iot:us-west-2:123456789012:stream/stream12345",
            "streamVersion": 1,
            "description": "This stream is used for Amazon FreeRTOS OTA Update
 12345."
        },
        {
            "streamId": "stream54321",
            "streamArn": "arn:aws:iot:us-west-2:123456789012:stream/stream54321",
            "streamVersion": 1,
            "description": "This stream is used for Amazon FreeRTOS OTA Update
 54321."
```

```
        }
    ]
}
```

Per ulteriori informazioni, consulta la ListStreamssezione AWS IoT API Reference.

- Per i dettagli sull'API, consulta ListStreams AWS CLICommand Reference.

`list-tags-for-resource`

Il seguente esempio di codice mostra come utilizzare`list-tags-for-resource`.

AWS CLI

Come visualizzare i tag e i relativi valori associati a una risorsa

L'esempio `list-tags-for-resource` seguente visualizza i tag e i valori associati al gruppo di oggetti `LightBulbs`.

```
aws iot list-tags-for-resource \
    --resource-arn "arn:aws:iot:us-west-2:094249569039:thinggroup/LightBulbs"
```

Output:

```
{
    "tags": [
        {
            "Key": "Assembly",
            "Value": "Fact1NW"
        },
        {
            "Key": "MyTag",
            "Value": "777"
        }
    ]
}
```

Per ulteriori informazioni, consulta Tagging Your AWS IoT Resources nella AWS IoT Developer Guide.

- Per i dettagli sull'API, consulta ListTagsForResource AWS CLICommand Reference.

`list-targets-for-policy`

Il seguente esempio di codice mostra come utilizzare`list-targets-for-policy`.

AWS CLI

Per elencare i principi associati a una policy AWS IoT

L'esempio `list-targets-for-policy` seguente elenca i certificati di dispositivo a cui è collegata la policy specificata.

```
aws iot list-targets-for-policy \
    --policy-name UpdateDeviceCertPolicy
```

Output:

```
{
    "targets": [
        "arn:aws:iot:us-
west-2:123456789012:cert/488b6a7f2acdeb00a77384e63c4e40b18b1b3caaae57b7272ba44c45e3448142",
        "arn:aws:iot:us-west-2:123456789012:cert/
d1eb269fb55a628552143c8f96eb3c258fcd5331ea113e766ba0c82bf225f0be"
    ]
}
```

Per ulteriori informazioni, consulta Gruppi di oggetti nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta ListTargetsForPolicy AWS CLICommand Reference.

`list-targets-for-security-profile`

Il seguente esempio di codice mostra come utilizzare`list-targets-for-security-profile`.

AWS CLI

Come elencare le destinazioni a cui è applicato un profilo di sicurezza

L'`list-targets-for-security-profile`esempio seguente elenca le destinazioni a cui viene applicato il profilo di sicurezza AWS IoT Device Defender denominato`PossibleIssue`.

```
aws iot list-targets-for-security-profile \
```

```
    --security-profile-name Testprofile
```

Output:

```
{
    "securityProfileTargets": [
        {
            "arn": "arn:aws:iot:us-west-2:123456789012:all/unregistered-things"
        },
        {
            "arn": "arn:aws:iot:us-west-2:123456789012:all/registered-things"
        }
    ]
}
```

Per ulteriori informazioni, consulta Comandi di rilevamento nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta ListTargetsForSecurityProfile AWS CLICommand Reference.

`list-thing-groups-for-thing`

Il seguente esempio di codice mostra come utilizzare`list-thing-groups-for-thing`.

AWS CLI

Come visualizzare i gruppi a cui appartiene un oggetto

L'esempio `list-thing-groups-for-thing` seguente elenca i gruppi a cui appartiene l'oggetto specificato.

```
aws iot list-thing-groups-for-thing \
    --thing-name MyLightBulb
```

Output:

```
{
    "thingGroups": [
        {
            "groupName": "DeadBulbs",
            "groupArn": "arn:aws:iot:us-west-2:123456789012:thinggroup/DeadBulbs"
```

```
        },
        {
            "groupName": "LightBulbs",
            "groupArn": "arn:aws:iot:us-west-2:123456789012:thinggroup/LightBulbs"
        }
    ]
}
```

Per ulteriori informazioni, consulta Gruppi di oggetti nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta ListThingGroupsForThing AWS CLICommand Reference.

`list-thing-groups`

Il seguente esempio di codice mostra come utilizzare`list-thing-groups`.

AWS CLI

Per elencare i gruppi di cose definiti nel tuo AWS account

L'`describe-thing-group`esempio seguente elenca tutti i gruppi di cose definiti nell' AWS account.

```
aws iot list-thing-groups
```

Output:

```
{
    "thingGroups": [
        {
            "groupName": "HalogenBulbs",
            "groupArn": "arn:aws:iot:us-west-2:123456789012:thinggroup/HalogenBulbs"
        },
        {
            "groupName": "LightBulbs",
            "groupArn": "arn:aws:iot:us-west-2:123456789012:thinggroup/LightBulbs"
        }
    ]
}
```

Per ulteriori informazioni, consulta Gruppi di oggetti nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta ListThingGroups AWS CLICommand Reference.

`list-thing-principals`

Il seguente esempio di codice mostra come utilizzare`list-thing-principals`.

AWS CLI

Come elencare i principali associati a un oggetto

L'esempio `list-thing-principals` seguente elenca i principali (certificati X.509, utenti IAM, gruppi, ruoli, identità Amazon Cognito o identità federate) associati all'oggetto specificato.

```
aws iot list-thing-principals \
    --thing-name MyRaspberryPi
```

Output:

```
{
    "principals": [
        "arn:aws:iot:us-
west-2:123456789012:cert/33475ac865079a5ffd5ecd44240640349293facc760642d7d8d5dbb6b4c86893"
    ]
}
```

Per ulteriori informazioni, consulta la ListThingPrincipalssezione AWS IoT API Reference.

- Per i dettagli sull'API, consulta ListThingPrincipals AWS CLICommand Reference.

`list-thing-types`

Il seguente esempio di codice mostra come utilizzare`list-thing-types`.

AWS CLI

Come elencare i tipi di oggetti definiti

L'`list-thing-types`esempio seguente visualizza un elenco di tipi di oggetti definiti nell' AWS account.

```
aws iot list-thing-types
```

Output:

```
{
    "thingTypes": [
        {
            "thingTypeName": "LightBulb",
            "thingTypeArn": "arn:aws:iot:us-west-2:123456789012:thingtype/
LightBulb",
            "thingTypeProperties": {
                "thingTypeDescription": "light bulb type",
                "searchableAttributes": [
                    "model",
                    "wattage"
                ]
            },
            "thingTypeMetadata": {
            "deprecated": false,
            "creationDate": 1559772562.498
            }
        }
    ]
}
```

Per ulteriori informazioni, consulta Tipi di oggetti nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta ListThingTypes AWS CLICommand Reference.

`list-things-in-billing-group`

Il seguente esempio di codice mostra come utilizzare`list-things-in-billing-group`.

AWS CLI

Come elencare gli oggetti in un gruppo di fatturazione

L'esempio `list-things-in-billing-group` seguente elenca gli oggetti presenti nel gruppo di fatturazione specificato.

```
aws iot list-things-in-billing-group \
    --billing-group-name GroupOne
```

Output:

```
{
    "things": [
        "MyOtherLightBulb",
        "MyLightBulb"
    ]
}
```

Per ulteriori informazioni, consulta Gruppi di fatturazione nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta ListThingsInBillingGroup AWS CLICommand Reference.

`list-things-in-thing-group`

Il seguente esempio di codice mostra come utilizzare`list-things-in-thing-group`.

AWS CLI

Come elencare gli oggetti che appartengono a un gruppo

L'esempio `list-things-in-thing-group` seguente elenca gli oggetti che appartengono al gruppo di oggetti specificato.

```
aws iot list-things-in-thing-group \
    --thing-group-name LightBulbs
```

Output:

```
{
    "things": [
        "MyLightBulb"
    ]
}
```

Per ulteriori informazioni, consulta Gruppi di oggetti nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta ListThingsInThingGroup AWS CLICommand Reference.

`list-things`

Il seguente esempio di codice mostra come utilizzare`list-things`.

AWS CLI

Esempio 1: come elencare tutti gli oggetti presenti nel registro

L'`list-things`esempio seguente elenca gli oggetti (dispositivi) definiti nel registro AWS IoT per il tuo AWS account.

```
aws iot list-things
```

Output:

```
{
    "things": [
        {
            "thingName": "ThirdBulb",
            "thingTypeName": "LightBulb",
            "thingArn": "arn:aws:iot:us-west-2:123456789012:thing/ThirdBulb",
            "attributes": {
                "model": "123",
                "wattage": "75"
            },
            "version": 2
        },
        {
            "thingName": "MyOtherLightBulb",
            "thingTypeName": "LightBulb",
            "thingArn": "arn:aws:iot:us-west-2:123456789012:thing/MyOtherLightBulb",
            "attributes": {
                "model": "123",
                "wattage": "75"
            },
            "version": 3
        },
        {
            "thingName": "MyLightBulb",
            "thingTypeName": "LightBulb",
            "thingArn": "arn:aws:iot:us-west-2:123456789012:thing/MyLightBulb",
            "attributes": {
                "model": "123",
                "wattage": "75"
            },
            "version": 1
        },
```

```
        {
        "thingName": "SampleIoTThing",
        "thingArn": "arn:aws:iot:us-west-2:123456789012:thing/SampleIoTThing",
        "attributes": {},
        "version": 1
        }
    ]
}
```

Esempio 2: come elencare gli oggetti definiti con un attributo specifico

L'esempio `list-things` seguente visualizza un elenco degli oggetti contenenti un attributo denominato `wattage`.

```
aws iot list-things \
    --attribute-name wattage
```

Output:

```
{
    "things": [
        {
            "thingName": "MyLightBulb",
            "thingTypeName": "LightBulb",
            "thingArn": "arn:aws:iot:us-west-2:123456789012:thing/MyLightBulb",
            "attributes": {
                "model": "123",
                "wattage": "75"
            },
            "version": 1
        },
        {
            "thingName": "MyOtherLightBulb",
            "thingTypeName": "LightBulb",
            "thingArn": "arn:aws:iot:us-west-2:123456789012:thing/MyOtherLightBulb",
            "attributes": {
                "model": "123",
                "wattage": "75"
            },
            "version": 3
        }
    ]
```

```
}
```

Per ulteriori informazioni, consulta Come gestire gli oggetti con il registro nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta ListThings AWS CLICommand Reference.

`list-topic-rule-destinations`

Il seguente esempio di codice mostra come utilizzare`list-topic-rule-destinations`.

AWS CLI

Come elencare le destinazioni delle regole degli argomenti

L'`list-topic-rule-destinations`esempio seguente elenca tutte le destinazioni delle regole tematiche che sono state definite nella AWS regione corrente.

```
aws iot list-topic-rule-destinations
```

Output:

```
{
    "destinationSummaries": [
        {
            "arn": "arn:aws:iot:us-west-2:123456789012:ruledestination/http/
a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
            "status": "ENABLED",
            "httpUrlSummary": {
                "confirmationUrl": "https://example.com"
            }
        }
    ]
}
```

Per ulteriori informazioni, consulta Utilizzo delle destinazioni delle regole degli argomenti nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, vedere ListTopicRuleDestinationsin AWS CLI Command Reference.

`list-topic-rules`

Il seguente esempio di codice mostra come utilizzare`list-topic-rules`.

AWS CLI

Come elencare le regole

L'esempio `list-topic-rules` seguente elenca tutte le regole che hai definito.

```
aws iot list-topic-rules
```

Output:

```
{
    "rules": [
        {
            "ruleArn": "arn:aws:iot:us-west-2:123456789012:rule/
MyRPiLowMoistureAlertRule",
            "ruleName": "MyRPiLowMoistureAlertRule",
            "topicPattern": "$aws/things/MyRPi/shadow/update/accepted",
            "createdAt": 1558624363.0,
            "ruleDisabled": false
        },
        {
            "ruleArn": "arn:aws:iot:us-west-2:123456789012:rule/
MyPlantPiMoistureAlertRule",
            "ruleName": "MyPlantPiMoistureAlertRule",
            "topicPattern": "$aws/things/MyPlantPi/shadow/update/accepted",
            "createdAt": 1541458459.0,
            "ruleDisabled": false
        }
    ]
}
```

Per ulteriori informazioni, consulta Visualizzazione delle regole nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta ListTopicRules AWS CLICommand Reference.

`list-v2-logging-levels`

Il seguente esempio di codice mostra come utilizzare`list-v2-logging-levels`.

AWS CLI

Come elencare i livelli di registrazione di log

L'esempio `list-v2-logging-levels` seguente elenca i livelli di registrazione di log configurati. Se non sono stati impostati livelli di registrazione di log, si verifica un errore `NotConfiguredException` quando esegui questo comando.

```
aws iot list-v2-logging-levels
```

Output:

```
{
    "logTargetConfigurations": [
        {
            "logTarget": {
                "targetType": "DEFAULT"
            },
            "logLevel": "ERROR"
        }
    ]
}
```

- Per i dettagli sull'API, consulta ListV2 LoggingLevels in AWS CLI Command Reference.

`list-violation-events`

Il seguente esempio di codice mostra come utilizzare. `list-violation-events`

AWS CLI

Come elencare le violazioni del profilo di sicurezza durante un periodo di tempo

L'`list-violation-events`esempio seguente elenca le violazioni che si sono verificate tra il 5 giugno 2019 e il 12 giugno 2019 per tutti i profili di sicurezza AWS IoT Device Defender per l' AWS account corrente e AWS la regione.

```
aws iot list-violation-events \
    --start-time 1559747125 \
    --end-time 1560351925
```

Output:


```
{
    "violationEvents": [
        {
            "violationId": "174db59167fa474c80a652ad1583fd44",
            "thingName": "iotconsole-1560269126751-1",
            "securityProfileName": "Testprofile",
            "behavior": {
                "name": "Authorization",
                "metric": "aws:num-authorization-failures",
                "criteria": {
                    "comparisonOperator": "greater-than",
                    "value": {
                        "count": 10
                    },
                    "durationSeconds": 300,
                    "consecutiveDatapointsToAlarm": 1,
                    "consecutiveDatapointsToClear": 1
                }
            },
            "metricValue": {
                "count": 0
            },
            "violationEventType": "in-alarm",
            "violationEventTime": 1560279000.0
        },
        {
            "violationId": "c8a9466a093d3b7b35cd44ca58bdbeab",
            "thingName": "TvnQoEoU",
            "securityProfileName": "Testprofile",
            "behavior": {
                "name": "CellularBandwidth",
                "metric": "aws:message-byte-size",
                "criteria": {
                    "comparisonOperator": "greater-than",
                    "value": {
                        "count": 128
                    },
                    "consecutiveDatapointsToAlarm": 1,
                    "consecutiveDatapointsToClear": 1
                }
            },
            "metricValue": {
```

```
                "count": 110
            },
            "violationEventType": "in-alarm",
            "violationEventTime": 1560276600.0
        },
        {
            "violationId": "74aa393adea02e6648f3ac362beed55e",
            "thingName": "iotconsole-1560269232412-2",
            "securityProfileName": "Testprofile",
            "behavior": {
                "name": "Authorization",
                "metric": "aws:num-authorization-failures",
                "criteria": {
                    "comparisonOperator": "greater-than",
                    "value": {
                        "count": 10
                    },
                    "durationSeconds": 300,
                    "consecutiveDatapointsToAlarm": 1,
                    "consecutiveDatapointsToClear": 1
                }
            },
            "metricValue": {
                "count": 0
            },
            "violationEventType": "in-alarm",
            "violationEventTime": 1560276600.0
        },
        {
            "violationId": "1e6ab5f7cf39a1466fcd154e1377e406",
            "thingName": "TvnQoEoU",
            "securityProfileName": "Testprofile",
            "behavior": {
                "name": "Authorization",
                "metric": "aws:num-authorization-failures",
                "criteria": {
                    "comparisonOperator": "greater-than",
                    "value": {
                        "count": 10
                    },
                    "durationSeconds": 300,
                    "consecutiveDatapointsToAlarm": 1,
                    "consecutiveDatapointsToClear": 1
                }
```

```
            },
            "metricValue": {
                "count": 0
            },
            "violationEventType": "in-alarm",
            "violationEventTime": 1560276600.0
        }
    ]
}
```

Per ulteriori informazioni, consulta Comandi di rilevamento nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta ListViolationEvents AWS CLICommand Reference.

`register-ca-certificate`

Il seguente esempio di codice mostra come utilizzare`register-ca-certificate`.

AWS CLI

Come registrare un certificato di autorità di certificazione (CA)

L'esempio `register-ca-certificate` seguente registra un certificato CA. Il comando fornisce il certificato CA e un certificato di verifica della chiave che dimostra che possiedi la chiave privata associata al certificato CA.

```
aws iot register-ca-certificate \
    --ca-certificate file://rootCA.pem \
    --verification-cert file://verificationCert.pem
```

Output:

```
{
    "certificateArn": "arn:aws:iot:us-west-2:123456789012:cacert/
f4efed62c0142f16af278166f61962501165c4f053629520742646005 8cd1467",
    "certificateId":
 "f4efed62c0142f16af278166f61962501165c4f053629520742646005 8cd1467"
 }
```

Per ulteriori informazioni, consulta Register CACertificate in the AWS IoT API Reference.

- Per i dettagli sull'API, consulta RegisterCaCertificate AWS CLICommand Reference.

`register-certificate`

Il seguente esempio di codice mostra come utilizzare`register-certificate`.

AWS CLI

Come registrare un certificato di dispositivo autofirmato

L'esempio `register-certificate` seguente registra il certificato di dispositivo `deviceCert.pem` firmato dal certificato CA `rootCA.pem`. Il certificato CA deve essere registrato prima di poterlo utilizzare per registrare un certificato di dispositivo autofirmato. Il certificato autofirmato deve essere firmato con lo stesso certificato CA passato a questo comando.

```
aws iot register-certificate \
    --certificate-pem file://deviceCert.pem \
    --ca-certificate-pem file://rootCA.pem
```

Output:

```
{
    "certificateArn": "arn:aws:iot:us-
west-2:123456789012:cert/488b6a7f2acdeb00a77384e63c4e40b18b1b3caaae57b7272ba44c45e3448142",
    "certificateId":
 "488b6a7f2acdeb00a77384e63c4e40b18b1b3caaae57b7272ba44c45e3448142"
 }
```

Per ulteriori informazioni, consulta la RegisterCertificatesezione AWS IoT API Reference.

- Per i dettagli sull'API, consulta RegisterCertificate AWS CLICommand Reference.

`register-thing`

Il seguente esempio di codice mostra come utilizzare`register-thing`.

AWS CLI

Come registrare un oggetto

L'esempio `register-thing` seguente registra un oggetto utilizzando un modello di provisioning.

```
aws iot register-thing \
    --template-body '{"Parameters":{"ThingName":
{"Type":"String"},"AWS::IoT::Certificate::Id":{"Type":"String"}},"Resources":
 {"certificate":{"Properties":{"CertificateId":
{"Ref":"AWS::IoT::Certificate::Id"},"Status":"Active"},"Type":"AWS::IoT::Certificate"},"poli
{"Properties":{"PolicyName":"MyIotPolicy"},"Type":"AWS::IoT::Policy"},"thing":
{"OverrideSettings":
{"AttributePayload":"MERGE","ThingGroups":"DO_NOTHING","ThingTypeName":"REPLACE"},"Propertie
{"AttributePayload":{},"ThingGroups":[],"ThingName":
{"Ref":"ThingName"},"ThingTypeName":"VirtualThings"},"Type":"AWS::IoT::Thing"}}' \
    --parameters '{"ThingName":"Register-thing-
trial-1","AWS::IoT::Certificate::Id":"799a9ea048a1e6aea42b55EXAMPLEf8697b4bafcd77a318a3068e3
```

Output:

```
{
    "certificatePem": "-----BEGIN CERTIFICATE-----
\nMIIDWTCCAkGgAwIBAgIUYLk81I35cIppobpw
HiOJ2jNjboIwDQYJKoZIhvcNAQEL
\nBQAwTTFLMEkGA1UECwxCQW1hem9uIFdlYiBTZXJ2aWNlcyBPPUFtYXpvbi
5jb20g\nSW5jLiBMPVNlYXR0bGUgU1Q9V2FzaGluZ3RvbiBDPVVTMB4XDTIwMDcyMzE2NDUw
\nOVoXDTQ5MTIzMT
IzNTk1OVowHjEcMBoGA1UEAwwTQVdTIElvVCBDZXJ0aWZpY2F0\nZTCCASIwDQYJKoZIhvcNAQEBBQADggEPADCC
AQoCggEBAO71uADhdBajqTmgrMV5\nmCFfBZQRMo1MdtVoZr2X+M4MzL
+RARrtUzH9a2SMAckeX8KeblIOTKzORI
RDXnyE
\n6lVOwjgAsd0ku22rFxex4eG2ikha7pYYkvuToqA7L3TxItRvfKrxRI4ZfJoFPip4\nKqiuBJVNOGKTcQ
Hd1RNOrddwwu6kFJLeKDmEXAMPLEdUF0N+qfR9yKnZQkm
+g6Q2\nGXu7u0W3hn6nlRN8qVoka0uW12p53xM7oHVz
Gf+cxKBxlbOhGkp6yCfTSkUBm3Sp\n9zLw35kiHXVm4EVpwgNlnk6XcIGIkw8a/iy4pzmvuGAANY1/uU/
zgCjymw
ZT5S30\nBV0CAwEAAaNgMF4wHwYDVR0jBBgwFoAUGx0tCcU3q2n1WXAuUCv6hugXjKswHQYD
\nVR0OBBYEFOVtvZ
9Aj2RYFnkX7Iu01XTRUdxgMAwGA1UdEwEB/wQCMAAwDgYDVR0P\nAQH/
BAQDAgeAMA0GCSqGSIb3DQEBCwUAA4IB
AQCXCQcpOtubS5ftOsDMTcpP/jNX
\nDHyArxmjpSc2aCdmm7WX59lTKWyAdxGAvqaDVWqTo0oXI7tZ8w7aINlGi5
pXnifx\n3SBebMUoBbTktrC97yUaeL025mCFv8emDnTR/fE7PTsBKjW0g/rrfpwBxZLXDFwN
\nnqkQjy3EDfifj2
6j0xYIqqWMPogyn4srOCKynS5wMJuQZlHQOnabVwnwK4Y0Mflp
\np9+4susFUR9aT3BT1AcIwqSpzhlKhh4Iz7ND
kRn4amsUT210jg/zOO1Ow+BTHcVQ\nJly8XDu0CWSu04q6SnaBzHmlySIajxuRTP/AdfRouP1OXe
+qlbPOBcvVvF
```

```
8o\n-----END CERTIFICATE-----\n",
    "resourceArns": {
        "certificate": "arn:aws:iot:us-
west-2:571032923833:cert/799a9ea048a1e6aea42b55EXAMPLEf8697b4bafcd77a318a3068e30404b9233c",
        "thing": "arn:aws:iot:us-west-2:571032923833:thing/Register-thing-trial-1"
    }
}
```

Per ulteriori informazioni, consulta Provisioning per utente attendibile nella Guida per gli sviluppatori di AWS IoT Core.

- Per i dettagli sull'API, consulta RegisterThing AWS CLICommand Reference.

`reject-certificate-transfer`

Il seguente esempio di codice mostra come utilizzare`reject-certificate-transfer`.

AWS CLI

Come rifiutare un trasferimento di certificato

L'`reject-certificate-transfer`esempio seguente rifiuta il trasferimento del certificato del dispositivo specificato da un altro AWS account.

```
aws iot reject-certificate-transfer \
    --certificate-
id f0f33678c7c9a046e5cc87b2b1a58dfa0beec26db78addd5e605d630e05c7fc8
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Trasferire un certificato a un altro account nella Guida per gli sviluppatori di AWS IoT Core.

- Per i dettagli sull'API, vedere RejectCertificateTransferin AWS CLI Command Reference.

`remove-thing-from-billing-group`

Il seguente esempio di codice mostra come utilizzare`remove-thing-from-billing-group`.

AWS CLI

Come rimuovere un oggetto da un gruppo di fatturazione

L'esempio `remove-thing-from-billing-group` seguente rimuove l'oggetto specificato da un gruppo di fatturazione.

```
aws iot remove-thing-from-billing-group \
    --billing-group-name GroupOne \
    --thing-name MyOtherLightBulb
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Gruppi di fatturazione nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta RemoveThingFromBillingGroup AWS CLICommand Reference.

`remove-thing-from-thing-group`

Il seguente esempio di codice mostra come utilizzare`remove-thing-from-thing-group`.

AWS CLI

Come rimuovere un oggetto da un gruppo di oggetti

L'esempio `remove-thing-from-thing-group` seguente rimuove l'oggetto specificato da un gruppo di oggetti.

```
aws iot remove-thing-from-thing-group \
    --thing-name bulb7 \
    --thing-group-name DeadBulbs
```

Questo comando non produce alcun output.

Per ulteriori informazioni, vedere Thing Groups < https://docs.aws.amazon.com/iot/ latest/ developerguide/thing -groups.html > nella AWS IoT Developer Guide.

- Per i dettagli sull'API, vedere RemoveThingFromThingGroupin AWS CLI Command Reference.

`replace-topic-rule`

Il seguente esempio di codice mostra come utilizzare`replace-topic-rule`.

AWS CLI

Come aggiornare la definizione di una regola dell'argomento

L'esempio `replace-topic-rule` seguente aggiorna la regola specificata per inviare un avviso SNS quando i valori del livello di umidità del terreno sono troppo bassi.

```
aws iot replace-topic-rule \
    --rule-name MyRPiLowMoistureAlertRule \
    --topic-rule-payload "{\"sql\": \"SELECT * FROM '$aws/things/MyRPi/shadow/
update/accepted' WHERE state.reported.moisture = 'low'\", \"description\": \"Sends
 an alert when soil moisture level readings are too low.\",\"actions\": [{\"sns
\":{\"targetArn\":\"arn:aws:sns:us-west-2:123456789012:MyRPiLowMoistureTopic\",
\"roleArn\":\"arn:aws:iam::123456789012:role/service-role/MyRPiLowMoistureTopicRole
\",\"messageFormat\": \"RAW\"}}],\"ruleDisabled\": false,\"awsIotSqlVersion\":
\"2016-03-23\"}"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Creating an AWS IoT Rule nella AWS IoT Developer Guide.

- Per i dettagli sull'API, consulta ReplaceTopicRule AWS CLICommand Reference.

`search-index`

Il seguente esempio di codice mostra come utilizzare`search-index`.

AWS CLI

Come eseguire una query sull'indice dell'oggetto

L'esempio `search-index` seguente esegue una query sull'indice `AWS_Things` per cercare gli oggetti il cui tipo è `LightBulb`.

```
aws iot search-index \
    --index-name "AWS_Things" \
    --query-string "thingTypeName:LightBulb"
```

Output:

```
{
    "things": [
        {
```

```
            "thingName": "MyLightBulb",
            "thingId": "40da2e73-c6af-406e-b415-15acae538797",
            "thingTypeName": "LightBulb",
            "thingGroupNames": [
                "LightBulbs",
                "DeadBulbs"
            ],
            "attributes": {
                "model": "123",
                "wattage": "75"
            },
            "connectivity": {
                "connected": false
            }
        },
        {
            "thingName": "ThirdBulb",
            "thingId": "615c8455-33d5-40e8-95fd-3ee8b24490af",
            "thingTypeName": "LightBulb",
            "attributes": {
                "model": "123",
                "wattage": "75"
            },
            "connectivity": {
                "connected": false
            }
        },
        {
            "thingName": "MyOtherLightBulb",
            "thingId": "6dae0d3f-40c1-476a-80c4-1ed24ba6aa11",
            "thingTypeName": "LightBulb",
            "attributes": {
                "model": "123",
                "wattage": "75"
            },
            "connectivity": {
                "connected": false
            }
        }
    ]
}
```

Per ulteriori informazioni, consulta Gestione dell'indicizzazione degli oggetti nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta SearchIndex AWS CLICommand Reference.

`set-default-authorizer`

Il seguente esempio di codice mostra come utilizzare`set-default-authorizer`.

AWS CLI

Come impostare un sistema di autorizzazione predefinito

L'esempio `set-default-authorizer` seguente imposta il sistema di autorizzazione personalizzato denominato `CustomAuthorizer` come sistema di autorizzazione predefinito.

```
aws iot set-default-authorizer \
    --authorizer-name CustomAuthorizer
```

Output:

```
{
    "authorizerName": "CustomAuthorizer",
    "authorizerArn": "arn:aws:iot:us-west-2:123456789012:authorizer/
CustomAuthorizer"
}
```

Per ulteriori informazioni, consulta la CreateDefaultAuthorizersezione AWS IoT API Reference.

- Per i dettagli sull'API, consulta SetDefaultAuthorizer AWS CLICommand Reference.

`set-default-policy-version`

Il seguente esempio di codice mostra come utilizzare`set-default-policy-version`.

AWS CLI

Come impostare la versione predefinita di una policy

L'esempio `set-default-policy-version` seguente imposta la versione predefinita su 2 per la policy denominata `UpdateDeviceCertPolicy`.

```
aws iot set-default-policy-version \
    --policy-name UpdateDeviceCertPolicy \
    --policy-version-id 2
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta SetDefaultPolicyVersion AWS CLICommand Reference.

`set-v2-logging-level`

Il seguente esempio di codice mostra come utilizzare`set-v2-logging-level`.

AWS CLI

Come impostare il livello di registrazione di log per un gruppo di oggetti

L'esempio `set-v2-logging-level` seguente imposta il livello di registrazione di log per registrare avvertenze per il gruppo di oggetti specificato.

```
aws iot set-v2-logging-level \
    --log-target "{\"targetType\":\"THING_GROUP\",\"targetName\":\"LightBulbs\"}" \
    --log-level WARN
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta SetV2 LoggingLevel in AWS CLI Command Reference.

`set-v2-logging-options`

Il seguente esempio di codice mostra come utilizzare. `set-v2-logging-options`

AWS CLI

Come impostare le opzioni di registrazione di log

L'esempio `set-v2-logging-options` seguente imposta il livello di dettaglio della registrazione di log predefinito su ERROR e specifica l'ARN da utilizzare per la registrazione.

```
aws iot set-v2-logging-options \
    --default-log-level ERROR \
```

```
    --role-arn "arn:aws:iam::094249569039:role/service-role/iotLoggingRole"
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta SetV2 LoggingOptions in AWS CLI Command Reference.

`start-audit-mitigation-actions-task`

Il seguente esempio di codice mostra come utilizzare. `start-audit-mitigation-actions-task`

AWS CLI

Come applicare un'azione di mitigazione agli esiti di un audit

L'esempio `start-audit-mitigation-actions-task` seguente applica l'azione `ResetPolicyVersionAction` (che cancella la policy) al singolo esito specificato.

```
aws iot start-audit-mitigation-actions-task \
    --task-id "myActionsTaskId" \
    --target "findingIds=[\"0edbaaec-2fe1-4cf5-abc9-d4c3e51f7464\"]" \
    --audit-check-to-actions-mapping
 "IOT_POLICY_OVERLY_PERMISSIVE_CHECK=[\"ResetPolicyVersionAction\"]" \
    --client-request-token "adhadhahda"
```

Output:

```
{
    "taskId": "myActionsTaskId"
}
```

Per ulteriori informazioni, consulta StartAuditMitigationActionsTask (Mitigation Action Commands) nella AWS IoT Developer Guide.

- Per i dettagli sull'API, consulta StartAuditMitigationActionsTask AWS CLICommand Reference.

`start-on-demand-audit-task`

Il seguente esempio di codice mostra come utilizzare`start-on-demand-audit-task`.

AWS CLI

Come avviare immediatamente un audit

L'`start-on-demand-audit-task`esempio seguente avvia un audit AWS IoT Device Defender ed esegue tre controlli dei certificati.

```
aws iot start-on-demand-audit-task \
    --target-check-
names CA_CERTIFICATE_EXPIRING_CHECK DEVICE_CERTIFICATE_EXPIRING_CHECK REVOKED_CA_CERTIFICATE
```

Output:

```
{
    "taskId": "a3aea009955e501a31b764abe1bebd3d"
}
```

Per ulteriori informazioni, consulta Comandi di auditing nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta StartOnDemandAuditTask AWS CLICommand Reference.

`tag-resource`

Il seguente esempio di codice mostra come utilizzare`tag-resource`.

AWS CLI

Come specificare una chiave e un valore di tag per una risorsa

L'esempio `tag-resource` seguente applica il tag con una chiave `Assembly` e il valore `Fact1NW` al gruppo di oggetti `LightBulbs`.

```
aws iot tag-resource \
    --tags Key=Assembly,Value="Fact1NW" \
    --resource-arn "arn:aws:iot:us-west-2:094249569039:thinggroup/LightBulbs"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Tagging Your AWS IoT Resources nella AWS IoT Developer Guide.

- Per i dettagli sull'API, consulta TagResource AWS CLICommand Reference.

`test-authorization`

Il seguente esempio di codice mostra come utilizzare`test-authorization`.

AWS CLI

Per testare le tue politiche AWS IoT

L'`test-authorization`esempio seguente verifica le politiche AWS IoT associate al principale specificato.

```
aws iot test-authorization \
    --auth-infos actionType=CONNECT,resources=arn:aws:iot:us-
east-1:123456789012:client/client1 \
    --principal arn:aws:iot:us-west-2:123456789012:cert/
aab1068f7f43ac3e3cae4b3a8aa3f308d2a750e6350507962e32c1eb465d9775
```

Output:

```
{
    "authResults": [
        {
            "authInfo": {
                "actionType": "CONNECT",
                "resources": [
                    "arn:aws:iot:us-east-1:123456789012:client/client1"
                ]
            },
            "allowed": {
                "policies": [
                    {
                        "policyName": "TestPolicyAllowed",
                        "policyArn": "arn:aws:iot:us-west-2:123456789012:policy/
TestPolicyAllowed"
                    }
                ]
            },
            "denied": {
                "implicitDeny": {
                    "policies": [
                        {
                            "policyName": "TestPolicyDenied",
                            "policyArn": "arn:aws:iot:us-west-2:123456789012:policy/
TestPolicyDenied"
                        }
                    ]
                },
```

```
                "explicitDeny": {
                    "policies": [
                        {
                            "policyName": "TestPolicyExplicitDenied",
                            "policyArn": "arn:aws:iot:us-west-2:123456789012:policy/
TestPolicyExplicitDenied"
                        }
                    ]
                }
            },
            "authDecision": "IMPLICIT_DENY",
            "missingContextValues": []
        }
    ]
}
```

Per ulteriori informazioni, consulta la TestAuthorizationsezione AWS IoT API Reference.

- Per i dettagli sull'API, consulta TestAuthorization AWS CLICommand Reference.

`test-invoke-authorizer`

Il seguente esempio di codice mostra come utilizzare`test-invoke-authorizer`.

AWS CLI

Come testare il sistema di autorizzazione personalizzato

L'esempio `test-invoke-authorizer` seguente testa il sistema di autorizzazione personalizzato.

```
aws iot test-invoke-authorizer \
    --authorizer-name IoTAuthorizer \
    --token allow \
    --token-signature "mE0GvaHqy9nER/
FdgtJX5lXYEJ3b3vE7t1gEszc0TKGgLKWXTnPkb2AbKnOAZ8lGyoN5dVtWDWVmr25m7+
+zjbYIMk2TBvyGXhOmvKFBPkdgyA43KL6SiZy0cTqlPMcQDsP7VX2rXr7CTowCxSNKphGXdQe0/
I5dQ+JO6KUaHwCmupt0/MejKtaNwiia064j6wprOAUwG5S1IYFuRd0X
+wfo8pb0DubAIX1Ua705kuhRUcTx4SxUShEYKmN4IDEvLB6FsIr0B2wvB7y4iPmcajxzGl02ExvyCUNctCV9dYlRRGJj
```

Output:

```
{
```

```
    "isAuthenticated": true,
    "principalId": "principalId",
    "policyDocuments": [
        "{"Version":"2012-10-17",       "Statement":
[{"Action":"iot:Publish","Effect":"Allow","Resource":"arn:aws:iot:us-
west-2:123456789012:topic/customauthtesting"}]}"
    ],
    "refreshAfterInSeconds": 600,
    "disconnectAfterInSeconds": 3600
}
```

Per ulteriori informazioni, consulta la TestInvokeAuthorizersezione AWS IoT API Reference.

- Per i dettagli sull'API, consulta TestInvokeAuthorizer AWS CLICommand Reference.

`transfer-certificate`

Il seguente esempio di codice mostra come utilizzare`transfer-certificate`.

AWS CLI

Per trasferire il certificato di un dispositivo su un altro AWS account

L'`transfer-certificate`esempio seguente trasferisce il certificato di un dispositivo su un altro AWS account. Il certificato e l' AWS account sono identificati da un ID.

```
aws iot transfer-certificate \
    --certificate-
id 488b6a7f2acdeb00a77384e63c4e40b18b1b3caaae57b7272ba44c45e3448142 \
    --target-aws-account 030714055129
```

Output:

```
{
    "transferredCertificateArn": "arn:aws:iot:us-
west-2:030714055129:cert/488b6a7f2acdeb00a77384e63c4e40b18b1b3caaae57b7272ba44c45e3448142"
}
```

Per ulteriori informazioni, consulta Trasferire un certificato a un altro account nella Guida per gli sviluppatori di AWS IoT Core.

- Per i dettagli sull'API, consulta TransferCertificate AWS CLICommand Reference.

`untag-resource`

Il seguente esempio di codice mostra come utilizzare`untag-resource`.

AWS CLI

Come rimuovere una chiave di tag da una risorsa

L'esempio `untag-resource` seguente rimuove il tag `MyTag` e il relativo valore dal gruppo di oggetti `LightBulbs`.

```
command
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Tagging Your AWS IoT Resources nella AWS IoT Developer Guide.

- Per i dettagli sull'API, consulta UntagResource AWS CLICommand Reference.

`update-account-audit-configuration`

Il seguente esempio di codice mostra come utilizzare`update-account-audit-configuration`.

AWS CLI

Esempio 1: come abilitare le notifiche di Amazon SNS per le notifiche di audit

L'`update-account-audit-configuration`esempio seguente abilita le notifiche di Amazon SNS per le notifiche di audit di AWS IoT Device Defender, specificando una destinazione e il ruolo utilizzato per scrivere su tale destinazione.

```
aws iot update-account-audit-configuration \
    --audit-notification-target-configurations "SNS={targetArn=\"arn:aws:sns:us-
west-2:123456789012:ddaudits\",roleArn=\"arn:aws:iam::123456789012:role/service-
role/AWSIoTDeviceDefenderAudit\",enabled=true}"
```

Questo comando non produce alcun output.

Esempio 2: come abilitare un controllo di audit

L'`update-account-audit-configuration`esempio seguente abilita il controllo di controllo AWS IoT Device Defender denominato.

`AUTHENTICATED_COGNITO_ROLE_OVERLY_PERMISSIVE_CHECK` Non è possibile disabilitare un controllo di controllo se fa parte `targetCheckNames` di uno o più audit programmati per l' AWS account.

```
aws iot update-account-audit-configuration \
    --audit-check-configurations
 "{\"AUTHENTICATED_COGNITO_ROLE_OVERLY_PERMISSIVE_CHECK\":{\"enabled\":true}}"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Comandi di auditing nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta UpdateAccountAuditConfiguration AWS CLICommand Reference.

`update-audit-suppression`

Il seguente esempio di codice mostra come utilizzare`update-audit-suppression`.

AWS CLI

Come aggiornare una soppressione degli esiti di audit

L'esempio `update-audit-suppression` seguente aggiorna la data di scadenza della soppressione degli esiti di un audit al 21/09/2020.

```
aws iot update-audit-suppression \
    --check-name DEVICE_CERTIFICATE_EXPIRING_CHECK \
    --resource-identifier deviceCertificateId=c7691e<shortened> \
    --no-suppress-indefinitely \
    --expiration-date 2020-09-21
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Soppressioni degli esiti degli audit nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta UpdateAuditSuppression AWS CLICommand Reference.

`update-authorizer`

Il seguente esempio di codice mostra come utilizzare`update-authorizer`.

AWS CLI

Come aggiornare un sistema di autorizzazione personalizzato

L'esempio `update-authorizer` seguente imposta lo stato di `CustomAuthorizer2` su `INACTIVE`.

```
aws iot update-authorizer \
    --authorizer-name CustomAuthorizer2 \
    --status INACTIVE
```

Output:

```
{
    "authorizerName": "CustomAuthorizer2",
    "authorizerArn": "arn:aws:iot:us-west-2:123456789012:authorizer/
CustomAuthorizer2"
}
```

Per ulteriori informazioni, consulta la UpdateAuthorizersezione AWS IoT API Reference.

- Per i dettagli sull'API, consulta UpdateAuthorizer AWS CLICommand Reference.

`update-billing-group`

Il seguente esempio di codice mostra come utilizzareup`date-billing-group`.

AWS CLI

Come aggiornare le informazioni su un gruppo di fatturazione

L'esempio `update-billing-group` seguente aggiorna la descrizione del gruppo di fatturazione specificato.

```
aws iot update-billing-group \
    --billing-group-name GroupOne \
    --billing-group-properties "billingGroupDescription=\"Primary bulb billing group
\""
```

Output:

```
{
    "version": 2
}
```

Per ulteriori informazioni, consulta Gruppi di fatturazione nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta UpdateBillingGroup AWS CLICommand Reference.

`update-ca-certificate`

Il seguente esempio di codice mostra come utilizzare`update-ca-certificate`.

AWS CLI

Come aggiornare un certificato di autorità di certificazione (CA)

L'esempio `update-ca-certificate` seguente imposta lo stato del certificato CA specificato su ACTIVE.

```
aws iot update-ca-certificate \
    --certificate-
id f4efed62c0142f16af278166f61962501165c4f053629520742646005 8cd1467 \
    --new-status ACTIVE
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Update CACertificate in the AWS IoT API Reference.

- Per i dettagli sull'API, consulta UpdateCaCertificate AWS CLICommand Reference.

`update-certificate`

Il seguente esempio di codice mostra come utilizzare`update-certificate`.

AWS CLI

Come aggiornare un certificato di dispositivo

L'esempio `update-certificate` seguente imposta lo stato del certificato di dispositivo specificato su INACTIVE.

```
aws iot update-certificate \
    --certificate-
id d1eb269fb55a628552143c8f96eb3c258fcd5331ea113e766ba0c82bf225f0be \
    --new-status INACTIVE
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta la UpdateCertificatesezione AWS IoT API Reference.

- Per i dettagli sull'API, consulta UpdateCertificate AWS CLICommand Reference.

`update-custom-metric`

Il seguente esempio di codice mostra come utilizzareu`pdate-custom-metric`.

AWS CLI

Come aggiornare una metrica personalizzata

L'esempio `update-custom-metric` seguente aggiorna una metrica personalizzata per avere un nuovo oggetto `display-name`.

```
aws iot update-custom-metric \
    --metric-name batteryPercentage \
    --display-name 'remaining battery percentage on device' \
    --region us-east-1
```

Output:

```
{
    "metricName": "batteryPercentage",
    "metricArn": "arn:aws:iot:us-east-1:1234564789012:custommetric/
batteryPercentage",
    "metricType": "number",
    "displayName": "remaining battery percentage on device",
    "creationDate": "2020-11-17T23:01:35.110000-08:00",
    "lastModifiedDate": "2020-11-17T23:02:12.879000-08:00"
}
```

Per ulteriori informazioni, consulta Parametri personalizzati nella Guida per gli sviluppatori di AWS IoT Core.

- Per i dettagli sull'API, consulta UpdateCustomMetric AWS CLICommand Reference.

`update-dimension`

Il seguente esempio di codice mostra come utilizzare`update-dimension`.

AWS CLI

Come aggiornare una dimensione

L'esempio `update-dimension` seguente aggiorna una dimensione.

```
aws iot update-dimension \
    --name TopicFilterForAuthMessages \
    --string-values device/${iot:ClientId}/auth
```

Output:

```
{
    "name": "TopicFilterForAuthMessages",
    "lastModifiedDate": 1585866222.317,
    "stringValues": [
        "device/${iot:ClientId}/auth"
    ],
    "creationDate": 1585854500.474,
    "type": "TOPIC_FILTER",
    "arn": "arn:aws:iot:us-west-2:1234564789012:dimension/
TopicFilterForAuthMessages"
}
```

Per ulteriori informazioni, consulta Parametri di ambito nei profili di sicurezza utilizzando le dimensioni nella Guida per gli sviluppatori di AWS IoT Core.

- Per i dettagli sull'API, consulta UpdateDimension AWS CLICommand Reference.

`update-domain-configuration`

Il seguente esempio di codice mostra come utilizzare`update-domain-configuration`.

AWS CLI

Come aggiornare una configurazione di dominio

L'esempio `update-domain-configuration` seguente disabilita la configurazione di dominio specificata.

```
aws iot update-domain-configuration \
    --domain-configuration-name "additionalDataDomain" \
    --domain-configuration-status "DISABLED"
```

Output:

```
{
    "domainConfigurationName": "additionalDataDomain",
    "domainConfigurationArn": "arn:aws:iot:us-
west-2:123456789012:domainconfiguration/additionalDataDomain/dikMh"
}
```

Per ulteriori informazioni, consulta Endpoint configurabili nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta UpdateDomainConfiguration AWS CLICommand Reference.

`update-dynamic-thing-group`

Il seguente esempio di codice mostra come utilizzare`update-dynamic-thing-group`.

AWS CLI

Come aggiornare un gruppo di oggetti dinamico

L'esempio `update-dynamic-thing-group` seguente aggiorna il gruppo di oggetti dinamici specificato. Fornisce una descrizione e aggiorna la stringa di query per modificare i criteri di appartenenza al gruppo.

```
aws iot update-dynamic-thing-group \
    --thing-group-name "RoomTooWarm"
    --thing-group-properties "thingGroupDescription=\"This thing group contains
 rooms warmer than 65F.\"" \
    --query-string "attributes.temperature>65"
```

Output:

```
{
```

```
    "version": 2
}
```

Per ulteriori informazioni, consulta Gruppi di oggetti dinamici nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta UpdateDynamicThingGroup AWS CLICommand Reference.

`update-event-configurations`

Il seguente esempio di codice mostra come utilizzare`update-event-configurations`.

AWS CLI

Come mostrare quali tipi di eventi vengono pubblicati

L'esempio `update-event-configurations` seguente aggiorna la configurazione per abilitare messaggi quando il certificato CA viene aggiunto, aggiornato o eliminato.

```
aws iot update-event-configurations \
    --event-configurations "{\"CA_CERTIFICATE\":{\"Enabled\":true}}"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Messaggi di eventi nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta UpdateEventConfigurations AWS CLICommand Reference.

`update-indexing-configuration`

Il seguente esempio di codice mostra come utilizzare`update-indexing-configuration`.

AWS CLI

Come abilitare l'indicizzazione degli oggetti

L'`update-indexing-configuration`esempio seguente abilita l'indicizzazione degli oggetti per supportare la ricerca nei dati di registro, nei dati shadow e nello stato di connettività degli oggetti utilizzando l'indice AWS_Things.

```
aws iot update-indexing-configuration
```

```
    --thing-indexing-
configuration thingIndexingMode=REGISTRY_AND_SHADOW,thingConnectivityIndexingMode=STATUS
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Gestione dell'indicizzazione degli oggetti nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, vedere UpdateIndexingConfigurationin AWS CLI Command Reference.

update-job

Il seguente esempio di codice mostra come utilizzareupdate-job.

AWS CLI

Come ottenere lo stato dettagliato di un processo

L'esempio update-job seguente ottiene lo stato dettagliato del progetto con ID example-job-01.

```
aws iot describe-job \
    --job-id "example-job-01"
```

Output:

```
{
    "job": {
        "jobArn": "arn:aws:iot:us-west-2:123456789012:job/example-job-01",
        "jobId": "example-job-01",
        "targetSelection": "SNAPSHOT",
        "status": "IN_PROGRESS",
        "targets": [
            "arn:aws:iot:us-west-2:123456789012:thing/MyRaspberryPi"
        ],
        "description": "example job test",
        "presignedUrlConfig": {},
        "jobExecutionsRolloutConfig": {},
        "createdAt": 1560787022.733,
        "lastUpdatedAt": 1560787026.294,
        "jobProcessDetails": {
            "numberOfCanceledThings": 0,
            "numberOfSucceededThings": 0,
```

```
            "numberOfFailedThings": 0,
            "numberOfRejectedThings": 0,
            "numberOfQueuedThings": 1,
            "numberOfInProgressThings": 0,
            "numberOfRemovedThings": 0,
            "numberOfTimedOutThings": 0
        },
        "timeoutConfig": {}
    }
}
```

Per ulteriori informazioni, consulta Creazione e gestione di processi (CLI) nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta UpdateJob AWS CLICommand Reference.

`update-mitigation-action`

Il seguente esempio di codice mostra come utilizzare`update-mitigation-action`.

AWS CLI

Come aggiornare un'azione di mitigazione

L'esempio `update-mitigation-action` seguente aggiorna l'azione di mitigazione specificata denominata `AddThingsToQuarantineGroupAction`, modifica il nome del gruppo di oggetti e imposta `overrideDynamicGroups` su `false`. È possibile verificare le modifiche utilizzando il comando `describe-mitigation-action`.

```
aws iot update-mitigation-action \
    --cli-input-json "{ \"actionName\": \"AddThingsToQuarantineGroupAction\",
 \"actionParams\": { \"addThingsToThingGroupParams\": {\"thingGroupNames\":
[\"QuarantineGroup2\"],\"overrideDynamicGroups\": false}}}"
```

Output:

```
{
    "actionArn": "arn:aws:iot:us-west-2:123456789012:mitigationaction/
AddThingsToQuarantineGroupAction",
    "actionId": "2fd2726d-98e1-4abf-b10f-09465ccd6bfa"
}
```

Per ulteriori informazioni, consulta UpdateMitigationAction (Mitigation Action Commands) nella AWS IoT Developer Guide.

- Per i dettagli sull'API, consulta UpdateMitigationAction AWS CLICommand Reference.

`update-provisioning-template`

Il seguente esempio di codice mostra come utilizzare`update-provisioning-template`.

AWS CLI

Come aggiornare un modello di provisioning

L'esempio `update-provisioning-template` seguente modifica la descrizione e l'ARN del ruolo per il modello di provisioning specificato e abilita il modello.

```
aws iot update-provisioning-template \
    --template-name widget-template \
    --enabled \
    --description "An updated provisioning template for widgets" \
    --provisioning-role-arn arn:aws:iam::504350838278:role/Provision_role
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Tunnel sicuro di AWS IoT nella Guida per gli sviluppatori di AWS IoT Core.

- Per i dettagli sull'API, consulta UpdateProvisioningTemplate AWS CLICommand Reference.

`update-role-alias`

Il seguente esempio di codice mostra come utilizzare`update-role-alias`.

AWS CLI

Come aggiornare un alias di ruolo

L'esempio `update-role-alias` seguente aggiorna l'alias di ruolo `LightBulbRole`.

```
aws iot update-role-alias \
    --role-alias LightBulbRole \
    --role-arn arn:aws:iam::123456789012:role/lightbulbrole-001
```

Output:

```
{
    "roleAlias": "LightBulbRole",
    "roleAliasArn": "arn:aws:iot:us-west-2:123456789012:rolealias/LightBulbRole"
}
```

Per ulteriori informazioni, consulta la UpdateRoleAliassezione AWS IoT API Reference.

- Per i dettagli sull'API, consulta UpdateRoleAlias AWS CLICommand Reference.

`update-scheduled-audit`

Il seguente esempio di codice mostra come utilizzareupdate-scheduled-audit.

AWS CLI

Come aggiornare una definizione di audit pianificato

L'update-scheduled-auditesempio seguente modifica i nomi dei controlli di destinazione per un audit pianificato di AWS IoT Device Defender.

```
aws iot update-scheduled-audit \
    --scheduled-audit-name WednesdayCertCheck \
    --target-check-
names CA_CERTIFICATE_EXPIRING_CHECK DEVICE_CERTIFICATE_EXPIRING_CHECK REVOKED_CA_CERTIFICATE
```

Output:

```
{
    "scheduledAuditArn": "arn:aws:iot:us-west-2:123456789012:scheduledaudit/
WednesdayCertCheck"
}
```

Per ulteriori informazioni, consulta Comandi di auditing nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta UpdateScheduledAudit AWS CLICommand Reference.

`update-security-profile`

Il seguente esempio di codice mostra come utilizzareupdate-security-profile.

AWS CLI

Come modificare un profilo di sicurezza

L'`update-security-profile`esempio seguente aggiorna sia la descrizione che i comportamenti per un profilo di sicurezza AWS IoT Device Defender.

```
aws iot update-security-profile \
    --security-profile-name PossibleIssue \
    --security-profile-description "Check to see if authorization fails 12 times in
 5 minutes or if cellular bandwidth exceeds 128"  \
    --behaviors "[{\"name\":\"CellularBandwidth\",\"metric\":\"aws:message-byte-size
\",\"criteria\":{\"comparisonOperator\":\"greater-than\",\"value\":{\"count\":128},
\"consecutiveDatapointsToAlarm\":1,\"consecutiveDatapointsToClear\":1}},{\"name
\":\"Authorization\",\"metric\":\"aws:num-authorization-failures\",\"criteria\":
{\"comparisonOperator\":\"less-than\",\"value\":{\"count\":12},\"durationSeconds
\":300,\"consecutiveDatapointsToAlarm\":1,\"consecutiveDatapointsToClear\":1}}]"
```

Output:

```
{
    "securityProfileName": "PossibleIssue",
    "securityProfileArn": "arn:aws:iot:us-west-2:123456789012:securityprofile/
PossibleIssue",
    "securityProfileDescription": "check to see if authorization fails 12 times in 5
 minutes or if cellular bandwidth exceeds 128",
    "behaviors": [
        {
            "name": "CellularBandwidth",
            "metric": "aws:message-byte-size",
            "criteria": {
                "comparisonOperator": "greater-than",
                "value": {
                    "count": 128
                },
                "consecutiveDatapointsToAlarm": 1,
                "consecutiveDatapointsToClear": 1
            }
        },
        {
            "name": "Authorization",
            "metric": "aws:num-authorization-failures",
            "criteria": {
```

```
                "comparisonOperator": "less-than",
                "value": {
                    "count": 12
                },
                "durationSeconds": 300,
                "consecutiveDatapointsToAlarm": 1,
                "consecutiveDatapointsToClear": 1
            }
        }
    ],
    "version": 2,
    "creationDate": 1560278102.528,
    "lastModifiedDate": 1560352711.207
}
```

Per ulteriori informazioni, consulta Comandi di rilevamento nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta AWS CLI Command UpdateSecurityProfileReference.

`update-stream`

Il seguente esempio di codice mostra come utilizzare`update-stream`.

AWS CLI

Come aggiornare un flusso

L'esempio `update-stream` seguente aggiorna un flusso esistente. La versione del flusso viene incrementata di 1.

```
aws iot update-stream \
    --cli-input-json file://update-stream.json
```

Contenuto di `update-stream.json`:

```
{
    "streamId": "stream12345",
    "description": "This stream is used for Amazon FreeRTOS OTA Update 12345.",
    "files": [
        {
            "fileId": 123,
```

```
            "s3Location": {
                "bucket":"codesign-ota-bucket",
                "key":"48c67f3c-63bb-4f92-a98a-4ee0fbc2bef6"
            }
        }
    ]
    "roleArn": "arn:aws:iam:us-west-2:123456789012:role/service-role/
my_ota_stream_role"
}
```

Output:

```
{
    "streamId": "stream12345",
    "streamArn": "arn:aws:iot:us-west-2:123456789012:stream/stream12345",
    "description": "This stream is used for Amazon FreeRTOS OTA Update 12345.",
    "streamVersion": 2
}
```

Per ulteriori informazioni, consulta la UpdateStreamsezione AWS IoT API Reference.

- Per i dettagli sull'API, consulta UpdateStream AWS CLICommand Reference.

`update-thing-group`

Il seguente esempio di codice mostra come utilizzareu`pdate-thing-group`.

AWS CLI

Come aggiornare la definizione di un gruppo di oggetti

L'esempio `update-thing-group` seguente aggiorna la definizione per il gruppo di oggetti specificato, modificando la descrizione e due attributi.

```
aws iot update-thing-group \
    --thing-group-name HalogenBulbs \
    --thing-group-properties "thingGroupDescription=\"Halogen bulb group\",
 attributePayload={attributes={Manufacturer=AnyCompany,wattage=60}}"
```

Output:

```
{
```

```
    "version": 2
}
```

Per ulteriori informazioni, consulta Gruppi di oggetti nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta UpdateThingGroup AWS CLICommand Reference.

`update-thing-groups-for-thing`

Il seguente esempio di codice mostra come utilizzare`update-thing-groups-for-thing`.

AWS CLI

Come modificare i gruppi a cui appartiene un oggetto

L'esempio `update-thing-groups-for-thing` seguente rimuove l'oggetto denominato `MyLightBulb` dal gruppo denominato `DeadBulbs` e contemporaneamente lo aggiunge al gruppo denominato `replaceableItems`.

```
aws iot update-thing-groups-for-thing \
    --thing-name MyLightBulb \
    --thing-groups-to-add "replaceableItems" \
    --thing-groups-to-remove "DeadBulbs"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Gruppi di oggetti nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta UpdateThingGroupsForThing AWS CLICommand Reference.

`update-thing`

Il seguente esempio di codice mostra come utilizzare`update-thing`.

AWS CLI

Come associare un oggetto a un tipo di oggetto

L'`update-thing`esempio seguente associa un oggetto nel registro AWS IoT a un tipo di oggetto. Quando crei l'associazione, fornisci valori per gli attributi definiti dal tipo di oggetto.

```
aws iot update-thing \
```

```
    --thing-name "MyOtherLightBulb" \
    --thing-type-name "LightBulb" \
    --attribute-payload "{"attributes": {"wattage":"75", "model":"123"}}"
```

Questo comando non produce alcun output. Per visualizzare il risultato, utilizza il comando `describe-thing`.

Per ulteriori informazioni, consulta Tipi di oggetti nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, vedere UpdateThingin AWS CLI Command Reference.

`update-topic-rule-destination`

Il seguente esempio di codice mostra come utilizzareu`pdate-topic-rule-destination`.

AWS CLI

Esempio 1: come abilitare la destinazione di una regola dell'argomento

L'esempio `update-topic-rule-destination` seguente abilita il traffico verso la destinazione di una regola dell'argomento.

```
aws iot update-topic-rule-destination \
    --arn "arn:aws:iot:us-west-2:123456789012:ruledestination/http/
a1b2c3d4-5678-90ab-cdef-11111EXAMPLE" \
    --status ENABLED
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Abilitazione della destinazione di una regola dell'argomento nella Guida per gli sviluppatori di AWS IoT.

Esempio 2: come disabilitare la destinazione di una regola dell'argomento

L'esempio `update-topic-rule-destination` seguente disabilita il traffico verso la destinazione di una regola dell'argomento.

```
aws iot update-topic-rule-destination \
    --arn "arn:aws:iot:us-west-2:123456789012:ruledestination/http/
a1b2c3d4-5678-90ab-cdef-11111EXAMPLE" \
    --status DISABLED
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Disabilitazione della destinazione di una regola dell'argomento nella Guida per gli sviluppatori di AWS IoT.

Esempio 3: come inviare un nuovo messaggio di conferma

L'esempio `update-topic-rule-destination` seguente invia un nuovo messaggio di conferma per la destinazione di una regola dell'argomento.

```
aws iot update-topic-rule-destination \
    --arn "arn:aws:iot:us-west-2:123456789012:ruledestination/http/
a1b2c3d4-5678-90ab-cdef-11111EXAMPLE" \
    --status IN_PROGRESS
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Invio di un nuovo messaggio di conferma nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta UpdateTopicRuleDestination AWS CLICommand Reference.

`validate-security-profile-behaviors`

Il seguente esempio di codice mostra come utilizzare`validate-security-profile-behaviors`.

AWS CLI

Esempio 1: come convalidare i parametri dei comportamenti per un profilo di sicurezza

L'`validate-security-profile-behaviors`esempio seguente convalida un set di comportamenti ben formato e corretto per un profilo di sicurezza IoT Device AWS Defender.

```
aws iot validate-security-profile-behaviors \
    --behaviors "[{\"name\":\"CellularBandwidth\",\"metric\":\"aws:message-byte-size
\",\"criteria\":{\"comparisonOperator\":\"greater-than\",\"value\":{\"count\":128},
\"consecutiveDatapointsToAlarm\":1,\"consecutiveDatapointsToClear\":1}},{\"name
\":\"Authorization\",\"metric\":\"aws:num-authorization-failures\",\"criteria\":
{\"comparisonOperator\":\"greater-than\",\"value\":{\"count\":12},\"durationSeconds
\":300,\"consecutiveDatapointsToAlarm\":1,\"consecutiveDatapointsToClear\":1}}]"
```

Output:

```
{
    "valid": true,
    "validationErrors": []
}
```

Esempio 2: come convalidare parametri dei comportamenti errati per un profilo di sicurezza

L'`validate-security-profile-behaviors`esempio seguente convalida un set di comportamenti che contiene un errore per un profilo di sicurezza AWS IoT Device Defender.

```
aws iot validate-security-profile-behaviors \
    --behaviors "[{\"name\":\"CellularBandwidth\",\"metric\":\"aws:message-byte-size
\",\"criteria\":{\"comparisonOperator\":\"greater-than\",\"value\":{\"count\":128},
\"consecutiveDatapointsToAlarm\":1,\"consecutiveDatapointsToClear\":1}},{\"name
\":\"Authorization\",\"metric\":\"aws:num-authorization-failures\",\"criteria\":
{\"comparisonOperator\":\"greater-than\",\"value\":{\"count\":12},\"durationSeconds
\":300,\"consecutiveDatapointsToAlarm\":100000,\"consecutiveDatapointsToClear
\":1}}]"
```

Output:

```
{
    "valid": false,
    "validationErrors": [
        {
            "errorMessage": "Behavior Authorization is malformed.
 consecutiveDatapointsToAlarm 100000 should be in range[1,10]"
        }
    ]
}
```

Per ulteriori informazioni, consulta Comandi di rilevamento nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta Command ValidateSecurityProfileBehaviorsReference AWS CLI .

AWS IoT Analytics esempi utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with AWS IoT Analytics.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`batch-put-message`

Il seguente esempio di codice mostra come utilizzare`batch-put-message`.

AWS CLI

Come inviare un messaggio a un canale

L'esempio `batch-put-message` seguente invia un messaggio al canale specificato.

```
aws iotanalytics batch-put-message \
    --cli-binary-format raw-in-base64-out \
    --cli-input-json file://batch-put-message.json
```

Contenuto di `batch-put-message.json`:

```
{
    "channelName": "mychannel",
    "messages": [
        {
            "messageId": "0001",
            "payload": "eyAidGVtcGVyYXR1cmUiOiAyMCB9"
        }
    ]
}
```

Output:

```
{
```

```
    "batchPutMessageErrorEntries": []
}
```

Per ulteriori informazioni, consulta il riferimento BatchPutMessageall'API AWS IoT Analytics.

- Per i dettagli sull'API, consulta BatchPutMessage AWS CLICommand Reference.

`cancel-pipeline-reprocessing`

Il seguente esempio di codice mostra come utilizzare`cancel-pipeline-reprocessing`.

AWS CLI

Come annullare la rielaborazione dei dati tramite una pipeline

L'esempio `cancel-pipeline-reprocessing` seguente annulla la rielaborazione dei dati tramite la pipeline specificata.

```
aws iotanalytics cancel-pipeline-reprocessing \
    --pipeline-name mypipeline \
    --reprocessing-id "6ad2764f-fb13-4de3-b101-4e74af03b043"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta il riferimento CancelPipelineReprocessingall'API AWS IoT Analytics.

- Per i dettagli sull'API, consulta CancelPipelineReprocessing AWS CLICommand Reference.

`create-channel`

Il seguente esempio di codice mostra come utilizzare`create-channel`.

AWS CLI

Come creare un canale

L'esempio `create-channel` seguente crea un canale con la configurazione specificata. Un canale raccoglie dati da un argomento MQTT e archivia i messaggi non elaborati prima di pubblicarli in una pipeline.

```
aws iotanalytics create-channel \
```

```
    --cli-input-json file://create-channel.json
```

Contenuto di `create-channel.json`:

```
{
    "channelName": "mychannel",
    "retentionPeriod": {
        "unlimited": true
    },
    "tags": [
        {
            "key": "Environment",
            "value": "Production"
        }
    ]
}
```

Output:

```
{
    "channelArn": "arn:aws:iotanalytics:us-west-2:123456789012:channel/mychannel",
    "channelName": "mychannel",
    "retentionPeriod": {
        "unlimited": true
    }
}
```

Per ulteriori informazioni, consulta il riferimento CreateChannelall'API AWS IoT Analytics.

- Per i dettagli sull'API, consulta CreateChannel AWS CLICommand Reference.

`create-dataset-content`

Il seguente esempio di codice mostra come utilizzare`create-dataset-content`.

AWS CLI

Come creare il contenuto di un set di dati

L'esempio `create-dataset-content` seguente crea i contenuti del set di dati specificato applicando `queryAction` (una query SQL) o `containerAction` (che esegue un'applicazione containerizzata).

```
aws iotanalytics create-dataset-content \
    --dataset-name mydataset
```

Output:

```
{
    "versionId": "d494b416-9850-4670-b885-ca22f1e89d62"
}
```

Per ulteriori informazioni, consulta il riferimento CreateDatasetContentall'API AWS IoT Analytics.

- Per i dettagli sull'API, consulta CreateDatasetContent AWS CLICommand Reference.

`create-dataset`

Il seguente esempio di codice mostra come utilizzare`create-dataset`.

AWS CLI

Come creare un set di dati

L'esempio `create-dataset` seguente crea un set di dati. Un set di dati archivia i dati recuperati da un datastore applicando una `queryAction` (una query SQL) o una `containerAction` (che esegue un'applicazione containerizzata). Questa operazione crea lo skeleton di un set di dati. Puoi popolare il set di dati manualmente chiamando `CreateDatasetContent` o automaticamente in base a un `trigger` specificato.

```
aws iotanalytics create-dataset \
    --cli-input-json file://create-dataset.json
```

Contenuto di `create-dataset.json`:

```
{
    "datasetName": "mydataset",
    "actions": [
        {
            "actionName": "myDatasetAction",
            "queryAction": {
                "sqlQuery": "SELECT * FROM mydatastore"
            }
        }
```

```
    ],
    "retentionPeriod": {
        "unlimited": true
    },
    "tags": [
        {
            "key": "Environment",
            "value": "Production"
        }
    ]
}
```

Output:

```
{
    "datasetName": "mydataset",
    "retentionPeriod": {
        "unlimited": true
    },
    "datasetArn": "arn:aws:iotanalytics:us-west-2:123456789012:dataset/mydataset"
}
```

Per ulteriori informazioni, consulta il riferimento CreateDatasetall'API AWS IoT Analytics.

- Per i dettagli sull'API, consulta CreateDataset AWS CLICommand Reference.

`create-datastore`

Il seguente esempio di codice mostra come utilizzare`create-datastore`.

AWS CLI

Come creare un datastore

L'esempio `create-datastore` seguente crea un datastore, che è un repository per i messaggi.

```
aws iotanalytics create-datastore \
    --cli-input-json file://create-datastore.json
```

Contenuto di `create-datastore.json`:

```
{
```

```
    "datastoreName": "mydatastore",
    "retentionPeriod": {
        "numberOfDays": 90
    },
    "tags": [
        {
            "key": "Environment",
            "value": "Production"
        }
    ]
}
```

Output:

```
{
    "datastoreName": "mydatastore",
    "datastoreArn": "arn:aws:iotanalytics:us-west-2:123456789012:datastore/
mydatastore",
    "retentionPeriod": {
        "numberOfDays": 90,
        "unlimited": false
    }
}
```

Per ulteriori informazioni, consulta il riferimento CreateDatastoreall'API AWS IoT Analytics.

- Per i dettagli sull'API, consulta CreateDatastore AWS CLICommand Reference.

`create-pipeline`

Il seguente esempio di codice mostra come utilizzare`create-pipeline`.

AWS CLI

Come creare una pipeline di IoT Analytics

L'esempio `create-pipeline` seguente crea una pipeline. Una pipeline utilizza i messaggi provenienti da un canale e permette di elaborarli prima di archiviarli in un datastore. È necessario specificare un canale e un'attività datastore e, facoltativamente, fino a 23 attività aggiuntive nell'array `pipelineActivities`.

```
aws iotanalytics create-pipeline \
```

```
    --cli-input-json file://create-pipeline.json
```

Contenuto di `create-pipeline.json`:

```
{
    "pipelineName": "mypipeline",
    "pipelineActivities": [
        {
            "channel": {
                "name": "myChannelActivity",
                "channelName": "mychannel",
                "next": "myMathActivity"
            }
        },
        {
            "datastore": {
                "name": "myDatastoreActivity",
                "datastoreName": "mydatastore"
            }
        },
        {
            "math": {
                "name": "myMathActivity",
                "math": "((temp - 32) * 5.0) / 9.0",
                "attribute": "tempC",
                "next": "myDatastoreActivity"
            }
        }
    ],
    "tags": [
        {
            "key": "Environment",
            "value": "Beta"
        }
    ]
}
```

Output:

```
{
    "pipelineArn": "arn:aws:iotanalytics:us-west-2:123456789012:pipeline/
mypipeline",
    "pipelineName": "mypipeline"
```

```
}
```

Per ulteriori informazioni, consulta il riferimento CreatePipelineall'API AWS IoT Analytics.

- Per i dettagli sull'API, consulta CreatePipeline AWS CLICommand Reference.

`delete-channel`

Il seguente esempio di codice mostra come utilizzare`delete-channel`.

AWS CLI

Come eliminare un canale di IoT Analytics

L'esempio `delete-channel` seguente elimina il canale specificato.

```
aws iotanalytics delete-channel \
    --channel-name mychannel
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta il riferimento DeleteChannelall'API AWS IoT Analytics.

- Per i dettagli sull'API, consulta DeleteChannel AWS CLICommand Reference.

`delete-dataset-content`

Il seguente esempio di codice mostra come utilizzare`delete-dataset-content`.

AWS CLI

Come eliminare il contenuto di un set di dati

L'esempio `delete-dataset-content` seguente elimina i contenuti del set di dati specificato.

```
aws iotanalytics delete-dataset-content \
    --dataset-name mydataset
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta il riferimento DeleteDatasetContentall'API AWS IoT Analytics.

- Per i dettagli sull'API, consulta DeleteDatasetContent AWS CLICommand Reference.

`delete-dataset`

Il seguente esempio di codice mostra come utilizzare`delete-dataset`.

AWS CLI

Come eliminare un set di dati

L'esempio `delete-dataset` seguente elimina il set di dati specificato. Non è necessario eliminare i contenuti del set di dati prima di eseguire questa operazione.

```
aws iotanalytics delete-dataset \
    --dataset-name mydataset
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta il riferimento DeleteDatasetall'API AWS IoT Analytics.

- Per i dettagli sull'API, consulta DeleteDataset AWS CLICommand Reference.

`delete-datastore`

Il seguente esempio di codice mostra come utilizzare`delete-datastore`.

AWS CLI

Come eliminare un datastore

L'esempio `delete-datastore` seguente elimina il datastore specificato.

```
aws iotanalytics delete-datastore \
    --datastore-name mydatastore
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta il riferimento DeleteDatastoreall'API AWS IoT Analytics.

- Per i dettagli sull'API, consulta DeleteDatastore AWS CLICommand Reference.

`delete-pipeline`

Il seguente esempio di codice mostra come utilizzare`delete-pipeline`.

AWS CLI

Come eliminare una pipeline

L'esempio `delete-pipeline` seguente elimina la pipeline specificata.

```
aws iotanalytics delete-pipeline \
    --pipeline-name mypipeline
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta il riferimento DeletePipelineall'API AWS IoT Analytics.

- Per i dettagli sull'API, consulta DeletePipeline AWS CLICommand Reference.

`describe-channel`

Il seguente esempio di codice mostra come utilizzare`describe-channel`.

AWS CLI

Come recuperare le informazioni su un canale

L'esempio `describe-channel` seguente mostra i dettagli, incluse le statistiche, del canale specificato.

```
aws iotanalytics describe-channel \
    --channel-name mychannel \
    --include-statistics
```

Output:

```
{
    "statistics": {
        "size": {
            "estimatedSizeInBytes": 402.0,
            "estimatedOn": 1561504380.0
        }
    },
    "channel": {
        "status": "ACTIVE",
```

```
        "name": "mychannel",
        "lastUpdateTime": 1557860351.001,
        "creationTime": 1557860351.001,
        "retentionPeriod": {
            "unlimited": true
        },
        "arn": "arn:aws:iotanalytics:us-west-2:123456789012:channel/mychannel"
    }
}
```

Per ulteriori informazioni, consulta il riferimento DescribeChannelall'API AWS IoT Analytics.

- Per i dettagli sull'API, consulta DescribeChannel AWS CLICommand Reference.

`describe-dataset`

Il seguente esempio di codice mostra come utilizzare`describe-dataset`.

AWS CLI

Come recuperare le informazioni su un set di dati

L'esempio `describe-dataset` seguente visualizza i dettagli della quota specificata.

```
aws iotanalytics describe-dataset \
    --dataset-name mydataset
```

Output:

```
{
    "dataset": {
        "status": "ACTIVE",
        "contentDeliveryRules": [],
        "name": "mydataset",
        "lastUpdateTime": 1557859240.658,
        "triggers": [],
        "creationTime": 1557859240.658,
        "actions": [
            {
                "actionName": "query_32",
                "queryAction": {
                    "sqlQuery": "SELECT * FROM mydatastore",
```

```
                    "filters": []
                }
            }
        ],
        "retentionPeriod": {
            "numberOfDays": 90,
            "unlimited": false
        },
        "arn": "arn:aws:iotanalytics:us-west-2:123456789012:dataset/mydataset"
    }
}
```

Per ulteriori informazioni, consulta il riferimento DescribeDatasetall'API AWS IoT Analytics.

- Per i dettagli sull'API, consulta DescribeDataset AWS CLICommand Reference.

`describe-datastore`

Il seguente esempio di codice mostra come utilizzare`describe-datastore`.

AWS CLI

Come recuperare le informazioni su un datastore

L'esempio `describe-datastore` seguente mostra i dettagli, incluse le statistiche, del datastore specificato.

```
aws iotanalytics describe-datastore \
    --datastore-name mydatastore \
    --include-statistics
```

Output:

```
{
    "datastore": {
        "status": "ACTIVE",
        "name": "mydatastore",
        "lastUpdateTime": 1557858971.02,
        "creationTime": 1557858971.02,
        "retentionPeriod": {
            "unlimited": true
        },
```

```
        "arn": "arn:aws:iotanalytics:us-west-2:123456789012:datastore/mydatastore"
    },
    "statistics": {
        "size": {
            "estimatedSizeInBytes": 397.0,
            "estimatedOn": 1561592040.0
        }
    }
}
```

Per ulteriori informazioni, consulta il riferimento DescribeDatastoreall'API AWS IoT Analytics.

- Per i dettagli sull'API, consulta DescribeDatastore AWS CLICommand Reference.

`describe-logging-options`

Il seguente esempio di codice mostra come utilizzare`describe-logging-options`.

AWS CLI

Come recuperare le opzioni di registrazione dei log attuali

L'`describe-logging-options`esempio seguente mostra le opzioni di registrazione correnti di AWS IoT Analytics.

```
aws iotanalytics describe-logging-options
```

Questo comando non produce alcun output. Output:

```
{
    "loggingOptions": {
        "roleArn": "arn:aws:iam::123456789012:role/service-role/myIoTAnalyticsRole",
        "enabled": true,
        "level": "ERROR"
    }
}
```

Per ulteriori informazioni, consulta il riferimento DescribeLoggingOptionsall'API AWS IoT Analytics.

- Per i dettagli sull'API, consulta DescribeLoggingOptions AWS CLICommand Reference.

`describe-pipeline`

Il seguente esempio di codice mostra come utilizzare`describe-pipeline`.

AWS CLI

Come recuperare le informazioni su una pipeline

L'esempio `describe-pipeline` seguente visualizza i dettagli della pipeline specificata.

```
aws iotanalytics describe-pipeline \
    --pipeline-name mypipeline
```

Output:

```
{
    "pipeline": {
        "activities": [
            {
                "channel": {
                    "channelName": "mychannel",
                    "name": "mychannel_28",
                    "next": "mydatastore_29"
                }
            },
            {
                "datastore": {
                    "datastoreName": "mydatastore",
                    "name": "mydatastore_29"
                }
            }
        ],
        "name": "mypipeline",
        "lastUpdateTime": 1561676362.515,
        "creationTime": 1557859124.432,
        "reprocessingSummaries": [
            {
                "status": "SUCCEEDED",
                "creationTime": 1561676362.189,
                "id": "6ad2764f-fb13-4de3-b101-4e74af03b043"
            }
        ],
        "arn": "arn:aws:iotanalytics:us-west-2:123456789012:pipeline/mypipeline"
```

```
    }
}
```

Per ulteriori informazioni, consulta il riferimento DescribePipelineall'API AWS IoT Analytics.

- Per i dettagli sull'API, consulta DescribePipeline AWS CLICommand Reference.

`get-dataset-content`

Il seguente esempio di codice mostra come utilizzare`get-dataset-content`.

AWS CLI

Come recuperare il contenuto di un set di dati

L'`get-dataset-content`esempio seguente recupera il contenuto di un set di dati come predefinito. URIs

```
aws iotanalytics get-dataset-content --dataset-name mydataset
```

Output:

```
{
    "status": {
        "state": "SUCCEEDED"
    },
    "timestamp": 1557863215.995,
    "entries": [
        {
            "dataURI": "https://aws-radiant-
dataset-12345678-1234-1234-1234-123456789012.s3.us-west-2.amazonaws.com/
results/12345678-e8b3-46ba-b2dd-efe8d86cf385.csv?X-Amz-Security-Token=...-Amz-
Algorithm=AWS4-HMAC-SHA256&X-Amz-Date=20190628T173437Z&X-Amz-SignedHeaders=host&X-
Amz-Expires=7200&X-Amz-Credential=...F20190628%2Fus-west-2%2Fs3%2Faws4_request&X-
Amz-Signature=..."
        }
    ]
}
```

Per ulteriori informazioni, consulta la guida GetDatasetContent.

- Per i dettagli sull'API, consulta GetDatasetContent AWS CLICommand Reference.

`list-channels`

Il seguente esempio di codice mostra come utilizzare`list-channels`.

AWS CLI

Come recuperare un elenco di canali

L'esempio `list-channels` seguente visualizza le informazioni di riepilogo per i canali disponibili.

```
aws iotanalytics list-channels
```

Output:

```
{
    "channelSummaries": [
        {
            "status": "ACTIVE",
            "channelName": "mychannel",
            "creationTime": 1557860351.001,
            "lastUpdateTime": 1557860351.001
        }
    ]
}
```

Per ulteriori informazioni, consulta il riferimento ListChannelsall'API AWS IoT Analytics.

- Per i dettagli sull'API, consulta ListChannels AWS CLICommand Reference.

`list-dataset-contents`

Il seguente esempio di codice mostra come utilizzare`list-dataset-contents`.

AWS CLI

Come elencare informazioni sui contenuti di un set di dati

L'esempio `list-dataset-contents` seguente elenca le informazioni sui contenuti del set di dati che sono stati creati.

```
aws iotanalytics list-dataset-contents \
```

```
    --dataset-name mydataset
```

Output:

```
{
    "datasetContentSummaries": [
        {
            "status": {
                "state": "SUCCEEDED"
            },
            "scheduleTime": 1557863215.995,
            "version": "b10ea2a9-66c1-4d99-8d1f-518113b738d0",
            "creationTime": 1557863215.995
        }
    ]
}
```

Per ulteriori informazioni, consulta il riferimento ListDatasetContentsall'API AWS IoT Analytics.

- Per i dettagli sull'API, consulta ListDatasetContents AWS CLICommand Reference.

`list-datasets`

Il seguente esempio di codice mostra come utilizzare`list-datasets`.

AWS CLI

Come recuperare le informazioni sui set di dati

L'esempio `list-datasets` seguente elenca le informazioni di riepilogo sui set di dati disponibili.

```
aws iotanalytics list-datasets
```

Output:

```
{
    "datasetSummaries": [
        {
            "status": "ACTIVE",
            "datasetName": "mydataset",
            "lastUpdateTime": 1557859240.658,
            "triggers": [],
```

```
            "creationTime": 1557859240.658,
            "actions": [
                {
                    "actionName": "query_32",
                    "actionType": "QUERY"
                }
            ]
        }
    ]
}
```

Per ulteriori informazioni, consulta il riferimento ListDatasetsall'API AWS IoT Analytics.

- Per i dettagli sull'API, consulta ListDatasets AWS CLICommand Reference.

`list-datastores`

Il seguente esempio di codice mostra come utilizzare`list-datastores`.

AWS CLI

Come recuperare un elenco di datastore

L'esempio `list-datastores` seguente visualizza le informazioni di riepilogo sui datastore disponibili.

```
aws iotanalytics list-datastores
```

Output:

```
{
    "datastoreSummaries": [
        {
            "status": "ACTIVE",
            "datastoreName": "mydatastore",
            "creationTime": 1557858971.02,
            "lastUpdateTime": 1557858971.02
        }
    ]
}
```

Per ulteriori informazioni, consulta il riferimento ListDatastoresall'API AWS IoT Analytics.

- Per i dettagli sull'API, consulta ListDatastores AWS CLICommand Reference.

`list-pipelines`

Il seguente esempio di codice mostra come utilizzare`list-pipelines`.

AWS CLI

Come recuperare un elenco di pipeline

L'esempio `list-pipelines` seguente visualizza un elenco di pipeline disponibili.

```
aws iotanalytics list-pipelines
```

Output:

```
{
    "pipelineSummaries": [
        {
            "pipelineName": "mypipeline",
            "creationTime": 1557859124.432,
            "lastUpdateTime": 1557859124.432,
            "reprocessingSummaries": []
        }
    ]
}
```

Per ulteriori informazioni, consulta il riferimento ListPipelinesall'API AWS IoT Analytics.

- Per i dettagli sull'API, consulta ListPipelines AWS CLICommand Reference.

`list-tags-for-resource`

Il seguente esempio di codice mostra come utilizzare`list-tags-for-resource`.

AWS CLI

Come elencare i tag per una risorsa

L'esempio `list-tags-for-resource` seguente elenca i tag associati alla risorsa specificata.

```
aws iotanalytics list-tags-for-resource \
```

```
    --resource-arn "arn:aws:iotanalytics:us-west-2:123456789012:channel/mychannel"
```

Output:

```
{
    "tags": [
        {
            "value": "bar",
            "key": "foo"
        }
    ]
}
```

Per ulteriori informazioni, consulta il riferimento ListTagsForResourceall'API AWS IoT Analytics.

- Per i dettagli sull'API, consulta ListTagsForResource AWS CLICommand Reference.

`put-logging-options`

Il seguente esempio di codice mostra come utilizzare`put-logging-options`.

AWS CLI

Come impostare o aggiornare le opzioni di registrazione dei log

L'`put-logging-options`esempio seguente imposta o aggiorna le opzioni di registrazione di AWS IoT Analytics. Se aggiorni il valore di qualsiasi campo `loggingOptions`, dovrai attendere fino a un minuto affinché la modifica abbia effetto. Inoltre, se modifichi la policy collegata al ruolo specificato nel campo "roleArn" (ad esempio per correggere una policy non valida), dovrai attendere fino a cinque minuti affinché la modifica abbia effetto.

```
aws iotanalytics put-logging-options \
    --cli-input-json file://put-logging-options.json
```

Contenuto di `put-logging-options.json`:

```
{
    "loggingOptions": {
        "roleArn": "arn:aws:iam::123456789012:role/service-role/myIoTAnalyticsRole",
        "level": "ERROR",
        "enabled": true
```

```
    }
}
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta il riferimento PutLoggingOptionsall'API AWS IoT Analytics.

- Per i dettagli sull'API, consulta PutLoggingOptions AWS CLICommand Reference.

`run-pipeline-activity`

Il seguente esempio di codice mostra come utilizzare`run-pipeline-activity`.

AWS CLI

Come simulare un'attività di pipeline

L'esempio `run-pipeline-activity` seguente simula i risultati dell'esecuzione di un'attività di pipeline sul payload di un messaggio.

```
aws iotanalytics run-pipeline-activity \
    --pipeline-activity file://maths.json \
    --payloads file://payloads.json
```

Contenuto di `maths.json`:

```
{
    "math": {
        "name": "MyMathActivity",
        "math": "((temp - 32) * 5.0) / 9.0",
        "attribute": "tempC"
    }
}
```

Contenuto di `payloads.json`:

```
[
    "{\"humidity\": 52, \"temp\": 68 }",
    "{\"humidity\": 52, \"temp\": 32 }"
]
```

Output:

```
{
    "logResult": "",
    "payloads": [
        "eyJodW1pZGl0eSI6NTIsInRlbXAiOjY4LCJ0ZW1wQyI6MjB9",
        "eyJodW1pZGl0eSI6NTIsInRlbXAiOjMyLCJ0ZW1wQyI6MH0="
    ]
}
```

Per ulteriori informazioni, consulta il riferimento RunPipelineActivityall'API AWS IoT Analytics.

- Per i dettagli sull'API, consulta RunPipelineActivity AWS CLICommand Reference.

`sample-channel-data`

Il seguente esempio di codice mostra come utilizzare`sample-channel-data`.

AWS CLI

Come recuperare messaggi di esempio da un canale

L'esempio `sample-channel-data` seguente recupera dal canale specificato un campione di messaggi elaborati durante l'intervallo di tempo specificato. Puoi recuperare fino a 10 messaggi.

```
aws iotanalytics sample-channel-data \
    --channel-name mychannel
```

Output:

```
{
    "payloads": [
        "eyAidGVtcGVyYXR1cmUiOiAyMCB9",
        "eyAiZm9vIjogImJhciIgfQ=="
    ]
}
```

Per ulteriori informazioni, consulta il riferimento SampleChannelDataall'API AWS IoT Analytics.

- Per i dettagli sull'API, consulta SampleChannelData AWS CLICommand Reference.

`start-pipeline-reprocessing`

Il seguente esempio di codice mostra come utilizzare`start-pipeline-reprocessing`.

AWS CLI

Come avviare la rielaborazione della pipeline

L'esempio `start-pipeline-reprocessing` seguente avvia la rielaborazione dei dati grezzi tramite la pipeline specificata.

```
aws iotanalytics start-pipeline-reprocessing \
    --pipeline-name mypipeline
```

Output:

```
{
    "reprocessingId": "6ad2764f-fb13-4de3-b101-4e74af03b043"
}
```

Per ulteriori informazioni, consulta il riferimento StartPipelineReprocessingall'API AWS IoT Analytics.

- Per i dettagli sull'API, consulta StartPipelineReprocessing AWS CLICommand Reference.

`tag-resource`

Il seguente esempio di codice mostra come utilizzare`tag-resource`.

AWS CLI

Come aggiungere o modificare i tag associati a una risorsa

L'esempio `tag-resource` seguente aggiunge o modifica i tag associati alla risorsa specificata.

```
aws iotanalytics tag-resource \
    --resource-arn "arn:aws:iotanalytics:us-west-2:123456789012:channel/mychannel" \
    --tags "[{\"key\": \"Environment\", \"value\": \"Production\"}]"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta il riferimento TagResourceall'API AWS IoT Analytics.

- Per i dettagli sull'API, consulta TagResource AWS CLICommand Reference.

`untag-resource`

Il seguente esempio di codice mostra come utilizzare`untag-resource`.

AWS CLI

Come rimuovere tag da una risorsa

L'esempio `untag-resource` seguente rimuove i tag con i nomi di chiave specificati dalla risorsa specificata.

```
aws iotanalytics untag-resource \
    --resource-arn "arn:aws:iotanalytics:us-west-2:123456789012:channel/mychannel" \
    --tag-keys "[\"Environment\"]"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta UntagResource < https://docs.aws.amazon.com/iotanalytics/latest/APIReference/API _ UntagResource .html > nel riferimento all'API AWS IoT Analytics.

- Per i dettagli sull'API, consulta UntagResource AWS CLICommand Reference.

`update-channel`

Il seguente esempio di codice mostra come utilizzare`update-channel`.

AWS CLI

Come modificare un canale

L'esempio `update-channel` seguente modifica le impostazioni del canale specificato.

```
aws iotanalytics update-channel \
    --cli-input-json file://update-channel.json
```

Contenuto di `update-channel.json`:

```
{
    "channelName": "mychannel",
    "retentionPeriod": {
        "numberOfDays": 92
    }
```

```
}
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta il riferimento UpdateChannelall'API AWS IoT Analytics.

- Per i dettagli sull'API, consulta UpdateChannel AWS CLICommand Reference.

`update-dataset`

Il seguente esempio di codice mostra come utilizzareupdate-dataset.

AWS CLI

Come aggiornare un set di dati

L'esempio update-dataset seguente modifica le impostazioni del set di dati specificato.

```
aws iotanalytics update-dataset \
    --cli-input-json file://update-dataset.json
```

Contenuto di update-dataset.json:

```
{
    "datasetName": "mydataset",
    "actions": [
        {
            "actionName": "myDatasetUpdateAction",
            "queryAction": {
                "sqlQuery": "SELECT * FROM mydatastore"
            }
        }
    ],
    "retentionPeriod": {
        "numberOfDays": 92
    }
}
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta UpdateDataset < https://docs.aws.amazon.com/iotanalytics/latest/APIReference/API _ UpdateDataset .html > nel riferimento all'API AWS IoT Analytics.

- Per i dettagli sull'API, consulta UpdateDataset AWS CLICommand Reference.

`update-datastore`

Il seguente esempio di codice mostra come utilizzare`update-datastore`.

AWS CLI

Come aggiornare un datastore

L'esempio `update-datastore` seguente modifica le impostazioni del datastore specificato.

```
aws iotanalytics update-datastore \
    --cli-input-json file://update-datastore.json
```

Contenuto di update-datastore.json:

```
{
    "datastoreName": "mydatastore",
    "retentionPeriod": {
        "numberOfDays": 93
    }
}
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta il riferimento UpdateDatastoreall'API AWS IoT Analytics.

- Per i dettagli sull'API, consulta UpdateDatastore AWS CLICommand Reference.

`update-pipeline`

Il seguente esempio di codice mostra come utilizzare`update-pipeline`.

AWS CLI

Come aggiornare una pipeline

L'esempio `update-pipeline` seguente modifica le impostazioni della pipeline specificata. È necessario specificare un canale e un'attività datastore e, facoltativamente, fino a 23 attività aggiuntive nell'array `pipelineActivities`.

```
aws iotanalytics update-pipeline \
    --cli-input-json file://update-pipeline.json
```

Contenuto di update-pipeline.json:

```
{
    "pipelineName": "mypipeline",
    "pipelineActivities": [
        {
            "channel": {
                "name": "myChannelActivity",
                "channelName": "mychannel",
                "next": "myMathActivity"
            }
        },
        {
            "datastore": {
                "name": "myDatastoreActivity",
                "datastoreName": "mydatastore"
            }
        },
        {
            "math": {
                "name": "myMathActivity",
                "math": "(((temp - 32) * 5.0) / 9.0) + 273.15",
                "attribute": "tempK",
                "next": "myDatastoreActivity"
            }
        }
    ]
}
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta il riferimento UpdatePipelineall'API AWS IoT Analytics.

- Per i dettagli sull'API, consulta UpdatePipeline AWS CLICommand Reference.

Esempi di Device Advisor che utilizzano AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with Device Advisor.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`create-suite-definition`

Il seguente esempio di codice mostra come utilizzare`create-suite-definition`.

AWS CLI

Esempio 1: come creare una suite di test IoT Device Advisor

L'`create-suite-definition`esempio seguente crea una suite di test Device Advisor nell' AWS IoT con la configurazione di definizione della suite specificata.

```
aws iotdeviceadvisor create-suite-definition \
    --suite-definition-configuration '{ \
        "suiteDefinitionName": "TestSuiteName", \
        "devices": [{"thingArn":"arn:aws:iot:us-east-1:123456789012:thing/
MyIotThing"}], \
        "intendedForQualification": false, \
        "rootGroup": "{\"configuration\":{},\"tests\":[{\"name\":\"MQTT Connect\",
\"configuration\":{\"EXECUTION_TIMEOUT\":120},\"tests\":[{\"name\":\"MQTT_Connect\",
\"configuration\":{},\"test\":{\"id\":\"MQTT_Connect\",\"testCase\":null,\"version
\":\"0.0.0\"}}]}]}", \
        "devicePermissionRoleArn": "arn:aws:iam::123456789012:role/Myrole"}'
```

Output:

```
{
    "suiteDefinitionId": "0jtsgio7yenu",
    "suiteDefinitionArn": "arn:aws:iotdeviceadvisor:us-
east-1:123456789012:suitedefinition/0jtsgio7yenu",
```

```
    "suiteDefinitionName": "TestSuiteName",
    "createdAt": "2022-12-02T11:38:13.263000-05:00"
}
```

Per ulteriori informazioni, consulta Creare una definizione di suite di test nella Guida per gli sviluppatori di AWS IoT Core.

Esempio 2: come creare una suite di test di qualificazione più recente di IoT Device Advisor

L'`create-suite-definition`esempio seguente crea una suite di test di qualificazione Device Advisor con la versione più recente dell' AWS IoT con la configurazione di definizione della suite specificata.

```
aws iotdeviceadvisor create-suite-definition \
    --suite-definition-configuration '{ \
        "suiteDefinitionName": "TestSuiteName", \
        "devices": [{"thingArn":"arn:aws:iot:us-east-1:123456789012:thing/
MyIotThing"}], \
        "intendedForQualification": true, \
        "rootGroup": "", \
        "devicePermissionRoleArn": "arn:aws:iam::123456789012:role/Myrole"}'
```

Output:

```
{
    "suiteDefinitionId": "txgsuolk2myj",
    "suiteDefinitionArn": "arn:aws:iotdeviceadvisor:us-
east-1:123456789012:suitedefinition/txgsuolk2myj",
    "suiteDefinitionName": "TestSuiteName",
    "createdAt": "2022-12-02T11:38:13.263000-05:00"
}
```

Per ulteriori informazioni, consulta Creare una definizione di suite di test nella Guida per gli sviluppatori di AWS IoT Core.

- Per i dettagli sull'API, vedere CreateSuiteDefinitionin AWS CLI Command Reference.

`delete-suite-definition`

Il seguente esempio di codice mostra come utilizzare`delete-suite-definition`.

AWS CLI

Come eliminare la suite di test IoT Device Advisor

L'esempio `delete-suite-definition` seguente elimina la suite di test Device Advisor con l'ID di definizione della suite specificato.

```
aws iotdeviceadvisor delete-suite-definition \
    --suite-definition-id 0jtsgio7yenu
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta la DeleteSuiteDefinitionsezione AWS IoT API Reference.

- Per i dettagli sull'API, consulta DeleteSuiteDefinition AWS CLICommand Reference.

`get-endpoint`

Il seguente esempio di codice mostra come utilizzare`get-endpoint`.

AWS CLI

Esempio 1: come ottenere informazioni su un endpoint a livello di account di IoT Device Advisor

L'esempio `get-endpoint` seguente ottiene le informazioni su un endpoint di test a livello di account di Device Advisor.

```
aws iotdeviceadvisor get-endpoint
```

Output:

```
{
    "endpoint": "t6y4c143x9sfo.deviceadvisor.iot.us-east-1.amazonaws.com"
}
```

Esempio 2: come ottenere informazioni su un endpoint a livello di dispositivo di IoT Device Advisor

L'esempio `get-endpoint` seguente ottiene le informazioni su un endpoint di test a livello di dispositivo di Device Advisor con il thing-arn o il certificate-arn specificato.

```
aws iotdeviceadvisor get-endpoint \
    --thing-arn arn:aws:iot:us-east-1:123456789012:thing/MyIotThing
```

Output:

```
{
    "endpoint": "tdb7719be5t6y4c143x9sfo.deviceadvisor.iot.us-east-1.amazonaws.com"
}
```

Per ulteriori informazioni, consulta Ottenere un endpoint di test nella Guida per gli sviluppatori di AWS IoT Core.

- Per i dettagli sull'API, consulta GetEndpoint AWS CLICommand Reference.

`get-suite-definition`

Il seguente esempio di codice mostra come utilizzare`get-suite-definition`.

AWS CLI

Come ottenere informazioni su una suite di test IoT Device Advisor

L'esempio `get-suite-definition` seguente ottiene le informazioni su una suite di test Device Advisor con l'ID di definizione della suite specificato.

```
aws iotdeviceadvisor get-suite-definition \
    --suite-definition-id qqcsmtyyjabl
```

Output:

```
{
    "suiteDefinitionId": "qqcsmtyyjabl",
    "suiteDefinitionArn": "arn:aws:iotdeviceadvisor:us-
east-1:123456789012:suitedefinition/qqcsmtyyjabl",
    "suiteDefinitionVersion": "v1",
    "latestVersion": "v1",
    "suiteDefinitionConfiguration": {
        "suiteDefinitionName": "MQTT connection",
        "devices": [],
        "intendedForQualification": false,
        "isLongDurationTest": false,
        "rootGroup": "{\"configuration\":{},\"tests\":[{\"id\":\"uta5d9j1kvwc\",
\"name\":\"Test group 1\",\"configuration\":{},\"tests\":[{\"id\":\"awr8pq5vc9yp\",
\"name\":\"MQTT Connect\",\"configuration\":{},\"test\":{\"id\":\"MQTT_Connect\",
\"testCase\":null,\"version\":\"0.0.0\"}}]}]}",
```

```
        "devicePermissionRoleArn": "arn:aws:iam::123456789012:role/Myrole",
        "protocol": "MqttV3_1_1"
    },
    "createdAt": "2022-11-11T22:28:52.389000-05:00",
    "lastModifiedAt": "2022-11-11T22:28:52.389000-05:00",
    "tags": {}
}
```

Per ulteriori informazioni, consulta Ottenere una definizione di suite di test nella Guida per gli sviluppatori di AWS IoT Core.

- Per i dettagli sull'API, consulta GetSuiteDefinition AWS CLICommand Reference.

`get-suite-run-report`

Il seguente esempio di codice mostra come utilizzare`get-suite-run-report`.

AWS CLI

Come ottenere informazioni sul report di esecuzione di una suite di test qualificata per IoT Device Advisor

L'esempio `get-suite-run-report` seguente ottiene il link per il download del rapporto relativo all'esecuzione di una suite di test qualificata per Device Advisor con successo con l'ID di definizione della suite e l'ID di esecuzione della suite specificati.

```
aws iotdeviceadvisor get-suite-run-report \
    --suite-definition-id ztvb5aek4w4x \
    --suite-run-id p6awv83nre6v
```

Output:

```
{
    "qualificationReportDownloadUrl": "https://senate-apn-reports-us-east-1-
prod.s3.amazonaws.com/report.downloadlink"
}
```

Per ulteriori informazioni, consulta Ottenere un rapporto di qualificazione per una suite di test di qualificazione riuscita nella Guida gli per sviluppatori di AWS IoT Core.

- Per i dettagli sull'API, consulta GetSuiteRunReport AWS CLICommand Reference.

`get-suite-run`

Il seguente esempio di codice mostra come utilizzare`get-suite-run`.

AWS CLI

Come ottenere informazioni sullo stato di esecuzione di una suite di test IoT Device Advisor

L'esempio `get-suite-run` seguente ottiene le informazioni sullo stato di esecuzione di una suite di test Device Advisor con l'ID di definizione della suite e l'ID di esecuzione della suite specificati.

```
aws iotdeviceadvisor get-suite-run \
    --suite-definition-id qqcsmtyyjabl \
    --suite-run-id nzlfyhaa18oa
```

Output:

```
{
    "suiteDefinitionId": "qqcsmtyyjabl",
    "suiteDefinitionVersion": "v1",
    "suiteRunId": "nzlfyhaa18oa",
    "suiteRunArn": "arn:aws:iotdeviceadvisor:us-east-1:123456789012:suiterun/
qqcsmtyyjabl/nzlfyhaa18oa",
    "suiteRunConfiguration": {
        "primaryDevice": {
            "thingArn": "arn:aws:iot:us-east-1:123456789012:thing/MyIotThing",
            "certificateArn": "arn:aws:iot:us-east-1:123456789012:cert/certFile"
        },
        "parallelRun": false
    },
    "testResult": {
        "groups": [
            {
                "groupId": "uta5d9j1kvwc",
                "groupName": "Test group 1",
                "tests": [
                    {
                        "testCaseRunId": "2ve2twrqyr0s",
                        "testCaseDefinitionId": "awr8pq5vc9yp",
                        "testCaseDefinitionName": "MQTT Connect",
                        "status": "PASS",
                        "startTime": "2022-11-12T00:01:53.693000-05:00",
```

```
                        "endTime": "2022-11-12T00:02:15.443000-05:00",
                        "logUrl": "https://console.aws.amazon.com/
cloudwatch/home?region=us-east-1#logEventViewer:group=/aws/iot/deviceadvisor/
qqcsmtyyjabl;stream=nzlfyhaa18oa_2ve2twrqyr0s",
                        "warnings": "null",
                        "failure": "null"
                    }
                ]
            }
        ]
    },
    "startTime": "2022-11-12T00:01:52.673000-05:00",
    "endTime": "2022-11-12T00:02:16.496000-05:00",
    "status": "PASS",
    "tags": {}
}
```

Per ulteriori informazioni, consulta Ottenere un'esecuzione della suite di test nella Guida gli per sviluppatori di AWS IoT Core.

- Per i dettagli sull'API, consulta GetSuiteRun AWS CLICommand Reference.

`list-suite-definitions`

Il seguente esempio di codice mostra come utilizzare`list-suite-definitions`.

AWS CLI

Esempio 1: come elencare le suite di test IoT Device Advisor create con le impostazioni specificate

L'`list-suite-definitions`esempio seguente elenca fino a 25 suite di test Device Advisor create in AWS IoT. Se hai più di 25 suite di test, nell'output verrà mostrato il valore "nextToken". Puoi utilizzare "nextToken" per mostrare il resto delle suite di test che hai creato.

```
aws iotdeviceadvisor list-suite-definitions
```

Output:

```
{
    "suiteDefinitionInformationList": [
```

```
        {
            "suiteDefinitionId": "3hsn88h4p2g5",
            "suiteDefinitionName": "TestSuite1",
            "defaultDevices": [
                {
                    "thingArn": "arn:aws:iot:us-east-1:123456789012:thing/
MyIotThing"
                }
            ],
            "intendedForQualification": false,
            "isLongDurationTest": false,
            "protocol": "MqttV3_1_1",
            "createdAt": "2022-11-17T14:15:56.830000-05:00"
        },
        {
            ......
        }
    ],
    "nextToken": "nextTokenValue"
}
```

Esempio 2: come elencare le suite di test IoT Device Advisor create con le impostazioni specificate

L'`list-suite-definitions`esempio seguente elenca le suite di test Device Advisor create in AWS IoT con il numero massimo di risultati specificato. Se hai più suite di test rispetto al numero massimo, nell'output verrà mostrato il valore "nextToken". Se è presente "nextToken", è possibile utilizzarlo per mostrare le suite di test create che non lo erano mai state prima.

```
aws iotdeviceadvisor list-suite-definitions \
    --max-result 1 \
    --next-token "nextTokenValue"
```

Output:

```
{
    "suiteDefinitionInformationList": [
        {
            "suiteDefinitionId": "ztvb5aew4w4x",
            "suiteDefinitionName": "TestSuite2",
            "defaultDevices": [],
            "intendedForQualification": true,
```

```
            "isLongDurationTest": false,
            "protocol": "MqttV3_1_1",
            "createdAt": "2022-11-17T14:15:56.830000-05:00"
        }
    ],
    "nextToken": "nextTokenValue"
}
```

Per ulteriori informazioni, consulta la ListSuiteDefinitionssezione AWS IoT API Reference.

- Per i dettagli sull'API, consulta ListSuiteDefinitions AWS CLICommand Reference.

`list-suite-runs`

Il seguente esempio di codice mostra come utilizzare`list-suite-runs`.

AWS CLI

Esempio 1: come elencare tutte le informazioni sullo stato di esecuzione della suite di test IoT Device Advisor specificata

L'esempio `list-suite-runs` seguente elenca tutte le informazioni sullo stato di esecuzione della suite di test Device Advisor con l'ID di definizione della suite specificato. Se sono presenti più di 25 esecuzioni della suite di test, nell'output verrà mostrato il valore "nextToken". Puoi utilizzare "nextToken" per mostrare il resto delle esecuzioni della suite di test.

```
aws iotdeviceadvisor list-suite-runs \
    --suite-definition-id ztvb5aew4w4x
```

Output:

```
{
    "suiteRunsList": [
        {
            "suiteDefinitionId": "ztvb5aew4w4x",
            "suiteDefinitionVersion": "v1",
            "suiteDefinitionName": "TestSuite",
            "suiteRunId": "p6awv89nre6v",
            "createdAt": "2022-12-01T16:33:14.212000-05:00",
            "startedAt": "2022-12-01T16:33:15.710000-05:00",
            "endAt": "2022-12-01T16:42:03.323000-05:00",
            "status": "PASS",
```

```
            "passed": 6,
            "failed": 0
        }
    ]
}
```

Esempio 2: come elencare le informazioni sullo stato di esecuzione della suite di test IoT Device Advisor specificata con le impostazioni specificate

L'esempio `list-suite-runs` seguente elenca le informazioni sullo stato di esecuzione della suite di test Device Advisor con l'ID di definizione della suite specificato e il numero massimo di risultati specificato. Se hai più esecuzioni della suite di test rispetto al numero massimo, nell'output verrà mostrato il valore "nextToken". Se è presente "nextToken", è possibile utilizzarlo per mostrare le esecuzioni della suite di test create che non lo erano mai state prima.

```
aws iotdeviceadvisor list-suite-runs \
    --suite-definition-id qqcsmtyyjaml \
    --max-result 1 \
    --next-token "nextTokenValue"
```

Output:

```
{
    "suiteRunsList": [
        {
            "suiteDefinitionId": "qqcsmtyyjaml",
            "suiteDefinitionVersion": "v1",
            "suiteDefinitionName": "MQTT connection",
            "suiteRunId": "gz9vm2s6d2jy",
            "createdAt": "2022-12-01T20:10:27.079000-05:00",
            "startedAt": "2022-12-01T20:10:28.003000-05:00",
            "endAt": "2022-12-01T20:10:45.084000-05:00",
            "status": "STOPPED",
            "passed": 0,
            "failed": 0
        }
    ],
    "nextToken": "nextTokenValue"
}
```

Per ulteriori informazioni, consulta la ListSuiteRunssezione AWS IoT API Reference.

- Per i dettagli sull'API, consulta ListSuiteRuns AWS CLICommand Reference.

`list-tags-for-resource`

Il seguente esempio di codice mostra come utilizzare`list-tags-for-resource`.

AWS CLI

Come elencare i tag allegati a una risorsa IoT Device Advisor

L'esempio `list-tags-for-resource` seguente elenca i tag associati a una risorsa Device Advisor. La risorsa Device Advisor può essere Suitedefinition-Arn o Suiterun-Arn.

```
aws iotdeviceadvisor list-tags-for-resource \
    --resource-arn arn:aws:iotdeviceadvisor:us-east-1:123456789012:suitedefinition/ba0uyjpg38ny
```

Output:

```
{
    "tags": {
        "TestTagKey": "TestTagValue"
    }
}
```

Per ulteriori informazioni, vedere il riferimento ListTagsForResourceall'API AWS IoT e i tipi di risorse definiti da AWS IoT Core Device Advisor nel Service Authorization Reference.

- Per i dettagli sull'API, consulta ListTagsForResource AWS CLICommand Reference.

`start-suite-run`

Il seguente esempio di codice mostra come utilizzare`start-suite-run`.

AWS CLI

Come avviare l'esecuzione di una suite di test IoT Device Advisor

L'`start-suite-run`esempio seguente elenca i widget disponibili nel tuo AWS account.

```
aws iotdeviceadvisor start-suite-run \
```

```
    --suite-definition-id qqcsmtyyjabl \
    --suite-definition-version v1 \
    --suite-run-configuration '{"primaryDevice":{"thingArn": "arn:aws:iot:us-
east-1:123456789012:thing/MyIotThing","certificateArn":"arn:aws:iot:us-
east-1:123456789012:cert/certFile"}}'
```

Output:

```
{
    "suiteRunId": "pwmucgw7lt9s",
    "suiteRunArn": "arn:aws:iotdeviceadvisor:us-east-1:123456789012:suiterun/
qqcsmtyyjabl/pwmucgw7lk9s",
    "createdAt": "2022-12-02T15:43:05.581000-05:00"
}
```

Per ulteriori informazioni, consulta Avviare una suite di test nella Guida per gli sviluppatori di AWS IoT Core.

- Per i dettagli sull'API, consulta StartSuiteRun AWS CLICommand Reference.

`stop-suite-run`

Il seguente esempio di codice mostra come utilizzare`stop-suite-run`.

AWS CLI

Come arrestare una suite di test IoT Device Advisor attualmente in esecuzione

L'esempio `stop-suite-run` seguente arresta una suite di test Device Advisor attualmente in esecuzione con l'ID di definizione della suite e l'ID di esecuzione della suite specificati.

```
aws iotdeviceadvisor stop-suite-run \
    --suite-definition-id qqcsmtyyjabl \
    --suite-run-id nzlfyhaa18oa
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Interrompere l'esecuzione di una suite di test nella Guida per gli sviluppatori di AWS IoT Core.

- Per i dettagli sull'API, consulta StopSuiteRun AWS CLICommand Reference.

`tag-resource`

Il seguente esempio di codice mostra come utilizzare`tag-resource`.

AWS CLI

Come aggiungere e modificare i tag esistenti di una risorsa IoT Device Advisor

L'esempio `tag-resource` seguente aggiunge e modifica i tag esistenti di una risorsa Device Advisor con la risorsa arn e i tag specificati. La risorsa Device Advisor può essere Suitedefinition-Arn o Suiterun-Arn.

```
aws iotdeviceadvisor tag-resource \
    --resource-arn arn:aws:iotdeviceadvisor:us-east-1:123456789012:suitedefinition/ba0uyjpg38ny \
    --tags '{"TagKey": "TagValue"}'
```

Questo comando non produce alcun output.

Per ulteriori informazioni, vedere il riferimento TagResourceall'API AWS IoT e i tipi di risorse definiti da AWS IoT Core Device Advisor nel Service Authorization Reference.

- Per i dettagli sull'API, consulta TagResource AWS CLICommand Reference.

`untag-resource`

Il seguente esempio di codice mostra come utilizzare`untag-resource`.

AWS CLI

Come rimuovere i tag esistenti da una risorsa IoT Device Advisor

L'esempio `untag-resource` seguente rimuove i tag esistenti da una risorsa Device Advisor con la risorsa arn e la chiave tag specificati. La risorsa Device Advisor può essere Suitedefinition-Arn o Suiterun-Arn.

```
aws iotdeviceadvisor untag-resource \
    --resource-arn arn:aws:iotdeviceadvisor:us-east-1:123456789012:suitedefinition/ba0uyjpg38ny \
    --tag-keys "TagKey"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, vedere il riferimento UntagResourceall'API AWS IoT e i tipi di risorse definiti da AWS IoT Core Device Advisor nel Service Authorization Reference.

- Per i dettagli sull'API, consulta UntagResource AWS CLICommand Reference.

`update-suite-definition`

Il seguente esempio di codice mostra come utilizzareu`pdate-suite-definition`.

AWS CLI

Esempio 1: come aggiornare una suite di test IoT Device Advisor

L'`update-suite-definition`esempio seguente aggiorna una suite di test Device Advisor nell' AWS IoT con l'ID di definizione della suite e la configurazione della definizione della suite specificati.

```
aws iotdeviceadvisor update-suite-definition \
    --suite-definition-id 3hsn88h4p2g5 \
    --suite-definition-configuration '{ \
        "suiteDefinitionName": "TestSuiteName", \
        "devices": [{"thingArn":"arn:aws:iot:us-east-1:123456789012:thing/
MyIotThing"}], \
        "intendedForQualification": false, \
        "rootGroup": "{\"configuration\":{},\"tests\":[{\"name\":\"MQTT Connect\",
\"configuration\":{\"EXECUTION_TIMEOUT\":120},\"tests\":[{\"name\":\"MQTT_Connect\",
\"configuration\":{},\"test\":{\"id\":\"MQTT_Connect\",\"testCase\":null,\"version
\":\"0.0.0\"}}]}]}", \
        "devicePermissionRoleArn": "arn:aws:iam::123456789012:role/Myrole"}'
```

Output:

```
{
    "suiteDefinitionId": "3hsn88h4p2g5",
    "suiteDefinitionName": "TestSuiteName",
    "suiteDefinitionVersion": "v3",
    "createdAt": "2022-11-17T14:15:56.830000-05:00",
    "lastUpdatedAt": "2022-12-02T16:02:45.857000-05:00"
}
```

Esempio 2: come aggiornare una suite di test di qualificazione di IoT Device Advisor

L'`update-suite-definition`esempio seguente aggiorna una suite di test di qualificazione Device Advisor nell' AWS IoT con l'ID di definizione della suite e la configurazione della definizione della suite specificati.

```
aws iotdeviceadvisor update-suite-definition \
    --suite-definition-id txgsuolk2myj \
    --suite-definition-configuration '{
        "suiteDefinitionName": "TestSuiteName", \
        "devices": [{"thingArn":"arn:aws:iot:us-east-1:123456789012:thing/
MyIotThing"}], \
        "intendedForQualification": true, \
        "rootGroup": "", \
        "devicePermissionRoleArn": "arn:aws:iam::123456789012:role/Myrole"}'
```

Output:

```
{
    "suiteDefinitionId": "txgsuolk2myj",
    "suiteDefinitionName": "TestSuiteName",
    "suiteDefinitionVersion": "v3",
    "createdAt": "2022-11-17T14:15:56.830000-05:00",
    "lastUpdatedAt": "2022-12-02T16:02:45.857000-05:00"
}
```

Per ulteriori informazioni, consulta la UpdateSuiteDefinitionsezione AWS IoT API Reference.

- Per i dettagli sull'API, consulta UpdateSuiteDefinition AWS CLICommand Reference.

AWS IoT data esempi utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with AWS IoT data.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`delete-thing-shadow`

Il seguente esempio di codice mostra come utilizzare`delete-thing-shadow`.

AWS CLI

Come eliminare il documento shadow di un dispositivo

L'esempio `delete-thing-shadow` seguente elimina l'intero documento shadow per il dispositivo denominato `MyRPi`.

```
aws iot-data delete-thing-shadow \
    --thing-name MyRPi \
    "output.txt"
```

Il comando non produce alcun output sullo schermo, ma `output.txt` contiene informazioni che confermano la versione e il timestamp del documento shadow eliminato.

```
{"version":2,"timestamp":1560270384}
```

Per ulteriori informazioni, consulta Uso delle copie shadow nella Guida per sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta DeleteThingShadow AWS CLICommand Reference.

`get-thing-shadow`

Il seguente esempio di codice mostra come utilizzare`get-thing-shadow`.

AWS CLI

Come ottenere un documento shadow per l'oggetto

L'esempio `get-thing-shadow` seguente ottiene il documento shadow per l'elemento IoT specificato.

```
aws iot-data get-thing-shadow \
```

```
    --thing-name MyRPi \
    output.txt
```

Il comando non restituisce output sullo schermo, ma di seguito è riportato il contenuto di `output.txt`:

```
{
  "state":{
    "reported":{
    "moisture":"low"
    }
  },
  "metadata":{
    "reported":{
      "moisture":{
        "timestamp":1560269319
      }
    }
  },
  "version":1,"timestamp":1560269405
}
```

Per ulteriori informazioni, consulta Flusso di dati del servizio Device Shadow nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta GetThingShadow AWS CLICommand Reference.

`update-thing-shadow`

Il seguente esempio di codice mostra come utilizzare`update-thing-shadow`.

AWS CLI

Come aggiornare il device shadow di un oggetto

L'esempio `update-thing-shadow` seguente modifica lo stato corrente del device shadow per l'oggetto specificato e lo salva nel file `output.txt`.

```
aws iot-data update-thing-shadow \
    --thing-name MyRPi \
    --payload "{"state":{"reported":{"moisture":"okay"}}}" \
```

```
    "output.txt"
```

Il comando non restituisce output sullo schermo, ma di seguito è riportato il contenuto di `output.txt`:

```
{
    "state": {
        "reported": {
            "moisture": "okay"
        }
    },
    "metadata": {
        "reported": {
            "moisture": {
                "timestamp": 1560270036
            }
        }
    },
    "version": 2,
    "timestamp": 1560270036
}
```

Per ulteriori informazioni, consulta Flusso di dati del servizio Device Shadow nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta UpdateThingShadow AWS CLICommand Reference.

AWS IoT Events esempi utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with AWS IoT Events.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`batch-put-message`

Il seguente esempio di codice mostra come utilizzare`batch-put-message`.

AWS CLI

Per inviare messaggi (input) a AWS IoT Events

L'`batch-put-message`esempio seguente invia una serie di messaggi al sistema AWS IoT Events. Ogni payload del messaggio viene trasformato nell'input specificato (`inputName`) e inserito in tutti i rilevatori che monitorano tale input. Se vengono inviati più messaggi, l'ordine in cui i messaggi vengono elaborati non è garantito. Per garantire l'ordine, è necessario inviare i messaggi uno alla volta e attendere una risposta corretta.

```
aws iotevents-data batch-put-message \
    --cli-input-json file://highPressureMessage.json
```

Contenuto di `highPressureMessage.json`:

```
{
    "messages": [
        {
            "messageId": "00001",
            "inputName": "PressureInput",
            "payload": "{\"motorid\": \"Fulton-A32\", \"sensorData\": {\"pressure\":
 80, \"temperature\": 39} }"
        }
    ]
}
```

Output:

```
{
    "BatchPutMessageErrorEntries": []
}
```

Per ulteriori informazioni, consulta BatchPutMessagela sezione AWS IoT Events API Reference.

- Per i dettagli sull'API, consulta BatchPutMessage AWS CLICommand Reference.

`batch-update-detector`

Il seguente esempio di codice mostra come utilizzare`batch-update-detector`.

AWS CLI

Come aggiornare un rilevatore (istanza)

L'esempio `batch-update-detector` seguente aggiorna lo stato, i valori delle variabili e le impostazioni del timer di uno o più rilevatori (istanze) di un modello di rilevatore specificato.

```
aws iotevents-data batch-update-detector \
    --cli-input-json file://budFulton-A32.json
```

Contenuto di `budFulton-A32.json`:

```
{
    "detectors": [
        {
            "messageId": "00001",
            "detectorModelName": "motorDetectorModel",
            "keyValue": "Fulton-A32",
            "state": {
                "stateName": "Normal",
                "variables": [
                    {
                        "name": "pressureThresholdBreached",
                        "value": "0"
                    }
                ],
                "timers": [
                ]
            }
        }
    ]
}
```

Output:

```
{
    "batchUpdateDetectorErrorEntries": []
}
```

Per ulteriori informazioni, consulta BatchUpdateDetectorla sezione AWS IoT Events API Reference.

- Per i dettagli sull'API, consulta BatchUpdateDetector AWS CLICommand Reference.

`create-detector-model`

Il seguente esempio di codice mostra come utilizzare`create-detector-model`.

AWS CLI

Come creare un modello di rivelatore

L'esempio `create-detector-model` seguente crea un modello di rilevatore con la sua configurazione specificata da un file di parametri.

```
aws iotevents create-detector-model  \
    --cli-input-json file://motorDetectorModel.json
```

Contenuto di `motorDetectorModel.json`:

```
{
    "detectorModelName": "motorDetectorModel",
    "detectorModelDefinition": {
        "states": [
            {
                "stateName": "Normal",
                "onEnter": {
                    "events": [
                        {
                            "eventName": "init",
                            "condition": "true",
                            "actions": [
                                {
                                    "setVariable": {
                                        "variableName": "pressureThresholdBreached",
                                        "value": "0"
                                    }
                                }
                            ]
                        }
                    ]
                },
```

```
                "onInput": {
                    "transitionEvents": [
                        {
                            "eventName": "Overpressurized",
                            "condition": "$input.PressureInput.sensorData.pressure
 &gt; 70",
                            "actions": [
                                {
                                    "setVariable": {
                                        "variableName": "pressureThresholdBreached",
                                        "value":
 "$variable.pressureThresholdBreached + 3"
                                    }
                                }
                            ],
                            "nextState": "Dangerous"
                        }
                    ]
                }
            },
            {
                "stateName": "Dangerous",
                "onEnter": {
                    "events": [
                        {
                            "eventName": "Pressure Threshold Breached",
                            "condition": "$variable.pressureThresholdBreached &gt;
 1",
                            "actions": [
                                {
                                    "sns": {
                                        "targetArn": "arn:aws:sns:us-
east-1:123456789012:underPressureAction"
                                    }
                                }
                            ]
                        }
                    ]
                },
                "onInput": {
                    "events": [
                        {
                            "eventName": "Overpressurized",
```

```
                            "condition": "$input.PressureInput.sensorData.pressure
&gt; 70",
                            "actions": [
                                {
                                    "setVariable": {
                                        "variableName": "pressureThresholdBreached",
                                        "value": "3"
                                    }
                                }
                            ]
                        },
                        {
                            "eventName": "Pressure Okay",
                            "condition": "$input.PressureInput.sensorData.pressure
&lt;= 70",
                            "actions": [
                                {
                                    "setVariable": {
                                        "variableName": "pressureThresholdBreached",
                                        "value":
"$variable.pressureThresholdBreached - 1"
                                    }
                                }
                            ]
                        }
                    ],
                    "transitionEvents": [
                        {
                            "eventName": "BackToNormal",
                            "condition": "$input.PressureInput.sensorData.pressure
&lt;= 70 &amp;&amp; $variable.pressureThresholdBreached &lt;= 1",
                            "nextState": "Normal"
                        }
                    ]
                },
                "onExit": {
                    "events": [
                        {
                            "eventName": "Normal Pressure Restored",
                            "condition": "true",
                            "actions": [
                                {
                                    "sns": {
```

```
                                        "targetArn": "arn:aws:sns:us-
east-1:123456789012:pressureClearedAction"
                                    }
                                }
                            ]
                        }
                    ]
                }
            }
        ],
        "initialStateName": "Normal"
    },
    "key": "motorid",
    "roleArn": "arn:aws:iam::123456789012:role/IoTEventsRole"
}
```

Output:

```
{
    "detectorModelConfiguration": {
        "status": "ACTIVATING",
        "lastUpdateTime": 1560796816.077,
        "roleArn": "arn:aws:iam::123456789012:role/IoTEventsRole",
        "creationTime": 1560796816.077,
        "detectorModelArn": "arn:aws:iotevents:us-west-2:123456789012:detectorModel/
motorDetectorModel",
        "key": "motorid",
        "detectorModelName": "motorDetectorModel",
        "detectorModelVersion": "1"
    }
}
```

Per ulteriori informazioni, consulta CreateDetectorModella sezione AWS IoT Events API Reference.

- Per i dettagli sull'API, consulta CreateDetectorModel AWS CLICommand Reference.

`create-input`

Il seguente esempio di codice mostra come utilizzare`create-input`.

AWS CLI

Come creare un input

L'esempio `create-input` seguente crea un input.

```
aws iotevents create-input  \
    --cli-input-json file://pressureInput.json
```

Contenuto di `pressureInput.json`:

```
{
    "inputName": "PressureInput",
    "inputDescription": "Pressure readings from a motor",
    "inputDefinition": {
        "attributes": [
            { "jsonPath": "sensorData.pressure" },
            { "jsonPath": "motorid" }
        ]
    }
}
```

Output:

```
{
    "inputConfiguration": {
        "status": "ACTIVE",
        "inputArn": "arn:aws:iotevents:us-west-2:123456789012:input/PressureInput",
        "lastUpdateTime": 1560795312.542,
        "creationTime": 1560795312.542,
        "inputName": "PressureInput",
        "inputDescription": "Pressure readings from a motor"
    }
}
```

Per ulteriori informazioni, consulta CreateInputla sezione AWS IoT Events API Reference.

- Per i dettagli sull'API, consulta CreateInput AWS CLICommand Reference.

`delete-detector-model`

Il seguente esempio di codice mostra come utilizzare`delete-detector-model`.

AWS CLI

Come eliminare un modello di rilevatore

L'esempio `delete-detector-model` seguente elimina un modello di rilevatore specificato. Vengono eliminate anche tutte le istanze attive del modello di rilevatore.

```
aws iotevents delete-detector-model \
    --detector-model-name motorDetectorModel
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta DeleteDetectorModella sezione AWS IoT Events API Reference.

- Per i dettagli sull'API, consulta DeleteDetectorModel AWS CLICommand Reference.

`delete-input`

Il seguente esempio di codice mostra come utilizzare`delete-input`.

AWS CLI

Come eliminare un input

L'esempio `delete-input` seguente elimina l'input specificato.

```
aws iotevents delete-input \
    --input-name PressureInput
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta DeleteInputla sezione AWS IoT Events API Reference.

- Per i dettagli sull'API, consulta DeleteInput AWS CLICommand Reference.

`describe-detector-model`

Il seguente esempio di codice mostra come utilizzare`describe-detector-model`.

AWS CLI

Come ottenere informazioni su un modello di rilevatore

L'esempio `describe-detector-model` seguente visualizza i dettagli di un modello di rilevatore specificato. Poiché il parametro `version` non è specificato, vengono restituite informazioni sulla versione più recente.

```
aws iotevents describe-detector-model \
    --detector-model-name motorDetectorModel
```

Output:

```
{
    "detectorModel": {
        "detectorModelConfiguration": {
            "status": "ACTIVE",
            "lastUpdateTime": 1560796816.077,
            "roleArn": "arn:aws:iam::123456789012:role/IoTEventsRole",
            "creationTime": 1560796816.077,
            "detectorModelArn": "arn:aws:iotevents:us-
west-2:123456789012:detectorModel/motorDetectorModel",
            "key": "motorid",
            "detectorModelName": "motorDetectorModel",
            "detectorModelVersion": "1"
        },
        "detectorModelDefinition": {
            "states": [
                {
                    "onInput": {
                        "transitionEvents": [
                            {
                                "eventName": "Overpressurized",
                                "actions": [
                                    {
                                        "setVariable": {
                                            "variableName":
 "pressureThresholdBreached",
                                            "value":
 "$variable.pressureThresholdBreached + 3"
                                        }
                                    }
                                ],
                                "condition":
 "$input.PressureInput.sensorData.pressure > 70",
                                "nextState": "Dangerous"
                            }
```

```
                        ],
                        "events": []
                    },
                    "stateName": "Normal",
                    "onEnter": {
                        "events": [
                            {
                                "eventName": "init",
                                "actions": [
                                    {
                                        "setVariable": {
                                            "variableName":
"pressureThresholdBreached",
                                            "value": "0"
                                        }
                                    }
                                ],
                                "condition": "true"
                            }
                        ]
                    },
                    "onExit": {
                        "events": []
                    }
                },
                {
                    "onInput": {
                        "transitionEvents": [
                            {
                                "eventName": "BackToNormal",
                                "actions": [],
                                "condition":
"$input.PressureInput.sensorData.pressure <= 70 &&
$variable.pressureThresholdBreached <= 1",
                                "nextState": "Normal"
                            }
                        ],
                        "events": [
                            {
                                "eventName": "Overpressurized",
                                "actions": [
                                    {
                                        "setVariable": {
```

```
                                            "variableName":
 "pressureThresholdBreached",
                                            "value": "3"
                                        }
                                    }
                                ],
                                "condition":
 "$input.PressureInput.sensorData.pressure > 70"
                            },
                            {
                                "eventName": "Pressure Okay",
                                "actions": [
                                    {
                                        "setVariable": {
                                            "variableName":
 "pressureThresholdBreached",
                                            "value":
 "$variable.pressureThresholdBreached - 1"
                                        }
                                    }
                                ],
                                "condition":
 "$input.PressureInput.sensorData.pressure <= 70"
                            }
                        ]
                    },
                    "stateName": "Dangerous",
                    "onEnter": {
                        "events": [
                            {
                                "eventName": "Pressure Threshold Breached",
                                "actions": [
                                    {
                                        "sns": {
                                            "targetArn": "arn:aws:sns:us-
east-1:123456789012:underPressureAction"
                                        }
                                    }
                                ],
                                "condition": "$variable.pressureThresholdBreached >
 1"
                            }
                        ]
                    },
```

```
                    "onExit": {
                        "events": [
                            {
                                "eventName": "Normal Pressure Restored",
                                "actions": [
                                    {
                                        "sns": {
                                            "targetArn": "arn:aws:sns:us-
east-1:123456789012:pressureClearedAction"
                                        }
                                    }
                                ],
                                "condition": "true"
                            }
                        ]
                    }
                }
            ],
            "initialStateName": "Normal"
        }
    }
}
```

Per ulteriori informazioni, consulta DescribeDetectorModella sezione AWS IoT Events API Reference.

- Per i dettagli sull'API, consulta DescribeDetectorModel AWS CLICommand Reference.

`describe-detector`

Il seguente esempio di codice mostra come utilizzare`describe-detector`.

AWS CLI

Come ottenere informazioni su un rilevatore (istanza).

L'esempio `describe-detector` seguente visualizza i dettagli del tipo di rilevatore (istanza) specificato.

```
aws iotevents-data describe-detector \
    --detector-model-name motorDetectorModel \
    --key-value "Fulton-A32"
```

Output:

```
{
    "detector": {
        "lastUpdateTime": 1560797852.776,
        "creationTime": 1560797852.775,
        "state": {
            "variables": [
                {
                    "name": "pressureThresholdBreached",
                    "value": "3"
                }
            ],
            "stateName": "Dangerous",
            "timers": []
        },
        "keyValue": "Fulton-A32",
        "detectorModelName": "motorDetectorModel",
        "detectorModelVersion": "1"
    }
}
```

Per ulteriori informazioni, consulta DescribeDetectorla sezione AWS IoT Events API Reference.

- Per i dettagli sull'API, consulta DescribeDetector AWS CLICommand Reference.

`describe-input`

Il seguente esempio di codice mostra come utilizzare`describe-input`.

AWS CLI

Come ottenere informazioni su un input

L'esempio `describe-input` seguente visualizza i dettagli dell'input specificato.

```
aws iotevents describe-input \
    --input-name PressureInput
```

Output:

```
{
    "input": {
```

```
        "inputConfiguration": {
            "status": "ACTIVE",
            "inputArn": "arn:aws:iotevents:us-west-2:123456789012:input/
PressureInput",
            "lastUpdateTime": 1560795312.542,
            "creationTime": 1560795312.542,
            "inputName": "PressureInput",
            "inputDescription": "Pressure readings from a motor"
        },
        "inputDefinition": {
            "attributes": [
                {
                    "jsonPath": "sensorData.pressure"
                },
                {
                    "jsonPath": "motorid"
                }
            ]
        }
    }
}
```

Per ulteriori informazioni, consulta DescribeInputla sezione AWS IoT Events API Reference.

- Per i dettagli sull'API, consulta DescribeInput AWS CLICommand Reference.

`describe-logging-options`

Il seguente esempio di codice mostra come utilizzare`describe-logging-options`.

AWS CLI

Come ottenere informazioni sulle impostazioni della registrazione di log

L'`describe-logging-options`esempio seguente recupera le impostazioni correnti delle opzioni di registrazione degli eventi AWS IoT.

```
aws iotevents describe-logging-options
```

Output:

```
{
    "loggingOptions": {
```

```
        "roleArn": "arn:aws:iam::123456789012:role/IoTEventsRole",
        "enabled": false,
        "level": "ERROR"
    }
}
```

Per ulteriori informazioni, consulta DescribeLoggingOptionsla sezione AWS IoT Events API Reference.

- Per i dettagli sull'API, consulta DescribeLoggingOptions AWS CLICommand Reference.

`list-detector-model-versions`

Il seguente esempio di codice mostra come utilizzare`list-detector-model-versions`.

AWS CLI

Come ottenere informazioni sulle versioni di un modello di rilevatore

L'esempio `list-detector-model-versions` seguente elenca tutte le versioni di un modello di rilevatore. Vengono restituiti solo i metadati associati a ciascuna versione del modello di rivelatore.

```
aws iotevents list-detector-model-versions \
    --detector-model-name motorDetectorModel
```

Output:

```
{
    "detectorModelVersionSummaries": [
        {
            "status": "ACTIVE",
            "lastUpdateTime": 1560796816.077,
            "roleArn": "arn:aws:iam::123456789012:role/IoTEventsRole",
            "creationTime": 1560796816.077,
            "detectorModelArn": "arn:aws:iotevents:us-
west-2:123456789012:detectorModel/motorDetectorModel",
            "detectorModelName": "motorDetectorModel",
            "detectorModelVersion": "1"
        }
    ]
}
```

Per ulteriori informazioni, consulta ListDetectorModelVersionsla sezione AWS IoT Events API Reference.

- Per i dettagli sull'API, consulta ListDetectorModelVersions AWS CLICommand Reference.

`list-detector-models`

Il seguente esempio di codice mostra come utilizzare`list-detector-models`.

AWS CLI

Come ottenere un elenco dei modelli di rilevatori

L'esempio `list-detector-models` seguente elenca i modelli di rilevatore creati. Vengono restituiti solo i metadati associati a ciascun modello di rivelatore.

```
aws iotevents list-detector-models
```

Output:

```
{
    "detectorModelSummaries": [
        {
            "detectorModelName": "motorDetectorModel",
            "creationTime": 1552072424.212
            "detectorModelDescription": "Detect overpressure in a motor."
        }
    ]
}
```

Per ulteriori informazioni, consulta ListDetectorModelsla sezione AWS IoT Events API Reference.

- Per i dettagli sull'API, consulta ListDetectorModels AWS CLICommand Reference.

`list-detectors`

Il seguente esempio di codice mostra come utilizzare`list-detectors`.

AWS CLI

Come ottenere un elenco di rilevatori per un modello di rilevatore

L'esempio `list-detectors` seguente elenca i rilevatori (le istanze di un modello di rilevatore) presenti nel tuo account.

```
aws iotevents-data list-detectors \
    --detector-model-name motorDetectorModel
```

Output:

```
{
    "detectorSummaries": [
        {
            "lastUpdateTime": 1558129925.2,
            "creationTime": 1552073155.527,
            "state": {
                "stateName": "Normal"
            },
            "keyValue": "Fulton-A32",
            "detectorModelName": "motorDetectorModel",
            "detectorModelVersion": "1"
        }
    ]
}
```

Per ulteriori informazioni, consulta ListDetectorsla sezione AWS IoT Events API Reference.

- Per i dettagli sull'API, consulta ListDetectors AWS CLICommand Reference.

`list-inputs`

Il seguente esempio di codice mostra come utilizzare`list-inputs`.

AWS CLI

Come elencare gli input

L'esempio `list-inputs` seguente elenca gli input creati nel tuo account.

```
aws iotevents list-inputs
```

Questo comando non produce alcun output. Output:

```
{
```

```
    {
        "status": "ACTIVE",
        "inputArn": "arn:aws:iotevents:us-west-2:123456789012:input/PressureInput",
        "lastUpdateTime": 1551742986.768,
        "creationTime": 1551742986.768,
        "inputName": "PressureInput",
        "inputDescription": "Pressure readings from a motor"
    }
}
```

Per ulteriori informazioni, consulta ListInputsla sezione AWS IoT Events API Reference.

- Per i dettagli sull'API, consulta ListInputs AWS CLICommand Reference.

`list-tags-for-resource`

Il seguente esempio di codice mostra come utilizzare`list-tags-for-resource`.

AWS CLI

Come elencare i tag assegnati a una risorsa.

L'esempio `list-tags-for-resource` seguente elenca i nomi e i valori delle chiavi dei tag assegnati alla risorsa.

```
aws iotevents list-tags-for-resource \
    --resource-arn "arn:aws:iotevents:us-west-2:123456789012:input/PressureInput"
```

Output:

```
{
    "tags": [
        {
            "value": "motor",
            "key": "deviceType"
        }
    ]
}
```

Per ulteriori informazioni, consulta ListTagsForResourcela sezione AWS IoT Events API Reference.

- Per i dettagli sull'API, consulta ListTagsForResource AWS CLICommand Reference.

`put-logging-options`

Il seguente esempio di codice mostra come utilizzare`put-logging-options`.

AWS CLI

Come impostare le opzioni di registrazione di log

L'`put-logging-options`esempio seguente imposta o aggiorna le opzioni di registrazione degli eventi AWS IoT. Se aggiorni il valore di un campo `loggingOptions` field, it can take up to one minute for the change to take effect. Also, if you change the policy attached to the role you specified in the ``roleArn` (ad esempio per correggere un criterio non valido) possono essere necessari fino a cinque minuti per rendere effettiva la modifica.

```
aws iotevents put-logging-options \
    --cli-input-json file://logging-options.json
```

Contenuto di `logging-options.json`:

```
{
    "loggingOptions": {
        "roleArn": "arn:aws:iam::123456789012:role/IoTEventsRole",
        "level": "DEBUG",
        "enabled": true,
        "detectorDebugOptions": [
            {
                "detectorModelName": "motorDetectorModel",
                "keyValue": "Fulton-A32"
            }
        ]
    }
}
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta PutLoggingOptionsla sezione AWS IoT Events API Reference.

- Per i dettagli sull'API, consulta PutLoggingOptions AWS CLICommand Reference.

`tag-resource`

Il seguente esempio di codice mostra come utilizzare`tag-resource`.

AWS CLI

Come aggiungere tag a una risorsa

L'esempio `tag-resource` seguente aggiunge o modifica (se la chiave `deviceType` esiste già) il tag associato alla risorsa specificata.

```
aws iotevents tag-resource \
    --cli-input-json file://pressureInput.tag.json
```

Contenuto di `pressureInput.tag.json`:

```
{
    "resourceArn": "arn:aws:iotevents:us-west-2:123456789012:input/PressureInput",
    "tags": [
        {
            "key": "deviceType",
            "value": "motor"
        }
    ]
}
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta TagResourcela sezione AWS IoT Events API Reference.

- Per i dettagli sull'API, consulta TagResource AWS CLICommand Reference.

`untag-resource`

Il seguente esempio di codice mostra come utilizzare`untag-resource`.

AWS CLI

Come rimuovere tag da una risorsa

L'esempio `untag-resource` seguente rimuove il tag con il nome di chiave specificato dalla risorsa specificata.

```
aws iotevents untag-resource \
    --resource-arn arn:aws:iotevents:us-west-2:123456789012:input/PressureInput \
    --tagkeys deviceType
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta UntagResourcela sezione AWS IoT Events API Reference.

- Per i dettagli sull'API, consulta UntagResource AWS CLICommand Reference.

`update-detector-model`

Il seguente esempio di codice mostra come utilizzareu`pdate-detector-model`.

AWS CLI

Come aggiornare un modello di rivelatore

L'esempio `update-detector-model` seguente aggiorna il modello di rilevatore specificato. I rilevatori (istanze) generati dalla versione precedente vengono eliminati e quindi ricreati all'arrivo di nuovi input.

```
aws iotevents update-detector-model \
    --cli-input-json file://motorDetectorModel.update.json
```

Contenuto di `motorDetectorModel.update.json`:

```
{
    "detectorModelName": "motorDetectorModel",
    "detectorModelDefinition": {
        "states": [
            {
                "stateName": "Normal",
                "onEnter": {
                    "events": [
                        {
                            "eventName": "init",
                            "condition": "true",
                            "actions": [
                                {
                                    "setVariable": {
                                        "variableName": "pressureThresholdBreached",
```

```
                                        "value": "0"
                                    }
                                }
                            ]
                        }
                    ]
                },
                "onInput": {
                    "transitionEvents": [
                        {
                            "eventName": "Overpressurized",
                            "condition": "$input.PressureInput.sensorData.pressure >
 70",
                            "actions": [
                                {
                                    "setVariable": {
                                        "variableName": "pressureThresholdBreached",
                                        "value":
 "$variable.pressureThresholdBreached + 3"
                                    }
                                }
                            ],
                            "nextState": "Dangerous"
                        }
                    ]
                }
            },
            {
                "stateName": "Dangerous",
                "onEnter": {
                    "events": [
                        {
                            "eventName": "Pressure Threshold Breached",
                            "condition": "$variable.pressureThresholdBreached > 1",
                            "actions": [
                                {
                                    "sns": {
                                        "targetArn": "arn:aws:sns:us-
east-1:123456789012:underPressureAction"
                                    }
                                }
                            ]
                        }
                    ]
```

```
                },
                "onInput": {
                    "events": [
                        {
                            "eventName": "Overpressurized",
                            "condition": "$input.PressureInput.sensorData.pressure >
70",
                            "actions": [
                                {
                                    "setVariable": {
                                        "variableName": "pressureThresholdBreached",
                                        "value": "3"
                                    }
                                }
                            ]
                        },
                        {
                            "eventName": "Pressure Okay",
                            "condition": "$input.PressureInput.sensorData.pressure
<= 70",
                            "actions": [
                                {
                                    "setVariable": {
                                        "variableName": "pressureThresholdBreached",
                                        "value":
"$variable.pressureThresholdBreached - 1"
                                    }
                                }
                            ]
                        }
                    ],
                    "transitionEvents": [
                        {
                            "eventName": "BackToNormal",
                            "condition": "$input.PressureInput.sensorData.pressure
<= 70 && $variable.pressureThresholdBreached <= 1",
                            "nextState": "Normal"
                        }
                    ]
                },
                "onExit": {
                    "events": [
                        {
                            "eventName": "Normal Pressure Restored",
```

```
                            "condition": "true",
                            "actions": [
                                {
                                    "sns": {
                                        "targetArn": "arn:aws:sns:us-
east-1:123456789012:pressureClearedAction"
                                    }
                                }
                            ]
                        }
                    ]
                }
            }
        ],
        "initialStateName": "Normal"
    },
    "roleArn": "arn:aws:iam::123456789012:role/IoTEventsRole"
}
```

Output:

```
{
    "detectorModelConfiguration": {
        "status": "ACTIVATING",
        "lastUpdateTime": 1560799387.719,
        "roleArn": "arn:aws:iam::123456789012:role/IoTEventsRole",
        "creationTime": 1560799387.719,
        "detectorModelArn": "arn:aws:iotevents:us-west-2:123456789012:detectorModel/
motorDetectorModel",
        "key": "motorid",
        "detectorModelName": "motorDetectorModel",
        "detectorModelVersion": "2"
    }
}
```

Per ulteriori informazioni, consulta UpdateDetectorModella sezione AWS IoT Events API Reference.

- Per i dettagli sull'API, consulta UpdateDetectorModel AWS CLICommand Reference.

`update-input`

Il seguente esempio di codice mostra come utilizzare`update-input`.

AWS CLI

Come aggiornare un input

L'esempio `update-input` seguente aggiorna l'input specificato con una nuova descrizione e definizione.

```
aws iotevents update-input \
    --cli-input-json file://pressureInput.json
```

Contenuto di `pressureInput.json`:

```
{
    "inputName": "PressureInput",
    "inputDescription": "Pressure readings from a motor",
    "inputDefinition": {
        "attributes": [
            { "jsonPath": "sensorData.pressure" },
            { "jsonPath": "motorid" }
        ]
    }
}
```

Output:

```
{
    "inputConfiguration": {
        "status": "ACTIVE",
        "inputArn": "arn:aws:iotevents:us-west-2:123456789012:input/PressureInput",
        "lastUpdateTime": 1560795976.458,
        "creationTime": 1560795312.542,
        "inputName": "PressureInput",
        "inputDescription": "Pressure readings from a motor"
    }
}
```

Per ulteriori informazioni, consulta UpdateInputla sezione AWS IoT Events API Reference.

- Per i dettagli sull'API, consulta UpdateInput AWS CLICommand Reference.

AWS IoT Events-Data esempi utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with AWS IoT Events-Data.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`batch-put-message`

Il seguente esempio di codice mostra come utilizzare`batch-put-message`.

AWS CLI

Per inviare messaggi (input) a AWS IoT Events

L'`batch-put-message`esempio seguente invia una serie di messaggi al sistema AWS IoT Events. Ogni payload del messaggio viene trasformato nell'input specificato (`inputName`) e inserito in tutti i rilevatori che monitorano tale input. Se vengono inviati più messaggi, l'ordine in cui i messaggi vengono elaborati non è garantito. Per garantire l'ordine, è necessario inviare i messaggi uno alla volta e attendere una risposta corretta.

```
aws iotevents-data batch-put-message \
    --cli-binary-format raw-in-base64-out \
    --cli-input-json file://highPressureMessage.json
```

Contenuto di `highPressureMessage.json`:

```
{
    "messages": [
        {
            "messageId": "00001",
```

```
            "inputName": "PressureInput",
            "payload": "{\"motorid\": \"Fulton-A32\", \"sensorData\": {\"pressure\":
 80, \"temperature\": 39} }"
        }
    ]
}
```

Output:

```
{
    "BatchPutMessageErrorEntries": []
}
```

Per ulteriori informazioni, consulta BatchPutMessagela AWS IoT Events Developer Guide*.

- Per i dettagli sull'API, consulta BatchPutMessage AWS CLICommand Reference.

`batch-update-detector`

Il seguente esempio di codice mostra come utilizzare`batch-update-detector`.

AWS CLI

Come aggiornare un rilevatore (istanza)

L'esempio `batch-update-detector` seguente aggiorna lo stato, i valori delle variabili e le impostazioni del timer di uno o più rilevatori (istanze) di un modello di rilevatore specificato.

```
aws iotevents-data batch-update-detector \
    --cli-input-json file://budFulton-A32.json
```

Contenuto di `budFulton-A32.json`:

```
{
    "detectors": [
        {
            "messageId": "00001",
            "detectorModelName": "motorDetectorModel",
            "keyValue": "Fulton-A32",
            "state": {
                "stateName": "Normal",
                "variables": [
```

```
                    {
                        "name": "pressureThresholdBreached",
                        "value": "0"
                    }
                ],
                "timers": [
                ]
            }
        }
    ]
}
```

Output:

```
{
    "batchUpdateDetectorErrorEntries": []
}
```

Per ulteriori informazioni, consulta BatchUpdateDetectorla AWS IoT Events Developer Guide*.

- Per i dettagli sull'API, consulta BatchUpdateDetector AWS CLICommand Reference.

`create-detector-model`

Il seguente esempio di codice mostra come utilizzare`create-detector-model`.

AWS CLI

Come creare un modello di rivelatore

L'esempio `create-detector-model` seguente crea un modello di rilevatore.

```
aws iotevents create-detector-model \
    --cli-input-json file://motorDetectorModel.json
```

Contenuto di `motorDetectorModel.json`:

```
{
    "detectorModelName": "motorDetectorModel",
    "detectorModelDefinition": {
        "states": [
            {
                "stateName": "Normal",
```

```
                "onEnter": {
                    "events": [
                        {
                            "eventName": "init",
                            "condition": "true",
                            "actions": [
                                {
                                    "setVariable": {
                                        "variableName": "pressureThresholdBreached",
                                        "value": "0"
                                    }
                                }
                            ]
                        }
                    ]
                },
                "onInput": {
                    "transitionEvents": [
                        {
                            "eventName": "Overpressurized",
                            "condition": "$input.PressureInput.sensorData.pressure
&gt; 70",
                            "actions": [
                                {
                                    "setVariable": {
                                        "variableName": "pressureThresholdBreached",
                                        "value":
"$variable.pressureThresholdBreached + 3"
                                    }
                                }
                            ],
                            "nextState": "Dangerous"
                        }
                    ]
                }
            },
            {
                "stateName": "Dangerous",
                "onEnter": {
                    "events": [
                        {
                            "eventName": "Pressure Threshold Breached",
                            "condition": "$variable.pressureThresholdBreached &gt;
1",
```

```
                            "actions": [
                                {
                                    "sns": {
                                        "targetArn": "arn:aws:sns:us-
east-1:123456789012:underPressureAction"
                                    }
                                }
                            ]
                        }
                    ]
                },
                "onInput": {
                    "events": [
                        {
                            "eventName": "Overpressurized",
                            "condition": "$input.PressureInput.sensorData.pressure
 &gt; 70",
                            "actions": [
                                {
                                    "setVariable": {
                                        "variableName": "pressureThresholdBreached",
                                        "value": "3"
                                    }
                                }
                            ]
                        },
                        {
                            "eventName": "Pressure Okay",
                            "condition": "$input.PressureInput.sensorData.pressure
 &lt;= 70",
                            "actions": [
                                {
                                    "setVariable": {
                                        "variableName": "pressureThresholdBreached",
                                        "value":
 "$variable.pressureThresholdBreached - 1"
                                    }
                                }
                            ]
                        }
                    ],
                    "transitionEvents": [
                        {
                            "eventName": "BackToNormal",
```

```
                            "condition": "$input.PressureInput.sensorData.pressure
 &lt;= 70 &amp;&amp; $variable.pressureThresholdBreached &lt;= 1",
                            "nextState": "Normal"
                        }
                    ]
                },
                "onExit": {
                    "events": [
                        {
                            "eventName": "Normal Pressure Restored",
                            "condition": "true",
                            "actions": [
                                {
                                    "sns": {
                                        "targetArn": "arn:aws:sns:us-
east-1:123456789012:pressureClearedAction"
                                    }
                                }
                            ]
                        }
                    ]
                }
            }
        ],
        "initialStateName": "Normal"
    },
    "key": "motorid",
    "roleArn": "arn:aws:iam::123456789012:role/IoTEventsRole"
}
```

Output:

```
{
    "detectorModelConfiguration": {
        "status": "ACTIVATING",
        "lastUpdateTime": 1560796816.077,
        "roleArn": "arn:aws:iam::123456789012:role/IoTEventsRole",
        "creationTime": 1560796816.077,
        "detectorModelArn": "arn:aws:iotevents:us-west-2:123456789012:detectorModel/
motorDetectorModel",
        "key": "motorid",
        "detectorModelName": "motorDetectorModel",
        "detectorModelVersion": "1"
```

```
    }
}
```

Per ulteriori informazioni, consulta CreateDetectorModella AWS IoT Events Developer Guide*.

- Per i dettagli sull'API, consulta CreateDetectorModel AWS CLICommand Reference.

`create-input`

Il seguente esempio di codice mostra come utilizzare`create-input`.

AWS CLI

Come creare un input

L'esempio `create-input` seguente crea un input.

```
aws iotevents create-input \
    --cli-input-json file://pressureInput.json
```

Contenuto di `pressureInput.json`:

```
{
    "inputName": "PressureInput",
    "inputDescription": "Pressure readings from a motor",
    "inputDefinition": {
        "attributes": [
                { "jsonPath": "sensorData.pressure" },
                { "jsonPath": "motorid" }
            ]
    }
}
```

Output:

```
{
    "inputConfiguration": {
        "status": "ACTIVE",
        "inputArn": "arn:aws:iotevents:us-west-2:123456789012:input/PressureInput",
        "lastUpdateTime": 1560795312.542,
        "creationTime": 1560795312.542,
        "inputName": "PressureInput",
        "inputDescription": "Pressure readings from a motor"
```

```
    }
}
```

Per ulteriori informazioni, consulta CreateInputla AWS IoT Events Developer Guide*.

- Per i dettagli sull'API, consulta CreateInput AWS CLICommand Reference.

`delete-detector-model`

Il seguente esempio di codice mostra come utilizzare`delete-detector-model`.

AWS CLI

Come eliminare un modello di rilevatore

L'esempio `delete-detector-model` seguente elimina un modello di rilevatore. Vengono eliminate anche tutte le istanze attive del modello di rilevatore.

```
aws iotevents delete-detector-model \
    --detector-model-name motorDetectorModel*
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta DeleteDetectorModella AWS IoT Events Developer Guide*.

- Per i dettagli sull'API, consulta DeleteDetectorModel AWS CLICommand Reference.

`delete-input`

Il seguente esempio di codice mostra come utilizzare`delete-input`.

AWS CLI

Come eliminare un input

L'esempio `delete-input` seguente elimina un input.

```
aws iotevents delete-input \
    --input-name PressureInput
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta DeleteInputla AWS IoT Events Developer Guide*.

- Per i dettagli sull'API, consulta DeleteInput AWS CLICommand Reference.

`describe-detector-model`

Il seguente esempio di codice mostra come utilizzare`describe-detector-model`.

AWS CLI

Come ottenere informazioni su un modello di rilevatore

L'esempio `describe-detector-model` seguente descrive un modello di rilevatore. Se il parametro `version` non è specificato, il comando restituisce le informazioni sulla versione più recente.

```
aws iotevents describe-detector-model \
    --detector-model-name motorDetectorModel
```

Output:

```
{
    "detectorModel": {
        "detectorModelConfiguration": {
            "status": "ACTIVE",
            "lastUpdateTime": 1560796816.077,
            "roleArn": "arn:aws:iam::123456789012:role/IoTEventsRole",
            "creationTime": 1560796816.077,
            "detectorModelArn": "arn:aws:iotevents:us-
west-2:123456789012:detectorModel/motorDetectorModel",
            "key": "motorid",
            "detectorModelName": "motorDetectorModel",
            "detectorModelVersion": "1"
        },
        "detectorModelDefinition": {
            "states": [
                {
                    "onInput": {
                        "transitionEvents": [
                            {
                                "eventName": "Overpressurized",
                                "actions": [
                                    {
                                        "setVariable": {
```

```
                                            "variableName":
"pressureThresholdBreached",
                                            "value":
"$variable.pressureThresholdBreached + 3"
                                        }
                                    }
                                ],
                                "condition":
"$input.PressureInput.sensorData.pressure > 70",
                                "nextState": "Dangerous"
                            }
                        ],
                        "events": []
                    },
                    "stateName": "Normal",
                    "onEnter": {
                        "events": [
                            {
                                "eventName": "init",
                                "actions": [
                                    {
                                        "setVariable": {
                                            "variableName":
"pressureThresholdBreached",
                                            "value": "0"
                                        }
                                    }
                                ],
                                "condition": "true"
                            }
                        ]
                    },
                    "onExit": {
                        "events": []
                    }
                },
                {
                    "onInput": {
                        "transitionEvents": [
                            {
                                "eventName": "BackToNormal",
                                "actions": [],
```

```
                                "condition":
"$input.PressureInput.sensorData.pressure <= 70 &&
$variable.pressureThresholdBreached <= 1",
                                "nextState": "Normal"
                            }
                        ],
                        "events": [
                            {
                                "eventName": "Overpressurized",
                                "actions": [
                                    {
                                        "setVariable": {
                                            "variableName":
"pressureThresholdBreached",
                                            "value": "3"
                                        }
                                    }
                                ],
                                "condition":
"$input.PressureInput.sensorData.pressure > 70"
                            },
                            {
                                "eventName": "Pressure Okay",
                                "actions": [
                                    {
                                        "setVariable": {
                                            "variableName":
"pressureThresholdBreached",
                                            "value":
"$variable.pressureThresholdBreached - 1"
                                        }
                                    }
                                ],
                                "condition":
"$input.PressureInput.sensorData.pressure <= 70"
                            }
                        ]
                    },
                    "stateName": "Dangerous",
                    "onEnter": {
                        "events": [
                            {
                                "eventName": "Pressure Threshold Breached",
                                "actions": [
```

```
                                    {
                                        "sns": {
                                            "targetArn": "arn:aws:sns:us-
east-1:123456789012:underPressureAction"
                                        }
                                    }
                                ],
                                "condition": "$variable.pressureThresholdBreached >
 1"
                            }
                        ]
                    },
                    "onExit": {
                        "events": [
                            {
                                "eventName": "Normal Pressure Restored",
                                "actions": [
                                    {
                                        "sns": {
                                            "targetArn": "arn:aws:sns:us-
east-1:123456789012:pressureClearedAction"
                                        }
                                    }
                                ],
                                "condition": "true"
                            }
                        ]
                    }
                }
            ],
            "initialStateName": "Normal"
        }
    }
}
```

Per ulteriori informazioni, consulta DescribeDetectorModella AWS IoT Events Developer Guide*.

- Per i dettagli sull'API, consulta DescribeDetectorModel AWS CLICommand Reference.

`describe-detector`

Il seguente esempio di codice mostra come utilizzare`describe-detector`.

AWS CLI

Come ottenere informazioni su un rilevatore (istanza)

L'esempio `describe-detector` seguente restituisce informazioni sul rilevatore specificato (istanza).

```
aws iotevents-data describe-detector \
    --detector-model-name motorDetectorModel \
    --key-value "Fulton-A32"
```

Output:

```
{
    "detector": {
        "lastUpdateTime": 1560797852.776,
        "creationTime": 1560797852.775,
        "state": {
            "variables": [
                {
                    "name": "pressureThresholdBreached",
                    "value": "3"
                }
            ],
            "stateName": "Dangerous",
            "timers": []
        },
        "keyValue": "Fulton-A32",
        "detectorModelName": "motorDetectorModel",
        "detectorModelVersion": "1"
    }
}
```

Per ulteriori informazioni, consulta DescribeDetectorla AWS IoT Events Developer Guide*.

- Per i dettagli sull'API, consulta DescribeDetector AWS CLICommand Reference.

`describe-input`

Il seguente esempio di codice mostra come utilizzare`describe-input`.

AWS CLI

Come ottenere informazioni su un input

L'esempio `describe-input` seguente recupera i dettagli di un input.

```
aws iotevents describe-input \
    --input-name PressureInput
```

Output:

```
{
    "input": {
        "inputConfiguration": {
            "status": "ACTIVE",
            "inputArn": "arn:aws:iotevents:us-west-2:123456789012:input/
PressureInput",
            "lastUpdateTime": 1560795312.542,
            "creationTime": 1560795312.542,
            "inputName": "PressureInput",
            "inputDescription": "Pressure readings from a motor"
        },
        "inputDefinition": {
            "attributes": [
                {
                    "jsonPath": "sensorData.pressure"
                },
                {
                    "jsonPath": "motorid"
                }
            ]
        }
    }
}
```

Per ulteriori informazioni, consulta DescribeInputla AWS IoT Events Developer Guide*.

- Per i dettagli sull'API, consulta DescribeInput AWS CLICommand Reference.

`describe-logging-options`

Il seguente esempio di codice mostra come utilizzare`describe-logging-options`.

AWS CLI

Come ottenere informazioni sulle impostazioni della registrazione di log

L'`describe-logging-options`esempio seguente recupera le opzioni di registrazione degli eventi AWS IoT Events correnti.

```
aws iotevents describe-logging-options
```

Output:

```
{
    "loggingOptions": {
        "roleArn": "arn:aws:iam::123456789012:role/IoTEventsRole",
        "enabled": false,
        "level": "ERROR"
    }
}
```

Per ulteriori informazioni, consulta DescribeLoggingOptionsla AWS IoT Events Developer Guide*.

- Per i dettagli sull'API, consulta DescribeLoggingOptions AWS CLICommand Reference.

`list-detector-model-versions`

Il seguente esempio di codice mostra come utilizzare`list-detector-model-versions`.

AWS CLI

Come ottenere informazioni sulle versioni di un modello di rilevatore

L'esempio `list-detector-model-versions` seguente elenca tutte le versioni di un modello di rilevatore. Vengono restituiti solo i metadati associati a ciascuna versione del modello di rivelatore.

```
aws iotevents list-detector-model-versions \
    --detector-model-name motorDetectorModel
```

Output:

```
{
    "detectorModelVersionSummaries": [
```

```
        {
            "status": "ACTIVE",
            "lastUpdateTime": 1560796816.077,
            "roleArn": "arn:aws:iam::123456789012:role/IoTEventsRole",
            "creationTime": 1560796816.077,
            "detectorModelArn": "arn:aws:iotevents:us-
west-2:123456789012:detectorModel/motorDetectorModel",
            "detectorModelName": "motorDetectorModel",
            "detectorModelVersion": "1"
        }
    ]
}
```

Per ulteriori informazioni, consulta ListDetectorModelVersionsla AWS IoT Events Developer Guide*.

- Per i dettagli sull'API, consulta ListDetectorModelVersions AWS CLICommand Reference.

`list-detector-models`

Il seguente esempio di codice mostra come utilizzare`list-detector-models`.

AWS CLI

Come ottenere un elenco dei modelli di rilevatori

L'esempio `list-detector-models` seguente elenca i modelli di rilevatore creati. Vengono restituiti solo i metadati associati a ciascun modello di rivelatore.

```
aws iotevents list-detector-models
```

Output:

```
{
    "detectorModelSummaries": [
        {
            "detectorModelName": "motorDetectorModel",
            "creationTime": 1552072424.212
            "detectorModelDescription": "Detect overpressure in a motor."
        }
    ]
}
```

Per ulteriori informazioni, consulta ListDetectorModelsla AWS IoT Events Developer Guide*.

- Per i dettagli sull'API, consulta ListDetectorModels AWS CLICommand Reference.

`list-detectors`

Il seguente esempio di codice mostra come utilizzare`list-detectors`.

AWS CLI

Come ottenere un elenco di rilevatori per un modello di rilevatore

L'esempio `list-detectors` seguente elenca i rilevatori (le istanze di un modello di rivelatore).

```
aws iotevents-data list-detectors \
    --detector-model-name motorDetectorModel
```

Output:

```
{
    "detectorSummaries": [
        {
            "lastUpdateTime": 1558129925.2,
            "creationTime": 1552073155.527,
            "state": {
                "stateName": "Normal"
            },
            "keyValue": "Fulton-A32",
            "detectorModelName": "motorDetectorModel",
            "detectorModelVersion": "1"
        }
    ]
}
```

Per ulteriori informazioni, consulta ListDetectorsla AWS IoT Events Developer Guide*.

- Per i dettagli sull'API, consulta ListDetectors AWS CLICommand Reference.

`list-inputs`

Il seguente esempio di codice mostra come utilizzare`list-inputs`.

AWS CLI

Come elencare gli input

L'esempio `list-inputs` seguente elenca gli input creati.

```
aws iotevents list-inputs
```

Output:

```
{
    "status": "ACTIVE",
    "inputArn": "arn:aws:iotevents:us-west-2:123456789012:input/PressureInput",
    "lastUpdateTime": 1551742986.768,
    "creationTime": 1551742986.768,
    "inputName": "PressureInput",
    "inputDescription": "Pressure readings from a motor"
}
```

Per ulteriori informazioni, consulta ListInputsla AWS IoT Events Developer Guide*.

- Per i dettagli sull'API, consulta ListInputs AWS CLICommand Reference.

`list-tags-for-resource`

Il seguente esempio di codice mostra come utilizzare`list-tags-for-resource`.

AWS CLI

Come elencare i tag assegnati a una risorsa

L'esempio `list-tags-for-resource` seguente elenca i tag (metadati) assegnati alla risorsa.

```
aws iotevents list-tags-for-resource \
    --resource-arn "arn:aws:iotevents:us-west-2:123456789012:input/PressureInput"
```

Output:

```
{
    "tags": [
        {
            "value": "motor",
            "key": "deviceType"
```

```
        }
    ]
}
```

Per ulteriori informazioni, consulta ListTagsForResourcela AWS IoT Events Developer Guide*.

- Per i dettagli sull'API, consulta ListTagsForResource AWS CLICommand Reference.

`put-logging-options`

Il seguente esempio di codice mostra come utilizzare`put-logging-options`.

AWS CLI

Come impostare le opzioni di registrazione di log

L'`list-tags-for-resource`esempio seguente imposta o aggiorna le opzioni di registrazione degli eventi AWS IoT. Se aggiorni il valore di qualsiasi campo `loggingOptions`, dovrai attendere fino a un minuto per rendere effettiva la modifica. Inoltre, se modifichi la policy collegata al ruolo specificato nel campo `roleArn` (ad esempio per correggere una policy non valida), dovrai attendere fino a cinque minuti per rendere effettiva la modifica.

```
aws iotevents put-logging-options \
    --cli-input-json file://logging-options.json
```

Contenuto di `logging-options.json`:

```
{
    "loggingOptions": {
        "roleArn": "arn:aws:iam::123456789012:role/IoTEventsRole",
        "level": "DEBUG",
        "enabled": true,
        "detectorDebugOptions": [
            {
                "detectorModelName": "motorDetectorModel",
                "keyValue": "Fulton-A32"
            }
        ]
    }
}
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta PutLoggingOptionsla AWS IoT Events Developer Guide*.

- Per i dettagli sull'API, consulta PutLoggingOptions AWS CLICommand Reference.

`tag-resource`

Il seguente esempio di codice mostra come utilizzare`tag-resource`.

AWS CLI

Come aggiungere tag a una risorsa

L'esempio `tag-resource` seguente aggiunge o modifica i tag di una determinata risorsa. I tag sono metadati utilizzabili per gestire una risorsa.

```
aws iotevents tag-resource \
    --cli-input-json file://pressureInput.tag.json
```

Contenuto di `pressureInput.tag.json`:

```
{
    "resourceArn": "arn:aws:iotevents:us-west-2:123456789012:input/PressureInput",
    "tags": [
        {
            "key": "deviceType",
            "value": "motor"
        }
    ]
}
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta TagResourcela AWS IoT Events Developer Guide*.

- Per i dettagli sull'API, consulta TagResource AWS CLICommand Reference.

`untag-resource`

Il seguente esempio di codice mostra come utilizzare`untag-resource`.

AWS CLI

Come rimuovere tag da una risorsa

L'esempio `untag-resource` seguente rimuove i tag specificati dalla risorsa.

```
aws iotevents untag-resource \
    --cli-input-json file://pressureInput.untag.json
```

Contenuto di `pressureInput.untag.json`:

```
{
    "resourceArn": "arn:aws:iotevents:us-west-2:123456789012:input/PressureInput",
    "tagKeys": [
            "deviceType"
    ]
}
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta UntagResourcela AWS IoT Events Developer Guide*.

- Per i dettagli sull'API, consulta UntagResource AWS CLICommand Reference.

`update-detector-model`

Il seguente esempio di codice mostra come utilizzareu`pdate-detector-model`.

AWS CLI

Come aggiornare un modello di rivelatore

L'esempio `update-detector-model` seguente aggiorna un modello di rilevatore. I rilevatori (istanze) generati dalla versione precedente vengono eliminati e quindi ricreati all'arrivo di nuovi input.

```
aws iotevents update-detector-model \
    --cli-input-json file://motorDetectorModel.update.json
```

Contenuto di motorDetectorModel .update.json:

```
{
  "detectorModelName": "motorDetectorModel",
  "detectorModelDefinition": {
    "states": [
      {
```

```
        "stateName": "Normal",
        "onEnter": {
          "events": [
            {
              "eventName": "init",
              "condition": "true",
              "actions": [
                {
                  "setVariable": {
                    "variableName": "pressureThresholdBreached",
                    "value": "0"
                  }
                }
              ]
            }
          ]
        },
        "onInput": {
          "transitionEvents": [
            {
              "eventName": "Overpressurized",
              "condition": "$input.PressureInput.sensorData.pressure > 70",
              "actions": [
                {
                  "setVariable": {
                    "variableName": "pressureThresholdBreached",
                    "value": "$variable.pressureThresholdBreached + 3"
                  }
                }
              ],
              "nextState": "Dangerous"
            }
          ]
        }
      },
      {
        "stateName": "Dangerous",
        "onEnter": {
          "events": [
            {
              "eventName": "Pressure Threshold Breached",
              "condition": "$variable.pressureThresholdBreached > 1",
              "actions": [
                {
```

```
                  "sns": {
                    "targetArn": "arn:aws:sns:us-
east-1:123456789012:underPressureAction"
                  }
                }
              ]
            }
          ]
        },
        "onInput": {
          "events": [
            {
              "eventName": "Overpressurized",
              "condition": "$input.PressureInput.sensorData.pressure > 70",
              "actions": [
                {
                  "setVariable": {
                    "variableName": "pressureThresholdBreached",
                    "value": "3"
                  }
                }
              ]
            },
            {
              "eventName": "Pressure Okay",
              "condition": "$input.PressureInput.sensorData.pressure <= 70",
              "actions": [
                {
                  "setVariable": {
                    "variableName": "pressureThresholdBreached",
                    "value": "$variable.pressureThresholdBreached - 1"
                  }
                }
              ]
            }
          ],
          "transitionEvents": [
            {
              "eventName": "BackToNormal",
              "condition": "$input.PressureInput.sensorData.pressure <= 70 &&
 $variable.pressureThresholdBreached <= 1",
              "nextState": "Normal"
            }
          ]
```

```
        },
        "onExit": {
          "events": [
            {
              "eventName": "Normal Pressure Restored",
              "condition": "true",
              "actions": [
                {
                  "sns": {
                    "targetArn": "arn:aws:sns:us-
east-1:123456789012:pressureClearedAction"
                  }
                }
              ]
            }
          ]
        }
      }
    ],
    "initialStateName": "Normal"
  },
  "roleArn": "arn:aws:iam::123456789012:role/IoTEventsRole"
}
```

Output:

```
{
    "detectorModelConfiguration": {
        "status": "ACTIVATING",
        "lastUpdateTime": 1560799387.719,
        "roleArn": "arn:aws:iam::123456789012:role/IoTEventsRole",
        "creationTime": 1560799387.719,
        "detectorModelArn": "arn:aws:iotevents:us-west-2:123456789012:detectorModel/
motorDetectorModel",
        "key": "motorid",
        "detectorModelName": "motorDetectorModel",
        "detectorModelVersion": "2"
    }
}
```

Per ulteriori informazioni, consulta UpdateDetectorModella AWS IoT Events Developer Guide*.

- Per i dettagli sull'API, consulta UpdateDetectorModel AWS CLICommand Reference.

`update-input`

Il seguente esempio di codice mostra come utilizzare`update-input`.

AWS CLI

Come aggiornare un input

L'esempio `update-input` seguente aggiorna un input.

```
aws iotevents update-input \
    --cli-input-json file://pressureInput.json
```

Contenuto di `pressureInput.json`:

```
{
    "inputName": "PressureInput",
    "inputDescription": "Pressure readings from a motor",
    "inputDefinition": {
        "attributes": [
            { "jsonPath": "sensorData.pressure" },
            { "jsonPath": "motorid" }
        ]
    }
}
```

Output:

```
{
    "inputConfiguration": {
        "status": "ACTIVE",
        "inputArn": "arn:aws:iotevents:us-west-2:123456789012:input/PressureInput",
        "lastUpdateTime": 1560795976.458,
        "creationTime": 1560795312.542,
        "inputName": "PressureInput",
        "inputDescription": "Pressure readings from a motor"
    }
}
```

Per ulteriori informazioni, consulta UpdateInputla AWS IoT Events Developer Guide*.

- Per i dettagli sull'API, consulta UpdateInput AWS CLICommand Reference.

AWS IoT Greengrass esempi utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with AWS IoT Greengrass.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`associate-role-to-group`

Il seguente esempio di codice mostra come utilizzare`associate-role-to-group`.

AWS CLI

Come associare un ruolo a un gruppo Greengrass

L'esempio `associate-role-to-group` seguente associa il ruolo IAM specificato a un gruppo Greengrass. Il ruolo di gruppo viene utilizzato dalle funzioni e dai connettori Lambda locali per accedere ai servizi AWS . Ad esempio, il ruolo del gruppo potrebbe concedere le autorizzazioni necessarie per l'integrazione dei CloudWatch log.

```
aws greengrass associate-role-to-group \
    --group-id 2494ee3f-7f8a-4e92-a78b-d205f808b84b \
    --role-arn arn:aws:iam::123456789012:role/GG-Group-Role
```

Output:

```
{
    "AssociatedAt": "2019-09-10T20:03:30Z"
}
```

Per ulteriori informazioni, consulta Configurare il ruolo del gruppo nella Guida per gli sviluppatori di AWS IoT Greengrass.

- Per i dettagli sull'API, consulta AWS CLI Command AssociateRoleToGroupReference.

`associate-service-role-to-account`

Il seguente esempio di codice mostra come utilizzare`associate-service-role-to-account`.

AWS CLI

Per associare un ruolo di servizio al tuo AWS account

L'`associate-service-role-to-account`esempio seguente associa un ruolo di servizio IAM, specificato dal relativo ARN, a AWS IoT Greengrass nel tuo account. AWS È necessario aver precedentemente creato il ruolo di servizio in IAM e associare ad esso un documento di policy che consenta a AWS IoT Greengrass di assumere questo ruolo.

```
aws greengrass associate-service-role-to-account \
    --role-arn "arn:aws:iam::123456789012:role/service-role/Greengrass_ServiceRole"
```

Output:

```
{
    "AssociatedAt": "2019-06-25T18:12:45Z"
}
```

Per ulteriori informazioni, consulta Ruolo del servizio Greengrass nella Guida per gli sviluppatori di AWS IoT Greengrass.

- Per i dettagli sull'API, consulta AssociateServiceRoleToAccount AWS CLICommand Reference.

`create-connector-definition-version`

Il seguente esempio di codice mostra come utilizzare`create-connector-definition-version`.

AWS CLI

Come creare una versione della definizione del connettore

L'esempio `create-connector-definition-version` seguente crea una versione della definizione del connettore e la associa alla definizione del connettore specificata. Tutti i connettori in una versione definiscono i valori per i relativi parametri.

```
aws greengrass create-connector-definition-version \
    --connector-definition-id "55d0052b-0d7d-44d6-b56f-21867215e118" \
    --connectors "[{\"Id\": \"MyTwilioNotificationsConnector\",
 \"ConnectorArn\": \"arn:aws:greengrass:us-west-2::/connectors/
TwilioNotifications/versions/2\", \"Parameters\": {\"TWILIO_ACCOUNT_SID
\": \"AC1a8d4204890840d7fc482aab38090d57\", \"TwilioAuthTokenSecretArn\":
 \"arn:aws:secretsmanager:us-west-2:123456789012:secret:greengrass-TwilioAuthToken-
ntSlp6\", \"TwilioAuthTokenSecretArn-ResourceId\": \"TwilioAuthToken\",
 \"DefaultFromPhoneNumber\": \"4254492999\"}}]"
```

Output:

```
{
    "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/
connectors/55d0052b-0d7d-44d6-b56f-21867215e118/versions/33f709a0-c825-49cb-9eea-
dc8964fbd635",
    "CreationTimestamp": "2019-06-24T20:46:30.134Z",
    "Id": "55d0052b-0d7d-44d6-b56f-21867215e118",
    "Version": "33f709a0-c825-49cb-9eea-dc8964fbd635"
}
```

- Per i dettagli sull'API, consulta CreateConnectorDefinitionVersion AWS CLICommand Reference.

`create-connector-definition`

Il seguente esempio di codice mostra come utilizzare`create-connector-definition`.

AWS CLI

Come creare una definizione di connettore

L'esempio `create-connector-definition` seguente crea una definizione di connettore e una versione iniziale della definizione di connettore. La versione iniziale contiene un connettore. Tutti i connettori in una versione definiscono i valori per i relativi parametri.

```
aws greengrass create-connector-definition \
```

```
    --name MySNSConnector \
    --initial-version "{\"Connectors\": [{\"Id\":\"MySNSConnector\",\"ConnectorArn
\":\"arn:aws:greengrass:us-west-2::/connectors/SNS/versions/1\",\"Parameters\":
 {\"DefaultSNSArn\":\"arn:aws:sns:us-west-2:123456789012:GGConnectorTopic\"}}]}"
```

Output:

```
{
    "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/
connectors/b5c4ebfd-f672-49a3-83cd-31c7216a7bb8",
    "CreationTimestamp": "2019-06-19T19:30:01.300Z",
    "Id": "b5c4ebfd-f672-49a3-83cd-31c7216a7bb8",
    "LastUpdatedTimestamp": "2019-06-19T19:30:01.300Z",
    "LatestVersion": "63c57963-c7c2-4a26-a7e2-7bf478ea2623",
    "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
definition/connectors/b5c4ebfd-f672-49a3-83cd-31c7216a7bb8/versions/63c57963-
c7c2-4a26-a7e2-7bf478ea2623",
    "Name": "MySNSConnector"
}
```

Per ulteriori informazioni, consulta Nozioni di base sui connettori Greengrass (CLI) nella Guida per gli sviluppatori di AWS IoT Greengrass.

- Per i dettagli sull'API, consulta CreateConnectorDefinition AWS CLICommand Reference.

`create-core-definition-version`

Il seguente esempio di codice mostra come utilizzare`create-core-definition-version`.

AWS CLI

Come creare una versione della definizione di core

L'esempio `create-core-definition-version` seguente crea una versione della definizione di core e la associa alla definizione di core specificata. La versione può contenere un solo core. Prima di poter creare un core, è necessario creare e fornire l'oggetto AWS IoT corrispondente. Questo processo include i seguenti comandi `iot`, che restituiscono `ThingArn` e `CertificateArn` necessari per il comando `create-core-definition-version`.

Crea l'oggetto AWS IoT che corrisponde al dispositivo principale:

```
aws iot create-thing \
```

```
    --thing-name "MyCoreDevice"
```

Output:

```
{
    "thingArn": "arn:aws:iot:us-west-2:123456789012:thing/MyCoreDevice",
    "thingName": "MyCoreDevice",
    "thingId": "cb419a19-9099-4515-9cec-e9b0e760608a"
}
```

Crea chiavi pubbliche e private e il certificato del dispositivo core per l'oggetto. Questo esempio utilizza il comando `create-keys-and-certificate` e richiede le autorizzazioni di scrittura nella directory corrente. In alternativa, puoi utilizzare il comando `create-certificate-from-csr`.

```
aws iot create-keys-and-certificate \
    --set-as-active \
    --certificate-pem-outfile "myCore.cert.pem" \
    --public-key-outfile "myCore.public.key" \
    --private-key-outfile "myCore.private.key"
```

Output:

```
{
    "certificateArn": "arn:aws:iot:us-
west-2:123456789012:cert/123a15ec415668c2349a76170b64ac0878231c1e21ec83c10e92a1EXAMPLExyz",
    "certificatePem": "-----BEGIN CERTIFICATE-----
\nMIIDWTCAkGgAwIBATgIUCgq6EGqou6zFqWgIZRndgQEFW+gwDQYJKoZIhvc...KdGewQS\n-----END
 CERTIFICATE-----\n",
    "keyPair": {
        "PublicKey": "-----BEGIN PUBLIC KEY-----
\nMIIBIjANBzrqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAqKpRgnn6yq26U3y...wIDAQAB\n-----END
 PUBLIC KEY-----\n",
        "PrivateKey": "-----BEGIN RSA PRIVATE KEY-----
\nMIIEowIABAKCAQEAqKpRgnn6yq26U3yt5YFZquyukfRjbMXDcNOK4rMCxDR...fvY4+te\n-----END
 RSA PRIVATE KEY-----\n"
    },
    "certificateId":
 "123a15ec415668c2349a76170b64ac0878231c1e21ec83c10e92a1EXAMPLExyz"
}
```

Crea una policy AWS IoT che consenta `iot` e `greengrass` agisca. Per semplificare, la seguente policy consente azioni su tutte le risorse, ma la policy dovrebbe essere più restrittiva.

```
aws iot create-policy \
    --policy-name "Core_Devices" \
    --policy-document "{\"Version\":\"2012-10-17\",\"Statement\":[{\"Effect
\":\"Allow\",\"Action\":[\"iot:Publish\",\"iot:Subscribe\",\"iot:Connect
\",\"iot:Receive\"],\"Resource\":[\"*\"]},{\"Effect\":\"Allow\",\"Action\":
[\"iot:GetThingShadow\",\"iot:UpdateThingShadow\",\"iot:DeleteThingShadow\"],
\"Resource\":[\"*\"]},{\"Effect\":\"Allow\",\"Action\":[\"greengrass:*\"],\"Resource
\":[\"*\"]}]}"
```

Output:

```
{
    "policyName": "Core_Devices",
    "policyArn": "arn:aws:iot:us-west-2:123456789012:policy/Core_Devices",
    "policyDocument": "{\"Version\":\"2012-10-17\",\"Statement\":[{\"Effect
\":\"Allow\",\"Action\":[\"iot:Publish\",\"iot:Subscribe\",\"iot:Connect
\",\"iot:Receive\"],\"Resource\":[\"*\"]},{\"Effect\":\"Allow\",\"Action\":
[\"iot:GetThingShadow\",\"iot:UpdateThingShadow\",\"iot:DeleteThingShadow\"],
\"Resource\":[\"*\"]},{\"Effect\":\"Allow\",\"Action\":[\"greengrass:*\"],\"Resource
\":[\"*\"]}]}",
    "policyVersionId": "1"
}
```

Collega la policy al certificato:

```
aws iot attach-policy \
    --policy-name "Core_Devices" \
    --target "arn:aws:iot:us-
west-2:123456789012:cert/123a15ec415668c2349a76170b64ac0878231c1e21ec83c10e92a1EXAMPLExyz"
```

Questo comando non produce alcun output.

Collega l'oggetto al certificato:

```
aws iot attach-thing-principal \
    --thing-name "MyCoreDevice" \
    --principal "arn:aws:iot:us-
west-2:123456789012:cert/123a15ec415668c2349a76170b64ac0878231c1e21ec83c10e92a1EXAMPLExyz"
```

Questo comando non produce alcun output.

Crea una versione della definizione core:

```
aws greengrass create-core-definition-version \
    --core-definition-id "582efe12-b05a-409e-9a24-a2ba1bcc4a12" \
    --cores "[{\"Id\":\"MyCoreDevice\",\"ThingArn\":\"arn:aws:iot:us-
west-2:123456789012:thing/MyCoreDevice\",\"CertificateArn\":\"arn:aws:iot:us-
west-2:123456789012:cert/123a15ec415668c2349a76170b64ac0878231c1e21ec83c10e92a1EXAMPLExyz
\",\"SyncShadow\":true}]"
```

Output:

```
{
    "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/
cores/582efe12-b05a-409e-9a24-a2ba1bcc4a12/versions/3fdc1190-2ce5-44de-b98b-
eec8f9571014",
    "Version": "3fdc1190-2ce5-44de-b98b-eec8f9571014",
    "CreationTimestamp": "2019-09-18T00:15:09.838Z",
    "Id": "582efe12-b05a-409e-9a24-a2ba1bcc4a12"
}
```

Per ulteriori informazioni, consulta Configurare AWS IoT Greengrass Core nella IoT AWS Greengrass Developer Guide.

- Per i dettagli sull'API, consulta AWS CLI Command CreateCoreDefinitionVersionReference.

`create-core-definition`

Il seguente esempio di codice mostra come utilizzare`create-core-definition`.

AWS CLI

Esempio 1: come creare una definizione di core vuota

L'esempio `create-core-definition` seguente crea una definizione di core Greengrass vuota (nessuna versione iniziale). Prima che il core sia utilizzabile, è necessario utilizzare il comando `create-core-definition-version` per fornire gli altri parametri per il core.

```
aws greengrass create-core-definition \
    --name cliGroup_Core
```

Output:

```
{
    "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/cores/
b5c08008-54cb-44bd-9eec-c121b04283b5",
    "CreationTimestamp": "2019-06-25T18:23:22.106Z",
    "Id": "b5c08008-54cb-44bd-9eec-c121b04283b5",
    "LastUpdatedTimestamp": "2019-06-25T18:23:22.106Z",
    "Name": "cliGroup_Core"
}
```

Esempio 2: come creare una definizione di core con una versione iniziale

L'esempio `create-core-definition` seguente crea una definizione di core che contiene una versione iniziale della definizione di core. La versione può contenere un solo core. Prima di poter creare un core, è necessario creare e fornire l'oggetto AWS IoT corrispondente. Questo processo include i seguenti comandi `iot`, che restituiscono `ThingArn` e `CertificateArn` necessari per il comando `create-core-definition`.

Crea l'oggetto AWS IoT che corrisponde al dispositivo principale:

```
aws iot create-thing \
    --thing-name "MyCoreDevice"
```

Output:

```
{
    "thingArn": "arn:aws:iot:us-west-2:123456789012:thing/MyCoreDevice",
    "thingName": "MyCoreDevice",
    "thingId": "cb419a19-9099-4515-9cec-e9b0e760608a"
}
```

Crea chiavi pubbliche e private e il certificato del dispositivo core per l'oggetto. Questo esempio utilizza il comando `create-keys-and-certificate` e richiede le autorizzazioni di scrittura nella directory corrente. In alternativa, puoi utilizzare il comando `create-certificate-from-csr`.

```
aws iot create-keys-and-certificate \
    --set-as-active \
    --certificate-pem-outfile "myCore.cert.pem" \
```

```
    --public-key-outfile "myCore.public.key" \
    --private-key-outfile "myCore.private.key"
```

Output:

```
{
    "certificateArn": "arn:aws:iot:us-
west-2:123456789012:cert/123a15ec415668c2349a76170b64ac0878231c1e21ec83c10e92a1EXAMPLExyz",
    "certificatePem": "-----BEGIN CERTIFICATE-----
\nMIIDWTCAkGgAwIBATgIUCgq6EGqou6zFqWgIZRndgQEFW+gwDQYJKoZIhvc...KdGewQS\n-----END
 CERTIFICATE-----\n",
    "keyPair": {
        "PublicKey": "-----BEGIN PUBLIC KEY-----
\nMIIBIjANBzrqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAqKpRgnn6yq26U3y...wIDAQAB\n-----END
 PUBLIC KEY-----\n",
        "PrivateKey": "-----BEGIN RSA PRIVATE KEY-----
\nMIIEowIABAKCAQEAqKpRgnn6yq26U3yt5YFZquyukfRjbMXDcNOK4rMCxDR...fvY4+te\n-----END
 RSA PRIVATE KEY-----\n"
    },
    "certificateId":
 "123a15ec415668c2349a76170b64ac0878231c1e21ec83c10e92a1EXAMPLExyz"
}
```

Crea una policy AWS IoT che consenta `iot` e `greengrass` agisca. Per semplificare, la seguente policy consente azioni su tutte le risorse, ma la policy dovrebbe essere più restrittiva.

```
aws iot create-policy \
    --policy-name "Core_Devices" \
    --policy-document "{\"Version\":\"2012-10-17\",\"Statement\":[{\"Effect
\":\"Allow\",\"Action\":[\"iot:Publish\",\"iot:Subscribe\",\"iot:Connect
\",\"iot:Receive\"],\"Resource\":[\"*\"]},{\"Effect\":\"Allow\",\"Action\":
[\"iot:GetThingShadow\",\"iot:UpdateThingShadow\",\"iot:DeleteThingShadow\"],
\"Resource\":[\"*\"]},{\"Effect\":\"Allow\",\"Action\":[\"greengrass:*\"],\"Resource
\":[\"*\"]}]}"
```

Output:

```
{
    "policyName": "Core_Devices",
    "policyArn": "arn:aws:iot:us-west-2:123456789012:policy/Core_Devices",
    "policyDocument": "{\"Version\":\"2012-10-17\",\"Statement\":[{\"Effect
\":\"Allow\",\"Action\":[\"iot:Publish\",\"iot:Subscribe\",\"iot:Connect
```

```
\",\"iot:Receive\"],\"Resource\":[\"*\"]},{\"Effect\":\"Allow\",\"Action\":
[\"iot:GetThingShadow\",\"iot:UpdateThingShadow\",\"iot:DeleteThingShadow\"],
\"Resource\":[\"*\"]},{\"Effect\":\"Allow\",\"Action\":[\"greengrass:*\"],\"Resource
\":[\"*\"]}]}",
    "policyVersionId": "1"
}
```

Collega la policy al certificato:

```
aws iot attach-policy \
    --policy-name "Core_Devices" \
    --target "arn:aws:iot:us-
west-2:123456789012:cert/123a15ec415668c2349a76170b64ac0878231c1e21ec83c10e92a1EXAMPLExyz"
```

Questo comando non produce alcun output.

Collega l'oggetto al certificato:

```
aws iot attach-thing-principal \
    --thing-name "MyCoreDevice" \
    --principal "arn:aws:iot:us-
west-2:123456789012:cert/123a15ec415668c2349a76170b64ac0878231c1e21ec83c10e92a1EXAMPLExyz"
```

Questo comando non produce alcun output.

Crea una definizione core:

```
aws greengrass create-core-definition \
    --name "MyCores" \
    --initial-version "{\"Cores\":[{\"Id\":\"MyCoreDevice\",\"ThingArn\":
\"arn:aws:iot:us-west-2:123456789012:thing/MyCoreDevice\",\"CertificateArn\":
\"arn:aws:iot:us-
west-2:123456789012:cert/123a15ec415668c2349a76170b64ac0878231c1e21ec83c10e92a1EXAMPLExyz
\",\"SyncShadow\":true}]}"
```

Output:

```
{
    "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/definition/cores/582efe12-b05a-409e-9a24-a2ba1bcc4a12/versions/
cc87b5b3-8f4b-465d-944c-1d6de5dbfcdb",
```

```
    "Name": "MyCores",
    "LastUpdatedTimestamp": "2019-09-18T00:11:06.197Z",
    "LatestVersion": "cc87b5b3-8f4b-465d-944c-1d6de5dbfcdb",
    "CreationTimestamp": "2019-09-18T00:11:06.197Z",
    "Id": "582efe12-b05a-409e-9a24-a2ba1bcc4a12",
    "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/
cores/582efe12-b05a-409e-9a24-a2ba1bcc4a12"
}
```

Per ulteriori informazioni, consulta Configurare AWS IoT Greengrass Core nella IoT AWS Greengrass Developer Guide.

- Per i dettagli sull'API, consulta AWS CLI Command CreateCoreDefinitionReference.

`create-deployment`

Il seguente esempio di codice mostra come utilizzare`create-deployment`.

AWS CLI

Come creare un'implementazione per una versione di un gruppo Greengrass

L'esempio `create-deployment` seguente distribuisce la versione specificata di un gruppo Greengrass.

```
aws greengrass create-deployment \
    --deployment-type NewDeployment \
    --group-id "ce2e7d01-3240-4c24-b8e6-f6f6e7a9eeca" \
    --group-version-id "dc40c1e9-e8c8-4d28-a84d-a9cad5f599c9"
```

Output:

```
{
    "DeploymentArn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
groups/ce2e7d01-3240-4c24-b8e6-f6f6e7a9eeca/deployments/bfceb608-4e97-45bc-
af5c-460144270308",
    "DeploymentId": "bfceb608-4e97-45bc-af5c-460144270308"
}
```

Per ulteriori informazioni, consulta Nozioni di base sui connettori (CLI) nella Guida per gli sviluppatori di AWS IoT Greengrass.

- Per i dettagli sull'API, consulta CreateDeployment AWS CLICommand Reference.

`create-device-definition-version`

Il seguente esempio di codice mostra come utilizzare`create-device-definition-version`.

AWS CLI

Come creare una versione di definizione del dispositivo

L'esempio `create-device-definition-version` seguente crea una versione della definizione del dispositivo e la associa alla definizione del dispositivo specificata. La versione definisce due dispositivi. Prima di poter creare un dispositivo Greengrass, è necessario creare e fornire il dispositivo AWS IoT corrispondente. Questo processo include i seguenti comandi `iot` che è necessario eseguire per ottenere le informazioni richieste per il comando Greengrass:

Crea l'oggetto AWS IoT che corrisponde al dispositivo:

```
aws iot create-thing \
    --thing-name "InteriorTherm"
```

Output:

```
{
    "thingArn": "arn:aws:iot:us-west-2:123456789012:thing/InteriorTherm",
    "thingName": "InteriorTherm",
    "thingId": "01d4763c-78a6-46c6-92be-7add080394bf"
}
```

Crea chiavi pubbliche e private e il certificato del dispositivo per l'oggetto. Questo esempio utilizza il comando `create-keys-and-certificate` e richiede le autorizzazioni di scrittura nella directory corrente. In alternativa, puoi utilizzare il comando `create-certificate-from-csr`.

```
aws iot create-keys-and-certificate \
    --set-as-active \
    --certificate-pem-outfile "myDevice.cert.pem" \
    --public-key-outfile "myDevice.public.key" \
    --private-key-outfile "myDevice.private.key"
```

Output:

```
{
    "certificateArn": "arn:aws:iot:us-
west-2:123456789012:cert/66a415ec415668c2349a76170b64ac0878231c1e21ec83c10e92a18bd568eb92",
    "certificatePem": "-----BEGIN CERTIFICATE-----
\nMIIDWTCAkGgAwIBATgIUCgq6EGqou6zFqWgIZRndgQEFW+gwDQYJKoZIhvc...KdGewQS\n-----END
 CERTIFICATE-----\n",
    "keyPair": {
        "PublicKey": "-----BEGIN PUBLIC KEY-----
\nMIIBIjANBzrqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAqKpRgnn6yq26U3y...wIDAQAB\n-----END
 PUBLIC KEY-----\n",
        "PrivateKey": "-----BEGIN RSA PRIVATE KEY-----
\nMIIEowIABAKCAQEAqKpRgnn6yq26U3yt5YFZquyukfRjbMXDcNOK4rMCxDR...fvY4+te\n-----END
 RSA PRIVATE KEY-----\n"
    },
    "certificateId":
 "66a415ec415668c2349a76170b64ac0878231c1e21ec83c10e92a18bd568eb92"
}
```

Crea una policy AWS IoT che consenta `iot` e `greengrass` agisca. Per semplicità, la seguente policy consente azioni su tutte le risorse, ma la tua policy può essere più restrittiva:

```
aws iot create-policy \
    --policy-name "GG_Devices" \
    --policy-document "{\"Version\":\"2012-10-17\",\"Statement\":[{\"Effect
\":\"Allow\",\"Action\":[\"iot:Publish\",\"iot:Subscribe\",\"iot:Connect
\",\"iot:Receive\"],\"Resource\":[\"*\"]},{\"Effect\":\"Allow\",\"Action\":
[\"iot:GetThingShadow\",\"iot:UpdateThingShadow\",\"iot:DeleteThingShadow\"],
\"Resource\":[\"*\"]},{\"Effect\":\"Allow\",\"Action\":[\"greengrass:*\"],\"Resource
\":[\"*\"]}]}"
```

Output:

```
{
    "policyName": "GG_Devices",
    "policyArn": "arn:aws:iot:us-west-2:123456789012:policy/GG_Devices",
    "policyDocument": "{\"Version\":\"2012-10-17\",\"Statement\":[{\"Effect
\":\"Allow\",\"Action\":[\"iot:Publish\",\"iot:Subscribe\",\"iot:Connect
\",\"iot:Receive\"],\"Resource\":[\"*\"]},{\"Effect\":\"Allow\",\"Action\":
[\"iot:GetThingShadow\",\"iot:UpdateThingShadow\",\"iot:DeleteThingShadow\"],
\"Resource\":[\"*\"]},{\"Effect\":\"Allow\",\"Action\":[\"greengrass:*\"],\"Resource
\":[\"*\"]}]}",
    "policyVersionId": "1"
```

```
}
```

Collega la policy al certificato:

```
aws iot attach-policy \
    --policy-name "GG_Devices" \
    --target "arn:aws:iot:us-
west-2:123456789012:cert/66a415ec415668c2349a76170b64ac0878231c1e21ec83c10e92a18bd568eb92"
```

Collega l'oggetto al certificato

```
aws iot attach-thing-principal \
    --thing-name "InteriorTherm" \
    --principal "arn:aws:iot:us-
west-2:123456789012:cert/66a415ec415668c2349a76170b64ac0878231c1e21ec83c10e92a18bd568eb92"
```

Dopo aver creato e configurato l'oggetto IoT come mostrato sopra, utilizza i comandi `ThingArn` e `CertificateArn` dei primi due comandi nell'esempio seguente.

```
aws greengrass create-device-definition-version \
    --device-definition-id "f9ba083d-5ad4-4534-9f86-026a45df1ccd" \
    --devices "[{\"Id\":\"InteriorTherm\",\"ThingArn\":\"arn:aws:iot:us-
west-2:123456789012:thing/InteriorTherm\",\"CertificateArn\":\"arn:aws:iot:us-
west-2:123456789012:cert/66a415ec415668c2349a76170b64ac0878231c1e21ec83c10e92a18bd568eb92\",
\"SyncShadow\":true},{\"Id\":\"ExteriorTherm\",\"ThingArn\":\"arn:aws:iot:us-
west-2:123456789012:thing/ExteriorTherm\",\"CertificateArn\":\"arn:aws:iot:us-
west-2:123456789012:cert/6c52ce1b47bde88a637e9ccdd45fe4e4c2c0a75a6866f8f63d980ee22fa51e02\",
\"SyncShadow\":true}]"
```

Output:

```
{
    "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
definition/devices/f9ba083d-5ad4-4534-9f86-026a45df1ccd/
versions/83c13984-6fed-447e-84d5-5b8aa45d5f71",
    "Version": "83c13984-6fed-447e-84d5-5b8aa45d5f71",
    "CreationTimestamp": "2019-09-11T00:15:09.838Z",
    "Id": "f9ba083d-5ad4-4534-9f86-026a45df1ccd"
}
```

- Per i dettagli sull'API, consulta CreateDeviceDefinitionVersion AWS CLICommand Reference.

`create-device-definition`

Il seguente esempio di codice mostra come utilizzare`create-device-definition`.

AWS CLI

Come creare una definizione di dispositivo

L'esempio `create-device-definition` seguente crea una definizione di dispositivo che contiene una versione iniziale della definizione di dispositivo. La versione iniziale definisce due dispositivi. Prima di poter creare un dispositivo Greengrass, è necessario creare e fornire il dispositivo AWS IoT corrispondente. Questo processo include i seguenti comandi `iot` che è necessario eseguire per ottenere le informazioni richieste per il comando Greengrass:

Crea l'oggetto AWS IoT che corrisponde al dispositivo:

```
aws iot create-thing \
    --thing-name "InteriorTherm"
```

Output:

```
{
    "thingArn": "arn:aws:iot:us-west-2:123456789012:thing/InteriorTherm",
    "thingName": "InteriorTherm",
    "thingId": "01d4763c-78a6-46c6-92be-7add080394bf"
}
```

Crea chiavi pubbliche e private e il certificato del dispositivo per l'oggetto. Questo esempio utilizza il comando `create-keys-and-certificate` e richiede le autorizzazioni di scrittura nella directory corrente. In alternativa, puoi utilizzare il comando `create-certificate-from-csr`.

```
aws iot create-keys-and-certificate \
    --set-as-active \
    --certificate-pem-outfile "myDevice.cert.pem" \
    --public-key-outfile "myDevice.public.key" \
    --private-key-outfile "myDevice.private.key"
```

Output:

```
{
    "certificateArn": "arn:aws:iot:us-
west-2:123456789012:cert/66a415ec415668c2349a76170b64ac0878231c1e21ec83c10e92a18bd568eb92",
```

```
    "certificatePem": "-----BEGIN CERTIFICATE-----
\nMIIDWTCAkGgAwIBATgIUCgq6EGqou6zFqWgIZRndgQEFW+gwDQYJKoZIhvc...KdGewQS\n-----END
 CERTIFICATE-----\n",
    "keyPair": {
        "PublicKey": "-----BEGIN PUBLIC KEY-----
\nMIIBIjANBzrqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAqKpRgnn6yq26U3y...wIDAQAB\n-----END
 PUBLIC KEY-----\n",
        "PrivateKey": "-----BEGIN RSA PRIVATE KEY-----
\nMIIEowIABAKCAQEAqKpRgnn6yq26U3yt5YFZquyukfRjbMXDcNOK4rMCxDR...fvY4+te\n-----END
 RSA PRIVATE KEY-----\n"
    },
    "certificateId":
 "66a415ec415668c2349a76170b64ac0878231c1e21ec83c10e92a18bd568eb92"
}
```

Crea una policy AWS IoT che consenta `iot` e `greengrass` agisca. Per semplicità, la seguente policy consente azioni su tutte le risorse, ma la tua policy può essere più restrittiva:

```
aws iot create-policy \
    --policy-name "GG_Devices" \
    --policy-document "{\"Version\":\"2012-10-17\",\"Statement\":[{\"Effect
\":\"Allow\",\"Action\":[\"iot:Publish\",\"iot:Subscribe\",\"iot:Connect
\",\"iot:Receive\"],\"Resource\":[\"*\"]},{\"Effect\":\"Allow\",\"Action\":
[\"iot:GetThingShadow\",\"iot:UpdateThingShadow\",\"iot:DeleteThingShadow\"],
\"Resource\":[\"*\"]},{\"Effect\":\"Allow\",\"Action\":[\"greengrass:*\"],\"Resource
\":[\"*\"]}]}"
```

Output:

```
{
    "policyName": "GG_Devices",
    "policyArn": "arn:aws:iot:us-west-2:123456789012:policy/GG_Devices",
    "policyDocument": "{\"Version\":\"2012-10-17\",\"Statement\":[{\"Effect
\":\"Allow\",\"Action\":[\"iot:Publish\",\"iot:Subscribe\",\"iot:Connect
\",\"iot:Receive\"],\"Resource\":[\"*\"]},{\"Effect\":\"Allow\",\"Action\":
[\"iot:GetThingShadow\",\"iot:UpdateThingShadow\",\"iot:DeleteThingShadow\"],
\"Resource\":[\"*\"]},{\"Effect\":\"Allow\",\"Action\":[\"greengrass:*\"],\"Resource
\":[\"*\"]}]}",
    "policyVersionId": "1"
}
```

Collega la policy al certificato:

```
aws iot attach-policy \
    --policy-name "GG_Devices" \
    --target "arn:aws:iot:us-
west-2:123456789012:cert/66a415ec415668c2349a76170b64ac0878231c1e21ec83c10e92a18bd568eb92"
```

Collega l'oggetto al certificato

```
aws iot attach-thing-principal \
    --thing-name "InteriorTherm" \
    --principal "arn:aws:iot:us-
west-2:123456789012:cert/66a415ec415668c2349a76170b64ac0878231c1e21ec83c10e92a18bd568eb92"
```

Dopo aver creato e configurato l'oggetto IoT come mostrato sopra, utilizza i comandi `ThingArn` e `CertificateArn` dei primi due comandi nell'esempio seguente.

```
aws greengrass create-device-definition \
    --name "Sensors" \
    --initial-version "{\"Devices\":[{\"Id\":\"InteriorTherm
\",\"ThingArn\":\"arn:aws:iot:us-west-2:123456789012:thing/
InteriorTherm\",\"CertificateArn\":\"arn:aws:iot:us-
west-2:123456789012:cert/66a415ec415668c2349a76170b64ac0878231c1e21ec83c10e92a18bd568eb92\",
\"SyncShadow\":true},{\"Id\":\"ExteriorTherm\",\"ThingArn\":\"arn:aws:iot:us-
west-2:123456789012:thing/ExteriorTherm\",\"CertificateArn\":\"arn:aws:iot:us-
west-2:123456789012:cert/6c52ce1b47bde88a637e9ccdd45fe4e4c2c0a75a6866f8f63d980ee22fa51e02\",
\"SyncShadow\":true}]}"
```

Output:

```
{
    "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/definition/devices/f9ba083d-5ad4-4534-9f86-026a45df1ccd/
versions/3b5cc510-58c1-44b5-9d98-4ad858ffa795",
    "Name": "Sensors",
    "LastUpdatedTimestamp": "2019-09-11T00:11:06.197Z",
    "LatestVersion": "3b5cc510-58c1-44b5-9d98-4ad858ffa795",
    "CreationTimestamp": "2019-09-11T00:11:06.197Z",
    "Id": "f9ba083d-5ad4-4534-9f86-026a45df1ccd",
    "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/
devices/f9ba083d-5ad4-4534-9f86-026a45df1ccd"
}
```

- Per i dettagli sull'API, consulta CreateDeviceDefinition AWS CLICommand Reference.

`create-function-definition-version`

Il seguente esempio di codice mostra come utilizzare`create-function-definition-version`.

AWS CLI

Come creare la versione della definizione di funzione

L'esempio `create-function-definition-version` seguente crea una nuova versione della definizione di funzione specificata. Questa versione specifica una singola funzione il cui ID è `Hello-World-function`, consente l'accesso al file system e specifica una dimensione massima della memoria e un periodo di timeout.

```
aws greengrass create-function-definition-version \
    --cli-input-json "{\"FunctionDefinitionId\": \"e626e8c9-3b8f-4bf3-9cdc-
d26ecdeb9fa3\",\"Functions\": [{\"Id\": \"Hello-World-function\", \"FunctionArn\":
 \""arn:aws:lambda:us-
west-2:123456789012:function:Greengrass_HelloWorld_Counter:gghw-alias"\",
\"FunctionConfiguration\": {\"Environment\": {\"AccessSysfs\": true},\"Executable\":
 \"greengrassHelloWorldCounter.function_handler\",\"MemorySize\": 16000,\"Pinned\":
 false,\"Timeout\": 25}}]}"
```

Output:

```
{
    "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
definition/functions/e626e8c9-3b8f-4bf3-9cdc-d26ecdeb9fa3/
versions/74abd1cc-637e-4abe-8684-9a67890f4043",
    "CreationTimestamp": "2019-06-25T22:03:43.376Z",
    "Id": "e626e8c9-3b8f-4bf3-9cdc-d26ecdeb9fa3",
    "Version": "74abd1cc-637e-4abe-8684-9a67890f4043"
}
```

- Per i dettagli sull'API, consulta CreateFunctionDefinitionVersion AWS CLICommand Reference.

`create-function-definition`

Il seguente esempio di codice mostra come utilizzare`create-function-definition`.

AWS CLI

Come creare una definizione di funzione Lambda

L'esempio `create-function-definition` seguente crea una definizione di funzione Lambda e una versione iniziale fornendo un elenco di funzioni Lambda (in questo caso, un elenco di una sola funzione denominata `TempMonitorFunction`) e le relative configurazioni. Prima di poter creare la definizione della funzione, è necessario l'ARN della funzione Lambda. Per creare la funzione e il relativo alias, utilizza i comandi `create-function` e `publish-version` di Lambda. Il `create-function` comando di Lambda richiede l'ARN del ruolo di esecuzione, anche se AWS IoT Greengrass non utilizza quel ruolo perché le autorizzazioni sono specificate nel ruolo del gruppo Greengrass. Puoi usare il comando `create-role` IAM per creare un ruolo vuoto per ottenere un ARN da utilizzare con `create-function` di Lambda oppure puoi utilizzare un ruolo di esecuzione esistente.

```
aws greengrass create-function-definition \
    --name MyGreengrassFunctions \
    --initial-version "{\"Functions\": [{\"Id\": \"TempMonitorFunction\",
 \"FunctionArn\": \"arn:aws:lambda:us-
west-2:123456789012:function:TempMonitor:GG_TempMonitor\", \"FunctionConfiguration
\": {\"Executable\": \"temp_monitor.function_handler\", \"MemorySize\": 16000,
\"Timeout\": 5}}]}"
```

Output:

```
{
    "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/
functions/3b0d0080-87e7-48c6-b182-503ec743a08b",
    "CreationTimestamp": "2019-06-19T22:24:44.585Z",
    "Id": "3b0d0080-87e7-48c6-b182-503ec743a08b",
    "LastUpdatedTimestamp": "2019-06-19T22:24:44.585Z",
    "LatestVersion": "67f918b9-efb4-40b0-b87c-de8c9faf085b",
    "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
definition/functions/3b0d0080-87e7-48c6-b182-503ec743a08b/versions/67f918b9-
efb4-40b0-b87c-de8c9faf085b",
    "Name": "MyGreengrassFunctions"
}
```

Per ulteriori informazioni, consulta Come configurare l'accesso alle risorse locali utilizzando l'interfaccia a riga di AWS comando nella AWS IoT Greengrass Developer Guide.

- Per i dettagli sull'API, consulta CreateFunctionDefinition AWS CLICommand Reference.

`create-group-certificate-authority`

Il seguente esempio di codice mostra come utilizzare`create-group-certificate-authority`.

AWS CLI

Come creare un'autorità di certificazione (CA) per un gruppo

L'esempio `create-group-certificate-authority` seguente crea o ruota una CA per il gruppo specificato.

```
aws greengrass create-group-certificate-authority \
    --group-id "8eaadd72-ce4b-4f15-892a-0cc4f3a343f1"
```

Output:

```
{
    "GroupCertificateAuthorityArn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/groups/8eaadd72-ce4b-4f15-892a-0cc4f3a343f1/certificateauthorities/
d31630d674c4437f6c5dbc0dca56312a902171ce2d086c38e509c8EXAMPLEcc5"
}
```

Per ulteriori informazioni, consulta Sicurezza di AWS IoT Greengrass nella Guida per gli sviluppatori di AWS IoT Greengrass.

- Per i dettagli sull'API, consulta CreateGroupCertificateAuthority AWS CLICommand Reference.

`create-group-version`

Il seguente esempio di codice mostra come utilizzare`create-group-version`.

AWS CLI

Come creare una versione di un gruppo Greengrass

L'esempio `create-group-version` seguente crea una versione di gruppo e la associa al gruppo specificato. La versione fa riferimento alle versioni core, di risorsa, di connettore, di funzione e di abbonamento che contengono le entità da includere in questa versione di gruppo. È necessario creare queste entità prima della versione del gruppo.

Per creare una definizione di risorsa con una versione iniziale, utilizza il comando `create-resource-definition`. Per creare una definizione di connettore con una versione iniziale,

utilizza il comando `create-connector-definition`. Per creare una definizione di funzione con una versione iniziale, utilizza il comando `create-function-definition`. Per creare una definizione di abbonamento con una versione iniziale, utilizza il comando `create-subscription-definition`. Per recuperare l'ARN dell'ultima versione di definizione core, utilizza il comando `get-group-version` e specifica l'ID dell'ultima versione di gruppo.

```
aws greengrass create-group-version \
    --group-id "ce2e7d01-3240-4c24-b8e6-f6f6e7a9eeca" \
    --core-definition-version-arn "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/definition/cores/6a630442-8708-4838-ad36-eb98849d975e/
versions/6c87151b-1fb4-4cb2-8b31-6ee715d8f8ba" \
    --resource-definition-version-arn "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/definition/resources/c8bb9ebc-c3fd-40a4-9c6a-568d75569d38/versions/
a5f94d0b-f6bc-40f4-bb78-7a1c5fe13ba1" \
    --connector-definition-version-arn "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/definition/connectors/55d0052b-0d7d-44d6-b56f-21867215e118/
versions/78a3331b-895d-489b-8823-17b4f9f418a0" \
    --function-definition-version-arn "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/definition/functions/3b0d0080-87e7-48c6-b182-503ec743a08b/
versions/67f918b9-efb4-40b0-b87c-de8c9faf085b" \
    --subscription-definition-version-arn "arn:aws:greengrass:us-
west-2:123456789012:/greengrass/definition/subscriptions/9d611d57-5d5d-44bd-a3b4-
feccbdd69112/versions/aa645c47-ac90-420d-9091-8c7ffa4f103f"
```

Output:

```
{
    "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/groups/
ce2e7d01-3240-4c24-b8e6-f6f6e7a9eeca/versions/e10b0459-4345-4a09-88a4-1af1f5d34638",
    "CreationTimestamp": "2019-06-20T18:42:47.020Z",
    "Id": "ce2e7d01-3240-4c24-b8e6-f6f6e7a9eeca",
    "Version": "e10b0459-4345-4a09-88a4-1af1f5d34638"
}
```

Per ulteriori informazioni, vedere Panoramica del modello a AWS oggetti del gruppo IoT Greengrass nella IoT AWS Greengrass Developer Guide.

- Per i dettagli sull'API, consulta AWS CLI Command CreateGroupVersionReference.

`create-group`

Il seguente esempio di codice mostra come utilizzare`create-group`.

AWS CLI

Come creare un gruppo Greengrass

L'esempio `create-group` seguente elimina un gruppo denominato `cli-created-group`.

```
aws greengrass create-group \
    --name cli-created-group
```

Output:

```
{
    "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
groups/4e22bd92-898c-436b-ade5-434d883ff749",
    "CreationTimestamp": "2019-06-25T18:07:17.688Z",
    "Id": "4e22bd92-898c-436b-ade5-434d883ff749",
    "LastUpdatedTimestamp": "2019-06-25T18:07:17.688Z",
    "Name": "cli-created-group"
}
```

Per ulteriori informazioni, vedere Panoramica del modello a AWS oggetti del gruppo IoT Greengrass nella IoT AWS Greengrass Developer Guide.

- Per i dettagli sull'API, consulta AWS CLI Command CreateGroupReference.

`create-logger-definition-version`

Il seguente esempio di codice mostra come utilizzare`create-logger-definition-version`.

AWS CLI

Come creare una versione della definizione di logger

L'esempio `create-logger-definition-version` seguente crea una versione della definizione di logger e la associa a una definizione di logger. La versione definisce quattro configurazioni di registrazione: 1) registri dei componenti di sistema sul file system del dispositivo principale, 2) registri delle funzioni Lambda definiti dall'utente sul file system del dispositivo principale, 3) registri dei componenti di sistema in Amazon Logs e 4) registri delle funzioni Lambda definiti dall'utente in Amazon Logs CloudWatch . CloudWatch Nota: per l'integrazione con Logs, il ruolo del gruppo deve concedere le autorizzazioni appropriate. CloudWatch

```
aws greengrass create-logger-definition-version \
    --logger-definition-id "a454b62a-5d56-4ca9-bdc4-8254e1662cb0" \
    --loggers "[{\"Id\":\"1\",\"Component\":\"GreengrassSystem\",\"Level\":\"ERROR
\",\"Space\":10240,\"Type\":\"FileSystem\"},{\"Id\":\"2\",\"Component\":\"Lambda
\",\"Level\":\"INFO\",\"Space\":10240,\"Type\":\"FileSystem\"},{\"Id\":\"3\",
\"Component\":\"GreengrassSystem\",\"Level\":\"WARN\",\"Type\":\"AWSCloudWatch\"},
{\"Id\":\"4\",\"Component\":\"Lambda\",\"Level\":\"INFO\",\"Type\":\"AWSCloudWatch
\"}]"
```

Output:

```
{
 "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/loggers/
a454b62a-5d56-4ca9-bdc4-8254e1662cb0/versions/49aedb1e-01a3-4d39-9871-3a052573f1ea",
 "Version": "49aedb1e-01a3-4d39-9871-3a052573f1ea",
 "CreationTimestamp": "2019-07-24T00:04:48.523Z",
 "Id": "a454b62a-5d56-4ca9-bdc4-8254e1662cb0"
}
```

Per ulteriori informazioni, consulta Monitoring with AWS IoT Greengrass Logs nella IoT AWS Greengrass Developer Guide.

- Per i dettagli sull'API, consulta Command CreateLoggerDefinitionVersionReference AWS CLI .

`create-logger-definition`

Il seguente esempio di codice mostra come utilizzare`create-logger-definition`.

AWS CLI

Come creare una definizione del logger

L'esempio `create-logger-definition` seguente crea una definizione di logger che contiene una versione iniziale della definizione di logger. La versione iniziale definisce tre configurazioni di registrazione: 1) registri dei componenti di sistema sul file system del dispositivo principale, 2) registri delle funzioni Lambda definiti dall'utente sul file system del dispositivo principale e 3) registri delle funzioni Lambda definiti dall'utente in Amazon Logs. CloudWatch Nota: per l'integrazione dei CloudWatch log, il ruolo del gruppo deve concedere le autorizzazioni appropriate.

```
aws greengrass create-logger-definition \
```

```
    --name "LoggingConfigs" \
    --initial-version "{\"Loggers\":[{\"Id\":\"1\",\"Component\":\"GreengrassSystem
\",\"Level\":\"ERROR\",\"Space\":10240,\"Type\":\"FileSystem\"},{\"Id\":
\"2\",\"Component\":\"Lambda\",\"Level\":\"INFO\",\"Space\":10240,\"Type\":
\"FileSystem\"},{\"Id\":\"3\",\"Component\":\"Lambda\",\"Level\":\"INFO\",\"Type\":
\"AWSCloudWatch\"}]}"
```

Output:

```
{
    "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
definition/loggers/a454b62a-5d56-4ca9-bdc4-8254e1662cb0/versions/de1d9854-1588-4525-
b25e-b378f60f2322",
    "Name": "LoggingConfigs",
    "LastUpdatedTimestamp": "2019-07-23T23:52:17.165Z",
    "LatestVersion": "de1d9854-1588-4525-b25e-b378f60f2322",
    "CreationTimestamp": "2019-07-23T23:52:17.165Z",
    "Id": "a454b62a-5d56-4ca9-bdc4-8254e1662cb0",
    "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/
loggers/a454b62a-5d56-4ca9-bdc4-8254e1662cb0"
}
```

Per ulteriori informazioni, consulta Monitoring with AWS IoT Greengrass Logs nella IoT AWS Greengrass Developer Guide.

- Per i dettagli sull'API, consulta Command CreateLoggerDefinitionReference AWS CLI .

`create-resource-definition-version`

Il seguente esempio di codice mostra come utilizzare`create-resource-definition-version`.

AWS CLI

Come creare una versione di una definizione di risorsa

L'`create-resource-definition-version`esempio seguente crea una nuova versione di a TwilioAuthToken.

```
aws greengrass create-resource-definition-version \
    --resource-definition-id "c8bb9ebc-c3fd-40a4-9c6a-568d75569d38" \
    --resources "[{\"Id\": \"TwilioAuthToken\",\"Name\": \"MyTwilioAuthToken
\",\"ResourceDataContainer\": {\"SecretsManagerSecretResourceData\": {\"ARN\":
```

```
 \"arn:aws:secretsmanager:us-west-2:123456789012:secret:greengrass-TwilioAuthToken-
ntSlp6\"}}}]"
```

Output:

```
{
    "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/
resources/c8bb9ebc-c3fd-40a4-9c6a-568d75569d38/versions/b3bcada0-5fb6-42df-
bf0b-1ee4f15e769e",
    "CreationTimestamp": "2019-06-24T21:17:25.623Z",
    "Id": "c8bb9ebc-c3fd-40a4-9c6a-568d75569d38",
    "Version": "b3bcada0-5fb6-42df-bf0b-1ee4f15e769e"
}
```

- Per i dettagli sull'API, vedere CreateResourceDefinitionVersionin AWS CLI Command Reference.

`create-resource-definition`

Il seguente esempio di codice mostra come utilizzare`create-resource-definition`.

AWS CLI

Come creare una definizione di risorsa

L'esempio `create-resource-definition` seguente crea una definizione di risorsa che contiene un elenco di risorse da utilizzare in un gruppo Greengrass. In questo esempio, viene inclusa una versione iniziale della definizione di risorsa fornendo un elenco di risorse. L'elenco include una risorsa per un token di autorizzazione Twilio e l'ARN per un segreto archiviato in Secrets AWS Manager. Prima di poter creare la definizione di risorsa, è necessario creare un segreto.

```
aws greengrass create-resource-definition \
    --name MyGreengrassResources \
    --initial-version "{\"Resources\": [{\"Id\": \"TwilioAuthToken
\",\"Name\": \"MyTwilioAuthToken\",\"ResourceDataContainer\":
 {\"SecretsManagerSecretResourceData\": {\"ARN\": \"arn:aws:secretsmanager:us-
west-2:123456789012:secret:greengrass-TwilioAuthToken-ntSlp6\"}}}]}"
```

Output:

```
{
    "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/
resources/c8bb9ebc-c3fd-40a4-9c6a-568d75569d38",
    "CreationTimestamp": "2019-06-19T21:51:28.212Z",
    "Id": "c8bb9ebc-c3fd-40a4-9c6a-568d75569d38",
    "LastUpdatedTimestamp": "2019-06-19T21:51:28.212Z",
    "LatestVersion": "a5f94d0b-f6bc-40f4-bb78-7a1c5fe13ba1",
    "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
definition/resources/c8bb9ebc-c3fd-40a4-9c6a-568d75569d38/versions/a5f94d0b-
f6bc-40f4-bb78-7a1c5fe13ba1",
    "Name": "MyGreengrassResources"
}
```

Per ulteriori informazioni, consulta Come configurare l'accesso alle risorse locali utilizzando l'interfaccia a riga di AWS comando nella AWS IoT Greengrass Developer Guide.

- Per i dettagli sull'API, consulta CreateResourceDefinition AWS CLICommand Reference.

`create-software-update-job`

Il seguente esempio di codice mostra come utilizzare`create-software-update-job`.

AWS CLI

Come creare un processo di aggiornamento del software per il core

L'`create-software-update-job`esempio seguente crea un processo di aggiornamento over-the-air (OTA) per aggiornare il software AWS IoT Greengrass Core sul core il cui nome è. `MyFirstGroup_Core` Questo comando richiede un ruolo IAM che consenta l'accesso ai pacchetti di aggiornamento software in Amazon S3 e includa `iot.amazonaws.com` come entità attendibile.

```
aws greengrass create-software-update-job \
    --update-targets-architecture armv7l \
    --update-targets [\"arn:aws:iot:us-west-2:123456789012:thing/MyFirstGroup_Core
\"] \
    --update-targets-operating-system raspbian \
    --software-to-update core \
    --s3-url-signer-role arn:aws:iam::123456789012:role/OTA_signer_role \
    --update-agent-log-level WARN
```

Output:

```
{
    "IotJobId": "GreengrassUpdateJob_30b353e3-3af7-4786-be25-4c446663c09e",
    "IotJobArn": "arn:aws:iot:us-west-2:123456789012:job/
GreengrassUpdateJob_30b353e3-3af7-4786-be25-4c446663c09e",
    "PlatformSoftwareVersion": "1.9.3"
}
```

Per ulteriori informazioni, consulta Aggiornamenti OTA del software AWS IoT Greengrass Core nella IoT AWS Greengrass Developer Guide.

- Per i dettagli sull'API, consulta AWS CLI Command CreateSoftwareUpdateJobReference.

`create-subscription-definition-version`

Il seguente esempio di codice mostra come utilizzare`create-subscription-definition-version`.

AWS CLI

Come creare una nuova versione della definizione di abbonamento

L'esempio `create-subscription-definition-version` seguente crea una nuova versione di una definizione di abbonamento che contiene tre abbonamenti: una notifica di attivazione, un input di temperatura e uno stato di output.

```
aws greengrass create-subscription-definition-version \
    --subscription-definition-id "9d611d57-5d5d-44bd-a3b4-feccbdd69112" \
    --subscriptions "[{\"Id\": \"TriggerNotification\", \"Source\":
 \"arn:aws:lambda:us-west-2:123456789012:function:TempMonitor:GG_TempMonitor
\", \"Subject\": \"twilio/txt\", \"Target\": \"arn:aws:greengrass:us-west-2::/
connectors/TwilioNotifications/versions/1\"},{\"Id\": \"TemperatureInput\", \"Source
\": \"cloud\", \"Subject\": \"temperature/input\", \"Target\": \"arn:aws:lambda:us-
west-2:123456789012:function:TempMonitor:GG_TempMonitor\"},{\"Id\": \"OutputStatus
\", \"Source\": \"arn:aws:greengrass:us-west-2::/connectors/TwilioNotifications/
versions/1\", \"Subject\": \"twilio/message/status\", \"Target\": \"cloud\"}]"
```

Output:

```
{
    "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/
subscriptions/9d611d57-5d5d-44bd-a3b4-feccbdd69112/versions/7b65dfae-50b6-4d0f-
b3e0-27728bfb0620",
```

```
    "CreationTimestamp": "2019-06-24T21:21:33.837Z",
    "Id": "9d611d57-5d5d-44bd-a3b4-feccbdd69112",
    "Version": "7b65dfae-50b6-4d0f-b3e0-27728bfb0620"
}
```

- Per i dettagli sull'API, consulta CreateSubscriptionDefinitionVersion AWS CLICommand Reference.

`create-subscription-definition`

Il seguente esempio di codice mostra come utilizzare`create-subscription-definition`.

AWS CLI

Come creare una definizione di abbonamento

L'esempio `create-subscription-definition` seguente crea una definizione di abbonamento e ne specifica la versione iniziale. La versione iniziale contiene tre abbonamenti: uno per l'argomento MQTT a cui è abbonato il connettore, uno per consentire a una funzione di ricevere letture di temperatura dall' AWS IoT e uno per consentire all' AWS IoT di ricevere informazioni sullo stato dal connettore. L'esempio fornisce l'ARN per l'alias della funzione Lambda creato in precedenza utilizzando il comando `create-alias` di Lambda.

```
aws greengrass create-subscription-definition \
    --initial-version "{\"Subscriptions\": [{\"Id\":
 \"TriggerNotification\", \"Source\": \"arn:aws:lambda:us-
west-2:123456789012:function:TempMonitor:GG_TempMonitor\", \"Subject\":
 \"twilio/txt\", \"Target\": \"arn:aws:greengrass:us-west-2::/connectors/
TwilioNotifications/versions/1\"},{\"Id\": \"TemperatureInput\", \"Source\":
 \"cloud\", \"Subject\": \"temperature/input\", \"Target\": \"arn:aws:lambda:us-
west-2:123456789012:function:TempMonitor:GG_TempMonitor\"},{\"Id\": \"OutputStatus
\", \"Source\": \"arn:aws:greengrass:us-west-2::/connectors/TwilioNotifications/
versions/1\", \"Subject\": \"twilio/message/status\", \"Target\": \"cloud\"}]}"
```

Output:

```
{
    "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/
subscriptions/9d611d57-5d5d-44bd-a3b4-feccbdd69112",
    "CreationTimestamp": "2019-06-19T22:34:26.677Z",
    "Id": "9d611d57-5d5d-44bd-a3b4-feccbdd69112",
```

```
    "LastUpdatedTimestamp": "2019-06-19T22:34:26.677Z",
    "LatestVersion": "aa645c47-ac90-420d-9091-8c7ffa4f103f",
    "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
definition/subscriptions/9d611d57-5d5d-44bd-a3b4-feccbdd69112/versions/aa645c47-
ac90-420d-9091-8c7ffa4f103f"
}
```

Per ulteriori informazioni, consulta Nozioni di base sui connettori (CLI) nella Guida per gli sviluppatori di AWS IoT Greengrass.

- Per i dettagli sull'API, vedere CreateSubscriptionDefinitionin Command Reference.AWS CLI

`delete-connector-definition`

Il seguente esempio di codice mostra come utilizzare`delete-connector-definition`.

AWS CLI

Come eliminare una definizione di connettore

L'esempio `delete-connector-definition` seguente elimina la definizione di connettore Greengrass specificata. Se si elimina una definizione di connettore utilizzata da un gruppo, tale gruppo non può essere distribuito correttamente.

```
aws greengrass delete-connector-definition \
    --connector-definition-id "b5c4ebfd-f672-49a3-83cd-31c7216a7bb8"
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta DeleteConnectorDefinition AWS CLICommand Reference.

`delete-core-definition`

Il seguente esempio di codice mostra come utilizzare`delete-core-definition`.

AWS CLI

Come eliminare una definizione core

L'esempio `delete-core-definition` seguente elimina la definizione core di Greengrass specificata, incluse tutte le versioni. Se elimini una definizione core associata a un gruppo Greengrass, quel gruppo non può essere distribuito correttamente.

```
aws greengrass delete-core-definition \
    --core-definition-id "ff36cc5f-9f98-4994-b468-9d9b6dc52abd"
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta DeleteCoreDefinition AWS CLICommand Reference.

`delete-device-definition`

Il seguente esempio di codice mostra come utilizzare`delete-device-definition`.

AWS CLI

Come eliminare una definizione di dispositivo

L'esempio `delete-device-definition` seguente elimina la definizione di dispositivo specificata, incluse tutte le relative versioni. Se si elimina una versione di definizione di dispositivo utilizzata da una versione di gruppo, la versione di gruppo non può essere distribuita correttamente.

```
aws greengrass delete-device-definition \
    --device-definition-id "f9ba083d-5ad4-4534-9f86-026a45df1ccd"
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta DeleteDeviceDefinition AWS CLICommand Reference.

`delete-function-definition`

Il seguente esempio di codice mostra come utilizzare`delete-function-definition`.

AWS CLI

Come eliminare una definizione di funzione

L'esempio `delete-function-definition` seguente elimina la definizione di funzione Greengrass specificata. Se si elimina una definizione di funzione utilizzata da un gruppo, tale gruppo non può essere distribuito correttamente.

```
aws greengrass delete-function-definition \
```

```
    --function-definition-id "fd4b906a-dff3-4c1b-96eb-52ebfcfac06a"
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta DeleteFunctionDefinition AWS CLICommand Reference.

`delete-group`

Il seguente esempio di codice mostra come utilizzare`delete-group`.

AWS CLI

Come eliminare un gruppo

L'esempio `delete-group` seguente elimina il gruppo Greengrass specificato.

```
aws greengrass delete-group \
    --group-id "4e22bd92-898c-436b-ade5-434d883ff749"
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta DeleteGroup AWS CLICommand Reference.

`delete-logger-definition`

Il seguente esempio di codice mostra come utilizzare`delete-logger-definition`.

AWS CLI

Come eliminare una definizione di logger

L'esempio `delete-logger-definition` seguente elimina la definizione di logger specificata, incluse tutte le versioni della definizione di logger. Se si elimina una versione di definizione di logger utilizzata da una versione di gruppo, la versione di gruppo non può essere distribuita correttamente.

```
aws greengrass delete-logger-definition \
    --logger-definition-id "a454b62a-5d56-4ca9-bdc4-8254e1662cb0"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Monitoring with AWS IoT Greengrass Logs nella IoT AWS Greengrass Developer Guide.

- Per i dettagli sull'API, consulta Command DeleteLoggerDefinitionReference AWS CLI .

`delete-resource-definition`

Il seguente esempio di codice mostra come utilizzare`delete-resource-definition`.

AWS CLI

Come eliminare una definizione di risorsa

L'esempio `delete-resource-definition` seguente elimina la definizione di risorsa specificata, incluse tutte le versioni della risorsa. Se si elimina una definizione di risorsa utilizzata da un gruppo, tale gruppo non può essere distribuito correttamente.

```
aws greengrass delete-resource-definition \
    --resource-definition-id "ad8c101d-8109-4b0e-b97d-9cc5802ab658"
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta DeleteResourceDefinition AWS CLICommand Reference.

`delete-subscription-definition`

Il seguente esempio di codice mostra come utilizzare`delete-subscription-definition`.

AWS CLI

Come eliminare una definizione di abbonamento

L'esempio `delete-subscription-definition` seguente elimina la definizione di abbonamento Greengrass specificata. Se si elimina un abbonamento utilizzato da un gruppo, tale gruppo non può essere distribuito correttamente.

```
aws greengrass delete-subscription-definition \
    --subscription-definition-id "cd6f1c37-d9a4-4e90-be94-01a7404f5967"
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta DeleteSubscriptionDefinition AWS CLICommand Reference.

`disassociate-role-from-group`

Il seguente esempio di codice mostra come utilizzare`disassociate-role-from-group`.

AWS CLI

Come dissociare il ruolo da un gruppo Greengrass

L'esempio `disassociate-role-from-group` seguente dissocia il ruolo IAM dal gruppo Greengrass specificato.

```
aws greengrass disassociate-role-from-group \
    --group-id 2494ee3f-7f8a-4e92-a78b-d205f808b84b
```

Output:

```
{
    "DisassociatedAt": "2019-09-10T20:05:49Z"
}
```

Per ulteriori informazioni, consulta Configurare il ruolo del gruppo nella Guida per gli sviluppatori di AWS IoT Greengrass.

- Per i dettagli sull'API, consulta DisassociateRoleFromGroup AWS CLICommand Reference.

`disassociate-service-role-from-account`

Il seguente esempio di codice mostra come utilizzare`disassociate-service-role-from-account`.

AWS CLI

Per dissociare un ruolo di servizio dal proprio account AWS

L'`disassociate-service-role-from-account`esempio seguente rimuove il ruolo di servizio associato all' AWS account. Se non si utilizza il ruolo di servizio in nessuna AWS regione, utilizzare il `delete-role-policy` comando per scollegare la politica `AWSGreengrassResourceAccessRolePolicy` gestita dal ruolo, quindi utilizzare il `delete-role` comando per eliminare il ruolo.

```
aws greengrass disassociate-service-role-from-account
```

Output:

```
{
    "DisassociatedAt": "2019-06-25T22:12:55Z"
}
```

Per ulteriori informazioni, consulta Ruolo del servizio Greengrass nella Guida per gli sviluppatori di AWS IoT Greengrass.

- Per i dettagli sull'API, consulta DisassociateServiceRoleFromAccount AWS CLICommand Reference.

`get-associated-role`

Il seguente esempio di codice mostra come utilizzare`get-associated-role`.

AWS CLI

Come associare il ruolo a un gruppo Greengrass

L'esempio `get-associated-role` seguente ottiene il ruolo IAM associato al gruppo Greengrass specificato. Il ruolo di gruppo viene utilizzato dalle funzioni e dai connettori Lambda locali per accedere ai servizi AWS .

```
aws greengrass get-associated-role \
    --group-id 2494ee3f-7f8a-4e92-a78b-d205f808b84b
```

Output:

```
{
    "RoleArn": "arn:aws:iam::123456789012:role/GG-Group-Role",
    "AssociatedAt": "2019-09-10T20:03:30Z"
}
```

Per ulteriori informazioni, consulta Configurare il ruolo del gruppo nella Guida per gli sviluppatori di AWS IoT Greengrass.

- Per i dettagli sull'API, consulta GetAssociatedRole AWS CLICommand Reference.

`get-bulk-deployment-status`

Il seguente esempio di codice mostra come utilizzare`get-bulk-deployment-status`.

AWS CLI

Come controllare lo stato dell'implementazione in blocco

L'esempio `get-bulk-deployment-status` seguente recupera le informazioni sullo stato per l'operazione di implementazione in blocco specificata. In questo esempio, il file che ha specificato i gruppi da distribuire ha un record di input non valido.

```
aws greengrass get-bulk-deployment-status \
    --bulk-deployment-id "870fb41b-6288-4e0c-bc76-a7ba4b4d3267"
```

Output:

```
{
    "BulkDeploymentMetrics": {
        "InvalidInputRecords": 1,
        "RecordsProcessed": 1,
        "RetryAttempts": 0
    },
    "BulkDeploymentStatus": "Completed",
    "CreatedAt": "2019-06-25T16:11:33.265Z",
    "tags": {}
}
```

Per ulteriori informazioni, consulta Creare implementazioni in blocco per i gruppi nella Guida per gli sviluppatori di AWS IoT Greengrass.

- Per i dettagli sull'API, consulta GetBulkDeploymentStatus AWS CLICommand Reference.

`get-connectivity-info`

Il seguente esempio di codice mostra come utilizzare`get-connectivity-info`.

AWS CLI

Come ottenere le informazioni di connettività per un core Greengrass

L'esempio `get-connectivity-info` seguente visualizza gli endpoint che i dispositivi possono utilizzare per connettersi al core Greengrass specificato. Le informazioni sulla connettività sono un

elenco di indirizzi IP o nomi di dominio, con i numeri di porta corrispondenti e i metadati opzionali definiti dal cliente.

```
aws greengrass get-connectivity-info \
    --thing-name "MyGroup_Core"
```

Output:

```
{
    "ConnectivityInfo": [
        {
            "Metadata": "",
            "PortNumber": 8883,
            "HostAddress": "127.0.0.1",
            "Id": "AUTOIP_127.0.0.1_0"
        },
        {
            "Metadata": "",
            "PortNumber": 8883,
            "HostAddress": "192.168.1.3",
            "Id": "AUTOIP_192.168.1.3_1"
        },
        {
            "Metadata": "",
            "PortNumber": 8883,
            "HostAddress": "::1",
            "Id": "AUTOIP_::1_2"
        },
        {
            "Metadata": "",
            "PortNumber": 8883,
            "HostAddress": "fe80::1e69:ed93:f5b:f6d",
            "Id": "AUTOIP_fe80::1e69:ed93:f5b:f6d_3"
        }
    ]
}
```

- Per i dettagli sull'API, consulta GetConnectivityInfo AWS CLICommand Reference.

`get-connector-definition-version`

Il seguente esempio di codice mostra come utilizzare`get-connector-definition-version`.

AWS CLI

Come recuperare informazioni su una versione specifica della definizione di un connettore

L'esempio `get-connector-definition-version` seguente recupera informazioni sulla versione specificata della definizione di connettore specificata. Per recuperare tutte le versioni IDs della definizione del connettore, utilizzate il `list-connector-definition-versions` comando. Per recuperare l'ID dell'ultima versione aggiunta alla definizione di connettore, utilizza il comando `get-connector-definition` e controlla la proprietà `LatestVersion`.

```
aws greengrass get-connector-definition-version \
    --connector-definition-id "b5c4ebfd-f672-49a3-83cd-31c7216a7bb8" \
    --connector-definition-version-id "63c57963-c7c2-4a26-a7e2-7bf478ea2623"
```

Output:

```
{
    "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/
connectors/b5c4ebfd-f672-49a3-83cd-31c7216a7bb8/versions/63c57963-c7c2-4a26-
a7e2-7bf478ea2623",
    "CreationTimestamp": "2019-06-19T19:30:01.300Z",
    "Definition": {
        "Connectors": [
            {
                "ConnectorArn": "arn:aws:greengrass:us-west-2::/connectors/SNS/
versions/1",
                "Id": "MySNSConnector",
                "Parameters": {
                    "DefaultSNSArn": "arn:aws:sns:us-
west-2:123456789012:GGConnectorTopic"
                }
            }
        ]
    },
    "Id": "b5c4ebfd-f672-49a3-83cd-31c7216a7bb8",
    "Version": "63c57963-c7c2-4a26-a7e2-7bf478ea2623"
}
```

Per ulteriori informazioni, consulta Integrazione con servizi e protocolli tramite i connettori Greengrass nella Guida per gli sviluppatori di AWS IoT Greengrass.

- Per i dettagli sull'API, vedere GetConnectorDefinitionVersionin AWS CLI Command Reference.

`get-connector-definition`

Il seguente esempio di codice mostra come utilizzare`get-connector-definition`.

AWS CLI

Come recuperare le informazioni su una definizione di connettore

L'esempio `get-connector-definition` seguente recupera le informazioni sulla definizione del connettore specificata. Per recuperare le definizioni IDs dei connettori, utilizzate il `list-connector-definitions` comando.

```
aws greengrass get-connector-definition \
    --connector-definition-id "b5c4ebfd-f672-49a3-83cd-31c7216a7bb8"
```

Output:

```
{
    "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/
connectors/b5c4ebfd-f672-49a3-83cd-31c7216a7bb8",
    "CreationTimestamp": "2019-06-19T19:30:01.300Z",
    "Id": "b5c4ebfd-f672-49a3-83cd-31c7216a7bb8",
    "LastUpdatedTimestamp": "2019-06-19T19:30:01.300Z",
    "LatestVersion": "63c57963-c7c2-4a26-a7e2-7bf478ea2623",
    "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
definition/connectors/b5c4ebfd-f672-49a3-83cd-31c7216a7bb8/versions/63c57963-
c7c2-4a26-a7e2-7bf478ea2623",
    "Name": "MySNSConnector",
    "tags": {}
}
```

Per ulteriori informazioni, consulta Integrazione con servizi e protocolli tramite i connettori Greengrass nella Guida per gli sviluppatori di AWS IoT Greengrass.

- Per i dettagli sull'API, consulta GetConnectorDefinition AWS CLICommand Reference.

`get-core-definition-version`

Il seguente esempio di codice mostra come utilizzare`get-core-definition-version`.

AWS CLI

Come recuperare dettagli su una versione specifica della definizione di core Greengrass

L'esempio `get-core-definition-version` seguente recupera le informazioni sulla versione specificata della definizione di core specificata. Per recuperare tutte le versioni della definizione principale, utilizzate il `list-core-definition-versions` comando. IDs Per recuperare l'ID dell'ultima versione aggiunta alla definizione core, utilizza il comando `get-core-definition` e controlla la proprietà `LatestVersion`.

```
aws greengrass get-core-definition-version \
    --core-definition-id "c906ed39-a1e3-4822-a981-7b9bd57b4b46"  \
    --core-definition-version-id "42aeeac3-fd9d-4312-a8fd-ffa9404a20e0"
```

Output:

```
{
    "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/cores/
c906ed39-a1e3-4822-a981-7b9bd57b4b46/versions/42aeeac3-fd9d-4312-a8fd-ffa9404a20e0",
    "CreationTimestamp": "2019-06-18T16:21:21.351Z",
    "Definition": {
        "Cores": [
            {
                "CertificateArn": "arn:aws:iot:us-
west-2:123456789012:cert/928dea7b82331b47c3ff77b0e763fc5e64e2f7c884e6ef391baed9b6b8e21b45",
                "Id": "1a39aac7-0885-4417-91f6-23e4cea6c511",
                "SyncShadow": false,
                "ThingArn": "arn:aws:iot:us-west-2:123456789012:thing/
GGGroup4Pi3_Core"
            }
        ]
    },
    "Id": "c906ed39-a1e3-4822-a981-7b9bd57b4b46",
    "Version": "42aeeac3-fd9d-4312-a8fd-ffa9404a20e0"
}
```

- Per i dettagli sull'API, consulta GetCoreDefinitionVersion AWS CLICommand Reference.

`get-core-definition`

Il seguente esempio di codice mostra come utilizzare`get-core-definition`.

AWS CLI

Come recuperare i dettagli per una definizione di core Greengrass

L'esempio `get-core-definition` seguente recupera le informazioni sulla definizione di core specificata. Per recuperare le IDs definizioni principali, utilizzate il `list-core-definitions` comando.

```
aws greengrass get-core-definition \
    --core-definition-id "c906ed39-a1e3-4822-a981-7b9bd57b4b46"
```

Output:

```
{
    "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/
cores/237d6916-27cf-457f-ba0c-e86cfb5d25cd",
    "CreationTimestamp": "2018-10-18T04:47:06.721Z",
    "Id": "237d6916-27cf-457f-ba0c-e86cfb5d25cd",
    "LastUpdatedTimestamp": "2018-10-18T04:47:06.721Z",
    "LatestVersion": "bd2cd6d4-2bc5-468a-8962-39e071e34b68",
    "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/definition/cores/237d6916-27cf-457f-ba0c-e86cfb5d25cd/versions/
bd2cd6d4-2bc5-468a-8962-39e071e34b68",
    "tags": {}
}
```

- Per i dettagli sull'API, consulta GetCoreDefinition AWS CLICommand Reference.

`get-deployment-status`

Il seguente esempio di codice mostra come utilizzare`get-deployment-status`.

AWS CLI

Come recuperare lo stato di un'implementazione

L'esempio `get-deployment-status` seguente recupera lo stato per l'implementazione specificata del gruppo Greengrass specificato. Per ottenere l'ID di implementazione, utilizza il comando `list-deployments` e specifica l'ID del gruppo.

```
aws greengrass get-deployment-status \
    --group-id "1013db12-8b58-45ff-acc7-704248f66731" \
    --deployment-id "1065b8a0-812b-4f21-9d5d-e89b232a530f"
```

Output:

```
{
    "DeploymentStatus": "Success",
    "DeploymentType": "NewDeployment",
    "UpdatedAt": "2019-06-18T17:04:44.761Z"
}
```

- Per i dettagli sull'API, consulta GetDeploymentStatus AWS CLICommand Reference.

`get-device-definition-version`

Il seguente esempio di codice mostra come utilizzare`get-device-definition-version`.

AWS CLI

Come ottenere una versione della definizione di dispositivo

L'esempio `get-device-definition-version` seguente recupera le informazioni sulla versione specificata della definizione di dispositivo specificata. Per recuperare tutte le versioni IDs della definizione del dispositivo, utilizzare il `list-device-definition-versions` comando. Per recuperare l'ID dell'ultima versione aggiunta alla definizione di dispositivo, utilizza il comando `get-device-definition` e controlla la proprietà `LatestVersion`.

```
aws greengrass get-device-definition-version \
    --device-definition-id "f9ba083d-5ad4-4534-9f86-026a45df1ccd" \
    --device-definition-version-id "83c13984-6fed-447e-84d5-5b8aa45d5f71"
```

Output:

```
{
    "Definition": {
        "Devices": [
            {
                "CertificateArn": "arn:aws:iot:us-
west-2:123456789012:cert/6c52ce1b47bde88a637e9ccdd45fe4e4c2c0a75a6866f8f63d980ee22fa51e02",
                "ThingArn": "arn:aws:iot:us-west-2:123456789012:thing/
ExteriorTherm",
                "SyncShadow": true,
                "Id": "ExteriorTherm"
            },
            {
```

```
                "CertificateArn": "arn:aws:iot:us-
west-2:123456789012:cert/66a415ec415668c2349a76170b64ac0878231c1e21ec83c10e92a18bd568eb92",
                "ThingArn": "arn:aws:iot:us-west-2:123456789012:thing/
InteriorTherm",
                "SyncShadow": true,
                "Id": "InteriorTherm"
            }
        ]
    },
    "Version": "83c13984-6fed-447e-84d5-5b8aa45d5f71",
    "CreationTimestamp": "2019-09-11T00:15:09.838Z",
    "Id": "f9ba083d-5ad4-4534-9f86-026a45df1ccd",
    "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
definition/devices/f9ba083d-5ad4-4534-9f86-026a45df1ccd/
versions/83c13984-6fed-447e-84d5-5b8aa45d5f71"
}
```

- Per i dettagli sull'API, vedere GetDeviceDefinitionVersionin AWS CLI Command Reference.

`get-device-definition`

Il seguente esempio di codice mostra come utilizzare`get-device-definition`.

AWS CLI

Come ottenere una definizione di dispositivo

L'esempio `get-device-definition` seguente recupera le informazioni sulla definizione di dispositivo specificata. Per recuperare le definizioni IDs del dispositivo, utilizzare il `list-device-definitions` comando.

```
aws greengrass get-device-definition \
    --device-definition-id "f9ba083d-5ad4-4534-9f86-026a45df1ccd"
```

Output:

```
{
    "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/definition/devices/f9ba083d-5ad4-4534-9f86-026a45df1ccd/
versions/83c13984-6fed-447e-84d5-5b8aa45d5f71",
    "Name": "TemperatureSensors",
    "tags": {},
```

```
    "LastUpdatedTimestamp": "2019-09-11T00:19:03.698Z",
    "LatestVersion": "83c13984-6fed-447e-84d5-5b8aa45d5f71",
    "CreationTimestamp": "2019-09-11T00:11:06.197Z",
    "Id": "f9ba083d-5ad4-4534-9f86-026a45df1ccd",
    "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/
devices/f9ba083d-5ad4-4534-9f86-026a45df1ccd"
}
```

- Per i dettagli sull'API, consulta GetDeviceDefinition AWS CLICommand Reference.

`get-function-definition-version`

Il seguente esempio di codice mostra come utilizzare`get-function-definition-version`.

AWS CLI

Come recuperare dettagli su una versione specifica di una funzione Lambda

L'esempio `get-function-definition-version` seguente recupera le informazioni sulla versione specificata della definizione di funzione specificata. Per recuperare tutte le versioni IDs della definizione della funzione, utilizzate il `list-function-definition-versions` comando. Per recuperare l'ID dell'ultima versione aggiunta alla definizione di funzione, utilizza il comando `get-function-definition` e controlla la proprietà `LatestVersion`.

```
aws greengrass get-function-definition-version \
    --function-definition-id "063f5d1a-1dd1-40b4-9b51-56f8993d0f85" \
    --function-definition-version-id "9748fda7-1589-4fcc-ac94-f5559e88678b"
```

Output:

```
{
    "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/
functions/063f5d1a-1dd1-40b4-9b51-56f8993d0f85/versions/9748fda7-1589-4fcc-ac94-
f5559e88678b",
    "CreationTimestamp": "2019-06-18T17:04:30.776Z",
    "Definition": {
        "Functions": [
            {
                "FunctionArn": "arn:aws:lambda:::function:GGIPDetector:1",
                "FunctionConfiguration": {
                    "Environment": {},
```

```
                    "MemorySize": 32768,
                    "Pinned": true,
                    "Timeout": 3
                },
                "Id": "26b69bdb-e547-46bc-9812-84ec04b6cc8c"
            },
            {
                "FunctionArn": "arn:aws:lambda:us-
west-2:123456789012:function:Greengrass_HelloWorld:GG_HelloWorld",
                "FunctionConfiguration": {
                    "EncodingType": "json",
                    "Environment": {
                        "Variables": {}
                    },
                    "MemorySize": 16384,
                    "Pinned": true,
                    "Timeout": 25
                },
                "Id": "384465a8-eedf-48c6-b793-4c35f7bfae9b"
            }
        ]
    },
    "Id": "063f5d1a-1dd1-40b4-9b51-56f8993d0f85",
    "Version": "9748fda7-1589-4fcc-ac94-f5559e88678b"
}
```

- Per i dettagli sull'API, vedere GetFunctionDefinitionVersionin AWS CLI Command Reference.

`get-function-definition`

Il seguente esempio di codice mostra come utilizzare`get-function-definition`.

AWS CLI

Come recuperare la definizione di funzione

L'esempio `get-function-definition` seguente visualizza i dettagli della definizione di istanza specificata. Per recuperare le definizioni IDs delle funzioni, utilizzate il `list-function-definitions` comando.

```
aws greengrass get-function-definition \
    --function-definition-id "063f5d1a-1dd1-40b4-9b51-56f8993d0f85"
```

Output:

```
{
    "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/
functions/063f5d1a-1dd1-40b4-9b51-56f8993d0f85",
    "CreationTimestamp": "2019-06-18T16:21:21.431Z",
    "Id": "063f5d1a-1dd1-40b4-9b51-56f8993d0f85",
    "LastUpdatedTimestamp": "2019-06-18T16:21:21.431Z",
    "LatestVersion": "9748fda7-1589-4fcc-ac94-f5559e88678b",
    "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/definition/functions/063f5d1a-1dd1-40b4-9b51-56f8993d0f85/
versions/9748fda7-1589-4fcc-ac94-f5559e88678b",
    "tags": {}
}
```

- Per i dettagli sull'API, consulta GetFunctionDefinition AWS CLICommand Reference.

`get-group-certificate-authority`

Il seguente esempio di codice mostra come utilizzare`get-group-certificate-authority`.

AWS CLI

Come recuperare la CA associata a un gruppo Greengrass

L'esempio `get-group-certificate-authority` seguente recupera l'autorità di certificazione (CA) associata al gruppo Greengrass specificato. Per ottenere l'ID dell'autorità di certificazione, utilizza il comando `list-group-certificate-authorities` e specifica l'ID del gruppo.

```
aws greengrass get-group-certificate-authority \
    --group-id "1013db12-8b58-45ff-acc7-704248f66731" \
    --certificate-authority-
id "f0430e1736ea8ed30cc5d5de9af67a7e3586bad9ae4d89c2a44163f65fdd8cf6"
```

Output:

```
{
    "GroupCertificateAuthorityArn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/groups/1013db12-8b58-45ff-acc7-704248f66731/certificateauthorities/
f0430e1736ea8ed30cc5d5de9af67a7e3586bad9ae4d89c2a44163f65fdd8cf6",
    "GroupCertificateAuthorityId":
 "f0430e1736ea8ed30cc5d5de9af67a7e3586bad9ae4d89c2a44163f65fdd8cf6",
```

```
    "PemEncodedCertificate": "-----BEGIN CERTIFICATE-----
MIICiTCCAfICCQD6m7oRw0uXOjANBgkqhkiG9w0BAQUFADCBWEXAMPLEGA1UEBhMC
VVMxCzAJBgNVBAgTAldBMRAwDEXAMPLEEwdTZWF0dGxlMQ8wDQYDVQQKEwZBbWF6
b24xFDASBgNVBAEXAMPLESBDb25zb2xlMRIwEAYDVQQDEwlUZXN0Q2lsYWMxHzAd
BgkqhkiG9w0BCQEWEG5vb25lQGFtYXpvbi5jEXAMPLENMTEwNDI1MjA0NTIxWhcN
MTIwNDI0MjA0EXAMPLEBiDELMAkGA1UEBhMCVVMxCzAJBgNVBAgTAldBMRAwDgYD
VQQHEwdTZWF0dGxlMQ8wDQYDVQQKEwZBbWEXAMPLEDASBgNVBAsTC0lBTSBDb25z
b2xlMRIwEAYDVQQDEwlUZXN0Q2lsYWEXAMPLEgkqhkiG9w0BCQEWEG5vb25lQGFt
YXpvbi5EXAMPLE8wDQYJKoZIhvcNAQEBBQADgY0AMIGJAoGBAMaK0dn+a4GmWIWJ
21uUSfwfEvySWtC2XADZ4nB+BLYgVIk60CEXAMPLE93vUEIO3IyNoH/f0wYK8m9T
rDHudUZg3qX4waLG5M43q7Wgc/MbQITxOUSQv7c7ugFFDzQGBzZswYEXAMPLEgpE
Ibb3OhjZnzcvQAaRHhdlQWIMm2nrAgMBAAEwDQYJKEXAMPLEAQEFBQADgYEAtCu4
nUhVVxYUntneD9+h8Mg9q6q+auNKyExzyLwaxlAoo7TJHidbtS4J5iNmZgXL0Fkb
FFBjvSfpJIlJ00zbhNYS5f6GuoEDmFJl0ZxBHjJnyp378OD8uTs7fLvjx79LjSTb
NYiytVbZPQUQ5Yaxu2jXnimvw3rrszlaEXAMPLE=
-----END CERTIFICATE-----\n"
}
```

- Per i dettagli sull'API, consulta GetGroupCertificateAuthority AWS CLICommand Reference.

`get-group-certificate-configuration`

Il seguente esempio di codice mostra come utilizzare`get-group-certificate-configuration`.

AWS CLI

Come recuperare la configurazione per l'autorità di certificazione utilizzata dal gruppo Greengrass

L'esempio `get-group-certificate-configuration` seguente recupera l'autorità di certificazione (CA) utilizzata dal gruppo Greengrass specificato.

```
aws greengrass get-group-certificate-configuration \
    --group-id "1013db12-8b58-45ff-acc7-704248f66731"
```

Output:

```
{
    "CertificateAuthorityExpiryInMilliseconds": 2524607999000,
    "CertificateExpiryInMilliseconds": 604800000,
    "GroupId": "1013db12-8b58-45ff-acc7-704248f66731"
}
```

- Per i dettagli sull'API, consulta GetGroupCertificateConfiguration AWS CLICommand Reference.

`get-group-version`

Il seguente esempio di codice mostra come utilizzare`get-group-version`.

AWS CLI

Come recuperare le informazioni su una versione di un gruppo Greengrass

L'esempio `get-group-version` seguente recupera le informazioni sulla versione specificata del gruppo specificato. Per recuperare tutte le versioni del gruppo, utilizzate il `list-group-versions` comando. IDs Per recuperare l'ID dell'ultima versione aggiunta al gruppo, utilizza il comando `get-group` e controlla la proprietà `LatestVersion`.

```
aws greengrass get-group-version \
    --group-id "1013db12-8b58-45ff-acc7-704248f66731"  \
    --group-version-id "115136b3-cfd7-4462-b77f-8741a4b00e5e"
```

Output:

```
{
    "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
groups/1013db12-8b58-45ff-acc7-704248f66731/versions/115136b3-cfd7-4462-
b77f-8741a4b00e5e",
    "CreationTimestamp": "2019-06-18T17:04:30.915Z",
    "Definition": {
        "CoreDefinitionVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/definition/cores/c906ed39-a1e3-4822-a981-7b9bd57b4b46/versions/42aeeac3-
fd9d-4312-a8fd-ffa9404a20e0",
        "FunctionDefinitionVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/definition/functions/063f5d1a-1dd1-40b4-9b51-56f8993d0f85/
versions/9748fda7-1589-4fcc-ac94-f5559e88678b",
        "SubscriptionDefinitionVersionArn": "arn:aws:greengrass:us-
west-2:123456789012:/greengrass/definition/subscriptions/70e49321-83d5-45d2-
bc09-81f4917ae152/versions/88ae8699-12ac-4663-ba3f-4d7f0519140b"
    },
    "Id": "1013db12-8b58-45ff-acc7-704248f66731",
    "Version": "115136b3-cfd7-4462-b77f-8741a4b00e5e"
}
```

- Per i dettagli sull'API, consulta GetGroupVersion AWS CLICommand Reference.

`get-group`

Il seguente esempio di codice mostra come utilizzare`get-group`.

AWS CLI

Come recuperare informazioni su un gruppo Greengrass

L'esempio `get-group` seguente recupera le informazioni sul gruppo Greengrass specificato. Per recuperare i IDs tuoi gruppi, usa il `list-groups` comando.

```
aws greengrass get-group \
    --group-id "1013db12-8b58-45ff-acc7-704248f66731"
```

Output:

```
{
    "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
groups/1013db12-8b58-45ff-acc7-704248f66731",
    "CreationTimestamp": "2019-06-18T16:21:21.457Z",
    "Id": "1013db12-8b58-45ff-acc7-704248f66731",
    "LastUpdatedTimestamp": "2019-06-18T16:21:21.457Z",
    "LatestVersion": "115136b3-cfd7-4462-b77f-8741a4b00e5e",
    "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
groups/1013db12-8b58-45ff-acc7-704248f66731/versions/115136b3-cfd7-4462-
b77f-8741a4b00e5e",
    "Name": "GGGroup4Pi3",
    "tags": {}
}
```

- Per i dettagli sull'API, consulta GetGroup AWS CLICommand Reference.

`get-logger-definition-version`

Il seguente esempio di codice mostra come utilizzare`get-logger-definition-version`.

AWS CLI

Come recuperare le informazioni su una versione di una definizione di logger

L'esempio `get-logger-definition-version` seguente recupera le informazioni sulla versione specificata della definizione di logger specificata. Per recuperare tutte le versioni IDs della definizione del logger, usa il `list-logger-definition-versions` comando. Per recuperare l'ID dell'ultima versione aggiunta alla definizione di logger, utilizza il comando `get-logger-definition` e controlla la proprietà `LatestVersion`.

```
aws greengrass get-logger-definition-version \
    --logger-definition-id "49eeeb66-f1d3-4e34-86e3-3617262abf23" \
    --logger-definition-version-id "5e3f6f64-a565-491e-8de0-3c0d8e0f2073"
```

Output:

```
{
    "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
definition/loggers/49eeeb66-f1d3-4e34-86e3-3617262abf23/versions/5e3f6f64-
a565-491e-8de0-3c0d8e0f2073",
    "CreationTimestamp": "2019-05-08T16:10:13.866Z",
    "Definition": {
        "Loggers": []
    },
    "Id": "49eeeb66-f1d3-4e34-86e3-3617262abf23",
    "Version": "5e3f6f64-a565-491e-8de0-3c0d8e0f2073"
}
```

- Per i dettagli sull'API, vedere GetLoggerDefinitionVersionin AWS CLI Command Reference.

`get-logger-definition`

Il seguente esempio di codice mostra come utilizzare`get-logger-definition`.

AWS CLI

Come recuperare le informazioni su una definizione di logger

L'esempio `get-logger-definition` seguente recupera le informazioni sulla definizione di logger specificata. Per recuperare le definizioni IDs del tuo logger, usa il `list-logger-definitions` comando.

```
aws greengrass get-logger-definition \
    --logger-definition-id "49eeeb66-f1d3-4e34-86e3-3617262abf23"
```

Output:

```
{
    "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/
loggers/49eeeb66-f1d3-4e34-86e3-3617262abf23",
    "CreationTimestamp": "2019-05-08T16:10:13.809Z",
    "Id": "49eeeb66-f1d3-4e34-86e3-3617262abf23",
    "LastUpdatedTimestamp": "2019-05-08T16:10:13.809Z",
    "LatestVersion": "5e3f6f64-a565-491e-8de0-3c0d8e0f2073",
    "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
definition/loggers/49eeeb66-f1d3-4e34-86e3-3617262abf23/versions/5e3f6f64-
a565-491e-8de0-3c0d8e0f2073",
    "tags": {}
}
```

- Per i dettagli sull'API, vedere GetLoggerDefinitionin AWS CLI Command Reference.

`get-resource-definition-version`

Il seguente esempio di codice mostra come utilizzare`get-resource-definition-version`.

AWS CLI

Come recuperare le informazioni su una versione specifica di una definizione di risorsa

L'esempio `get-resource-definition-version` seguente recupera le informazioni sulla versione specificata della definizione di risorsa specificata. Per recuperare tutte le versioni IDs della definizione della risorsa, utilizzate il `list-resource-definition-versions` comando. Per recuperare l'ID dell'ultima versione aggiunta alla definizione di risorsa, utilizza il comando `get-resource-definition` e controlla la proprietà `LatestVersion`.

```
aws greengrass get-resource-definition-version \
    --resource-definition-id "ad8c101d-8109-4b0e-b97d-9cc5802ab658" \
    --resource-definition-version-id "26e8829a-491a-464d-9c87-664bf6f6f2be"
```

Output:

```
{
    "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
definition/resources/ad8c101d-8109-4b0e-b97d-9cc5802ab658/
versions/26e8829a-491a-464d-9c87-664bf6f6f2be",
```

```
    "CreationTimestamp": "2019-06-19T16:40:59.392Z",
    "Definition": {
        "Resources": [
            {
                "Id": "26ff3f7b-839a-4217-9fdc-a218308b3963",
                "Name": "usb-port",
                "ResourceDataContainer": {
                    "LocalDeviceResourceData": {
                        "GroupOwnerSetting": {
                            "AutoAddGroupOwner": false
                        },
                        "SourcePath": "/dev/bus/usb"
                    }
                }
            }
        ]
    },
    "Id": "ad8c101d-8109-4b0e-b97d-9cc5802ab658",
    "Version": "26e8829a-491a-464d-9c87-664bf6f6f2be"
}
```

- Per i dettagli sull'API, consulta GetResourceDefinitionVersion AWS CLICommand Reference.

`get-resource-definition`

Il seguente esempio di codice mostra come utilizzare`get-resource-definition`.

AWS CLI

Come recuperare le informazioni sulla definizione di risorsa

L'esempio `get-resource-definition` seguente recupera informazioni sulla definizione di risorsa specificata. Per recuperare le definizioni IDs delle risorse, usa il `list-resource-definitions` comando.

```
aws greengrass get-resource-definition \
    --resource-definition-id "ad8c101d-8109-4b0e-b97d-9cc5802ab658"
```

Output:

```
{
```

```
    "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/
resources/ad8c101d-8109-4b0e-b97d-9cc5802ab658",
    "CreationTimestamp": "2019-06-19T16:40:59.261Z",
    "Id": "ad8c101d-8109-4b0e-b97d-9cc5802ab658",
    "LastUpdatedTimestamp": "2019-06-19T16:40:59.261Z",
    "LatestVersion": "26e8829a-491a-464d-9c87-664bf6f6f2be",
    "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/definition/resources/ad8c101d-8109-4b0e-b97d-9cc5802ab658/
versions/26e8829a-491a-464d-9c87-664bf6f6f2be",
    "tags": {}
}
```

- Per i dettagli sull'API, consulta GetResourceDefinition AWS CLICommand Reference.

`get-service-role-for-account`

Il seguente esempio di codice mostra come utilizzare`get-service-role-for-account`.

AWS CLI

Come recuperare i dettagli relativi al ruolo di servizio collegato all'account

L'`get-service-role-for-account`esempio seguente recupera le informazioni sul ruolo di servizio associato all' AWS account.

```
aws greengrass get-service-role-for-account
```

Output:

```
{
    "AssociatedAt": "2018-10-18T15:59:20Z",
    "RoleArn": "arn:aws:iam::123456789012:role/service-role/Greengrass_ServiceRole"
}
```

Per ulteriori informazioni, consulta Ruolo del servizio Greengrass nella Guida per gli sviluppatori di AWS IoT Greengrass.

- Per i dettagli sull'API, consulta GetServiceRoleForAccount AWS CLICommand Reference.

`get-subscription-definition-version`

Il seguente esempio di codice mostra come utilizzare`get-subscription-definition-version`.

AWS CLI

Come recuperare le informazioni su una versione specifica di una definizione di abbonamento

L'esempio `get-subscription-definition-version` seguente recupera le informazioni sulla versione specificata della definizione di abbonamento specificata. Per recuperare tutte le versioni IDs della definizione di sottoscrizione, utilizzare il `list-subscription-definition-versions` comando. Per recuperare l'ID dell'ultima versione aggiunta alla definizione di abbonamento, utilizza il comando `get-subscription-definition` e controlla la proprietà `LatestVersion`.

```
aws greengrass get-subscription-definition-version \
    --subscription-definition-id "70e49321-83d5-45d2-bc09-81f4917ae152" \
    --subscription-definition-version-id "88ae8699-12ac-4663-ba3f-4d7f0519140b"
```

Output:

```
{
    "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/
subscriptions/70e49321-83d5-45d2-bc09-81f4917ae152/versions/88ae8699-12ac-4663-
ba3f-4d7f0519140b",
    "CreationTimestamp": "2019-06-18T17:03:52.499Z",
    "Definition": {
        "Subscriptions": [
            {
                "Id": "692c4484-d89f-4f64-8edd-1a041a65e5b6",
                "Source": "arn:aws:lambda:us-
west-2:123456789012:function:Greengrass_HelloWorld:GG_HelloWorld",
                "Subject": "hello/world",
                "Target": "cloud"
            }
        ]
    },
    "Id": "70e49321-83d5-45d2-bc09-81f4917ae152",
    "Version": "88ae8699-12ac-4663-ba3f-4d7f0519140b"
}
```

- Per i dettagli sull'API, consulta GetSubscriptionDefinitionVersion AWS CLICommand Reference.

`get-subscription-definition`

Il seguente esempio di codice mostra come utilizzare`get-subscription-definition`.

AWS CLI

Come recuperare le informazioni su una definizione di abbonamento

L'esempio `get-subscription-definition` seguente recupera le informazioni sulla definizione di abbonamento specificata. Per recuperare le definizioni IDs delle sottoscrizioni, utilizzare il `list-subscription-definitions` comando.

```
aws greengrass get-subscription-definition \
    --subscription-definition-id "70e49321-83d5-45d2-bc09-81f4917ae152"
```

Output:

```
{
    "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/definition/
subscriptions/70e49321-83d5-45d2-bc09-81f4917ae152",
    "CreationTimestamp": "2019-06-18T17:03:52.392Z",
    "Id": "70e49321-83d5-45d2-bc09-81f4917ae152",
    "LastUpdatedTimestamp": "2019-06-18T17:03:52.392Z",
    "LatestVersion": "88ae8699-12ac-4663-ba3f-4d7f0519140b",
    "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/definition/subscriptions/70e49321-83d5-45d2-bc09-81f4917ae152/
versions/88ae8699-12ac-4663-ba3f-4d7f0519140b",
    "tags": {}
}
```

- Per i dettagli sull'API, consulta GetSubscriptionDefinition AWS CLICommand Reference.

`get-thing-runtime-configuration`

Il seguente esempio di codice mostra come utilizzare`get-thing-runtime-configuration`.

AWS CLI

Come recuperare la configurazione di runtime di un core Greengrass

L'esempio `get-thing-runtime-configuration` seguente recupera la configurazione di runtime di un core Greengrass. Prima di poter recuperare la configurazione di runtime, è

necessario utilizzare il comando `update-thing-runtime-configuration` per creare una configurazione di runtime per il core.

```
aws greengrass get-thing-runtime-configuration \
    --thing-name SampleGreengrassCore
```

Output:

```
{
    "RuntimeConfiguration": {
        "TelemetryConfiguration": {
            "ConfigurationSyncStatus": "OutOfSync",
            "Telemetry": "On"
        }
    }
}
```

Per ulteriori informazioni, consulta Configurazione delle impostazioni di telemetria nella Guida per gli sviluppatori di AWS IoT Greengrass.

- Per i dettagli sull'API, consulta GetThingRuntimeConfiguration AWS CLICommand Reference.

`list-bulk-deployment-detailed-reports`

Il seguente esempio di codice mostra come utilizzare`list-bulk-deployment-detailed-reports`.

AWS CLI

Come elencare le informazioni sulle singole implementazioni in un'implementazione in blocco

L'esempio `list-bulk-deployment-detailed-reports` seguente visualizza le informazioni sulle singole implementazioni in un'operazione di implementazione in blocco, incluso lo stato.

```
aws greengrass list-bulk-deployment-detailed-reports \
    --bulk-deployment-id 42ce9c42-489b-4ed4-b905-8996aa50ef9d
```

Output:

```
{
    "Deployments": [
```

```
        {
            "DeploymentType": "NewDeployment",
            "DeploymentStatus": "Success",
            "DeploymentId": "123456789012:a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
            "DeploymentArn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/groups/a1b2c3d4-5678-90ab-cdef-EXAMPLE33333/
deployments/123456789012:123456789012:a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
            "GroupArn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/groups/a1b2c3d4-5678-90ab-cdef-EXAMPLE33333/
versions/123456789012:a1b2c3d4-5678-90ab-cdef-EXAMPLE44444",
            "CreatedAt": "2020-01-21T21:34:16.501Z"
        },
        {
            "DeploymentType": "NewDeployment",
            "DeploymentStatus": "InProgress",
            "DeploymentId": "123456789012:a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
            "DeploymentArn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/groups/a1b2c3d4-5678-90ab-cdef-EXAMPLE55555/
deployments/123456789012:123456789012:a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
            "GroupArn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
groups/a1b2c3d4-5678-90ab-cdef-EXAMPLE55555/versions/a1b2c3d4-5678-90ab-cdef-
EXAMPLE66666",
            "CreatedAt": "2020-01-21T21:34:16.486Z"
        },
        ...
    ]
}
```

Per ulteriori informazioni, consulta Creare implementazioni in blocco per i gruppi nella Guida per gli sviluppatori di AWS IoT Greengrass.

- Per i dettagli sull'API, consulta ListBulkDeploymentDetailedReports AWS CLICommand Reference.

`list-bulk-deployments`

Il seguente esempio di codice mostra come utilizzare`list-bulk-deployments`.

AWS CLI

Come elencare le implementazioni in blocco

L'esempio `list-bulk-deployments` seguente elenca tutte le implementazioni in blocco.

```
aws greengrass list-bulk-deployments
```

Output:

```
{
    "BulkDeployments": [
        {
            "BulkDeploymentArn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/bulk/deployments/870fb41b-6288-4e0c-bc76-a7ba4b4d3267",
            "BulkDeploymentId": "870fb41b-6288-4e0c-bc76-a7ba4b4d3267",
            "CreatedAt": "2019-06-25T16:11:33.265Z"
        }
    ]
}
```

Per ulteriori informazioni, consulta Creare implementazioni in blocco per i gruppi nella Guida per gli sviluppatori di AWS IoT Greengrass.

- Per i dettagli sull'API, consulta ListBulkDeployments AWS CLICommand Reference.

`list-connector-definition-versions`

Il seguente esempio di codice mostra come utilizzare`list-connector-definition-versions`.

AWS CLI

Come elencare le versioni disponibili per una definizione di connettore

L'esempio `list-connector-definition-versions` seguente elenca le versioni disponibili per la definizione di connettore specificata. Utilizza il comando `list-connector-definitions` per ottenere l'ID della definizione di connettore.

```
aws greengrass list-connector-definition-versions \
    --connector-definition-id "b5c4ebfd-f672-49a3-83cd-31c7216a7bb8"
```

Output:

```
{
    "Versions": [
        {
```

```
            "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
definition/connectors/b5c4ebfd-f672-49a3-83cd-31c7216a7bb8/versions/63c57963-
c7c2-4a26-a7e2-7bf478ea2623",
            "CreationTimestamp": "2019-06-19T19:30:01.300Z",
            "Id": "b5c4ebfd-f672-49a3-83cd-31c7216a7bb8",
            "Version": "63c57963-c7c2-4a26-a7e2-7bf478ea2623"
        }
    ]
}
```

Per ulteriori informazioni, consulta Integrazione con servizi e protocolli tramite i connettori Greengrass nella Guida per gli sviluppatori di AWS IoT Greengrass.

- Per i dettagli sull'API, consulta ListConnectorDefinitionVersions AWS CLICommand Reference.

`list-connector-definitions`

Il seguente esempio di codice mostra come utilizzare`list-connector-definitions`.

AWS CLI

Come elencare i connettori Greengrass definiti

L'`list-connector-definitions`esempio seguente elenca tutti i connettori Greengrass definiti per il tuo AWS account.

```
aws greengrass list-connector-definitions
```

Output:

```
{
    "Definitions": [
        {
            "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
definition/connectors/b5c4ebfd-f672-49a3-83cd-31c7216a7bb8",
            "CreationTimestamp": "2019-06-19T19:30:01.300Z",
            "Id": "b5c4ebfd-f672-49a3-83cd-31c7216a7bb8",
            "LastUpdatedTimestamp": "2019-06-19T19:30:01.300Z",
            "LatestVersion": "63c57963-c7c2-4a26-a7e2-7bf478ea2623",
            "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/definition/connectors/b5c4ebfd-f672-49a3-83cd-31c7216a7bb8/
versions/63c57963-c7c2-4a26-a7e2-7bf478ea2623",
```

```
            "Name": "MySNSConnector"
        }
    ]
}
```

Per ulteriori informazioni, consulta Integrazione con servizi e protocolli tramite i connettori Greengrass nella Guida per gli sviluppatori di AWS IoT Greengrass.

- Per i dettagli sull'API, consulta ListConnectorDefinitions AWS CLICommand Reference.

`list-core-definition-versions`

Il seguente esempio di codice mostra come utilizzare`list-core-definition-versions`.

AWS CLI

Come elencare le versioni di una definizione di core Greengrass

L'esempio `list-core-definitions` seguente elenca tutte le versioni della definizione core Greengrass specificata. Puoi utilizzare il comando `list-core-definitions` per ottenere l'ID della versione.

```
aws greengrass list-core-definition-versions \
    --core-definition-id "eaf280cb-138c-4d15-af36-6f681a1348f7"
```

Output:

```
{
    "Versions": [
        {
            "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
definition/cores/eaf280cb-138c-4d15-af36-6f681a1348f7/versions/467c36e4-c5da-440c-
a97b-084e62593b4c",
            "CreationTimestamp": "2019-06-18T16:14:17.709Z",
            "Id": "eaf280cb-138c-4d15-af36-6f681a1348f7",
            "Version": "467c36e4-c5da-440c-a97b-084e62593b4c"
        }
    ]
}
```

- Per i dettagli sull'API, consulta ListCoreDefinitionVersions AWS CLICommand Reference.

`list-core-definitions`

Il seguente esempio di codice mostra come utilizzare`list-core-definitions`.

AWS CLI

Come elencare le definizioni di core Greengrass

L'`list-core-definitions`esempio seguente elenca tutte le definizioni principali di Greengrass per il tuo AWS account.

```
aws greengrass list-core-definitions
```

Output:

```
{
    "Definitions": [
        {
            "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
definition/cores/0507843c-c1ef-4f06-b051-817030df7e7d",
            "CreationTimestamp": "2018-10-17T04:30:32.786Z",
            "Id": "0507843c-c1ef-4f06-b051-817030df7e7d",
            "LastUpdatedTimestamp": "2018-10-17T04:30:32.786Z",
            "LatestVersion": "bcdf9e86-3793-491e-93af-3cdfbf4e22b7",
            "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/definition/cores/0507843c-c1ef-4f06-b051-817030df7e7d/versions/
bcdf9e86-3793-491e-93af-3cdfbf4e22b7"
        },
        {
            "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
definition/cores/31c22500-3509-4271-bafd-cf0655cda438",
            "CreationTimestamp": "2019-06-18T16:24:16.064Z",
            "Id": "31c22500-3509-4271-bafd-cf0655cda438",
            "LastUpdatedTimestamp": "2019-06-18T16:24:16.064Z",
            "LatestVersion": "2f350395-6d09-4c8a-8336-9ae5b57ace84",
            "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/definition/cores/31c22500-3509-4271-bafd-cf0655cda438/
versions/2f350395-6d09-4c8a-8336-9ae5b57ace84"
        },
        {
            "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
definition/cores/c906ed39-a1e3-4822-a981-7b9bd57b4b46",
            "CreationTimestamp": "2019-06-18T16:21:21.351Z",
```

```
            "Id": "c906ed39-a1e3-4822-a981-7b9bd57b4b46",
            "LastUpdatedTimestamp": "2019-06-18T16:21:21.351Z",
            "LatestVersion": "42aeeac3-fd9d-4312-a8fd-ffa9404a20e0",
            "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/definition/cores/c906ed39-a1e3-4822-a981-7b9bd57b4b46/versions/42aeeac3-
fd9d-4312-a8fd-ffa9404a20e0"
        },
        {
            "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
definition/cores/eaf280cb-138c-4d15-af36-6f681a1348f7",
            "CreationTimestamp": "2019-06-18T16:14:17.709Z",
            "Id": "eaf280cb-138c-4d15-af36-6f681a1348f7",
            "LastUpdatedTimestamp": "2019-06-18T16:14:17.709Z",
            "LatestVersion": "467c36e4-c5da-440c-a97b-084e62593b4c",
            "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/definition/cores/eaf280cb-138c-4d15-af36-6f681a1348f7/versions/467c36e4-
c5da-440c-a97b-084e62593b4c"
        }
    ]
}
```

- Per i dettagli sull'API, consulta ListCoreDefinitions AWS CLICommand Reference.

`list-deployments`

Il seguente esempio di codice mostra come utilizzare`list-deployments`.

AWS CLI

Come elencare le implementazioni per un gruppo Greengrass

L'esempio `list-deployments` seguente elenca le implementazioni per il gruppo Greengrass specificato. Puoi utilizzare il comando `list-groups` sul dispositivo per cercare l'ID di gruppo.

```
aws greengrass list-deployments \
    --group-id "1013db12-8b58-45ff-acc7-704248f66731"
```

Output:

```
{
    "Deployments": [
        {
```

```
            "CreatedAt": "2019-06-18T17:04:32.702Z",
            "DeploymentId": "1065b8a0-812b-4f21-9d5d-e89b232a530f",
            "DeploymentType": "NewDeployment",
            "GroupArn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
groups/1013db12-8b58-45ff-acc7-704248f66731/versions/115136b3-cfd7-4462-
b77f-8741a4b00e5e"
        }
    ]
}
```

- Per i dettagli sull'API, consulta ListDeployments AWS CLICommand Reference.

`list-device-definition-versions`

Il seguente esempio di codice mostra come utilizzare`list-device-definition-versions`.

AWS CLI

Come elencare le versioni di una definizione di dispositivo

L'esempio `list-device-definition-versions` seguente visualizza le versioni della definizione di dispositivo associate alla definizione di dispositivo specificata.

```
aws greengrass list-device-definition-versions \
    --device-definition-id "f9ba083d-5ad4-4534-9f86-026a45df1ccd"
```

Output:

```
{
    "Versions": [
        {
            "Version": "83c13984-6fed-447e-84d5-5b8aa45d5f71",
            "CreationTimestamp": "2019-09-11T00:15:09.838Z",
            "Id": "f9ba083d-5ad4-4534-9f86-026a45df1ccd",
            "Arn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/definition/devices/f9ba083d-5ad4-4534-9f86-026a45df1ccd/
versions/83c13984-6fed-447e-84d5-5b8aa45d5f71"
        },
        {
            "Version": "3b5cc510-58c1-44b5-9d98-4ad858ffa795",
            "CreationTimestamp": "2019-09-11T00:11:06.197Z",
            "Id": "f9ba083d-5ad4-4534-9f86-026a45df1ccd",
```

```
            "Arn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/definition/devices/f9ba083d-5ad4-4534-9f86-026a45df1ccd/
versions/3b5cc510-58c1-44b5-9d98-4ad858ffa795"
        }
    ]
}
```

- Per i dettagli sull'API, consulta ListDeviceDefinitionVersions AWS CLICommand Reference.

`list-device-definitions`

Il seguente esempio di codice mostra come utilizzare`list-device-definitions`.

AWS CLI

Come elencare le definizioni di dispositivo

L'`list-device-definitions`esempio seguente mostra i dettagli sulle definizioni dei dispositivi nell' AWS account nella AWS regione specificata.

```
aws greengrass list-device-definitions \
    --region us-west-2
```

Output:

```
{
    "Definitions": [
        {
            "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/definition/devices/50f3274c-3f0a-4f57-b114-6f46085281ab/versions/
c777b0f5-1059-449b-beaa-f003ebc56c34",
            "LastUpdatedTimestamp": "2019-06-14T15:42:09.059Z",
            "LatestVersion": "c777b0f5-1059-449b-beaa-f003ebc56c34",
            "CreationTimestamp": "2019-06-14T15:42:09.059Z",
            "Id": "50f3274c-3f0a-4f57-b114-6f46085281ab",
            "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
definition/devices/50f3274c-3f0a-4f57-b114-6f46085281ab"
        },
        {
            "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/definition/devices/e01951c9-6134-479a-969a-1a15cac11c40/
versions/514d57aa-4ee6-401c-9fac-938a9f7a51e5",
```

```
            "Name": "TestDeviceDefinition",
            "LastUpdatedTimestamp": "2019-04-16T23:17:43.245Z",
            "LatestVersion": "514d57aa-4ee6-401c-9fac-938a9f7a51e5",
            "CreationTimestamp": "2019-04-16T23:17:43.245Z",
            "Id": "e01951c9-6134-479a-969a-1a15cac11c40",
            "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
definition/devices/e01951c9-6134-479a-969a-1a15cac11c40"
        },
        {
            "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/definition/devices/f9ba083d-5ad4-4534-9f86-026a45df1ccd/
versions/83c13984-6fed-447e-84d5-5b8aa45d5f71",
            "Name": "TemperatureSensors",
            "LastUpdatedTimestamp": "2019-09-10T00:19:03.698Z",
            "LatestVersion": "83c13984-6fed-447e-84d5-5b8aa45d5f71",
            "CreationTimestamp": "2019-09-11T00:11:06.197Z",
            "Id": "f9ba083d-5ad4-4534-9f86-026a45df1ccd",
            "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
definition/devices/f9ba083d-5ad4-4534-9f86-026a45df1ccd"
        }
    ]
}
```

- Per i dettagli sull'API, consulta ListDeviceDefinitions AWS CLICommand Reference.

`list-function-definition-versions`

Il seguente esempio di codice mostra come utilizzare`list-function-definition-versions`.

AWS CLI

Come elencare le versioni di una funzione Lambda

L'esempio `list-function-definition-versions` seguente visualizza l'elenco di versioni per la funzione Lambda specificata. Per ottenere l'ID, puoi utilizzare il comando `list-function-definitions`.

```
aws greengrass list-function-definition-versions \
    --function-definition-id "063f5d1a-1dd1-40b4-9b51-56f8993d0f85"
```

Output:

```
{
    "Versions": [
        {
            "Arn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/definition/functions/063f5d1a-1dd1-40b4-9b51-56f8993d0f85/
versions/9748fda7-1589-4fcc-ac94-f5559e88678b",
            "CreationTimestamp": "2019-06-18T17:04:30.776Z",
            "Id": "063f5d1a-1dd1-40b4-9b51-56f8993d0f85",
            "Version": "9748fda7-1589-4fcc-ac94-f5559e88678b"
        },
        {
            "Arn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/definition/functions/063f5d1a-1dd1-40b4-9b51-56f8993d0f85/
versions/9b08df77-26f2-4c29-93d2-769715edcfec",
            "CreationTimestamp": "2019-06-18T17:02:44.087Z",
            "Id": "063f5d1a-1dd1-40b4-9b51-56f8993d0f85",
            "Version": "9b08df77-26f2-4c29-93d2-769715edcfec"
        },
        {
            "Arn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/definition/functions/063f5d1a-1dd1-40b4-9b51-56f8993d0f85/
versions/4236239f-94f7-4b90-a2f8-2a24c829d21e",
            "CreationTimestamp": "2019-06-18T17:01:42.284Z",
            "Id": "063f5d1a-1dd1-40b4-9b51-56f8993d0f85",
            "Version": "4236239f-94f7-4b90-a2f8-2a24c829d21e"
        },
        {
            "Arn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/definition/functions/063f5d1a-1dd1-40b4-9b51-56f8993d0f85/
versions/343408bb-549a-4fbe-b043-853643179a39",
            "CreationTimestamp": "2019-06-18T16:21:21.431Z",
            "Id": "063f5d1a-1dd1-40b4-9b51-56f8993d0f85",
            "Version": "343408bb-549a-4fbe-b043-853643179a39"
        }
    ]
}
```

- Per i dettagli sull'API, consulta ListFunctionDefinitionVersions AWS CLICommand Reference.

`list-function-definitions`

Il seguente esempio di codice mostra come utilizzare`list-function-definitions`.

AWS CLI

Come elencare le funzioni Lambda

L'`list-function-definitions`esempio seguente elenca tutte le funzioni Lambda definite per il tuo AWS account.

```
aws greengrass list-function-definitions
```

Output:

```
{
    "Definitions": [
        {
            "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
definition/functions/017970a5-8952-46dd-b1c1-020b3ae8e960",
            "CreationTimestamp": "2018-10-17T04:30:32.884Z",
            "Id": "017970a5-8952-46dd-b1c1-020b3ae8e960",
            "LastUpdatedTimestamp": "2018-10-17T04:30:32.884Z",
            "LatestVersion": "4380b302-790d-4ed8-92bf-02e88afecb15",
            "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/definition/functions/017970a5-8952-46dd-b1c1-020b3ae8e960/
versions/4380b302-790d-4ed8-92bf-02e88afecb15"
        },
        {
            "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
definition/functions/063f5d1a-1dd1-40b4-9b51-56f8993d0f85",
            "CreationTimestamp": "2019-06-18T16:21:21.431Z",
            "Id": "063f5d1a-1dd1-40b4-9b51-56f8993d0f85",
            "LastUpdatedTimestamp": "2019-06-18T16:21:21.431Z",
            "LatestVersion": "9748fda7-1589-4fcc-ac94-f5559e88678b",
            "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/definition/functions/063f5d1a-1dd1-40b4-9b51-56f8993d0f85/
versions/9748fda7-1589-4fcc-ac94-f5559e88678b"
        },
        {
            "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
definition/functions/6598e653-a262-440c-9967-e2697f64da7b",
            "CreationTimestamp": "2019-06-18T16:24:16.123Z",
            "Id": "6598e653-a262-440c-9967-e2697f64da7b",
            "LastUpdatedTimestamp": "2019-06-18T16:24:16.123Z",
            "LatestVersion": "38bc6ccd-98a2-4ce7-997e-16c84748fae4",
```

```
            "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/definition/functions/6598e653-a262-440c-9967-e2697f64da7b/
versions/38bc6ccd-98a2-4ce7-997e-16c84748fae4"
        },
        {
            "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
definition/functions/c668df84-fad2-491b-95f4-655d2cad7885",
            "CreationTimestamp": "2019-06-18T16:14:17.784Z",
            "Id": "c668df84-fad2-491b-95f4-655d2cad7885",
            "LastUpdatedTimestamp": "2019-06-18T16:14:17.784Z",
            "LatestVersion": "37dd68c4-a64f-40ba-aa13-71fecc3ebded",
            "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/definition/functions/c668df84-fad2-491b-95f4-655d2cad7885/
versions/37dd68c4-a64f-40ba-aa13-71fecc3ebded"
        }
    ]
}
```

- Per i dettagli sull'API, consulta ListFunctionDefinitions AWS CLICommand Reference.

`list-group-certificate-authorities`

Il seguente esempio di codice mostra come utilizzare`list-group-certificate-authorities`.

AWS CLI

Per elencare la corrente CAs di un gruppo

L'`list-group-certificate-authorities`esempio seguente elenca le autorità di certificazione correnti (CAs) per il gruppo Greengrass specificato.

```
aws greengrass list-group-certificate-authorities \
    --group-id "1013db12-8b58-45ff-acc7-704248f66731"
```

Output:

```
{
    "GroupCertificateAuthorities": [
        {
            "GroupCertificateAuthorityArn": "arn:aws:greengrass:us-
west-2:123456789012:/greengrass/groups/1013db12-8b58-45ff-acc7-704248f66731/
```

```
certificateauthorities/
f0430e1736ea8ed30cc5d5de9af67a7e3586bad9ae4d89c2a44163f65fdd8cf6",
            "GroupCertificateAuthorityId":
 "f0430e1736ea8ed30cc5d5de9af67a7e3586bad9ae4d89c2a44163f65fdd8cf6"
        }
    ]
}
```

- Per i dettagli sull'API, vedere ListGroupCertificateAuthoritiesin AWS CLI Command Reference.

`list-group-versions`

Il seguente esempio di codice mostra come utilizzare`list-group-versions`.

AWS CLI

Come elencare le versioni di un gruppo Greengrass

L'esempio `list-group-versions` seguente elenca tutte le versioni del gruppo Greengrass specificato.

```
aws greengrass list-group-versions \
    --group-id "1013db12-8b58-45ff-acc7-704248f66731"
```

Output:

```
{
    "Versions": [
        {
            "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
groups/1013db12-8b58-45ff-acc7-704248f66731/versions/115136b3-cfd7-4462-
b77f-8741a4b00e5e",
            "CreationTimestamp": "2019-06-18T17:04:30.915Z",
            "Id": "1013db12-8b58-45ff-acc7-704248f66731",
            "Version": "115136b3-cfd7-4462-b77f-8741a4b00e5e"
        },
        {
            "Arn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/groups/1013db12-8b58-45ff-acc7-704248f66731/versions/4340669d-
d14d-44e3-920c-46c928750750",
            "CreationTimestamp": "2019-06-18T17:03:52.663Z",
            "Id": "1013db12-8b58-45ff-acc7-704248f66731",
```

```
            "Version": "4340669d-d14d-44e3-920c-46c928750750"
        },
        {
            "Arn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/groups/1013db12-8b58-45ff-acc7-704248f66731/
versions/1b06e099-2d5b-4f10-91b9-78c4e060f5da",
            "CreationTimestamp": "2019-06-18T17:02:44.189Z",
            "Id": "1013db12-8b58-45ff-acc7-704248f66731",
            "Version": "1b06e099-2d5b-4f10-91b9-78c4e060f5da"
        },
        {
            "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
groups/1013db12-8b58-45ff-acc7-704248f66731/versions/2d3f27f1-3b43-4554-
ab7a-73ec30477efe",
            "CreationTimestamp": "2019-06-18T17:01:42.401Z",
            "Id": "1013db12-8b58-45ff-acc7-704248f66731",
            "Version": "2d3f27f1-3b43-4554-ab7a-73ec30477efe"
        },
        {
            "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
groups/1013db12-8b58-45ff-acc7-704248f66731/versions/d20f7ae9-3444-4c1c-b025-
e2ede23cdd31",
            "CreationTimestamp": "2019-06-18T16:21:21.457Z",
            "Id": "1013db12-8b58-45ff-acc7-704248f66731",
            "Version": "d20f7ae9-3444-4c1c-b025-e2ede23cdd31"
        }
    ]
}
```

- Per i dettagli sull'API, consulta ListGroupVersions AWS CLICommand Reference.

`list-groups`

Il seguente esempio di codice mostra come utilizzare`list-groups`.

AWS CLI

Come elencare i gruppi Greengrass

L'`list-groups`esempio seguente elenca tutti i gruppi Greengrass definiti nel tuo AWS account.

```
aws greengrass list-groups
```

Output:


```
{
    "Groups": [
        {
            "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
groups/1013db12-8b58-45ff-acc7-704248f66731",
            "CreationTimestamp": "2019-06-18T16:21:21.457Z",
            "Id": "1013db12-8b58-45ff-acc7-704248f66731",
            "LastUpdatedTimestamp": "2019-06-18T16:21:21.457Z",
            "LatestVersion": "115136b3-cfd7-4462-b77f-8741a4b00e5e",
            "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/groups/1013db12-8b58-45ff-acc7-704248f66731/versions/115136b3-cfd7-4462-
b77f-8741a4b00e5e",
            "Name": "GGGroup4Pi3"
        },
        {
            "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
groups/1402daf9-71cf-4cfe-8be0-d5e80526d0d8",
            "CreationTimestamp": "2018-10-31T21:52:46.603Z",
            "Id": "1402daf9-71cf-4cfe-8be0-d5e80526d0d8",
            "LastUpdatedTimestamp": "2018-10-31T21:52:46.603Z",
            "LatestVersion": "749af901-60ab-456f-a096-91b12d983c29",
            "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/groups/1402daf9-71cf-4cfe-8be0-d5e80526d0d8/versions/749af901-60ab-456f-
a096-91b12d983c29",
            "Name": "MyTestGroup"
        },
        {
            "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
groups/504b5c8d-bbed-4635-aff1-48ec5b586db5",
            "CreationTimestamp": "2018-12-31T21:39:36.771Z",
            "Id": "504b5c8d-bbed-4635-aff1-48ec5b586db5",
            "LastUpdatedTimestamp": "2018-12-31T21:39:36.771Z",
            "LatestVersion": "46911e8e-f9bc-4898-8b63-59c7653636ec",
            "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/groups/504b5c8d-bbed-4635-aff1-48ec5b586db5/versions/46911e8e-
f9bc-4898-8b63-59c7653636ec",
            "Name": "smp-ggrass-group"
        }
    ]
}
```

- Per i dettagli sull'API, consulta ListGroups AWS CLICommand Reference.

`list-logger-definition-versions`

Il seguente esempio di codice mostra come utilizzare`list-logger-definition-versions`.

AWS CLI

Come ottenere un elenco di versioni di una definizione di logger

L'esempio `list-logger-definition-versions` seguente ottiene un elenco di tutte le versioni della definizione di logger specificata.

```
aws greengrass list-logger-definition-versions \
    --logger-definition-id "49eeeb66-f1d3-4e34-86e3-3617262abf23"
```

Output:

```
{
    "Versions": [
        {
            "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
definition/loggers/49eeeb66-f1d3-4e34-86e3-3617262abf23/versions/5e3f6f64-
a565-491e-8de0-3c0d8e0f2073",
            "CreationTimestamp": "2019-05-08T16:10:13.866Z",
            "Id": "49eeeb66-f1d3-4e34-86e3-3617262abf23",
            "Version": "5e3f6f64-a565-491e-8de0-3c0d8e0f2073"
        },
        {
            "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
definition/loggers/49eeeb66-f1d3-4e34-86e3-3617262abf23/versions/3ec6d3af-eb85-48f9-
a16d-1c795fe696d7",
            "CreationTimestamp": "2019-05-08T16:10:13.809Z",
            "Id": "49eeeb66-f1d3-4e34-86e3-3617262abf23",
            "Version": "3ec6d3af-eb85-48f9-a16d-1c795fe696d7"
        }
    ]
}
```

- Per i dettagli sull'API, consulta ListLoggerDefinitionVersions AWS CLICommand Reference.

`list-logger-definitions`

Il seguente esempio di codice mostra come utilizzare`list-logger-definitions`.

AWS CLI

Come ottenere un elenco di definizioni di logger

L'`list-logger-definitions`esempio seguente elenca tutte le definizioni dei logger per il tuo AWS account.

```
aws greengrass list-logger-definitions
```

Output:

```
{
    "Definitions": [
        {
            "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
definition/loggers/49eeeb66-f1d3-4e34-86e3-3617262abf23",
            "CreationTimestamp": "2019-05-08T16:10:13.809Z",
            "Id": "49eeeb66-f1d3-4e34-86e3-3617262abf23",
            "LastUpdatedTimestamp": "2019-05-08T16:10:13.809Z",
            "LatestVersion": "5e3f6f64-a565-491e-8de0-3c0d8e0f2073",
            "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/definition/loggers/49eeeb66-f1d3-4e34-86e3-3617262abf23/
versions/5e3f6f64-a565-491e-8de0-3c0d8e0f2073"
        }
    ]
}
```

- Per i dettagli sull'API, consulta ListLoggerDefinitions AWS CLICommand Reference.

`list-resource-definition-versions`

Il seguente esempio di codice mostra come utilizzare`list-resource-definition-versions`.

AWS CLI

Come elencare le versioni di una definizione di risorsa

L'esempio `list-resource-definition-versions` seguente elenca le versioni per la risorsa Greengrass specificata.

```
aws greengrass list-resource-definition-versions \
    --resource-definition-id "ad8c101d-8109-4b0e-b97d-9cc5802ab658"
```

Output:

```
{
    "Versions": [
        {
            "Arn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/definition/resources/ad8c101d-8109-4b0e-b97d-9cc5802ab658/
versions/26e8829a-491a-464d-9c87-664bf6f6f2be",
            "CreationTimestamp": "2019-06-19T16:40:59.392Z",
            "Id": "ad8c101d-8109-4b0e-b97d-9cc5802ab658",
            "Version": "26e8829a-491a-464d-9c87-664bf6f6f2be"
        },
        {
            "Arn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/definition/resources/ad8c101d-8109-4b0e-b97d-9cc5802ab658/
versions/432d92f6-12de-4ec9-a704-619a942a62aa",
            "CreationTimestamp": "2019-06-19T16:40:59.261Z",
            "Id": "ad8c101d-8109-4b0e-b97d-9cc5802ab658",
            "Version": "432d92f6-12de-4ec9-a704-619a942a62aa"
        }
    ]
}
```

- Per i dettagli sull'API, consulta ListResourceDefinitionVersions AWS CLICommand Reference.

`list-resource-definitions`

Il seguente esempio di codice mostra come utilizzare`list-resource-definitions`.

AWS CLI

Come elencare le risorse definite

L'`list-resource-definitions`esempio seguente elenca le risorse definite per l'utilizzo di AWS IoT Greengrass.

```
aws greengrass list-resource-definitions
```

Output:

```
{
    "Definitions": [
        {
            "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
definition/resources/ad8c101d-8109-4b0e-b97d-9cc5802ab658",
            "CreationTimestamp": "2019-06-19T16:40:59.261Z",
            "Id": "ad8c101d-8109-4b0e-b97d-9cc5802ab658",
            "LastUpdatedTimestamp": "2019-06-19T16:40:59.261Z",
            "LatestVersion": "26e8829a-491a-464d-9c87-664bf6f6f2be",
            "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/definition/resources/ad8c101d-8109-4b0e-b97d-9cc5802ab658/
versions/26e8829a-491a-464d-9c87-664bf6f6f2be"
        },
        {
            "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
definition/resources/c8bb9ebc-c3fd-40a4-9c6a-568d75569d38",
            "CreationTimestamp": "2019-06-19T21:51:28.212Z",
            "Id": "c8bb9ebc-c3fd-40a4-9c6a-568d75569d38",
            "LastUpdatedTimestamp": "2019-06-19T21:51:28.212Z",
            "LatestVersion": "a5f94d0b-f6bc-40f4-bb78-7a1c5fe13ba1",
            "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/definition/resources/c8bb9ebc-c3fd-40a4-9c6a-568d75569d38/versions/
a5f94d0b-f6bc-40f4-bb78-7a1c5fe13ba1",
            "Name": "MyGreengrassResources"
        }
    ]
}
```

- Per i dettagli sull'API, consulta ListResourceDefinitions AWS CLICommand Reference.

`list-subscription-definition-versions`

Il seguente esempio di codice mostra come utilizzare`list-subscription-definition-versions`.

AWS CLI

Come elencare le versioni di una definizione di abbonamento

L'esempio `list-subscription-definition-versions` seguente elenca tutte le versioni dell'abbonamento specificate. Puoi utilizzare il comando `list-subscription-definitions` per cercare l'ID dell'abbonamento.

```
aws greengrass list-subscription-definition-versions \
    --subscription-definition-id "70e49321-83d5-45d2-bc09-81f4917ae152"
```

Output:

```
{
    "Versions": [
        {
            "Arn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/definition/subscriptions/70e49321-83d5-45d2-bc09-81f4917ae152/
versions/88ae8699-12ac-4663-ba3f-4d7f0519140b",
            "CreationTimestamp": "2019-06-18T17:03:52.499Z",
            "Id": "70e49321-83d5-45d2-bc09-81f4917ae152",
            "Version": "88ae8699-12ac-4663-ba3f-4d7f0519140b"
        },
        {
            "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
definition/subscriptions/70e49321-83d5-45d2-bc09-81f4917ae152/versions/7e320ba3-
c369-4069-a2f0-90acb7f219d6",
            "CreationTimestamp": "2019-06-18T17:03:52.392Z",
            "Id": "70e49321-83d5-45d2-bc09-81f4917ae152",
            "Version": "7e320ba3-c369-4069-a2f0-90acb7f219d6"
        }
    ]
}
```

- Per i dettagli sull'API, consulta ListSubscriptionDefinitionVersions AWS CLICommand Reference.

`list-subscription-definitions`

Il seguente esempio di codice mostra come utilizzare`list-subscription-definitions`.

AWS CLI

Come ottenere un elenco delle definizioni di abbonamenti

L'`list-subscription-definitions`esempio seguente elenca tutti gli abbonamenti AWS IoT Greengrass definiti nell'account. AWS

```
aws greengrass list-subscription-definitions
```

Output:

```
{
    "Definitions": [
        {
            "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
definition/subscriptions/70e49321-83d5-45d2-bc09-81f4917ae152",
            "CreationTimestamp": "2019-06-18T17:03:52.392Z",
            "Id": "70e49321-83d5-45d2-bc09-81f4917ae152",
            "LastUpdatedTimestamp": "2019-06-18T17:03:52.392Z",
            "LatestVersion": "88ae8699-12ac-4663-ba3f-4d7f0519140b",
            "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/definition/subscriptions/70e49321-83d5-45d2-bc09-81f4917ae152/
versions/88ae8699-12ac-4663-ba3f-4d7f0519140b"
        },
        {
            "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
definition/subscriptions/cd6f1c37-d9a4-4e90-be94-01a7404f5967",
            "CreationTimestamp": "2018-10-18T15:45:34.024Z",
            "Id": "cd6f1c37-d9a4-4e90-be94-01a7404f5967",
            "LastUpdatedTimestamp": "2018-10-18T15:45:34.024Z",
            "LatestVersion": "d1cf8fac-284f-4f6a-98fe-a2d36d089373",
            "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/definition/subscriptions/cd6f1c37-d9a4-4e90-be94-01a7404f5967/versions/
d1cf8fac-284f-4f6a-98fe-a2d36d089373"
        },
        {
            "Arn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
definition/subscriptions/fa81bc84-3f59-4377-a84b-5d0134da359b",
            "CreationTimestamp": "2018-10-22T17:09:31.429Z",
            "Id": "fa81bc84-3f59-4377-a84b-5d0134da359b",
            "LastUpdatedTimestamp": "2018-10-22T17:09:31.429Z",
            "LatestVersion": "086d1b08-b25a-477c-a16f-6f9b3a9c295a",
            "LatestVersionArn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/definition/subscriptions/fa81bc84-3f59-4377-a84b-5d0134da359b/
versions/086d1b08-b25a-477c-a16f-6f9b3a9c295a"
        }
    ]
}
```

- Per i dettagli sull'API, consulta AWS CLI Command ListSubscriptionDefinitionsReference.

`list-tags-for-resource`

Il seguente esempio di codice mostra come utilizzare`list-tags-for-resource`.

AWS CLI

Come elencare i tag collegati a una risorsa

L'esempio `list-tags-for-resource` seguente elenca i tag e i relativi valori associati alla risorsa specificata.

```
aws greengrass list-tags-for-resource \
    --resource-arn "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
definition/resources/ad8c101d-8109-4b0e-b97d-9cc5802ab658"
```

Output:

```
{
    "tags": {
        "ResourceSubType": "USB",
        "ResourceType": "Device"
    }
}
```

Per ulteriori informazioni, consulta Tagging delle risorse Greengrass nella Guida per gli sviluppatori di AWS IoT Greengrass.

- Per i dettagli sull'API, consulta ListTagsForResource AWS CLICommand Reference.

`reset-deployments`

Il seguente esempio di codice mostra come utilizzare`reset-deployments`.

AWS CLI

Come eliminare le informazioni sull'implementazione per un gruppo Greengrass

L'esempio `reset-deployments` seguente ripulisce le informazioni di implementazione per il gruppo Greengrass specificato. Quando si aggiunge `--force option`, le informazioni sull'implementazione vengono reimpostate senza attendere la risposta del dispositivo core.

```
aws greengrass reset-deployments \
```

```
    --group-id "1402daf9-71cf-4cfe-8be0-d5e80526d0d8" \
    --force
```

Output:

```
{
    "DeploymentArn": "arn:aws:greengrass:us-west-2:123456789012:/
greengrass/groups/1402daf9-71cf-4cfe-8be0-d5e80526d0d8/
deployments/7dd4e356-9882-46a3-9e28-6d21900c011a",
    "DeploymentId": "7dd4e356-9882-46a3-9e28-6d21900c011a"
}
```

Per ulteriori informazioni, consulta Reimpostazione delle implementazioni nella Guida per gli sviluppatori di AWS IoT Greengrass.

- Per i dettagli sull'API, consulta ResetDeployments AWS CLICommand Reference.

`start-bulk-deployment`

Il seguente esempio di codice mostra come utilizzare`start-bulk-deployment`.

AWS CLI

Come avviare un'operazione di implementazione in blocco

L'esempio `start-bulk-deployment` seguente avvia un'operazione di implementazione in blocco, utilizzando un file archiviato in un bucket S3 per specificare i gruppi da implementare.

```
aws greengrass start-bulk-deployment \
    --cli-input-json "{\"InputFileUri\":\"https://gg-group-deployment1.s3-us-
west-2.amazonaws.com/MyBulkDeploymentInputFile.txt\", \"ExecutionRoleArn\":
\"arn:aws:iam::123456789012:role/ggCreateDeploymentRole\",\"AmznClientToken\":
\"yourAmazonClientToken\"}"
```

Output:

```
{
    "BulkDeploymentArn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
bulk/deployments/870fb41b-6288-4e0c-bc76-a7ba4b4d3267",
    "BulkDeploymentId": "870fb41b-6288-4e0c-bc76-a7ba4b4d3267"
}
```

Per ulteriori informazioni, consulta Creare implementazioni in blocco per i gruppi nella Guida per gli sviluppatori di AWS IoT Greengrass.

- Per i dettagli sull'API, consulta StartBulkDeployment AWS CLICommand Reference.

`stop-bulk-deployment`

Il seguente esempio di codice mostra come utilizzare`stop-bulk-deployment`.

AWS CLI

Come interrompere un'implementazione in blocco

L'esempio `stop-bulk-deployment` seguente interrompe l'implementazione in blocco specificata. Se si tenta di interrompere un'implementazione in blocco che è completa, viene visualizzato un errore: `InvalidInputException: Cannot change state of finished execution.`

```
aws greengrass stop-bulk-deployment \
    --bulk-deployment-id "870fb41b-6288-4e0c-bc76-a7ba4b4d3267"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Creare implementazioni in blocco per i gruppi nella Guida per gli sviluppatori di AWS IoT Greengrass.

- Per i dettagli sull'API, consulta StopBulkDeployment AWS CLICommand Reference.

`tag-resource`

Il seguente esempio di codice mostra come utilizzare`tag-resource`.

AWS CLI

Come applicare tag a una risorsa

L'esempio `tag-resource` seguente applica due tag `ResourceType` e `ResourceSubType` alla risorsa Greengrass specificata. Questa operazione può sia aggiungere nuovi tag e valori sia aggiornare il valore dei tag esistenti. Pr rimuovere un tag, utilizza il comando `untag-resource`.

```
aws greengrass tag-resource \
```

```
    --resource-arn "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
definition/resources/ad8c101d-8109-4b0e-b97d-9cc5802ab658" \
    --tags "ResourceType=Device,ResourceSubType=USB"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Tagging delle risorse Greengrass nella Guida per gli sviluppatori di AWS IoT Greengrass.

- Per i dettagli sull'API, consulta TagResource AWS CLICommand Reference.

`untag-resource`

Il seguente esempio di codice mostra come utilizzare`untag-resource`.

AWS CLI

Come rimuovere un tag e il relativo valore da una risorsa

L'esempio `untag-resource` seguente rimuove il tag con chiave `Category` dal gruppo Greengrass specificato. Se la chiave `Category` non esiste per la risorsa specificata, non viene restituito alcun errore.

```
aws greengrass untag-resource \
    --resource-arn "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
groups/1013db12-8b58-45ff-acc7-704248f66731" \
    --tag-keys "Category"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Tagging delle risorse Greengrass nella Guida per gli sviluppatori di AWS IoT Greengrass.

- Per i dettagli sull'API, consulta UntagResource AWS CLICommand Reference.

`update-connectivity-info`

Il seguente esempio di codice mostra come utilizzare`update-connectivity-info`.

AWS CLI

Come aggiornare le informazioni sulla connettività per il core Greengrass

L'esempio `update-connectivity-info` seguente modifica gli endpoint che i dispositivi possono utilizzare per connettersi al core Greengrass specificato. Le informazioni sulla connettività sono un elenco di indirizzi IP o nomi di dominio, con i numeri di porta corrispondenti e i metadati opzionali definiti dal cliente. Se la rete locale viene modificata, potrebbe essere necessario aggiornare le informazioni sulla connettività.

```
aws greengrass update-connectivity-info \
    --thing-name "MyGroup_Core" \
    --connectivity-info "[{\"Metadata\":\"\",\"PortNumber\":8883,\"HostAddress\":
\"127.0.0.1\",\"Id\":\"localhost_127.0.0.1_0\"},{\"Metadata\":\"\",\"PortNumber
\":8883,\"HostAddress\":\"192.168.1.3\",\"Id\":\"localIP_192.168.1.3\"}]"
```

Output:

```
{
    "Version": "312de337-59af-4cf9-a278-2a23bd39c300"
}
```

- Per i dettagli sull'API, consulta UpdateConnectivityInfo AWS CLICommand Reference.

`update-connector-definition`

Il seguente esempio di codice mostra come utilizzare`update-connector-definition`.

AWS CLI

Come aggiornare il nome della definizione di un connettore

L'esempio `update-connector-definition` seguente aggiorna il nome della definizione di connettore specificata. Se desideri aggiornare i dettagli del connettore, utilizza il comando `create-connector-definition-version` per creare una nuova versione.

```
aws greengrass update-connector-definition \
    --connector-definition-id "55d0052b-0d7d-44d6-b56f-21867215e118" \
    --name "GreengrassConnectors2019"
```

Per ulteriori informazioni, consulta Integrazione con servizi e protocolli tramite i connettori nella Guida per gli sviluppatori di AWS IoT Greengrass.

- Per i dettagli sull'API, consulta UpdateConnectorDefinition AWS CLICommand Reference.

`update-core-definition`

Il seguente esempio di codice mostra come utilizzare`update-core-definition`.

AWS CLI

Come creare una definizione core

L'esempio `update-core-definition` seguente modifica il nome della definizione di core specificata. Puoi aggiornare solo la proprietà `name` di una definizione core.

```
aws greengrass update-core-definition \
    --core-definition-id "582efe12-b05a-409e-9a24-a2ba1bcc4a12" \
    --name "MyCoreDevices"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Configurare AWS IoT Greengrass Core nella IoT AWS Greengrass Developer Guide.

- Per i dettagli sull'API, consulta AWS CLI Command UpdateCoreDefinitionReference.

`update-device-definition`

Il seguente esempio di codice mostra come utilizzare`update-device-definition`.

AWS CLI

Come aggiornare una definizione di dispositivo

L'esempio `update-device-definition` seguente modifica il nome della definizione di dispositivo specificata. Puoi aggiornare solo la proprietà `name` di una definizione di dispositivo.

```
aws greengrass update-device-definition \
    --device-definition-id "f9ba083d-5ad4-4534-9f86-026a45df1ccd" \
    --name "TemperatureSensors"
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta UpdateDeviceDefinition AWS CLICommand Reference.

update-function-definition

Il seguente esempio di codice mostra come utilizzareupdate-function-definition.

AWS CLI

Come aggiornare il nome di una definizione di funzione

L'esempio update-function-definition seguente aggiorna il nome della definizione di funzione specificata. Se desideri aggiornare i dettagli per la funzione, utilizza il comando create-function-definition-version per creare una nuova versione.

```
aws greengrass update-function-definition \
    --function-definition-id "e47952bd-dea9-4e2c-a7e1-37bbe8807f46" \
    --name ObsoleteFunction
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Eseguire funzioni Lambda locali nella Guida per gli sviluppatori di AWS IoT Greengrass.

- Per i dettagli sull'API, consulta UpdateFunctionDefinition AWS CLICommand Reference.

update-group-certificate-configuration

Il seguente esempio di codice mostra come utilizzareupdate-group-certificate-configuration.

AWS CLI

Come aggiornare la scadenza dei certificati di un gruppo

L'esempio update-group-certificate-configuration seguente imposta una scadenza di 10 giorni per i certificati generati per il gruppo specificato.

```
aws greengrass update-group-certificate-configuration \
    --group-id "8eaadd72-ce4b-4f15-892a-0cc4f3a343f1" \
    --certificate-expiry-in-milliseconds 864000000
```

Output:

```
{
    "CertificateExpiryInMilliseconds": 864000000,
    "CertificateAuthorityExpiryInMilliseconds": 2524607999000,
    "GroupId": "8eaadd72-ce4b-4f15-892a-0cc4f3a343f1"
}
```

Per ulteriori informazioni, consulta Sicurezza di AWS IoT Greengrass nella Guida per gli sviluppatori di AWS IoT Greengrass.

- Per i dettagli sull'API, consulta UpdateGroupCertificateConfiguration AWS CLICommand Reference.

`update-group`

Il seguente esempio di codice mostra come utilizzare`update-group`.

AWS CLI

Come aggiornare il nome del gruppo

L'esempio `update-group` seguente aggiorna il nome del gruppo Greengrass specificato. Se desideri aggiornare i dettagli del gruppo, utilizza il comando `create-group-version` per creare una nuova versione.

```
aws greengrass update-group \
    --group-id "1402daf9-71cf-4cfe-8be0-d5e80526d0d8" \
    --name TestGroup4of6
```

Per ulteriori informazioni, consulta Configure AWS IoT Greengrass on IoT nella AWS IoT AWS Greengrass Developer Guide.

- Per i dettagli sull'API, consulta AWS CLI Command UpdateGroupReference.

`update-logger-definition`

Il seguente esempio di codice mostra come utilizzare`update-logger-definition`.

AWS CLI

Come aggiornare una definizione di logger

L'esempio `update-logger-definition` seguente modifica il nome della definizione del logger specificata. Puoi aggiornare solo la proprietà `name` di una definizione di logger.

```
aws greengrass update-logger-definition \
    --logger-definition-id "a454b62a-5d56-4ca9-bdc4-8254e1662cb0" \
    --name "LoggingConfigsForSensors"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Monitoring with AWS IoT Greengrass Logs nella IoT AWS Greengrass Developer Guide.

- Per i dettagli sull'API, consulta Command UpdateLoggerDefinitionReference AWS CLI .

`update-resource-definition`

Il seguente esempio di codice mostra come utilizzare`update-resource-definition`.

AWS CLI

Come aggiornare il nome di una definizione di risorsa

L'esempio `update-resource-definition` seguente aggiorna il nome per la definizione di risorsa specificata. Se desideri modificare i dettagli della risorsa, utilizza il comando `create-resource-definition-version` per creare una nuova versione.

```
aws greengrass update-resource-definition \
    --resource-definition-id "c8bb9ebc-c3fd-40a4-9c6a-568d75569d38" \
    --name GreengrassConnectorResources
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Accesso alle risorse locali con funzioni Lambda e connettori nella Guida per gli sviluppatori di AWS IoT Greengrass.

- Per i dettagli sull'API, consulta UpdateResourceDefinition AWS CLICommand Reference.

`update-subscription-definition`

Il seguente esempio di codice mostra come utilizzare`update-subscription-definition`.

AWS CLI

Come aggiornare il nome di una definizione di abbonamento

L'esempio `update-subscription-definition` seguente aggiorna il nome per la definizione di abbonamento specificata. Se desideri modificare i dettagli dell'abbonamento, utilizza il comando `create-subscription-definition-version` per creare una nuova versione.

```
aws greengrass update-subscription-definition \
    --subscription-definition-id "fa81bc84-3f59-4377-a84b-5d0134da359b" \
    --name "ObsoleteSubscription"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta il titolo nella guida.

- Per i dettagli sull'API, consulta UpdateSubscriptionDefinition AWS CLICommand Reference.

`update-thing-runtime-configuration`

Il seguente esempio di codice mostra come utilizzare`update-thing-runtime-configuration`.

AWS CLI

Come attivare la telemetria nella configurazione di runtime di un core Greengrass

L'esempio `update-thing-runtime-configuration` seguente aggiorna la configurazione di runtime di un core Greengrass per attivare la telemetria.

```
aws greengrass update-thing-runtime-configuration \
    --thing-name SampleGreengrassCore \
    --telemetry-configuration {\"Telemetry\":\"On\"}
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Configurazione delle impostazioni di telemetria nella Guida per gli sviluppatori di AWS IoT Greengrass.

- Per i dettagli sull'API, consulta UpdateThingRuntimeConfiguration AWS CLICommand Reference.

AWS IoT Greengrass V2 esempi utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with AWS IoT Greengrass V2.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`associate-service-role-to-account`

Il seguente esempio di codice mostra come utilizzare`associate-service-role-to-account`.

AWS CLI

Per associare il ruolo di servizio Greengrass al tuo account AWS

L'`associate-service-role-to-account`esempio seguente associa un ruolo di servizio a AWS IoT Greengrass per AWS il tuo account.

```
aws greengrassv2 associate-service-role-to-account \
    --role-arn arn:aws:iam::123456789012:role/service-role/Greengrass_ServiceRole
```

Output:

```
{
    "associatedAt": "2022-01-19T19:21:53Z"
}
```

Per ulteriori informazioni, consulta Ruolo di servizio di Greengrass nella Guida per gli sviluppatori di AWS IoT Greengrass V2.

- Per i dettagli sull'API, consulta AWS CLI Command AssociateServiceRoleToAccountReference.

`batch-associate-client-device-with-core-device`

Il seguente esempio di codice mostra come utilizzare`batch-associate-client-device-with-core-device`.

AWS CLI

Come associare i dispositivi client a un dispositivo principale

L'esempio `batch-associate-client-device-with-core-device` seguente associa due dispositivi client a un dispositivo principale.

```
aws greengrassv2 batch-associate-client-device-with-core-device \
  --core-device-thing-name MyGreengrassCore \
  --entries thingName=MyClientDevice1 thingName=MyClientDevice2
```

Output:

```
{
    "errorEntries": []
}
```

Per ulteriori informazioni, consulta Interazione con i dispositivi IoT locali nella Guida per gli sviluppatori di AWS IoT Greengrass V2.

- Per i dettagli sull'API, consulta BatchAssociateClientDeviceWithCoreDevice AWS CLICommand Reference.

`batch-disassociate-client-device-from-core-device`

Il seguente esempio di codice mostra come utilizzare`batch-disassociate-client-device-from-core-device`.

AWS CLI

Come dissociare i dispositivi client da un dispositivo principale

L'esempio `batch-disassociate-client-device-from-core-device` seguente dissocia due dispositivi client da un dispositivo principale.

```
aws greengrassv2 batch-disassociate-client-device-from-core-device \
  --core-device-thing-name MyGreengrassCore \
  --entries thingName=MyClientDevice1 thingName=MyClientDevice2
```

Output:

```
{
    "errorEntries": []
}
```

Per ulteriori informazioni, consulta Interazione con i dispositivi IoT locali nella Guida per gli sviluppatori di AWS IoT Greengrass V2.

- Per i dettagli sull'API, consulta BatchDisassociateClientDeviceFromCoreDevice AWS CLICommand Reference.

`cancel-deployment`

Il seguente esempio di codice mostra come utilizzare`cancel-deployment`.

AWS CLI

Come annullare un'implementazione

L'esempio `cancel-deployment` seguente arresta un'implementazione continua in un gruppo di oggetti.

```
aws greengrassv2 cancel-deployment \
    --deployment-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Output:

```
{
    "message": "SUCCESS"
}
```

Per ulteriori informazioni, consulta Annullamento delle implementazioni nella Guida per gli sviluppatori di AWS IoT Greengrass V2.

- Per i dettagli sull'API, consulta CancelDeployment AWS CLICommand Reference.

`create-component-version`

Il seguente esempio di codice mostra come utilizzare`create-component-version`.

AWS CLI

Esempio 1: come creare una versione di un componente da una ricetta

L'esempio `create-component-version` seguente crea una versione di un componente Hello World da un file di ricetta.

```
aws greengrassv2 create-component-version \
    --inline-recipe fileb://com.example.HelloWorld-1.0.0.json
```

Contenuto di `com.example.HelloWorld-1.0.0.json`:

```
{
    "RecipeFormatVersion": "2020-01-25",
    "ComponentName": "com.example.HelloWorld",
    "ComponentVersion": "1.0.0",
    "ComponentDescription": "My first AWS IoT Greengrass component.",
    "ComponentPublisher": "Amazon",
    "ComponentConfiguration": {
        "DefaultConfiguration": {
            "Message": "world"
        }
    },
    "Manifests": [
        {
            "Platform": {
                "os": "linux"
            },
            "Lifecycle": {
                "Run": "echo 'Hello {configuration:/Message}'"
            }
        }
    ]
}
```

Output:

```
{
```

```
    "arn": "arn:aws:greengrass:us-
west-2:123456789012:components:com.example.HelloWorld:versions:1.0.0",
    "componentName": "com.example.HelloWorld",
    "componentVersion": "1.0.0",
    "creationTimestamp": "2021-01-07T16:24:33.650000-08:00",
    "status": {
        "componentState": "REQUESTED",
        "message": "NONE",
        "errors": {}
    }
}
```

Per ulteriori informazioni, consulta Creazione di componenti personalizzati nella Guida per gli sviluppatori di AWS IoT Greengrass V2.https://docs.aws.amazon.com/greengrass/v2/developerguide/upload-components.html

Esempio 2: creare una versione di un componente da una funzione AWS Lambda

L'`create-component-version`esempio seguente crea una versione di un componente Hello World da una funzione AWS Lambda.

```
aws greengrassv2 create-component-version \
    --cli-input-json file://lambda-function-component.json
```

Contenuto di `lambda-function-component.json`:

```
{
    "lambdaFunction": {
        "lambdaArn": "arn:aws:lambda:us-
west-2:123456789012:function:HelloWorldPythonLambda:1",
        "componentName": "com.example.HelloWorld",
        "componentVersion": "1.0.0",
        "componentLambdaParameters": {
            "eventSources": [
                {
                    "topic": "hello/world/+",
                    "type": "IOT_CORE"
                }
            ]
        }
    }
}
```

Output:

```
{
    "arn": "arn:aws:greengrass:us-
west-2:123456789012:components:com.example.HelloWorld:versions:1.0.0",
    "componentName": "com.example.HelloWorld",
    "componentVersion": "1.0.0",
    "creationTimestamp": "2021-01-07T17:05:27.347000-08:00",
    "status": {
        "componentState": "REQUESTED",
        "message": "NONE",
        "errors": {}
    }
}
```

Per ulteriori informazioni, consulta Run AWS Lambda functions nella IoT AWS Greengrass V2 Developer Guide.

- Per i dettagli sull'API, consulta Command CreateComponentVersionReference AWS CLI .

`create-deployment`

Il seguente esempio di codice mostra come utilizzare`create-deployment`.

AWS CLI

Esempio 1: come creare un'implementazione

L'`create-deployment`esempio seguente distribuisce l'interfaccia a riga di comando AWS IoT Greengrass su un dispositivo principale.

```
aws greengrassv2 create-deployment \
    --cli-input-json file://cli-deployment.json
```

Contenuto di `cli-deployment.json`:

```
{
    "targetArn": "arn:aws:iot:us-west-2:123456789012:thing/MyGreengrassCore",
    "deploymentName": "Deployment for MyGreengrassCore",
    "components": {
        "aws.greengrass.Cli": {
            "componentVersion": "2.0.3"
```

```
        }
    },
    "deploymentPolicies": {
        "failureHandlingPolicy": "DO_NOTHING",
        "componentUpdatePolicy": {
            "timeoutInSeconds": 60,
            "action": "NOTIFY_COMPONENTS"
        },
        "configurationValidationPolicy": {
            "timeoutInSeconds": 60
        }
    },
    "iotJobConfiguration": {}
}
```

Output:

```
{
    "deploymentId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
}
```

Per ulteriori informazioni, consulta Creazioni delle implementazioni nella Guida per gli sviluppatori di AWS IoT Greengrass V2.

Esempio 2: come creare un'implementazione che aggiorni le configurazioni dei componenti

L'`create-deployment`esempio seguente distribuisce il componente AWS IoT Greengrass nucleus su un gruppo di dispositivi principali. Questa implementazione applica i seguenti aggiornamenti di configurazione per il componente nucleus:

Reimposta le impostazioni proxy dei dispositivi di destinazione ai valori predefiniti senza impostazioni proxy. Reimposta le impostazioni MQTT dei dispositivi di destinazione ai valori predefiniti. Imposta le opzioni JVM per la JVM del componente nucleus. Imposta il livello di registrazione dei log per il componente nucleus.

```
aws greengrassv2 create-deployment \
    --cli-input-json file://nucleus-deployment.json
```

Contenuto di `nucleus-deployment.json`:

```
{
```

```
    "targetArn": "arn:aws:iot:us-west-2:123456789012:thinggroup/
MyGreengrassCoreGroup",
    "deploymentName": "Deployment for MyGreengrassCoreGroup",
    "components": {
        "aws.greengrass.Nucleus": {
            "componentVersion": "2.0.3",
            "configurationUpdate": {
                "reset": [
                    "/networkProxy",
                    "/mqtt"
                ],
                "merge": "{\"jvmOptions\":\"-Xmx64m\",\"logging\":{\"level\":\"WARN
\"}}"
            }
        }
    },
    "deploymentPolicies": {
        "failureHandlingPolicy": "ROLLBACK",
        "componentUpdatePolicy": {
            "timeoutInSeconds": 60,
            "action": "NOTIFY_COMPONENTS"
        },
        "configurationValidationPolicy": {
            "timeoutInSeconds": 60
        }
    },
    "iotJobConfiguration": {}
}
```

Output:

```
{
    "deploymentId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "iotJobId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
    "iotJobArn": "arn:aws:iot:us-west-2:123456789012:job/a1b2c3d4-5678-90ab-cdef-
EXAMPLE22222"
}
```

Per ulteriori informazioni, consulta Creazioni delle implementazioni nella Guida per gli sviluppatori di AWS IoT Greengrass V2.https://docs.aws.amazon.com/greengrass/v2/developerguide/update-component-configurations.html

- Per i dettagli sull'API, consulta Command CreateDeploymentReference AWS CLI .

`delete-component`

Il seguente esempio di codice mostra come utilizzare`delete-component`.

AWS CLI

Come eliminare una versione del componente

L'esempio `delete-component` seguente elimina un componente Hello World.

```
aws greengrassv2 delete-component \
    --arn arn:aws:greengrass:us-
west-2:123456789012:components:com.example.HelloWorld:versions:1.0.0
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Gestione dei componenti nella Guida per gli sviluppatori di AWS IoT Greengrass V2.

- Per i dettagli sull'API, consulta DeleteComponent AWS CLICommand Reference.

`delete-core-device`

Il seguente esempio di codice mostra come utilizzare`delete-core-device`.

AWS CLI

Come eliminare un dispositivo principale

L'`delete-core-device`esempio seguente elimina un dispositivo principale AWS IoT Greengrass.

```
aws greengrassv2 delete-core-device \
    --core-device-thing-name MyGreengrassCore
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Disinstallare il software AWS IoT Greengrass Core nella Guida per sviluppatori IoT AWS Greengrass V2.

- Per i dettagli sull'API, consulta Command DeleteCoreDeviceReference AWS CLI .

`describe-component`

Il seguente esempio di codice mostra come utilizzare`describe-component`.

AWS CLI

Come descrivere una versione del componente

L'esempio `describe-component` seguente descrive un componente Hello World.

```
aws greengrassv2 describe-component \
    --arn arn:aws:greengrass:us-
west-2:123456789012:components:com.example.HelloWorld:versions:1.0.0
```

Output:

```
{
    "arn": "arn:aws:greengrass:us-
west-2:123456789012:components:com.example.HelloWorld:versions:1.0.0",
    "componentName": "com.example.HelloWorld",
    "componentVersion": "1.0.0",
    "creationTimestamp": "2021-01-07T17:12:11.133000-08:00",
    "publisher": "Amazon",
    "description": "My first AWS IoT Greengrass component.",
    "status": {
        "componentState": "DEPLOYABLE",
        "message": "NONE",
        "errors": {}
    },
    "platforms": [
        {
            "attributes": {
                "os": "linux"
            }
        }
    ]
}
```

Per ulteriori informazioni, consulta Gestione dei componenti nella Guida per gli sviluppatori di AWS IoT Greengrass V2.

- Per i dettagli sull'API, consulta DescribeComponent AWS CLICommand Reference.

`disassociate-service-role-from-account`

Il seguente esempio di codice mostra come utilizzare`disassociate-service-role-from-account`.

AWS CLI

Per dissociare il ruolo di servizio Greengrass dal tuo account AWS

L'`disassociate-service-role-from-account`esempio seguente dissocia il ruolo del servizio Greengrass da IoT Greengrass per il AWS tuo account. AWS

```
aws greengrassv2 disassociate-service-role-from-account
```

Output:

```
{
    "disassociatedAt": "2022-01-19T19:26:09Z"
}
```

Per ulteriori informazioni, consulta Ruolo di servizio di Greengrass nella Guida per gli sviluppatori di AWS IoT Greengrass V2.

- Per i dettagli sull'API, consulta DisassociateServiceRoleFromAccountCommand Reference.AWS CLI

`get-component-version-artifact`

Il seguente esempio di codice mostra come utilizzare`get-component-version-artifact`.

AWS CLI

Come ottenere un URL per scaricare un artefatto del componente

L'esempio `get-component-version-artifact` seguente ottiene un URL per scaricare la console di debug locale del file JAR del componente.

```
aws greengrassv2 get-component-version-artifact \
    --arn arn:aws:greengrass:us-
west-2:aws:components:aws.greengrass.LocalDebugConsole:versions:2.0.3 \
    --artifact-name "Uvt6ZEzQ9TKiAuLbfXBX_APdY0TWks3uc46tHFHTzBM=/
aws.greengrass.LocalDebugConsole.jar"
```

Output:

```
{
    "preSignedUrl": "https://evergreencomponentmanageme-
artifactbucket7410c9ef-g18n1iya8kwr.s3.us-west-2.amazonaws.com/public/
aws.greengrass.LocalDebugConsole/2.0.3/s3/ggv2-component-releases-prod-pdx/
EvergreenHttpDebugView/2ffc496ba41b39568968b22c582b4714a937193ee7687a45527238e696672521/
aws.greengrass.LocalDebugConsole/aws.greengrass.LocalDebugConsole.jar?X-Amz-
Security-Token=KwfLKSdEXAMPLE..."
}
```

Per ulteriori informazioni, consulta Gestione dei componenti nella Guida per gli sviluppatori di AWS IoT Greengrass V2.

- Per i dettagli sull'API, consulta GetComponentVersionArtifact AWS CLICommand Reference.

`get-component`

Il seguente esempio di codice mostra come utilizzare`get-component`.

AWS CLI

Esempio 1: come scaricare la ricetta di un componente in formato YAML (Linux, macOS o Unix)

L'esempio `get-component` seguente scarica la ricetta di un componente Hello World in un file in formato YAML. Questo comando funziona nel modo seguente:

Utilizza i parametri `--output` e `--query` per controllare l'output del comando. Questi parametri estraggono il blob della ricetta dall'output del comando. Per ulteriori informazioni sul controllo dell'output dei comandi, consulta Controllo dell'output dei comandi nella Guida per l'utente dell'interfaccia a riga di comando AWS . Utilizza la utility `base64`. Questa utility decodifica il blob estratto nel testo originale. Il blob restituito da un comando `get-component` eseguito correttamente è un testo con codifica Base64. È necessario decodificare questo blob per ottenere il testo originale. Salva il testo decodificato in un file. La sezione finale del comando (> `com.example.HelloWorld-1.0.0.json`) salva il testo decodificato in un file.

```
aws greengrassv2 get-component \
    --arn arn:aws:greengrass:us-
west-2:123456789012:components:com.example.HelloWorld:versions:1.0.0 \
    --recipe-output-format YAML \
    --query recipe \
```

```
    --output text | base64 --decode > com.example.HelloWorld-1.0.0.json
```

Per ulteriori informazioni, consulta Gestione dei componenti nella Guida per gli sviluppatori di AWS IoT Greengrass V2.

Esempio 2: come scaricare la ricetta di un componente in formato YAML (Windows CMD)

L'esempio `get-component` seguente scarica la ricetta di un componente Hello World in un file in formato YAML. Questo comando utilizza l'utility `certutil`.

```
aws greengrassv2 get-component ^
    --arn arn:aws:greengrass:us-
west-2:675946970638:components:com.example.HelloWorld:versions:1.0.0 ^
    --recipe-output-format YAML ^
    --query recipe ^
    --output text > com.example.HelloWorld-1.0.0.yaml.b64

certutil -
decode com.example.HelloWorld-1.0.0.yaml.b64 com.example.HelloWorld-1.0.0.yaml
```

Per ulteriori informazioni, consulta Gestione dei componenti nella Guida per gli sviluppatori di AWS IoT Greengrass V2.

Esempio 3: Per scaricare la ricetta di un componente in formato YAML (Windows) PowerShell

L'esempio `get-component` seguente scarica la ricetta di un componente Hello World in un file in formato YAML. Questo comando utilizza l'utility `certutil`.

```
aws greengrassv2 get-component `
    --arn arn:aws:greengrass:us-
west-2:675946970638:components:com.example.HelloWorld:versions:1.0.0 `
    --recipe-output-format YAML `
    --query recipe `
    --output text > com.example.HelloWorld-1.0.0.yaml.b64

certutil -
decode com.example.HelloWorld-1.0.0.yaml.b64 com.example.HelloWorld-1.0.0.yaml
```

Per ulteriori informazioni, consulta Gestione dei componenti nella Guida per gli sviluppatori di AWS IoT Greengrass V2.

- Per i dettagli sull'API, consulta AWS CLI Command GetComponentReference.

`get-connectivity-info`

Il seguente esempio di codice mostra come utilizzare`get-connectivity-info`.

AWS CLI

Come ottenere le informazioni di connettività per un dispositivo core Greengrass

L'esempio `get-connectivity-info` seguente ottiene le informazioni di connettività per un dispositivo Greengrass principale. I dispositivi client utilizzano queste informazioni per connettersi al broker MQTT che funziona su questo dispositivo principale.

```
aws greengrassv2 get-connectivity-info \
    --thing-name MyGreengrassCore
```

Output:

```
{
    "connectivityInfo": [
        {
            "id": "localIP_192.0.2.0",
            "hostAddress": "192.0.2.0",
            "portNumber": 8883
        }
    ]
}
```

Per ulteriori informazioni, consulta Gestione degli endpoint dei dispositivi di base nella Guida per gli sviluppatori di AWS IoT Greengrass V2.

- Per i dettagli sull'API, consulta GetConnectivityInfo AWS CLICommand Reference.

`get-core-device`

Il seguente esempio di codice mostra come utilizzare`get-core-device`.

AWS CLI

Come ottenere un dispositivo principale

L'`get-core-device`esempio seguente ottiene informazioni su un dispositivo principale AWS IoT Greengrass.

```
aws greengrassv2 get-core-device \
    --core-device-thing-name MyGreengrassCore
```

Output:

```
{
    "coreDeviceThingName": "MyGreengrassCore",
    "coreVersion": "2.0.3",
    "platform": "linux",
    "architecture": "amd64",
    "status": "HEALTHY",
    "lastStatusUpdateTimestamp": "2021-01-08T04:57:58.838000-08:00",
    "tags": {}
}
```

Per ulteriori informazioni, consulta Controllo degli stati dei dispositivi di base nella Guida per gli sviluppatori di AWS IoT Greengrass V2.

- Per i dettagli sull'API, vedere GetCoreDevicein AWS CLI Command Reference.

`get-deployment`

Il seguente esempio di codice mostra come utilizzare`get-deployment`.

AWS CLI

Come ottenere un'implementazione

L'`get-deployment`esempio seguente ottiene informazioni sull'implementazione del componente AWS IoT Greengrass nucleus su un gruppo di dispositivi principali.

```
aws greengrassv2 get-deployment \
    --deployment-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Output:

```
{
    "targetArn": "arn:aws:iot:us-west-2:123456789012:thinggroup/
MyGreengrassCoreGroup",
    "revisionId": "14",
```

```
    "deploymentId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "deploymentName": "Deployment for MyGreengrassCoreGroup",
    "deploymentStatus": "ACTIVE",
    "iotJobId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
    "iotJobArn": "arn:aws:iot:us-west-2:123456789012:job/a1b2c3d4-5678-90ab-cdef-
EXAMPLE22222",
    "components": {
        "aws.greengrass.Nucleus": {
            "componentVersion": "2.0.3",
            "configurationUpdate": {
                "merge": "{\"jvmOptions\":\"-Xmx64m\",\"logging\":{\"level\":\"WARN
\"}}",
                "reset": [
                    "/networkProxy",
                    "/mqtt"
                ]
            }
        }
    },
    "deploymentPolicies": {
        "failureHandlingPolicy": "ROLLBACK",
        "componentUpdatePolicy": {
            "timeoutInSeconds": 60,
            "action": "NOTIFY_COMPONENTS"
        },
        "configurationValidationPolicy": {
            "timeoutInSeconds": 60
        }
    },
    "iotJobConfiguration": {},
    "creationTimestamp": "2021-01-07T17:21:20.691000-08:00",
    "isLatestForTarget": false,
    "tags": {}
}
```

Per ulteriori informazioni, consulta Distribuzione dei componenti ai dispositivi nella Guida per gli sviluppatori di AWS IoT Greengrass V2.

- Per i dettagli sull'API, consulta AWS CLI Command GetDeploymentReference.

`get-service-role-for-account`

Il seguente esempio di codice mostra come utilizzare`get-service-role-for-account`.

AWS CLI

Per ottenere il ruolo di servizio Greengrass per il tuo account AWS

L'`get-service-role-for-account`esempio seguente ottiene il ruolo di servizio associato a AWS IoT Greengrass per il tuo AWS account.

```
aws greengrassv2 get-service-role-for-account
```

Output:

```
{
    "associatedAt": "2022-01-19T19:21:53Z",
    "roleArn": "arn:aws:iam::123456789012:role/service-role/Greengrass_ServiceRole"
}
```

Per ulteriori informazioni, consulta Ruolo di servizio di Greengrass nella Guida per gli sviluppatori di AWS IoT Greengrass V2.

- Per i dettagli sull'API, consulta GetServiceRoleForAccount AWS CLICommand Reference.

`list-client-devices-associated-with-core-device`

Il seguente esempio di codice mostra come utilizzare`list-client-devices-associated-with-core-device`.

AWS CLI

Come elencare i dispositivi client a un dispositivo principale

L'esempio `list-client-devices-associated-with-core-device` seguente elenca tutti i dispositivi client associati a un dispositivo principale.

```
aws greengrassv2 list-client-devices-associated-with-core-device \
  --core-device-thing-name MyTestGreengrassCore
```

Output:

```
{
    "associatedClientDevices": [
```

```
        {
            "thingName": "MyClientDevice2",
            "associationTimestamp": "2021-07-12T16:33:55.843000-07:00"
        },
        {
            "thingName": "MyClientDevice1",
            "associationTimestamp": "2021-07-12T16:33:55.843000-07:00"
        }
    ]
}
```

Per ulteriori informazioni, consulta Interazione con i dispositivi IoT locali nella Guida per gli sviluppatori di AWS IoT Greengrass V2.

- Per i dettagli sull'API, consulta ListClientDevicesAssociatedWithCoreDevice AWS CLICommand Reference.

`list-component-versions`

Il seguente esempio di codice mostra come utilizzare`list-component-versions`.

AWS CLI

Come elencare le versioni di un componente

L'esempio `list-component-versions` seguente elenca tutte le versioni di un componente Hello World.

```
aws greengrassv2 list-component-versions \
    --arn arn:aws:greengrass:us-
west-2:123456789012:components:com.example.HelloWorld
```

Output:

```
{
    "componentVersions": [
        {
            "componentName": "com.example.HelloWorld",
            "componentVersion": "1.0.1",
            "arn": "arn:aws:greengrass:us-
west-2:123456789012:components:com.example.HelloWorld:versions:1.0.1"
        },
```

```
        {
            "componentName": "com.example.HelloWorld",
            "componentVersion": "1.0.0",
            "arn": "arn:aws:greengrass:us-
west-2:123456789012:components:com.example.HelloWorld:versions:1.0.0"
        }
    ]
}
```

Per ulteriori informazioni, consulta Gestione dei componenti nella Guida per gli sviluppatori di AWS IoT Greengrass V2.

- Per i dettagli sull'API, consulta ListComponentVersions AWS CLICommand Reference.

`list-components`

Il seguente esempio di codice mostra come utilizzare`list-components`.

AWS CLI

Come elencare i componenti

L'`list-components`esempio seguente elenca ogni componente e la sua versione più recente definita nell' AWS account nella regione corrente.

```
aws greengrassv2 list-components
```

Output:

```
{
    "components": [
        {
            "arn": "arn:aws:greengrass:us-
west-2:123456789012:components:com.example.HelloWorld",
            "componentName": "com.example.HelloWorld",
            "latestVersion": {
                "arn": "arn:aws:greengrass:us-
west-2:123456789012:components:com.example.HelloWorld:versions:1.0.1",
                "componentVersion": "1.0.1",
                "creationTimestamp": "2021-01-08T16:51:07.352000-08:00",
                "description": "My first AWS IoT Greengrass component.",
                "publisher": "Amazon",
```

```
                "platforms": [
                    {
                        "attributes": {
                            "os": "linux"
                        }
                    }
                ]
            }
        }
    ]
}
```

Per ulteriori informazioni, consulta Gestione dei componenti nella Guida per gli sviluppatori di AWS IoT Greengrass V2.

- Per i dettagli sull'API, consulta ListComponents AWS CLICommand Reference.

`list-core-devices`

Il seguente esempio di codice mostra come utilizzare`list-core-devices`.

AWS CLI

Come elencare i dispositivi principali

L'`list-core-devices`esempio seguente elenca i dispositivi principali AWS IoT Greengrass presenti nel tuo AWS account nella regione corrente.

```
aws greengrassv2 list-core-devices
```

Output:

```
{
    "coreDevices": [
        {
            "coreDeviceThingName": "MyGreengrassCore",
            "status": "HEALTHY",
            "lastStatusUpdateTimestamp": "2021-01-08T04:57:58.838000-08:00"
        }
    ]
}
```

Per ulteriori informazioni, consulta Controllo degli stati dei dispositivi di base nella Guida per gli sviluppatori di AWS IoT Greengrass V2.

- Per i dettagli sull'API, consulta ListCoreDevices AWS CLICommand Reference.

`list-deployments`

Il seguente esempio di codice mostra come utilizzare`list-deployments`.

AWS CLI

Come elencare le implementazioni

L'`list-deployments`esempio seguente elenca l'ultima revisione di ogni distribuzione definita nel tuo AWS account nella regione corrente.

```
aws greengrassv2 list-deployments
```

Output:

```
{
    "deployments": [
        {
            "targetArn": "arn:aws:iot:us-west-2:123456789012:thinggroup/
MyGreengrassCoreGroup",
            "revisionId": "14",
            "deploymentId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
            "deploymentName": "Deployment for MyGreengrassCoreGroup",
            "creationTimestamp": "2021-01-07T17:21:20.691000-08:00",
            "deploymentStatus": "ACTIVE",
            "isLatestForTarget": false
        },
        {
            "targetArn": "arn:aws:iot:us-west-2:123456789012:thing/
MyGreengrassCore",
            "revisionId": "1",
            "deploymentId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
            "deploymentName": "Deployment for MyGreengrassCore",
            "creationTimestamp": "2021-01-06T16:10:42.407000-08:00",
            "deploymentStatus": "COMPLETED",
            "isLatestForTarget": false
        }
    ]
```

```
}
```

Per ulteriori informazioni, consulta Distribuzione dei componenti ai dispositivi nella Guida per gli sviluppatori di AWS IoT Greengrass V2.

- Per i dettagli sull'API, consulta ListDeployments AWS CLICommand Reference.

`list-effective-deployments`

Il seguente esempio di codice mostra come utilizzare`list-effective-deployments`.

AWS CLI

Come elencare i processi di implementazione

L'`list-effective-deployments`esempio seguente elenca le implementazioni che si applicano a un dispositivo core IoT AWS Greengrass.

```
aws greengrassv2 list-effective-deployments \
    --core-device-thing-name MyGreengrassCore
```

Output:

```
{
    "effectiveDeployments": [
        {
            "deploymentId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
            "deploymentName": "Deployment for MyGreengrassCore",
            "iotJobId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE33333",
            "targetArn": "arn:aws:iot:us-west-2:123456789012:thing/
MyGreengrassCore",
            "coreDeviceExecutionStatus": "COMPLETED",
            "reason": "SUCCESSFUL",
            "creationTimestamp": "2021-01-06T16:10:42.442000-08:00",
            "modifiedTimestamp": "2021-01-08T17:21:27.830000-08:00"
        },
        {
            "deploymentId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
            "deploymentName": "Deployment for MyGreengrassCoreGroup",
            "iotJobId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE44444",
            "iotJobArn": "arn:aws:iot:us-west-2:123456789012:job/a1b2c3d4-5678-90ab-
cdef-EXAMPLE44444",
```

```
            "targetArn": "arn:aws:iot:us-west-2:123456789012:thinggroup/
MyGreengrassCoreGroup",
            "coreDeviceExecutionStatus": "SUCCEEDED",
            "reason": "SUCCESSFUL",
            "creationTimestamp": "2021-01-07T17:19:20.394000-08:00",
            "modifiedTimestamp": "2021-01-07T17:21:20.721000-08:00"
        }
    ]
}
```

Per ulteriori informazioni, consulta Controllo degli stati dei dispositivi di base nella Guida per gli sviluppatori di AWS IoT Greengrass V2.

- Per i dettagli sull'API, consulta Command ListEffectiveDeploymentsReference AWS CLI .

`list-installed-components`

Il seguente esempio di codice mostra come utilizzare`list-installed-components`.

AWS CLI

Come elencare i componenti installati in un dispositivo principale

L'`list-installed-components`esempio seguente elenca i componenti installati su un dispositivo core AWS IoT Greengrass.

```
aws greengrassv2 list-installed-components \
    --core-device-thing-name MyGreengrassCore
```

Output:

```
{
    "installedComponents": [
        {
            "componentName": "aws.greengrass.Cli",
            "componentVersion": "2.0.3",
            "lifecycleState": "RUNNING",
            "isRoot": true
        },
        {
            "componentName": "aws.greengrass.Nucleus",
            "componentVersion": "2.0.3",
```

```
            "lifecycleState": "FINISHED",
            "isRoot": true
        }
    ]
}
```

Per ulteriori informazioni, consulta Controllo degli stati dei dispositivi di base nella Guida per gli sviluppatori di AWS IoT Greengrass V2.

- Per i dettagli sull'API, consulta ListInstalledComponents AWS CLICommand Reference.

`list-tags-for-resource`

Il seguente esempio di codice mostra come utilizzare`list-tags-for-resource`.

AWS CLI

Come elencare i tag per una risorsa

L'`list-tags-for-resource`esempio seguente elenca tutti i tag per un dispositivo core AWS IoT Greengrass.

```
aws greengrassv2 list-tags-for-resource \
    --resource-arn arn:aws:greengrass:us-
west-2:123456789012:coreDevices:MyGreengrassCore
```

Output:

```
{
    "tags": {
        "Owner": "richard-roe"
    }
}
```

Per ulteriori informazioni, consulta Associazione di tag alle risorse nella Guida per gli sviluppatori di AWS IoT Greengrass V2.

- Per i dettagli sull'API, vedere ListTagsForResourcein AWS CLI Command Reference.

`tag-resource`

Il seguente esempio di codice mostra come utilizzare`tag-resource`.

AWS CLI

Come aggiungere un tag a una risorsa

L'`tag-resource`esempio seguente aggiunge un tag owner a un dispositivo core AWS IoT Greengrass. È possibile utilizzare questo tag per controllare l'accesso al dispositivo principale in base al proprietario.

```
aws greengrassv2 tag-resource \
    --resource-arn arn:aws:greengrass:us-
west-2:123456789012:coreDevices:MyGreengrassCore \
    --tags Owner=richard-roe
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Associazione di tag alle risorse nella Guida per gli sviluppatori di AWS IoT Greengrass V2.

- Per i dettagli sull'API, vedere TagResourcein AWS CLI Command Reference.

`untag-resource`

Il seguente esempio di codice mostra come utilizzare`untag-resource`.

AWS CLI

Come rimuovere un tag da una risorsa

L'`untag-resource`esempio seguente rimuove un tag proprietario da un dispositivo principale AWS IoT Greengrass.

```
aws iotsitewise untag-resource \
    --resource-arn arn:aws:greengrass:us-
west-2:123456789012:coreDevices:MyGreengrassCore \
    --tag-keys Owner
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Associazione di tag alle risorse nella Guida per gli sviluppatori di AWS IoT Greengrass V2.

- Per i dettagli sull'API, vedere UntagResourcein AWS CLI Command Reference.

`update-connectivity-info`

Il seguente esempio di codice mostra come utilizzareu`pdate-connectivity-info`.

AWS CLI

Come aggiornare le informazioni di connettività per un dispositivo core Greengrass

L'esempio `update-connectivity-info` seguente ottiene le informazioni di connettività per un dispositivo Greengrass principale. I dispositivi client utilizzano queste informazioni per connettersi al broker MQTT che funziona su questo dispositivo principale.

```
aws greengrassv2 update-connectivity-info \
    --thing-name MyGreengrassCore \
    --cli-input-json file://core-device-connectivity-info.json
```

Contenuto di `core-device-connectivity-info.json`:

```
{
    "connectivityInfo": [
        {
            "hostAddress": "192.0.2.0",
            "portNumber": 8883,
            "id": "localIP_192.0.2.0"
        }
    ]
}
```

Output:

```
{
    "version": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
}
```

Per ulteriori informazioni, consulta Gestione degli endpoint dei dispositivi di base nella Guida per gli sviluppatori di AWS IoT Greengrass V2.

- Per i dettagli sull'API, consulta UpdateConnectivityInfo AWS CLICommand Reference.

AWS IoT Processi SDK rilascio esempi utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with AWS IoT Processi SDK rilascio.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`describe-job-execution`

Il seguente esempio di codice mostra come utilizzare`describe-job-execution`.

AWS CLI

Come ottenere i dettagli di un'esecuzione del processo

L'esempio `describe-job-execution` seguente recupera i dettagli dell'ultima esecuzione del processo e dell'oggetto specificati.

```
aws iot-jobs-data describe-job-execution \
    --job-id SampleJob \
    --thing-name MotionSensor1 \
    --endpoint-url https://1234567890abcd.jobs.iot.us-west-2.amazonaws.com
```

Output:

```
{
    "execution": {
        "approximateSecondsBeforeTimedOut": 88,
        "executionNumber": 2939653338,
        "jobId": "SampleJob",
```

```
        "lastUpdatedAt": 1567701875.743,
        "queuedAt": 1567701902.444,
        "status": "QUEUED",
        "thingName": "MotionSensor1 ",
        "versionNumber": 3
    }
}
```

Per ulteriori informazioni, consulta Dispositivi e servizio Jobs nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta DescribeJobExecution AWS CLICommand Reference.

`get-pending-job-executions`

Il seguente esempio di codice mostra come utilizzare`get-pending-job-executions`.

AWS CLI

Come ottenere l'elenco di tutti i processi per un oggetto che non si trovano in uno stato terminale

L'esempio `get-pending-job-executions` seguente visualizza un elenco di tutti i processi che non si trovano in uno stato terminale per l'oggetto specifico.

```
aws iot-jobs-data get-pending-job-executions \
    --thing-name MotionSensor1
    --endpoint-url https://1234567890abcd.jobs.iot.us-west-2.amazonaws.com
```

Output:

```
{
    "inProgressJobs": [
    ],
    "queuedJobs": [
        {
            "executionNumber": 2939653338,
            "jobId": "SampleJob",
            "lastUpdatedAt": 1567701875.743,
            "queuedAt": 1567701902.444,
            "versionNumber": 3
      }
    ]
```

```
}
```

Per ulteriori informazioni, consulta Dispositivi e servizio Jobs nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta GetPendingJobExecutions AWS CLICommand Reference.

`start-next-pending-job-execution`

Il seguente esempio di codice mostra come utilizzare`start-next-pending-job-execution`.

AWS CLI

Come recuperare e avviare l'esecuzione del processo in sospeso successiva per un oggetto

L'esempio `start-next-pending-job-execution` seguente recupera e avvia l'esecuzione del processo successivo il cui stato è IN_PROGRESS o QUEUED per l'oggetto specificato.

```
aws iot-jobs-data start-next-pending-job-execution \
    --thing-name MotionSensor1
    --endpoint-url https://1234567890abcd.jobs.iot.us-west-2.amazonaws.com
```

Output:

```
{
    "execution": {
        "approximateSecondsBeforeTimedOut": 88,
        "executionNumber": 2939653338,
        "jobId": "SampleJob",
        "lastUpdatedAt": 1567714853.743,
        "queuedAt": 1567701902.444,
        "startedAt": 1567714871.690,
        "status": "IN_PROGRESS",
        "thingName": "MotionSensor1 ",
        "versionNumber": 3
   }
}
```

Per ulteriori informazioni, consulta Dispositivi e servizio Jobs nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta StartNextPendingJobExecution AWS CLICommand Reference.

`update-job-execution`

Il seguente esempio di codice mostra come utilizzare`update-job-execution`.

AWS CLI

Come aggiornare lo stato dell'esecuzione di un processo

L'esempio `update-job-execution` seguente aggiorna lo stato di un processo e di un oggetto specificati.

```
aws iot-jobs-data update-job-execution \
    --job-id SampleJob \
    --thing-name MotionSensor1 \
    --status REMOVED \
    --endpoint-url https://1234567890abcd.jobs.iot.us-west-2.amazonaws.com
```

Output:

```
{
    "executionState": {
        "status": "REMOVED",
        "versionNumber": 3
    },
}
```

Per ulteriori informazioni, consulta Dispositivi e servizio Jobs nella Guida per gli sviluppatori di AWS IoT.

- Per i dettagli sull'API, consulta UpdateJobExecution AWS CLICommand Reference.

AWS IoT SiteWise esempi utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with AWS IoT SiteWise.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`associate-assets`

Il seguente esempio di codice mostra come utilizzare`associate-assets`.

AWS CLI

Come associare un asset secondario a un asset principale

L'esempio `associate-assets` seguente associa un asset turbina eolica a un asset parco eolico, dove il modello di asset turbina eolica è presente come gerarchia nel modello di asset parco eolico.

```
aws iotsitewise associate-assets \
    --asset-id a1b2c3d4-5678-90ab-cdef-44444EXAMPLE \
    --hierarchy-id a1b2c3d4-5678-90ab-cdef-77777EXAMPLE \
    --child-asset-id a1b2c3d4-5678-90ab-cdef-33333EXAMPLE
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Associating assets nella AWS IoT SiteWise User Guide.

- Per i dettagli sull'API, consulta AssociateAssets AWS CLICommand Reference.

`batch-associate-project-assets`

Il seguente esempio di codice mostra come utilizzare`batch-associate-project-assets`.

AWS CLI

Come associare un asset a un progetto

L'esempio `batch-associate-project-assets` seguente associa un asset di tipo parco eolico a un progetto.

```
aws iotsitewise batch-associate-project-assets \
    --project-id a1b2c3d4-5678-90ab-cdef-eeeeeEXAMPLE \
```

```
    --asset-ids a1b2c3d4-5678-90ab-cdef-44444EXAMPLE
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Aggiungere risorse ai progetti nella Guida all'applicazione AWS IoT SiteWise Monitor.

- Per i dettagli sull'API, consulta BatchAssociateProjectAssets AWS CLICommand Reference.

`batch-disassociate-project-assets`

Il seguente esempio di codice mostra come utilizzare`batch-disassociate-project-assets`.

AWS CLI

Come dissociare un asset da un progetto

L'esempio `batch-disassociate-project-assets` seguente disassocia un asset di tipo parco eolico da un progetto.

```
aws iotsitewise batch-disassociate-project-assets \
    --project-id a1b2c3d4-5678-90ab-cdef-eeeeeEXAMPLE \
    --asset-ids a1b2c3d4-5678-90ab-cdef-44444EXAMPLE
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Aggiungere risorse ai progetti nella Guida all'applicazione AWS IoT SiteWise Monitor.

- Per i dettagli sull'API, consulta BatchDisassociateProjectAssets AWS CLICommand Reference.

`batch-put-asset-property-value`

Il seguente esempio di codice mostra come utilizzare`batch-put-asset-property-value`.

AWS CLI

Come inviare dati alle proprietà degli asset

L'esempio `batch-put-asset-property-value` seguente invia i dati di alimentazione e temperatura alle proprietà degli asset identificate dagli alias di proprietà.

```
aws iotsitewise batch-put-asset-property-value \
    --cli-input-json file://batch-put-asset-property-value.json
```

Contenuto di `batch-put-asset-property-value.json`:

```
{
    "entries": [
        {
            "entryId": "1575691200-company-windfarm-3-turbine-7-power",
            "propertyAlias": "company-windfarm-3-turbine-7-power",
            "propertyValues": [
                {
                    "value": {
                        "doubleValue": 4.92
                    },
                    "timestamp": {
                        "timeInSeconds": 1575691200
                    },
                    "quality": "GOOD"
                }
            ]
        },
        {
            "entryId": "1575691200-company-windfarm-3-turbine-7-temperature",
            "propertyAlias": "company-windfarm-3-turbine-7-temperature",
            "propertyValues": [
                {
                    "value": {
                        "integerValue": 38
                    },
                    "timestamp": {
                        "timeInSeconds": 1575691200
                    }
                }
            ]
        }
    ]
}
```

Output:

```
{
```

```
    "errorEntries": []
}
```

Per ulteriori informazioni, consulta Ingestione di dati utilizzando l' SiteWise API IoT nella AWSAWS IoT SiteWise User Guide.

- Per i dettagli sull'API, consulta AWS CLI Command BatchPutAssetPropertyValueReference.

`create-access-policy`

Il seguente esempio di codice mostra come utilizzare`create-access-policy`.

AWS CLI

Esempio 1: come concedere a un utente l'accesso amministrativo a un portale

L'esempio `create-access-policy` seguente crea una policy di accesso che concede a un utente l'accesso amministrativo a un portale web per una società che gestisce parchi eolici.

```
aws iotsitewise create-access-policy \
    --cli-input-json file://create-portal-administrator-access-policy.json
```

Contenuto di `create-portal-administrator-access-policy.json`:

```
{
    "accessPolicyIdentity": {
        "user": {
            "id": "a1b2c3d4e5-a1b2c3d4-5678-90ab-cdef-bbbbbEXAMPLE"
        }
    },
    "accessPolicyPermission": "ADMINISTRATOR",
    "accessPolicyResource": {
        "portal": {
            "id": "a1b2c3d4-5678-90ab-cdef-aaaaaEXAMPLE"
        }
    }
}
```

Output:

```
{
    "accessPolicyId": "a1b2c3d4-5678-90ab-cdef-cccccEXAMPLE",
```

```
    "accessPolicyArn": "arn:aws:iotsitewise:us-west-2:123456789012:access-policy/
a1b2c3d4-5678-90ab-cdef-cccccEXAMPLE"
}
```

Per ulteriori informazioni, consulta Aggiungere o rimuovere amministratori del portale nella Guida per l' SiteWise utente AWS IoT.

Esempio 2: come concedere a un utente l'accesso in sola lettura a un progetto

L'esempio `create-access-policy` seguente crea una policy di accesso che concede a un utente l'accesso in sola lettura a un progetto di parco eolico.

```
aws iotsitewise create-access-policy \
    --cli-input-json file://create-project-viewer-access-policy.json
```

Contenuto di `create-project-viewer-access-policy.json`:

```
{
    "accessPolicyIdentity": {
        "user": {
            "id": "a1b2c3d4e5-a1b2c3d4-5678-90ab-cdef-bbbbbEXAMPLE"
        }
    },
    "accessPolicyPermission": "VIEWER",
    "accessPolicyResource": {
        "project": {
            "id": "a1b2c3d4-5678-90ab-cdef-eeeeeEXAMPLE"
        }
    }
}
```

Output:

```
{
    "accessPolicyId": "a1b2c3d4-5678-90ab-cdef-dddddEXAMPLE",
    "accessPolicyArn": "arn:aws:iotsitewise:us-west-2:123456789012:access-policy/
a1b2c3d4-5678-90ab-cdef-dddddEXAMPLE"
}
```

Per ulteriori informazioni, consulta Assegnazione dei visualizzatori di progetti nella Guida all'applicazione AWS IoT SiteWise Monitor.

- Per i dettagli sull'API, consulta AWS CLI Command CreateAccessPolicyReference.

`create-asset-model`

Il seguente esempio di codice mostra come utilizzare`create-asset-model`.

AWS CLI

Come creare un modello di asset

L'esempio `create-asset-model` seguente crea un modello di asset che definisce una turbina eolica con le seguenti proprietà:

Numero di serie - Il numero di serie di una turbina eolica Potenza generata - Il flusso di dati sulla potenza generata da una turbina eolica Temperatura C - Il flusso di dati della temperatura da una turbina eolica in CelsiusTemperature F - I dati di temperatura mappati da Celsius a Fahrenheit

```
aws iotsitewise create-asset-model \
    --cli-input-json file://create-wind-turbine-model.json
```

Contenuto di `create-wind-turbine-model.json`:

```
{
    "assetModelName": "Wind Turbine Model",
    "assetModelDescription": "Represents a wind turbine",
    "assetModelProperties": [
        {
            "name": "Serial Number",
            "dataType": "STRING",
            "type": {
                "attribute": {}
            }
        },
        {
            "name": "Generated Power",
            "dataType": "DOUBLE",
            "unit": "kW",
            "type": {
                "measurement": {}
            }
        },
        {
```

```
            "name": "Temperature C",
            "dataType": "DOUBLE",
            "unit": "Celsius",
            "type": {
                "measurement": {}
            }
        },
        {
            "name": "Temperature F",
            "dataType": "DOUBLE",
            "unit": "Fahrenheit",
            "type": {
                "transform": {
                    "expression": "temp_c * 9 / 5 + 32",
                    "variables": [
                        {
                            "name": "temp_c",
                            "value": {
                                "propertyId": "Temperature C"
                            }
                        }
                    ]
                }
            }
        },
        {
            "name": "Total Generated Power",
            "dataType": "DOUBLE",
            "unit": "kW",
            "type": {
                "metric": {
                    "expression": "sum(power)",
                    "variables": [
                        {
                            "name": "power",
                            "value": {
                                "propertyId": "Generated Power"
                            }
                        }
                    ],
                    "window": {
                        "tumbling": {
                            "interval": "1h"
                        }
```

```
                    }
                }
            }
        }
    ]
}
```

Output:

```
{
    "assetModelId": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
    "assetModelArn": "arn:aws:iotsitewise:us-west-2:123456789012:asset-model/
a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
    "assetModelStatus": {
        "state": "CREATING"
    }
}
```

Per ulteriori informazioni, consulta Definizione dei modelli di asset nella AWS IoT SiteWise User Guide.

- Per i dettagli sull'API, consulta CreateAssetModel AWS CLICommand Reference.

`create-asset`

Il seguente esempio di codice mostra come utilizzare`create-asset`.

AWS CLI

Come creare un asset

L'esempio `create-asset` seguente crea un asset turbina eolica a partire da un modello di asset turbina eolica.

```
aws iotsitewise create-asset \
    --asset-model-id a1b2c3d4-5678-90ab-cdef-11111EXAMPLE \
    --asset-name "Wind Turbine 1"
```

Output:

```
{
```

```
    "assetId": "a1b2c3d4-5678-90ab-cdef-33333EXAMPLE",
    "assetArn": "arn:aws:iotsitewise:us-west-2:123456789012:asset/
a1b2c3d4-5678-90ab-cdef-33333EXAMPLE",
    "assetStatus": {
        "state": "CREATING"
    }
}
```

Per ulteriori informazioni, consulta Creazione di risorse nella Guida per SiteWise l'utente AWS IoT.

- Per i dettagli sull'API, consulta CreateAsset AWS CLICommand Reference.

`create-dashboard`

Il seguente esempio di codice mostra come utilizzare`create-dashboard`.

AWS CLI

Come creare una dashboard

L'esempio `create-dashboard` seguente crea una dashboard con un grafico a linee che mostra la potenza totale generata per un parco eolico.

```
aws iotsitewise create-dashboard \
    --project-id a1b2c3d4-5678-90ab-cdef-eeeeeEXAMPLE \
    --dashboard-name "Wind Farm" \
    --dashboard-definition file://create-wind-farm-dashboard.json
```

Contenuto di `create-wind-farm-dashboard.json`:

```
{
    "widgets": [
        {
            "type": "monitor-line-chart",
            "title": "Generated Power",
            "x": 0,
            "y": 0,
            "height": 3,
            "width": 3,
            "metrics": [
                {
```

```
                    "label": "Power",
                    "type": "iotsitewise",
                    "assetId": "a1b2c3d4-5678-90ab-cdef-44444EXAMPLE",
                    "propertyId": "a1b2c3d4-5678-90ab-cdef-99999EXAMPLE"
                }
            ]
        }
    ]
}
```

Output:

```
{
    "dashboardId": "a1b2c3d4-5678-90ab-cdef-fffffEXAMPLE",
    "dashboardArn": "arn:aws:iotsitewise:us-west-2:123456789012:dashboard/
a1b2c3d4-5678-90ab-cdef-fffffEXAMPLE"
}
```

Per ulteriori informazioni, consulta Creazione di dashboard (CLI) nella Guida per l'utente AWS SiteWise IoT.

- Per i dettagli sull'API, consulta AWS CLI Command CreateDashboardReference.

`create-gateway`

Il seguente esempio di codice mostra come utilizzare`create-gateway`.

AWS CLI

Come creare un gateway

L'`create-gateway`esempio seguente crea un gateway che funziona su AWS IoT Greengrass.

```
aws iotsitewise create-gateway \
    --gateway-name ExampleCorpGateway \
    --gateway-platform greengrass={groupArn=arn:aws:greengrass:us-
west-2:123456789012:/greengrass/groups/a1b2c3d4-5678-90ab-cdef-1b1b1EXAMPLE}
```

Output:

```
{
```

```
    "gatewayId": "a1b2c3d4-5678-90ab-cdef-1a1a1EXAMPLE",
    "gatewayArn": "arn:aws:iotsitewise:us-west-2:123456789012:gateway/
a1b2c3d4-5678-90ab-cdef-1a1a1EXAMPLE"
}
```

Per ulteriori informazioni, consulta Configurazione di un gateway nella Guida per l' SiteWise utente AWS IoT.

- Per i dettagli sull'API, consulta CreateGateway AWS CLICommand Reference.

`create-portal`

Il seguente esempio di codice mostra come utilizzare`create-portal`.

AWS CLI

Come creare un portale

L'esempio `create-portal` seguente crea un portale web per una società che gestisce parchi eolici. È possibile creare portali solo nella stessa regione in cui è stato abilitato il AWS Single Sign-On.

```
aws iotsitewise create-portal \
    --portal-name WindFarmPortal \
    --portal-description "A portal that contains wind farm projects for Example
 Corp." \
    --portal-contact-email support@example.com \
    --role-arn arn:aws:iam::123456789012:role/service-role/
MySiteWiseMonitorServiceRole
```

Output:

```
{
    "portalId": "a1b2c3d4-5678-90ab-cdef-aaaaaEXAMPLE",
    "portalArn": "arn:aws:iotsitewise:us-west-2:123456789012:portal/
a1b2c3d4-5678-90ab-cdef-aaaaaEXAMPLE",
    "portalStartUrl": "https://a1b2c3d4-5678-90ab-cdef-
aaaaaEXAMPLE.app.iotsitewise.aws",
    "portalStatus": {
        "state": "CREATING"
    },
```

```
    "ssoApplicationId": "ins-a1b2c3d4-EXAMPLE"
}
```

Per ulteriori informazioni, consulta la sezione Introduzione a AWS IoT SiteWise Monitor nella AWS IoT SiteWise User Guide e Enabling AWS SSO nella AWS IoT SiteWise User Guide.

- Per i dettagli sull'API, consulta CreatePortal AWS CLICommand Reference.

`create-project`

Il seguente esempio di codice mostra come utilizzare`create-project`.

AWS CLI

Come creare un progetto

L'esempio `create-project` seguente crea un progetto di parco eolico.

```
aws iotsitewise create-project \
    --portal-id a1b2c3d4-5678-90ab-cdef-aaaaaEXAMPLE \
    --project-name "Wind Farm 1" \
    --project-description "Contains asset visualizations for Wind Farm #1 for
 Example Corp."
```

Output:

```
{
    "projectId": "a1b2c3d4-5678-90ab-cdef-eeeeeEXAMPLE",
    "projectArn": "arn:aws:iotsitewise:us-west-2:123456789012:project/
a1b2c3d4-5678-90ab-cdef-eeeeeEXAMPLE"
}
```

Per ulteriori informazioni, consulta Creazione di progetti nella Guida all'applicazione AWS IoT SiteWise Monitor.

- Per i dettagli sull'API, consulta CreateProject AWS CLICommand Reference.

`delete-access-policy`

Il seguente esempio di codice mostra come utilizzare`delete-access-policy`.

AWS CLI

Come revocare l'accesso di un utente a un progetto o portale

L'esempio `delete-access-policy` seguente elimina una policy di accesso che concede a un utente l'accesso amministrativo a un portale.

```
aws iotsitewise delete-access-policy \
    --access-policy-id a1b2c3d4-5678-90ab-cdef-cccccEXAMPLE
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Aggiungere o rimuovere amministratori del portale nella Guida per l' SiteWise utente AWS IoT.

- Per i dettagli sull'API, consulta DeleteAccessPolicy AWS CLICommand Reference.

`delete-asset-model`

Il seguente esempio di codice mostra come utilizzare`delete-asset-model`.

AWS CLI

Come eliminare un modello di asset

L'esempio `delete-asset-model` seguente elimina un modello di asset turbina eolica.

```
aws iotsitewise delete-asset-model \
    --asset-model-id a1b2c3d4-5678-90ab-cdef-11111EXAMPLE
```

Output:

```
{
    "assetModelStatus": {
        "state": "DELETING"
    }
}
```

Per ulteriori informazioni, consulta Eliminazione dei modelli di asset nella Guida per l' SiteWise utente AWS IoT.

- Per i dettagli sull'API, consulta DeleteAssetModel AWS CLICommand Reference.

`delete-asset`

Il seguente esempio di codice mostra come utilizzare`delete-asset`.

AWS CLI

Come eliminare un asset

L'esempio `delete-asset` seguente elimina un asset turbina eolica.

```
aws iotsitewise delete-asset \
    --asset-id a1b2c3d4-5678-90ab-cdef-33333EXAMPLE
```

Output:

```
{
    "assetStatus": {
        "state": "DELETING"
    }
}
```

Per ulteriori informazioni, consulta Eliminazione delle risorse nella Guida per l' SiteWise utente AWS IoT.

- Per i dettagli sull'API, consulta DeleteAsset AWS CLICommand Reference.

`delete-dashboard`

Il seguente esempio di codice mostra come utilizzare`delete-dashboard`.

AWS CLI

Come eliminare una dashboard

L'esempio `delete-dashboard` seguente elimina una dashboard di turbina eolica.

```
aws iotsitewise delete-dashboard \
    --dashboard-id a1b2c3d4-5678-90ab-cdef-fffffEXAMPLE
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Eliminazione dei dashboard nella Guida all'applicazione AWS IoT SiteWise Monitor.

- Per i dettagli sull'API, consulta AWS CLI Command DeleteDashboardReference.

`delete-gateway`

Il seguente esempio di codice mostra come utilizzare`delete-gateway`.

AWS CLI

Come eliminare un gateway

L'esempio `delete-gateway` seguente elimina un gateway.

```
aws iotsitewise delete-gateway \
    --gateway-id a1b2c3d4-5678-90ab-cdef-1a1a1EXAMPLE
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Ingestione di dati utilizzando un gateway nella AWS IoT SiteWise User Guide.

- Per i dettagli sull'API, consulta AWS CLI Command DeleteGatewayReference.

`delete-portal`

Il seguente esempio di codice mostra come utilizzare`delete-portal`.

AWS CLI

Come eliminare un portale

L'esempio `delete-portal` seguente elimina un portale web per una società che gestisce parchi eolici.

```
aws iotsitewise delete-portal \
    --portal-id a1b2c3d4-5678-90ab-cdef-aaaaaEXAMPLE
```

Output:

```
{
    "portalStatus": {
        "state": "DELETING"
    }
}
```

Per ulteriori informazioni, consulta Eliminazione di un portale nella Guida per l' SiteWise utente AWS IoT.

- Per i dettagli sull'API, consulta DeletePortal AWS CLICommand Reference.

`delete-project`

Il seguente esempio di codice mostra come utilizzare`delete-project`.

AWS CLI

Come eliminare un progetto

L'esempio `delete-project` seguente elimina un progetto di parco eolico.

```
aws iotsitewise delete-project \
    --project-id a1b2c3d4-5678-90ab-cdef-eeeeeEXAMPLE
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Eliminazione di progetti nella Guida all'applicazione AWS IoT SiteWise Monitor.

- Per i dettagli sull'API, consulta DeleteProject AWS CLICommand Reference.

`describe-access-policy`

Il seguente esempio di codice mostra come utilizzare`describe-access-policy`.

AWS CLI

Come descrivere una policy di accesso

L'esempio `describe-access-policy` seguente descrive una policy di accesso che concede a un utente l'accesso amministrativo a un portale web per una società che gestisce parchi eolici.

```
aws iotsitewise describe-access-policy \
    --access-policy-id a1b2c3d4-5678-90ab-cdef-cccccEXAMPLE
```

Output:

```
{
    "accessPolicyId": "a1b2c3d4-5678-90ab-cdef-cccccEXAMPLE",
    "accessPolicyArn": "arn:aws:iotsitewise:us-west-2:123456789012:access-policy/
a1b2c3d4-5678-90ab-cdef-cccccEXAMPLE",
    "accessPolicyIdentity": {
        "user": {
            "id": "a1b2c3d4e5-a1b2c3d4-5678-90ab-cdef-bbbbbEXAMPLE"
        }
    },
    "accessPolicyResource": {
        "portal": {
            "id": "a1b2c3d4-5678-90ab-cdef-aaaaaEXAMPLE"
        }
    },
    "accessPolicyPermission": "ADMINISTRATOR",
    "accessPolicyCreationDate": "2020-02-20T22:35:15.552880124Z",
    "accessPolicyLastUpdateDate": "2020-02-20T22:35:15.552880124Z"
}
```

Per ulteriori informazioni, consulta Aggiungere o rimuovere amministratori del portale nella Guida per l' SiteWise utente AWS IoT.

- Per i dettagli sull'API, consulta DescribeAccessPolicy AWS CLICommand Reference.

`describe-asset-model`

Il seguente esempio di codice mostra come utilizzare`describe-asset-model`.

AWS CLI

Come descrivere un modello di asset

L'esempio `describe-asset-model` seguente descrive un modello di asset di tipo parco eolico.

```
aws iotsitewise describe-asset-model \
    --asset-model-id a1b2c3d4-5678-90ab-cdef-22222EXAMPLE
```

Output:

```
{
    "assetModelId": "a1b2c3d4-5678-90ab-cdef-22222EXAMPLE",
    "assetModelArn": "arn:aws:iotsitewise:us-west-2:123456789012:asset-model/
a1b2c3d4-5678-90ab-cdef-22222EXAMPLE",
    "assetModelName": "Wind Farm Model",
    "assetModelDescription": "Represents a wind farm that comprises many wind
 turbines",
    "assetModelProperties": [
        {
            "id": "a1b2c3d4-5678-90ab-cdef-99999EXAMPLE",
            "name": "Total Generated Power",
            "dataType": "DOUBLE",
            "unit": "kW",
            "type": {
                "metric": {
                    "expression": "sum(power)",
                    "variables": [
                        {
                            "name": "power",
                            "value": {
                                "propertyId": "a1b2c3d4-5678-90ab-
cdef-66666EXAMPLE",
                                "hierarchyId": "a1b2c3d4-5678-90ab-
cdef-77777EXAMPLE"
                            }
                        }
                    ],
                    "window": {
                        "tumbling": {
                            "interval": "1h"
                        }
                    }
                }
            }
        },
        {
            "id": "a1b2c3d4-5678-90ab-cdef-88888EXAMPLE",
            "name": "Region",
            "dataType": "STRING",
            "type": {
                "attribute": {
                    "defaultValue": " "
```

```
                }
            }
        }
    ],
    "assetModelHierarchies": [
        {
            "id": "a1b2c3d4-5678-90ab-cdef-77777EXAMPLE",
            "name": "Wind Turbines",
            "childAssetModelId": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE"
        }
    ],
    "assetModelCreationDate": 1575671284.0,
    "assetModelLastUpdateDate": 1575671988.0,
    "assetModelStatus": {
        "state": "ACTIVE"
    }
}
```

Per ulteriori informazioni, consulta Descrivere un modello di asset specifico nella AWS IoT SiteWise User Guide.

- Per i dettagli sull'API, consulta DescribeAssetModel AWS CLICommand Reference.

`describe-asset-property`

Il seguente esempio di codice mostra come utilizzare`describe-asset-property`.

AWS CLI

Come descrivere una proprietà di un asset

L'esempio `describe-asset-property` seguente descrive la proprietà della potenza totale generata da un impianto eolico.

```
aws iotsitewise describe-asset-property \
    --asset-id a1b2c3d4-5678-90ab-cdef-44444EXAMPLE \
    --property-id a1b2c3d4-5678-90ab-cdef-99999EXAMPLE
```

Output:

```
{
    "assetId": "a1b2c3d4-5678-90ab-cdef-44444EXAMPLE",
```

```
    "assetName": "Wind Farm 1",
    "assetModelId": "a1b2c3d4-5678-90ab-cdef-22222EXAMPLE",
    "assetProperty": {
        "id": "a1b2c3d4-5678-90ab-cdef-99999EXAMPLE",
        "name": "Total Generated Power",
        "notification": {
            "topic": "$aws/sitewise/asset-models/a1b2c3d4-5678-90ab-
cdef-22222EXAMPLE/assets/a1b2c3d4-5678-90ab-cdef-44444EXAMPLE/properties/
a1b2c3d4-5678-90ab-cdef-99999EXAMPLE",
            "state": "DISABLED"
        },
        "dataType": "DOUBLE",
        "unit": "kW",
        "type": {
            "metric": {
                "expression": "sum(power)",
                "variables": [
                    {
                        "name": "power",
                        "value": {
                            "propertyId": "a1b2c3d4-5678-90ab-cdef-66666EXAMPLE",
                            "hierarchyId": "a1b2c3d4-5678-90ab-cdef-77777EXAMPLE"
                        }
                    }
                ],
                "window": {
                    "tumbling": {
                        "interval": "1h"
                    }
                }
            }
        }
    }
}
```

Per ulteriori informazioni, consulta Descrivere una proprietà specifica dell'asset nella AWS IoT SiteWise User Guide.

- Per i dettagli sull'API, consulta DescribeAssetProperty AWS CLICommand Reference.

`describe-asset`

Il seguente esempio di codice mostra come utilizzare`describe-asset`.

AWS CLI

Come descrivere un asset

L'esempio `describe-asset` seguente descrive un asset di tipo parco eolico.

```
aws iotsitewise describe-asset \
    --asset-id a1b2c3d4-5678-90ab-cdef-44444EXAMPLE
```

Output:

```
{
    "assetId": "a1b2c3d4-5678-90ab-cdef-44444EXAMPLE",
    "assetArn": "arn:aws:iotsitewise:us-west-2:123456789012:asset/
a1b2c3d4-5678-90ab-cdef-44444EXAMPLE",
    "assetName": "Wind Farm 1",
    "assetModelId": "a1b2c3d4-5678-90ab-cdef-22222EXAMPLE",
    "assetProperties": [
        {
            "id": "a1b2c3d4-5678-90ab-cdef-88888EXAMPLE",
            "name": "Region",
            "dataType": "STRING"
        },
        {
            "id": "a1b2c3d4-5678-90ab-cdef-99999EXAMPLE",
            "name": "Total Generated Power",
            "dataType": "DOUBLE",
            "unit": "kW"
        }
    ],
    "assetHierarchies": [
        {
            "id": "a1b2c3d4-5678-90ab-cdef-77777EXAMPLE",
            "name": "Wind Turbines"
        }
    ],
    "assetCreationDate": 1575672453.0,
    "assetLastUpdateDate": 1575672453.0,
    "assetStatus": {
        "state": "ACTIVE"
    }
}
```

Per ulteriori informazioni, consulta Descrivere un asset specifico nella AWS IoT SiteWise User Guide.

- Per i dettagli sull'API, consulta DescribeAsset AWS CLICommand Reference.

`describe-dashboard`

Il seguente esempio di codice mostra come utilizzare`describe-dashboard`.

AWS CLI

Come descrivere una dashboard

L'esempio `describe-dashboard` seguente descrive la dashboard di parco eolico specificata.

```
aws iotsitewise describe-dashboard \
    --dashboard-id a1b2c3d4-5678-90ab-cdef-fffffEXAMPLE
```

Output:

```
{
    "dashboardId": "a1b2c3d4-5678-90ab-cdef-fffffEXAMPLE",
    "dashboardArn": "arn:aws:iotsitewise:us-west-2:123456789012:dashboard/
a1b2c3d4-5678-90ab-cdef-fffffEXAMPLE",
    "dashboardName": "Wind Farm",
    "projectId": "a1b2c3d4-5678-90ab-cdef-eeeeeEXAMPLE",
    "dashboardDefinition": "{\"widgets\":[{\"type\":\"monitor-line-chart\",\"title
\":\"Generated Power\",\"x\":0,\"y\":0,\"height\":3,\"width\":3,\"metrics\":
[{\"label\":\"Power\",\"type\":\"iotsitewise\",\"assetId\":\"a1b2c3d4-5678-90ab-
cdef-44444EXAMPLE\",\"propertyId\":\"a1b2c3d4-5678-90ab-cdef-99999EXAMPLE\"}]}]}",
    "dashboardCreationDate": "2020-05-01T20:32:12.228476348Z",
    "dashboardLastUpdateDate": "2020-05-01T20:32:12.228476348Z"
}
```

Per ulteriori informazioni, consulta Visualizzazione dei dashboard nella Guida all'applicazione AWS IoT SiteWise Monitor.

- Per i dettagli sull'API, consulta DescribeDashboard AWS CLICommand Reference.

`describe-gateway-capability-configuration`

Il seguente esempio di codice mostra come utilizzare`describe-gateway-capability-configuration`.

AWS CLI

Come descrivere una funzionalità gateway

L'esempio `describe-gateway-capability-configuration` seguente descrive una funzionalità di origine OPC-UA.

```
aws iotsitewise describe-gateway-capability-configuration \
    --gateway-id a1b2c3d4-5678-90ab-cdef-1a1a1EXAMPLE \
    --capability-namespace "iotsitewise:opcuacollector:1"
```

Output:

```
{
    "gatewayId": "a1b2c3d4-5678-90ab-cdef-1a1a1EXAMPLE",
    "capabilityNamespace": "iotsitewise:opcuacollector:1",
    "capabilityConfiguration": "{\"sources\":[{\"name\":\"Wind Farm #1\",
\"endpoint\":{\"certificateTrust\":{\"type\":\"TrustAny\"},\"endpointUri
\":\"opc.tcp://203.0.113.0:49320\",\"securityPolicy\":\"BASIC256\",
\"messageSecurityMode\":\"SIGN_AND_ENCRYPT\",\"identityProvider\":
{\"type\":\"Username\",\"usernameSecretArn\":\"arn:aws:secretsmanager:us-
east-1:123456789012:secret:greengrass-factory1-auth-3QNDmM\"},\"nodeFilterRules\":
[]},\"measurementDataStreamPrefix\":\"\"}]}",
    "capabilitySyncStatus": "IN_SYNC"
}
```

Per ulteriori informazioni, consulta Configurazione delle fonti di dati nella Guida per l' SiteWise utente AWS IoT.

- Per i dettagli sull'API, consulta DescribeGatewayCapabilityConfiguration AWS CLICommand Reference.

`describe-gateway`

Il seguente esempio di codice mostra come utilizzare`describe-gateway`.

AWS CLI

Come descrivere un gateway

L'esempio `describe-gateway` seguente descrive un gateway.

```
aws iotsitewise describe-gateway \
    --gateway-id a1b2c3d4-5678-90ab-cdef-1a1a1EXAMPLE
```

Output:

```
{
    "gatewayId": "a1b2c3d4-5678-90ab-cdef-1a1a1EXAMPLE",
    "gatewayName": "ExampleCorpGateway",
    "gatewayArn": "arn:aws:iotsitewise:us-west-2:123456789012:gateway/
a1b2c3d4-5678-90ab-cdef-1a1a1EXAMPLE",
    "gatewayPlatform": {
        "greengrass": {
            "groupArn": "arn:aws:greengrass:us-west-2:123456789012:/greengrass/
groups/a1b2c3d4-5678-90ab-cdef-1b1b1EXAMPLE"
        }
    },
    "gatewayCapabilitySummaries": [
        {
            "capabilityNamespace": "iotsitewise:opcuacollector:1",
            "capabilitySyncStatus": "IN_SYNC"
        }
    ],
    "creationDate": 1588369971.457,
    "lastUpdateDate": 1588369971.457
}
```

Per ulteriori informazioni, consulta Ingestione di dati utilizzando un gateway nella AWS IoT SiteWise User Guide.

- Per i dettagli sull'API, consulta AWS CLI Command DescribeGatewayReference.

`describe-logging-options`

Il seguente esempio di codice mostra come utilizzare`describe-logging-options`.

AWS CLI

Per recuperare le opzioni di SiteWise registrazione AWS IoT correnti

L'`describe-logging-options`esempio seguente recupera le opzioni di SiteWise registrazione AWS IoT correnti per il tuo AWS account nella regione corrente.

```
aws iotsitewise describe-logging-options
```

Output:

```
{
    "loggingOptions": {
        "level": "INFO"
    }
}
```

Per ulteriori informazioni, consulta Monitoring AWS IoT SiteWise with Amazon CloudWatch Logs nella AWS IoT SiteWise User Guide.

- Per i dettagli sull'API, consulta DescribeLoggingOptions AWS CLICommand Reference.

`describe-portal`

Il seguente esempio di codice mostra come utilizzare`describe-portal`.

AWS CLI

Come descrivere un portale

L'esempio `describe-portal` seguente descrive un portale web per una società che gestisce parchi eolici.

```
aws iotsitewise describe-portal \
    --portal-id a1b2c3d4-5678-90ab-cdef-aaaaaEXAMPLE
```

Output:

```
{
    "portalId": "a1b2c3d4-5678-90ab-cdef-aaaaaEXAMPLE",
```

```
    "portalArn": "arn:aws:iotsitewise:us-west-2:123456789012:portal/
a1b2c3d4-5678-90ab-cdef-aaaaaEXAMPLE",
    "portalName": "WindFarmPortal",
    "portalDescription": "A portal that contains wind farm projects for Example
 Corp.",
    "portalClientId": "E-a1b2c3d4e5f6_a1b2c3d4e5f6EXAMPLE",
    "portalStartUrl": "https://a1b2c3d4-5678-90ab-cdef-
aaaaaEXAMPLE.app.iotsitewise.aws",
    "portalContactEmail": "support@example.com",
    "portalStatus": {
        "state": "ACTIVE"
    },
    "portalCreationDate": "2020-02-04T23:01:52.90248068Z",
    "portalLastUpdateDate": "2020-02-04T23:01:52.90248078Z",
    "roleArn": "arn:aws:iam::123456789012:role/MySiteWiseMonitorServiceRole"
}
```

Per ulteriori informazioni, consulta Amministrare i portali nella Guida per SiteWise l'utente AWS IoT.

- Per i dettagli sull'API, consulta AWS CLI Command DescribePortalReference.

`describe-project`

Il seguente esempio di codice mostra come utilizzare`describe-project`.

AWS CLI

Come descrivere un progetto

L'esempio `describe-project` seguente descrive un progetto di parco eolico.

```
aws iotsitewise describe-project \
    --project-id a1b2c3d4-5678-90ab-cdef-eeeeeEXAMPLE
```

Output:

```
{
    "projectId": "a1b2c3d4-5678-90ab-cdef-eeeeeEXAMPLE",
    "projectArn": "arn:aws:iotsitewise:us-west-2:123456789012:project/
a1b2c3d4-5678-90ab-cdef-eeeeeEXAMPLE",
    "projectName": "Wind Farm 1",
```

```
    "portalId": "a1b2c3d4-5678-90ab-cdef-aaaaaEXAMPLE",
    "projectDescription": "Contains asset visualizations for Wind Farm #1 for
 Example Corp.",
    "projectCreationDate": "2020-02-20T21:58:43.362246001Z",
    "projectLastUpdateDate": "2020-02-20T21:58:43.362246095Z"
}
```

Per ulteriori informazioni, consulta Visualizzazione dei dettagli del progetto nella Guida all'applicazione AWS IoT SiteWise Monitor.

- Per i dettagli sull'API, consulta DescribeProject AWS CLICommand Reference.

`disassociate-assets`

Il seguente esempio di codice mostra come utilizzare`disassociate-assets`.

AWS CLI

Come dissociare un asset secondario da un asset principale

L'esempio `disassociate-assets` seguente disassocia un asset turbina eolica da un asset di tipo parco eolico.

```
aws iotsitewise disassociate-assets \
    --asset-id a1b2c3d4-5678-90ab-cdef-44444EXAMPLE \
    --hierarchy-id a1b2c3d4-5678-90ab-cdef-77777EXAMPLE \
    --child-asset-id a1b2c3d4-5678-90ab-cdef-33333EXAMPLE
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Associating assets nella AWS IoT SiteWise User Guide.

- Per i dettagli sull'API, consulta DisassociateAssets AWS CLICommand Reference.

`get-asset-property-aggregates`

Il seguente esempio di codice mostra come utilizzare`get-asset-property-aggregates`.

AWS CLI

Come recuperare la media aggregata e i valori di conteggio di una proprietà di un asset

L'esempio `get-asset-property-aggregates` seguente recupera la potenza totale media di un asset turbina eolica e il conteggio dei punti dati della potenza totale per un periodo di 1 ora.

```
aws iotsitewise get-asset-property-aggregates \
    --asset-id a1b2c3d4-5678-90ab-cdef-33333EXAMPLE \
    --property-id a1b2c3d4-5678-90ab-cdef-66666EXAMPLE \
    --start-date 1580849400 \
    --end-date 1580853000 \
    --aggregate-types AVERAGE COUNT \
    --resolution 1h
```

Output:

```
{
    "aggregatedValues": [
        {
            "timestamp": 1580850000.0,
            "quality": "GOOD",
            "value": {
                "average": 8723.46538886233,
                "count": 12.0
            }
        }
    ]
}
```

Per ulteriori informazioni, consulta Interrogazione degli aggregati di proprietà degli asset nella AWS IoT SiteWise User Guide.

- Per i dettagli sull'API, consulta AWS CLI Command GetAssetPropertyAggregatesReference.

`get-asset-property-value-history`

Il seguente esempio di codice mostra come utilizzare`get-asset-property-value-history`.

AWS CLI

Come recuperare i valori storici di una proprietà di un asset

L'esempio `get-asset-property-value-history` seguente recupera i valori di potenza totale di un asset turbina eolica per un periodo di 20 minuti.

```
aws iotsitewise get-asset-property-value-history \
    --asset-id a1b2c3d4-5678-90ab-cdef-33333EXAMPLE \
    --property-id a1b2c3d4-5678-90ab-cdef-66666EXAMPLE \
    --start-date 1580851800 \
    --end-date 1580853000
```

Output:

```
{
    "assetPropertyValueHistory": [
        {
            "value": {
                "doubleValue": 7217.787046814844
            },
            "timestamp": {
                "timeInSeconds": 1580852100,
                "offsetInNanos": 0
            },
            "quality": "GOOD"
        },
        {
            "value": {
                "doubleValue": 6941.242811875451
            },
            "timestamp": {
                "timeInSeconds": 1580852400,
                "offsetInNanos": 0
            },
            "quality": "GOOD"
        },
        {
            "value": {
                "doubleValue": 6976.797662266717
            },
            "timestamp": {
                "timeInSeconds": 1580852700,
                "offsetInNanos": 0
            },
            "quality": "GOOD"
        },
        {
            "value": {
                "doubleValue": 6890.8677520453875
```

```
            },
            "timestamp": {
                "timeInSeconds": 1580853000,
                "offsetInNanos": 0
            },
            "quality": "GOOD"
        }
    ]
}
```

Per ulteriori informazioni, consulta Interrogazione dei valori storici delle proprietà degli asset nella AWS IoT SiteWise User Guide.

- Per i dettagli sull'API, consulta GetAssetPropertyValueHistory AWS CLICommand Reference.

`get-asset-property-value`

Il seguente esempio di codice mostra come utilizzare`get-asset-property-value`.

AWS CLI

Come recuperare il valore corrente di una proprietà di asset

L'esempio `get-asset-property-value` seguente recupera la potenza totale corrente di un asset turbina eolica.

```
aws iotsitewise get-asset-property-value \
    --asset-id a1b2c3d4-5678-90ab-cdef-33333EXAMPLE \
    --property-id a1b2c3d4-5678-90ab-cdef-66666EXAMPLE
```

Output:

```
{
    "propertyValue": {
        "value": {
            "doubleValue": 6890.8677520453875
        },
        "timestamp": {
            "timeInSeconds": 1580853000,
            "offsetInNanos": 0
        },
        "quality": "GOOD"
    }
```

```
}
```

Per ulteriori informazioni, consulta Interrogazione dei valori delle proprietà degli asset correnti nella AWS IoT SiteWise User Guide.

- Per i dettagli sull'API, consulta GetAssetPropertyValue AWS CLICommand Reference.

`list-access-policies`

Il seguente esempio di codice mostra come utilizzare`list-access-policies`.

AWS CLI

Come elencare tutte le policy di accesso

L'esempio `list-access-policies` seguente elenca tutte le policy di accesso per un utente che è amministratore del portale.

```
aws iotsitewise list-access-policies \
    --identity-type USER \
    --identity-id a1b2c3d4e5-a1b2c3d4-5678-90ab-cdef-bbbbbEXAMPLE
```

Output:

```
{
    "accessPolicySummaries": [
        {
            "id": "a1b2c3d4-5678-90ab-cdef-cccccEXAMPLE",
            "identity": {
                "user": {
                    "id": "a1b2c3d4e5-a1b2c3d4-5678-90ab-cdef-bbbbbEXAMPLE"
                }
            },
            "resource": {
                "portal": {
                    "id": "a1b2c3d4-5678-90ab-cdef-aaaaaEXAMPLE"
                }
            },
            "permission": "ADMINISTRATOR"
        }
    ]
}
```

Per ulteriori informazioni, consulta Amministrare i portali nella Guida per SiteWise l'utente AWS IoT.

- Per i dettagli sull'API, consulta AWS CLI Command ListAccessPoliciesReference.

`list-asset-models`

Il seguente esempio di codice mostra come utilizzare`list-asset-models`.

AWS CLI

Come elencare tutti i modelli di asset

L'`list-asset-models`esempio seguente elenca tutti i modelli di asset definiti nel vostro AWS account nella regione corrente.

```
aws iotsitewise list-asset-models
```

Output:

```
{
    "assetModelSummaries": [
        {
            "id": "a1b2c3d4-5678-90ab-cdef-22222EXAMPLE",
            "arn": "arn:aws:iotsitewise:us-west-2:123456789012:asset-model/
a1b2c3d4-5678-90ab-cdef-22222EXAMPLE",
            "name": "Wind Farm Model",
            "description": "Represents a wind farm that comprises many wind
 turbines",
            "creationDate": 1575671284.0,
            "lastUpdateDate": 1575671988.0,
            "status": {
                "state": "ACTIVE"
            }
        },
        {
            "id": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
            "arn": "arn:aws:iotsitewise:us-west-2:123456789012:asset-model/
a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
            "name": "Wind Turbine Model",
            "description": "Represents a wind turbine manufactured by Example Corp",
            "creationDate": 1575671207.0,
```

```
            "lastUpdateDate": 1575686273.0,
            "status": {
                "state": "ACTIVE"
            }
        }
    ]
}
```

Per ulteriori informazioni, consulta Elenco di tutti i modelli di asset nella AWS IoT SiteWise User Guide.

- Per i dettagli sull'API, consulta ListAssetModels AWS CLICommand Reference.

`list-assets`

Il seguente esempio di codice mostra come utilizzare`list-assets`.

AWS CLI

Esempio 1: come elencare tutti gli asset di primo livello

L'`list-assets`esempio seguente elenca tutte le risorse che sono di primo livello nell'albero della gerarchia delle risorse e definite nel tuo AWS account nella regione corrente.

```
aws iotsitewise list-assets \
    --filter TOP_LEVEL
```

Output:

```
{
    "assetSummaries": [
        {
            "id": "a1b2c3d4-5678-90ab-cdef-44444EXAMPLE",
            "arn": "arn:aws:iotsitewise:us-west-2:123456789012:asset/
a1b2c3d4-5678-90ab-cdef-44444EXAMPLE",
            "name": "Wind Farm 1",
            "assetModelId": "a1b2c3d4-5678-90ab-cdef-22222EXAMPLE",
            "creationDate": 1575672453.0,
            "lastUpdateDate": 1575672453.0,
            "status": {
                "state": "ACTIVE"
            },
```

```
            "hierarchies": [
                {
                    "id": "a1b2c3d4-5678-90ab-cdef-77777EXAMPLE",
                    "name": "Wind Turbines"
                }
            ]
        }
    ]
}
```

Per ulteriori informazioni, consulta Elencare gli asset nella AWS IoT SiteWise User Guide.

Esempio 2: come elencare tutti gli asset in base a un modello di asset

L'`list-assets`esempio seguente elenca tutti gli asset in base a un modello di asset e definiti nel tuo AWS account nella regione corrente.

```
aws iotsitewise list-assets \
    --asset-model-id a1b2c3d4-5678-90ab-cdef-11111EXAMPLE
```

Output:

```
{
    "assetSummaries": [
        {
            "id": "a1b2c3d4-5678-90ab-cdef-33333EXAMPLE",
            "arn": "arn:aws:iotsitewise:us-west-2:123456789012:asset/
a1b2c3d4-5678-90ab-cdef-33333EXAMPLE",
            "name": "Wind Turbine 1",
            "assetModelId": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
            "creationDate": 1575671550.0,
            "lastUpdateDate": 1575686308.0,
            "status": {
                "state": "ACTIVE"
            },
            "hierarchies": []
        }
    ]
}
```

Per ulteriori informazioni, consulta Elencare gli asset nella AWS IoT SiteWise User Guide.

- Per i dettagli sull'API, consulta ListAssets AWS CLICommand Reference.

`list-associated-assets`

Il seguente esempio di codice mostra come utilizzare`list-associated-assets`.

AWS CLI

Come elencare tutti gli asset associati a un asset in una gerarchia specifica

L'esempio `list-associated-assets` seguente elenca tutti gli asset turbina eolica associati all'asset parco eolico specificato.

```
aws iotsitewise list-associated-assets \
    --asset-id a1b2c3d4-5678-90ab-cdef-44444EXAMPLE \
    --hierarchy-id a1b2c3d4-5678-90ab-cdef-77777EXAMPLE
```

Output:

```
{
    "assetSummaries": [
        {
            "id": "a1b2c3d4-5678-90ab-cdef-33333EXAMPLE",
            "arn": "arn:aws:iotsitewise:us-west-2:123456789012:asset/
a1b2c3d4-5678-90ab-cdef-33333EXAMPLE",
            "name": "Wind Turbine 1",
            "assetModelId": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
            "creationDate": 1575671550.0,
            "lastUpdateDate": 1575686308.0,
            "status": {
                "state": "ACTIVE"
            },
            "hierarchies": []
        }
    ]
}
```

Per ulteriori informazioni, consulta Elencare gli asset associati a un asset specifico nella AWS IoT SiteWise User Guide.

- Per i dettagli sull'API, consulta ListAssociatedAssets AWS CLICommand Reference.

`list-dashboards`

Il seguente esempio di codice mostra come utilizzare`list-dashboards`.

AWS CLI

Come elencare tutte le dashboard di un progetto

L'esempio `list-dashboards` seguente elenca tutte le dashboard definite in un progetto.

```
aws iotsitewise list-dashboards \
    --project-id a1b2c3d4-5678-90ab-cdef-eeeeeEXAMPLE
```

Output:

```
{
    "dashboardSummaries": [
        {
            "id": "a1b2c3d4-5678-90ab-cdef-fffffEXAMPLE",
            "name": "Wind Farm",
            "creationDate": "2020-05-01T20:32:12.228476348Z",
            "lastUpdateDate": "2020-05-01T20:32:12.228476348Z"
        }
    ]
}
```

Per ulteriori informazioni, consulta Visualizzazione dei dashboard nella Guida all'applicazione AWS IoT SiteWise Monitor.

- Per i dettagli sull'API, consulta ListDashboards AWS CLICommand Reference.

`list-gateways`

Il seguente esempio di codice mostra come utilizzare`list-gateways`.

AWS CLI

Come elencare tutti i gateway

L'`list-gateways`esempio seguente elenca tutti i gateway definiti nell' AWS account nella regione corrente.

```
aws iotsitewise list-gateways
```

Output:

```
{
    "gatewaySummaries": [
        {
            "gatewayId": "a1b2c3d4-5678-90ab-cdef-1a1a1EXAMPLE",
            "gatewayName": "ExampleCorpGateway",
            "gatewayCapabilitySummaries": [
                {
                    "capabilityNamespace": "iotsitewise:opcuacollector:1",
                    "capabilitySyncStatus": "IN_SYNC"
                }
            ],
            "creationDate": 1588369971.457,
            "lastUpdateDate": 1588369971.457
        }
    ]
}
```

Per ulteriori informazioni, consulta Ingestione di dati utilizzando un gateway nella AWS IoT SiteWise User Guide.

- Per i dettagli sull'API, consulta AWS CLI Command ListGatewaysReference.

`list-portals`

Il seguente esempio di codice mostra come utilizzare`list-portals`.

AWS CLI

Come elencare tutti i portali

L'`list-portals`esempio seguente elenca tutti i portali definiti nel tuo AWS account nella regione corrente.

```
aws iotsitewise list-portals
```

Output:

```
{
    "portalSummaries": [
        {
            "id": "a1b2c3d4-5678-90ab-cdef-aaaaaEXAMPLE",
            "name": "WindFarmPortal",
```

```
            "description": "A portal that contains wind farm projects for Example
 Corp.",
            "startUrl": "https://a1b2c3d4-5678-90ab-cdef-
aaaaaEXAMPLE.app.iotsitewise.aws",
            "creationDate": "2020-02-04T23:01:52.90248068Z",
            "lastUpdateDate": "2020-02-04T23:01:52.90248078Z",
            "roleArn": "arn:aws:iam::123456789012:role/service-role/
MySiteWiseMonitorServiceRole"
        }
    ]
}
```

Per ulteriori informazioni, consulta Amministrare i portali nella Guida per SiteWise l'utente AWS IoT.

- Per i dettagli sull'API, consulta AWS CLI Command ListPortalsReference.

`list-project-assets`

Il seguente esempio di codice mostra come utilizzare`list-project-assets`.

AWS CLI

Come elencare tutti gli asset associati a un progetto

L'esempio `list-project-assets` seguente elenca tutti gli asset associati a un progetto di parco eolico.

```
aws iotsitewise list-projects \
    --project-id a1b2c3d4-5678-90ab-cdef-eeeeeEXAMPLE
```

Output:

```
{
    "assetIds": [
        "a1b2c3d4-5678-90ab-cdef-44444EXAMPLE"
    ]
}
```

Per ulteriori informazioni, consulta Aggiungere risorse ai progetti nella Guida all'applicazione AWS IoT SiteWise Monitor.

- Per i dettagli sull'API, consulta ListProjectAssets AWS CLICommand Reference.

`list-projects`

Il seguente esempio di codice mostra come utilizzare`list-projects`.

AWS CLI

Come elencare tutti i progetti in un portale

L'esempio `list-projects` seguente elenca tutti i progetti definiti in un portale.

```
aws iotsitewise list-projects \
    --portal-id a1b2c3d4-5678-90ab-cdef-aaaaaEXAMPLE
```

Output:

```
{
    "projectSummaries": [
        {
            "id": "a1b2c3d4-5678-90ab-cdef-eeeeeEXAMPLE",
            "name": "Wind Farm 1",
            "description": "Contains asset visualizations for Wind Farm #1 for
 Example Corp.",
            "creationDate": "2020-02-20T21:58:43.362246001Z",
            "lastUpdateDate": "2020-02-20T21:58:43.362246095Z"
        }
    ]
}
```

Per ulteriori informazioni, consulta Visualizzazione dei dettagli del progetto nella Guida all'applicazione AWS IoT SiteWise Monitor.

- Per i dettagli sull'API, consulta ListProjects AWS CLICommand Reference.

`list-tags-for-resource`

Il seguente esempio di codice mostra come utilizzare`list-tags-for-resource`.

AWS CLI

Come elencare tutti i tag per una risorsa

L'esempio `list-tags-for-resource` seguente elenca tutti i tag relativi a un asset turbina eolica.

```
aws iotsitewise list-tags-for-resource \
    --resource-arn arn:aws:iotsitewise:us-west-2:123456789012:asset/
a1b2c3d4-5678-90ab-cdef-33333EXAMPLE
```

Output:

```
{
    "tags": {
        "Owner": "richard-roe"
    }
}
```

Per ulteriori informazioni, consulta Tagging your resources nella AWS IoT SiteWise User Guide.

- Per i dettagli sull'API, consulta ListTagsForResource AWS CLICommand Reference.

`put-logging-options`

Il seguente esempio di codice mostra come utilizzare`put-logging-options`.

AWS CLI

Come specificare il livello di registrazione di log

L'`put-logging-options`esempio seguente abilita la registrazione dei `INFO` livelli in AWS IoT SiteWise. Altri livelli includono `DEBUG` e `OFF`.

```
aws iotsitewise put-logging-options \
    --logging-options level=INFO
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Monitoring AWS IoT SiteWise with Amazon CloudWatch Logs nella AWS IoT SiteWise User Guide.

- Per i dettagli sull'API, consulta PutLoggingOptions AWS CLICommand Reference.

`tag-resource`

Il seguente esempio di codice mostra come utilizzare`tag-resource`.

AWS CLI

Come aggiungere un tag a una risorsa

L'esempio `tag-resource` seguente aggiunge un tag owner a un asset turbina eolica. Ciò consente di controllare l'accesso all'asset in base al proprietario.

```
aws iotsitewise tag-resource \
    --resource-arn arn:aws:iotsitewise:us-west-2:123456789012:asset/
a1b2c3d4-5678-90ab-cdef-33333EXAMPLE \
    --tags Owner=richard-roe
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Tagging your resources nella AWS IoT SiteWise User Guide.

- Per i dettagli sull'API, consulta TagResource AWS CLICommand Reference.

`untag-resource`

Il seguente esempio di codice mostra come utilizzare`untag-resource`.

AWS CLI

Come rimuovere un tag da una risorsa

L'esempio `untag-resource` seguente rimuove un tag owner da un asset turbina eolica.

```
aws iotsitewise untag-resource \
    --resource-arn arn:aws:iotsitewise:us-west-2:123456789012:asset/
a1b2c3d4-5678-90ab-cdef-33333EXAMPLE \
    --tag-keys Owner
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Tagging your resources nella AWS IoT SiteWise User Guide.

- Per i dettagli sull'API, consulta UntagResource AWS CLICommand Reference.

`update-access-policy`

Il seguente esempio di codice mostra come utilizzare`update-access-policy`.

AWS CLI

Come concedere a un visualizzatore di un progetto la proprietà del progetto

L'esempio `update-access-policy` seguente aggiorna una policy di accesso che concede a un visualizzatore di un progetto la proprietà del progetto.

```
aws iotsitewise update-access-policy \
    --access-policy-id a1b2c3d4-5678-90ab-cdef-dddddEXAMPLE \
    --cli-input-json file://update-project-viewer-access-policy.json
```

Contenuto di `update-project-viewer-access-policy.json`:

```
{
    "accessPolicyIdentity": {
        "user": {
            "id": "a1b2c3d4e5-a1b2c3d4-5678-90ab-cdef-bbbbbEXAMPLE"
        }
    },
    "accessPolicyPermission": "ADMINISTRATOR",
    "accessPolicyResource": {
        "project": {
            "id": "a1b2c3d4-5678-90ab-cdef-eeeeeEXAMPLE"
        }
    }
}
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Assegnazione dei proprietari dei progetti nella Guida all'applicazione AWS IoT SiteWise Monitor.

- Per i dettagli sull'API, consulta UpdateAccessPolicy AWS CLICommand Reference.

`update-asset-model`

Il seguente esempio di codice mostra come utilizzare`update-asset-model`.

AWS CLI

Come aggiornare un modello di asset

L'esempio `update-asset-model` seguente descrive un modello di asset di tipo parco eolico. Questo esempio include il modello esistente IDs e le definizioni, poiché `update-asset-model` sovrascrive il modello esistente con il nuovo modello.

```
aws iotsitewise update-asset-model \
    --cli-input-json file://update-wind-farm-model.json
```

Contenuto di `update-wind-farm-model.json`:

```
{
    "assetModelName": "Wind Farm Model",
    "assetModelDescription": "Represents a wind farm that comprises many wind
 turbines",
    "assetModelProperties": [
        {
            "id": "a1b2c3d4-5678-90ab-cdef-88888EXAMPLE",
            "name": "Region",
            "dataType": "STRING",
            "type": {
                "attribute": {}
            }
        },
        {
            "id": "a1b2c3d4-5678-90ab-cdef-99999EXAMPLE",
            "name": "Total Generated Power",
            "dataType": "DOUBLE",
            "unit": "kW",
            "type": {
                "metric": {
                    "expression": "sum(power)",
                    "variables": [
                        {
                            "name": "power",
                            "value": {
                                "hierarchyId": "a1b2c3d4-5678-90ab-
cdef-77777EXAMPLE",
                                "propertyId": "a1b2c3d4-5678-90ab-cdef-66666EXAMPLE"
                            }
                        }
                    ],
                    "window": {
                        "tumbling": {
                            "interval": "1h"
```

```
                        }
                    }
                }
            }
        }
    ],
    "assetModelHierarchies": [
        {
            "id": "a1b2c3d4-5678-90ab-cdef-77777EXAMPLE",
            "name": "Wind Turbines",
            "childAssetModelId": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE"
        }
    ]
}
```

Output:

```
{
    "assetModelId": "a1b2c3d4-5678-90ab-cdef-22222EXAMPLE",
    "assetModelArn": "arn:aws:iotsitewise:us-west-2:123456789012:asset-model/
a1b2c3d4-5678-90ab-cdef-22222EXAMPLE",
    "assetModelStatus": {
        "state": "CREATING"
    }
}
```

Per ulteriori informazioni, consulta Updating asset models nella AWS IoT SiteWise User Guide.

- Per i dettagli sull'API, consulta UpdateAssetModel AWS CLICommand Reference.

`update-asset-property`

Il seguente esempio di codice mostra come utilizzare`update-asset-property`.

AWS CLI

Esempio 1: come aggiornare l'alias di una proprietà di asset

L'esempio `update-asset-property` seguente aggiorna l'alias della proprietà di alimentazione di un asset turbina eolica.

```
aws iotsitewise update-asset-property \
    --asset-id a1b2c3d4-5678-90ab-cdef-33333EXAMPLE \
```

```
    --property-id a1b2c3d4-5678-90ab-cdef-55555EXAMPLE \
    --property-alias "/examplecorp/windfarm/1/turbine/1/power" \
    --property-notification-state DISABLED
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Mappatura dei flussi di dati industriali alle proprietà degli asset nella AWS IoT SiteWise User Guide.

Esempio 2: come abilitare le notifiche delle proprietà di asset

L'esempio `update-asset-property` seguente abilita le notifiche di aggiornamento delle proprietà di alimentazione di un asset turbina eolica. Gli aggiornamenti dei valori delle proprietà vengono pubblicati nell'argomento MQTT `$aws/sitewise/asset-models/<assetModelId>/assets/<assetId>/properties/<propertyId>`, dove ogni ID viene sostituito dalla proprietà, dall'asset e dall'ID del modello della proprietà dell'asset.

```
aws iotsitewise update-asset-property \
    --asset-id a1b2c3d4-5678-90ab-cdef-33333EXAMPLE \
    --property-id a1b2c3d4-5678-90ab-cdef-66666EXAMPLE \
    --property-notification-state ENABLED \
    --property-alias "/examplecorp/windfarm/1/turbine/1/power"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Interagire con altri servizi nella Guida per l' SiteWise utente AWS IoT.

- Per i dettagli sull'API, consulta UpdateAssetProperty AWS CLICommand Reference.

`update-asset`

Il seguente esempio di codice mostra come utilizzare`update-asset`.

AWS CLI

Come aggiornare il nome di un asset

L'esempio `update-asset` seguente aggiorna il nome dell'asset turbina eolica.

```
aws iotsitewise update-asset \
    --asset-id a1b2c3d4-5678-90ab-cdef-33333EXAMPLE \
```

```
    --asset-name "Wind Turbine 2"
```

Output:

```
{
    "assetStatus": {
        "state": "UPDATING"
    }
}
```

Per ulteriori informazioni, consulta Updating assets nella AWS IoT SiteWise User Guide.

- Per i dettagli sull'API, consulta UpdateAsset AWS CLICommand Reference.

`update-dashboard`

Il seguente esempio di codice mostra come utilizzare`update-dashboard`.

AWS CLI

Come aggiornare una dashboard

L'esempio `update-dashboard` seguente modifica il titolo di un grafico a linee di una dashboard che mostra la potenza totale generata per un parco eolico.

```
aws iotsitewise update-dashboard \
    --project-id a1b2c3d4-5678-90ab-cdef-fffffEXAMPLE \
    --dashboard-name "Wind Farm" \
    --dashboard-definition file://update-wind-farm-dashboard.json
```

Contenuto di `update-wind-farm-dashboard.json`:

```
{
    "widgets": [
        {
            "type": "monitor-line-chart",
            "title": "Total Generated Power",
            "x": 0,
            "y": 0,
            "height": 3,
            "width": 3,
            "metrics": [
```

```
                {
                    "label": "Power",
                    "type": "iotsitewise",
                    "assetId": "a1b2c3d4-5678-90ab-cdef-44444EXAMPLE",
                    "propertyId": "a1b2c3d4-5678-90ab-cdef-99999EXAMPLE"
                }
            ]
        }
    ]
}
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Creazione di dashboard (CLI) nella Guida per l'utente AWS SiteWise IoT.

- Per i dettagli sull'API, consulta AWS CLI Command UpdateDashboardReference.

`update-gateway-capability-configuration`

Il seguente esempio di codice mostra come utilizzareu`pdate-gateway-capability-configuration`.

AWS CLI

Come aggiornare la funzionalità di un gateway

L'esempio `update-gateway-capability-configuration` seguente configura un'origine OPC-UA con le seguenti proprietà:

Considera attendibile qualsiasi certificato.Utilizza l'algoritmo Basic256 per proteggere i messaggi.Utilizza la SignAndEncrypt modalità per proteggere le connessioni.Utilizza le credenziali di autenticazione memorizzate in un segreto di Secrets Manager. AWS

```
aws iotsitewise update-gateway-capability-configuration \
    --gateway-id a1b2c3d4-5678-90ab-cdef-1a1a1EXAMPLE \
    --capability-namespace "iotsitewise:opcuacollector:1" \
    --capability-configuration file://opc-ua-capability-configuration.json
```

Contenuto di `opc-ua-capability-configuration.json`:

```
{
```

```
    "sources": [
        {
            "name": "Wind Farm #1",
            "endpoint": {
                "certificateTrust": {
                    "type": "TrustAny"
                },
                "endpointUri": "opc.tcp://203.0.113.0:49320",
                "securityPolicy": "BASIC256",
                "messageSecurityMode": "SIGN_AND_ENCRYPT",
                "identityProvider": {
                    "type": "Username",
                    "usernameSecretArn": "arn:aws:secretsmanager:us-
west-2:123456789012:secret:greengrass-windfarm1-auth-1ABCDE"
                },
                "nodeFilterRules": []
            },
            "measurementDataStreamPrefix": ""
        }
    ]
}
```

Output:

```
{
    "capabilityNamespace": "iotsitewise:opcuacollector:1",
    "capabilitySyncStatus": "OUT_OF_SYNC"
}
```

Per ulteriori informazioni, consulta Configurazione delle fonti di dati nella Guida per l' SiteWise utente AWS IoT.

- Per i dettagli sull'API, consulta UpdateGatewayCapabilityConfiguration AWS CLICommand Reference.

`update-gateway`

Il seguente esempio di codice mostra come utilizzare`update-gateway`.

AWS CLI

Come aggiornare il nome di un gateway

L'esempio `update-gateway` seguente aggiorna il nome di un gateway.

```
aws iotsitewise update-gateway \
    --gateway-id a1b2c3d4-5678-90ab-cdef-1a1a1EXAMPLE \
    --gateway-name ExampleCorpGateway1
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Ingestione di dati utilizzando un gateway nella AWS IoT SiteWise User Guide.

- Per i dettagli sull'API, consulta AWS CLI Command UpdateGatewayReference.

`update-portal`

Il seguente esempio di codice mostra come utilizzare`update-portal`.

AWS CLI

Come aggiornare i dettagli di un portale

L'esempio `update-portal` seguente aggiorna un portale web per una società che gestisce parchi eolici.

```
aws iotsitewise update-portal \
    --portal-id a1b2c3d4-5678-90ab-cdef-aaaaaEXAMPLE \
    --portal-name WindFarmPortal \
    --portal-description "A portal that contains wind farm projects for Example
 Corp." \
    --portal-contact-email support@example.com \
    --role-arn arn:aws:iam::123456789012:role/MySiteWiseMonitorServiceRole
```

Output:

```
{
    "portalStatus": {
        "state": "UPDATING"
    }
}
```

Per ulteriori informazioni, consulta Amministrare i portali nella Guida per SiteWise l'utente AWS IoT.

- Per i dettagli sull'API, consulta AWS CLI Command UpdatePortalReference.

`update-project`

Il seguente esempio di codice mostra come utilizzareu`pdate-project`.

AWS CLI

Come aggiornare i dettagli di un progetto

L'esempio `update-project` seguente aggiorna un progetto di parco eolico.

```
aws iotsitewise update-project \
    --project-id a1b2c3d4-5678-90ab-cdef-eeeeeEXAMPLE \
    --project-name "Wind Farm 1" \
    --project-description "Contains asset visualizations for Wind Farm #1 for
 Example Corp."
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Modifica dei dettagli del progetto nella Guida all'applicazione AWS IoT SiteWise Monitor.

- Per i dettagli sull'API, consulta UpdateProject AWS CLICommand Reference.

AWS IoT Things Graph esempi utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with AWS IoT Things Graph.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`associate-entity-to-thing`

Il seguente esempio di codice mostra come utilizzare`associate-entity-to-thing`.

AWS CLI

Come associare un oggetto a un dispositivo

L'esempio `associate-entity-to-thing` seguente associa un oggetto a un dispositivo. L'esempio utilizza un dispositivo con sensore di movimento che si trova nel namespace pubblico.

```
aws iotthingsgraph associate-entity-to-thing \
    --thing-name "MotionSensorName" \
    --entity-id "urn:tdm:aws/examples:Device:HCSR501MotionSensor"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Creazione e caricamento di modelli nella Guida per l'utente di AWS IoT Things Graph.

- Per i dettagli sull'API, consulta AssociateEntityToThing AWS CLICommand Reference.

`create-flow-template`

Il seguente esempio di codice mostra come utilizzare`create-flow-template`.

AWS CLI

Come creare un flusso

L'esempio `create-flow-template` seguente crea un flusso (di lavoro). Il valore di `MyFlowDefinition` è il GraphQL che modella il flusso.

```
aws iotthingsgraph create-flow-template \
    --definition language=GRAPHQL,text="MyFlowDefinition"
```

Output:

```
{
    "summary": {
        "createdAt": 1559248067.545,
```

```
        "id": "urn:tdm:us-west-2/123456789012/default:Workflow:MyFlow",
        "revisionNumber": 1
    }
}
```

Per ulteriori informazioni, consulta Utilizzo dei flussi nella Guida per l'utente di AWS IoT Things Graph.

- Per i dettagli sull'API, consulta CreateFlowTemplate AWS CLICommand Reference.

`create-system-instance`

Il seguente esempio di codice mostra come utilizzare`create-system-instance`.

AWS CLI

Come creare un'istanza di sistema

L'esempio `create-system-instance` seguente crea un'istanza di sistema. Il valore di `MySystemInstanceDefinition` è il GraphQL che modella l'istanza di sistema.

```
aws iotthingsgraph create-system-instance -\
    -definition language=GRAPHQL,text="MySystemInstanceDefinition" \
    --target CLOUD \
    --flow-actions-role-arn myRoleARN
```

Output:

```
{
    "summary": {
        "id": "urn:tdm:us-west-2/123456789012/default:Deployment:Room218",
        "arn": "arn:aws:iotthingsgraph:us-west-2:123456789012:Deployment/default/
Room218",
        "status": "NOT_DEPLOYED",
        "target": "CLOUD",
        "createdAt": 1559249315.208,
        "updatedAt": 1559249315.208
    }
}
```

Per ulteriori informazioni, consulta Utilizzo di configurazioni di sistemi e flussi nella Guida per l'utente di AWS IoT Things Graph.

- Per i dettagli sull'API, consulta CreateSystemInstance AWS CLICommand Reference.

`create-system-template`

Il seguente esempio di codice mostra come utilizzare`create-system-template`.

AWS CLI

Come creare un sistema

L'esempio `create-system-template` seguente crea un sistema. Il valore di MySystemDefinition è il GraphQL che modella il sistema.

```
aws iotthingsgraph create-system-template \
    --definition language=GRAPHQL,text="MySystemDefinition"
```

Output:

```
{
    "summary": {
        "createdAt": 1559249776.254,
        "id": "urn:tdm:us-west-2/123456789012/default:System:MySystem",
        "arn": "arn:aws:iotthingsgraph:us-west-2:123456789012:System/default/
MySystem",
        "revisionNumber": 1
    }
}
```

Per ulteriori informazioni, consulta Creazione dei sistemi nella Guida per l'utente di AWS IoT Things Graph.

- Per i dettagli sull'API, consulta CreateSystemTemplate AWS CLICommand Reference.

`delete-flow-template`

Il seguente esempio di codice mostra come utilizzare`delete-flow-template`.

AWS CLI

Come eliminare un flusso

L'esempio `delete-flow-template` seguente elimina un flusso (di lavoro).

```
aws iotthingsgraph delete-flow-template \
    --id "urn:tdm:us-west-2/123456789012/default:Workflow:MyFlow"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Lifecycle Management for AWS IoT Things Graph Entities, Flows, Systems and Deployments nella AWS IoT Things Graph User Guide.

- Per i dettagli sull'API, consulta DeleteFlowTemplateCommand Reference.AWS CLI

`delete-namespace`

Il seguente esempio di codice mostra come utilizzare`delete-namespace`.

AWS CLI

Come eliminare un namespace

L'esempio `delete-namespace` seguente elimina un namespace.

```
aws iotthingsgraph delete-namespace
```

Output:

```
{
   "namespaceArn": "arn:aws:iotthingsgraph:us-west-2:123456789012",
   "namespaceName": "us-west-2/123456789012/default"
}
```

Per ulteriori informazioni, consulta Lifecycle Management for AWS IoT Things Graph Entities, Flows, Systems and Deployments nella AWS IoT Things Graph User Guide.

- Per i dettagli sull'API, consulta DeleteNamespaceCommand Reference.AWS CLI

`delete-system-instance`

Il seguente esempio di codice mostra come utilizzare`delete-system-instance`.

AWS CLI

Come eliminare un'istanza di sistema

L'esempio `delete-system-instance` seguente elimina un'istanza di sistema.

```
aws iotthingsgraph delete-system-instance \
    --id "urn:tdm:us-west-2/123456789012/default:Deployment:Room218"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Lifecycle Management for AWS IoT Things Graph Entities, Flows, Systems and Deployments nella AWS IoT Things Graph User Guide.

- Per i dettagli sull'API, consulta DeleteSystemInstanceCommand Reference.AWS CLI

`delete-system-template`

Il seguente esempio di codice mostra come utilizzare`delete-system-template`.

AWS CLI

Come eliminare un sistema

L'esempio `delete-system-template` seguente elimina un sistema.

```
aws iotthingsgraph delete-system-template \
    --id "urn:tdm:us-west-2/123456789012/default:System:MySystem"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Lifecycle Management for AWS IoT Things Graph Entities, Flows, Systems and Deployments nella AWS IoT Things Graph User Guide.

- Per i dettagli sull'API, consulta DeleteSystemTemplateCommand Reference.AWS CLI

`deploy-system-instance`

Il seguente esempio di codice mostra come utilizzare`deploy-system-instance`.

AWS CLI

Come implementare un'istanza di sistema

L'esempio `delete-system-template` seguente implementa un'istanza di sistema.

```
aws iotthingsgraph deploy-system-instance \
    --id "urn:tdm:us-west-2/123456789012/default:Deployment:Room218"
```

Output:

```
{
   "summary": {
      "arn": "arn:aws:iotthingsgraph:us-west-2:123456789012:Deployment:Room218",
      "createdAt": 1559249776.254,
      "id": "urn:tdm:us-west-2/123456789012/default:Deployment:Room218",
      "status": "DEPLOYED_IN_TARGET",
      "target": "CLOUD",
      "updatedAt": 1559249776.254
   }
}
```

Per ulteriori informazioni, consulta Utilizzo di configurazioni di sistemi e flussi nella Guida per l'utente di AWS IoT Things Graph.

- Per i dettagli sull'API, consulta DeploySystemInstance AWS CLICommand Reference.

`deprecate-flow-template`

Il seguente esempio di codice mostra come utilizzare`deprecate-flow-template`.

AWS CLI

Come deprecare un flusso

L'esempio `deprecate-flow-template` seguente rende obsoleto un flusso (di lavoro).

```
aws iotthingsgraph deprecate-flow-template \
    --id "urn:tdm:us-west-2/123456789012/default:Workflow:MyFlow"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Lifecycle Management for AWS IoT Things Graph Entities, Flows, Systems and Deployments nella AWS IoT Things Graph User Guide.

- Per i dettagli sull'API, consulta DeprecateFlowTemplateCommand Reference.AWS CLI

`deprecate-system-template`

Il seguente esempio di codice mostra come utilizzare`deprecate-system-template`.

AWS CLI

Come deprecare un sistema

L'esempio `deprecate-system-template` seguente rende obsoleto un sistema.

```
aws iotthingsgraph deprecate-system-template \
    --id "urn:tdm:us-west-2/123456789012/default:System:MySystem"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Lifecycle Management for AWS IoT Things Graph Entities, Flows, Systems and Deployments nella AWS IoT Things Graph User Guide.

- Per i dettagli sull'API, consulta DeprecateSystemTemplateCommand Reference.AWS CLI

`describe-namespace`

Il seguente esempio di codice mostra come utilizzare`describe-namespace`.

AWS CLI

Come ottenere una descrizione del namespace corrente

L'esempio `describe-namespace` seguente ottiene una descrizione del tuo namespace.

```
aws iotthingsgraph describe-namespace
```

Output:

```
{
    "namespaceName": "us-west-2/123456789012/default",
    "trackingNamespaceName": "aws",
    "trackingNamespaceVersion": 1,
    "namespaceVersion": 5
}
```

Per ulteriori informazioni, consulta Namespace nella Guida per l'utente di AWS IoT Things Graph.

- Per i dettagli sull'API, consulta DescribeNamespace AWS CLICommand Reference.

`dissociate-entity-from-thing`

Il seguente esempio di codice mostra come utilizzare`dissociate-entity-from-thing`.

AWS CLI

Come dissociare un oggetto da un dispositivo

L'esempio `dissociate-entity-from-thing` seguente dissocia un oggetto da un dispositivo.

```
aws iotthingsgraph dissociate-entity-from-thing \
    --thing-name "MotionSensorName" \
    --entity-type "DEVICE"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Creazione e caricamento di modelli nella Guida per l'utente di AWS IoT Things Graph.

- Per i dettagli sull'API, consulta DissociateEntityFromThing AWS CLICommand Reference.

`get-entities`

Il seguente esempio di codice mostra come utilizzare`get-entities`.

AWS CLI

Come ottenere le definizioni delle entità

L'esempio `get-entities` seguente ottiene una definizione per un modello di dispositivo.

```
aws iotthingsgraph get-entities \
    --ids "urn:tdm:aws/examples:DeviceModel:MotionSensor"
```

Output:

```
{
```

```
    "descriptions": [
        {
            "id": "urn:tdm:aws/examples:DeviceModel:MotionSensor",
            "type": "DEVICE_MODEL",
            "createdAt": 1559256190.599,
            "definition": {
                "language": "GRAPHQL",
                "text": "##\n# Specification of motion sensor devices interface.\n##
\ntype MotionSensor @deviceModel(id: \"urn:tdm:aws/examples:deviceModel:MotionSensor
\",\n        capability: \"urn:tdm:aws/examples:capability:MotionSensorCapability\")
 {ignore:void}"
            }
        }
    ]
}
```

Per ulteriori informazioni, consulta Creazione e caricamento di modelli nella Guida per l'utente di AWS IoT Things Graph.

- Per i dettagli sull'API, consulta GetEntities AWS CLICommand Reference.

`get-flow-template-revisions`

Il seguente esempio di codice mostra come utilizzare`get-flow-template-revisions`.

AWS CLI

Come ottenere informazioni sulle revisioni di un flusso

L'esempio `get-flow-template-revisions` seguente ottiene informazioni di revisione su un flusso (di lavoro).

```
aws iotthingsgraph get-flow-template-revisions \
    --id urn:tdm:us-west-2/123456789012/default:Workflow:MyFlow
```

Output:

```
{
    "summaries": [
        {
            "id": "urn:tdm:us-west-2/123456789012/default:Workflow:MyFlow",
            "revisionNumber": 1,
```

```
            "createdAt": 1559247540.292
        }
    ]
}
```

Per ulteriori informazioni, consulta Utilizzo dei flussi nella Guida per l'utente di AWS IoT Things Graph.

- Per i dettagli sull'API, consulta GetFlowTemplateRevisions AWS CLICommand Reference.

`get-flow-template`

Il seguente esempio di codice mostra come utilizzare`get-flow-template`.

AWS CLI

Come ottenere una definizione del flusso

L'esempio `get-flow-template` seguente ottiene una definizione per un flusso (di lavoro).

```
aws iotthingsgraph get-flow-template \
    --id "urn:tdm:us-west-2/123456789012/default:Workflow:MyFlow"
```

Output:

```
{
    "description": {
        "summary": {
            "id": "urn:tdm:us-west-2/123456789012/default:Workflow:MyFlow",
            "revisionNumber": 1,
            "createdAt": 1559247540.292
        },
        "definition": {
            "language": "GRAPHQL",
            "text": "{\nquery MyFlow($camera: string!, $screen: string!)
 @workflowType(id: \"urn:tdm:us-west-2/123456789012/default:Workflow:MyFlow\")
 @annotation(type: \"tgc:FlowEvent\", id: \"sledged790c1b2bcd949e09da0c9bfc077f79d
\", x: 1586, y: 653) @triggers(definition: \"{MotionSensor(description:
 \\\"\\\") @position(x: 1045, y: 635.6666564941406) {\\n  condition(expr:
 \\\"devices[name == \\\\\\\"motionSensor\\\\\\\"].events[name == \\\
\\\\"StateChanged\\\\\\\"].lastEvent\\\")\\n  action(expr: \\\"\\\")\
\n}}\") {\n  variables {\n    cameraResult @property(id: \"urn:tdm:aws/
```

```
examples:property:CameraStateProperty\")\n  }\n  steps {\n    step(name: \"Camera
\", outEvent: [\"sledged790c1b2bcd949e09da0c9bfc077f79d\"]) @position(x: 1377,
 y: 638.6666564941406) {\n      DeviceActivity(deviceModel: \"urn:tdm:aws/
examples:deviceModel:Camera\", out: \"cameraResult\", deviceId: \"${camera}\")
 {\n        capture\n      }\n    }\n    step(name: \"Screen\", inEvent:
 [\"sledged790c1b2bcd949e09da0c9bfc077f79d\"]) @position(x: 1675.6666870117188,
 y: 637.9999847412109) {\n      DeviceActivity(deviceModel: \"urn:tdm:aws/
examples:deviceModel:Screen\", deviceId: \"${screen}\") {\n        display(imageUrl:
 \"${cameraResult.lastClickedImage}\")\n      }\n    }\n  }\n}\n}"
        },
        "validatedNamespaceVersion": 5
    }
}
```

Per ulteriori informazioni, consulta Utilizzo dei flussi nella Guida per l'utente di AWS IoT Things Graph.

- Per i dettagli sull'API, consulta GetFlowTemplate AWS CLICommand Reference.

`get-namespace-deletion-status`

Il seguente esempio di codice mostra come utilizzare`get-namespace-deletion-status`.

AWS CLI

Come ottenere lo stato dell'attività di eliminazione del namespace

L'esempio `get-namespace-deletion-status` seguente ottiene lo stato dell'operazione di eliminazione del namespace.

```
aws iotthingsgraph get-namespace-deletion-status
```

Output:

```
{
   "namespaceArn": "arn:aws:iotthingsgraph:us-west-2:123456789012",
   "namespaceName": "us-west-2/123456789012/default"
   "status": "SUCCEEDED "
}
```

Per ulteriori informazioni, consulta Namespace nella Guida per l'utente di AWS IoT Things Graph.

- Per i dettagli sull'API, consulta GetNamespaceDeletionStatus AWS CLICommand Reference.

`get-system-instance`

Il seguente esempio di codice mostra come utilizzare`get-system-instance`.

AWS CLI

Come ottenere un'istanza di sistema

L'esempio `get-system-instance` seguente ottiene una definizione per un'istanza di sistema.

```
aws iotthingsgraph get-system-instance \
    --id "urn:tdm:us-west-2/123456789012/default:Deployment:Room218"
```

Output:

```
{
    "description": {
        "summary": {
            "id": "urn:tdm:us-west-2/123456789012/default:Deployment:Room218",
            "arn": "arn:aws:iotthingsgraph:us-west-2:123456789012:Deployment/
default/Room218",
            "status": "NOT_DEPLOYED",
            "target": "CLOUD",
            "createdAt": 1559249315.208,
            "updatedAt": 1559249315.208
        },
        "definition": {
            "language": "GRAPHQL",
            "text": "{\r\nquery Room218 @deployment(id: \"urn:tdm:us-
west-2/123456789012/default:Deployment:Room218\", systemId: \"urn:tdm:us-
west-2/123456789012/default:System:SecurityFlow\") {\r\n    motionSensor(deviceId:
 \"MotionSensorName\")\r\n    screen(deviceId: \"ScreenName\")\r\n
 camera(deviceId: \"CameraName\") \r\n    triggers {MotionEventTrigger(description:
 \"a trigger\") {  \r\n    condition(expr: \"devices[name ==
 'motionSensor'].events[name == 'StateChanged'].lastEvent\") \r\n    action(expr:
 \"ThingsGraph.startFlow('SecurityFlow', bindings[name == 'camera'].deviceId,
 bindings[name == 'screen'].deviceId)\")\r\n    }\r\n   }\r\n  }\r\n  }"
        },
        "metricsConfiguration": {
            "cloudMetricEnabled": false
        },
```

```
        "validatedNamespaceVersion": 5,
        "flowActionsRoleArn": "arn:aws:iam::123456789012:role/ThingsGraphRole"
    }
}
```

Per ulteriori informazioni, consulta Utilizzo di configurazioni di sistemi e flussi nella Guida per l'utente di AWS IoT Things Graph.

- Per i dettagli sull'API, consulta GetSystemInstance AWS CLICommand Reference.

`get-system-template-revisions`

Il seguente esempio di codice mostra come utilizzare`get-system-template-revisions`.

AWS CLI

Come ottenere informazioni su un sistema

L'esempio `get-system-template-revisions` seguente ottiene informazioni sulla revisione di un sistema.

```
aws iotthingsgraph get-system-template-revisions \
    --id "urn:tdm:us-west-2/123456789012/default:System:MySystem"
```

Output:

```
{
    "summaries": [
        {
            "id": "urn:tdm:us-west-2/123456789012/default:System:MySystem",
            "arn": "arn:aws:iotthingsgraph:us-west-2:123456789012:System/default/
MySystem",
            "revisionNumber": 1,
            "createdAt": 1559247540.656
        }
    ]
}
```

Per ulteriori informazioni, consulta Utilizzo di configurazioni di sistemi e flussi nella Guida per l'utente di AWS IoT Things Graph.

- Per i dettagli sull'API, consulta GetSystemTemplateRevisions AWS CLICommand Reference.

`get-system-template`

Il seguente esempio di codice mostra come utilizzare`get-system-template`.

AWS CLI

Come ottenere un sistema

L'esempio `get-system-template` seguente ottiene una definizione per un sistema.

```
aws iotthingsgraph get-system-template \
    --id "urn:tdm:us-west-2/123456789012/default:System:MySystem"
```

Output:

```
{
    "description": {
        "summary": {
            "id": "urn:tdm:us-west-2/123456789012/default:System:MySystem",
            "arn": "arn:aws:iotthingsgraph:us-west-2:123456789012:System/default/
MyFlow",
            "revisionNumber": 1,
            "createdAt": 1559247540.656
        },
        "definition": {
            "language": "GRAPHQL",
            "text": "{\ntype MySystem @systemType(id: \"urn:tdm:us-
west-2/123456789012/default:System:MySystem\", description: \"\") {\n  camera:
 Camera @thing(id: \"urn:tdm:aws/examples:deviceModel:Camera\")\n  screen:
 Screen @thing(id: \"urn:tdm:aws/examples:deviceModel:Screen\")\n  motionSensor:
 MotionSensor @thing(id: \"urn:tdm:aws/examples:deviceModel:MotionSensor
\")\n  MyFlow: MyFlow @workflow(id: \"urn:tdm:us-west-2/123456789012/
default:Workflow:MyFlow\")\n}\n}"
        },
        "validatedNamespaceVersion": 5
    }
}
```

Per ulteriori informazioni, consulta Utilizzo di configurazioni di sistemi e flussi nella Guida per l'utente di AWS IoT Things Graph.

- Per i dettagli sull'API, consulta GetSystemTemplate AWS CLICommand Reference.

`get-upload-status`

Il seguente esempio di codice mostra come utilizzare`get-upload-status`.

AWS CLI

Come ottenere lo stato del caricamento dell'entità

L'esempio `get-upload-status` seguente ottiene lo stato dell'operazione di caricamento dell'entità. Il valore di `MyUploadId` è il valore ID restituito dall'operazione `upload-entity-definitions`.

```
aws iotthingsgraph get-upload-status \
    --upload-id "MyUploadId"
```

Output:

```
{
    "namespaceName": "us-west-2/123456789012/default",
    "namespaceVersion": 5,
    "uploadId": "f6294f1e-b109-4bbe-9073-f451a2dda2da",
    "uploadStatus": "SUCCEEDED"
}
```

Per ulteriori informazioni, consulta Modellazione delle entità nella Guida per l'utente di AWS IoT Things Graph.

- Per i dettagli sull'API, consulta GetUploadStatus AWS CLICommand Reference.

`list-flow-execution-messages`

Il seguente esempio di codice mostra come utilizzare`list-flow-execution-messages`.

AWS CLI

Come ottenere informazioni sugli eventi durante l'esecuzione di un flusso

L'esempio `list-flow-execution-messages` seguente ottiene informazioni sugli eventi in un'esecuzione di un flusso.

```
aws iotthingsgraph list-flow-execution-messages \
```

```
    --flow-execution-id "urn:tdm:us-west-2/123456789012/
default:Workflow:SecurityFlow_2019-05-11T19:39:55.317Z_MotionSensor_69b151ad-
a611-42f5-ac21-fe537f9868ad"
```

Output:

```
{
    "messages": [
        {
         "eventType": "EXECUTION_STARTED",
         "messageId": "f6294f1e-b109-4bbe-9073-f451a2dda2da",
         "payload": "Flow execution started",
         "timestamp": 1559247540.656
        }
    ]
}
```

Per ulteriori informazioni, consulta Utilizzo dei flussi nella Guida per l'utente di AWS IoT Things Graph.

- Per i dettagli sull'API, consulta ListFlowExecutionMessages AWS CLICommand Reference.

`list-tags-for-resource`

Il seguente esempio di codice mostra come utilizzare`list-tags-for-resource`.

AWS CLI

Come elencare tutti i tag per una risorsa

L'`list-tags-for-resource`esempio seguente elenca tutti i tag per una risorsa AWS IoT Things Graph.

```
aws iotthingsgraph list-tags-for-resource \
    --resource-arn "arn:aws:iotthingsgraph:us-west-2:123456789012:Deployment/
default/Room218"
```

Output:

```
{
   "tags": [
      {
```

```
            "key": "Type",
            "value": "Residential"
        }
    ]
}
```

Per ulteriori informazioni, consulta Tagging Your AWS IoT Things Graph Resources nella AWS IoT Things Graph User Guide.

- Per i dettagli sull'API, consulta ListTagsForResource AWS CLICommand Reference.

`search-entities`

Il seguente esempio di codice mostra come utilizzare`search-entities`.

AWS CLI

Come cercare le entità

L'esempio `search-entities` seguente cerca tutte le entità di tipo `EVENT`.

```
aws iotthingsgraph search-entities \
    --entity-types "EVENT"
```

Output:

```
{
    "descriptions": [
        {
            "id": "urn:tdm:aws/examples:Event:MotionSensorEvent",
            "type": "EVENT",
            "definition": {
                "language": "GRAPHQL",
                "text": "##\n# Description of events emitted by motion
 sensor.\n##\ntype MotionSensorEvent @eventType(id: \"urn:tdm:aws/
examples:event:MotionSensorEvent\",\n            payload: \"urn:tdm:aws/
examples:property:MotionSensorStateProperty\") {ignore:void}"
            }
        },
        {
            "id": "urn:tdm:us-west-2/123456789012/
default:Event:CameraClickedEventV2",
            "type": "EVENT",
```

```
            "definition": {
                "language": "GRAPHQL",
                "text": "type CameraClickedEventV2 @eventType(id: \"urn:tdm:us-
west-2/123456789012/default:event:CameraClickedEventV2\",\r\npayload:
 \"urn:tdm:aws:Property:Boolean\"){ignore:void}"
            }
        },
        {
            "id": "urn:tdm:us-west-2/123456789012/
default:Event:MotionSensorEventV2",
            "type": "EVENT",
            "definition": {
                "language": "GRAPHQL",
                "text": "# Event emitted by the motion sensor.\r\ntype
 MotionSensorEventV2 @eventType(id: \"urn:tdm:us-west-2/123456789012/
default:event:MotionSensorEventV2\",\r\npayload: \"urn:tdm:us-west-2/123456789012/
default:property:MotionSensorStateProperty2\") {ignore:void}"
            }
        }
    ],
    "nextToken": "urn:tdm:us-west-2/123456789012/default:Event:MotionSensorEventV2"
}
```

Per ulteriori informazioni, consulta Riferimento al modello di dati di AWS IoT Things Graph nella Guida per l'utente di AWS IoT Things Graph.

- Per i dettagli sull'API, consulta SearchEntities AWS CLICommand Reference.

`search-flow-executions`

Il seguente esempio di codice mostra come utilizzare`search-flow-executions`.

AWS CLI

Come cercare le esecuzioni del flusso

L'esempio `search-flow-executions` seguente cerca tutte le esecuzioni di un flusso in un'istanza di sistema specificata.

```
aws iotthingsgraph search-flow-executions \
    --system-instance-id "urn:tdm:us-west-2/123456789012/default:Deployment:Room218"
```

Output:

```
{
   "summaries": [
      {
         "createdAt": 1559247540.656,
         "flowExecutionId": "f6294f1e-b109-4bbe-9073-f451a2dda2da",
         "flowTemplateId": "urn:tdm:us-west-2/123456789012/default:Workflow:MyFlow",
         "status": "RUNNING ",
         "systemInstanceId": "urn:tdm:us-west-2/123456789012/
default:System:MySystem",
         "updatedAt": 1559247540.656
      }
   ]
}
```

Per ulteriori informazioni, consulta Utilizzo di configurazioni di sistemi e flussi nella Guida per l'utente di AWS IoT Things Graph.

- Per i dettagli sull'API, consulta SearchFlowExecutions AWS CLICommand Reference.

`search-flow-templates`

Il seguente esempio di codice mostra come utilizzare`search-flow-templates`.

AWS CLI

Come cercare i flussi (o i flussi di lavoro)

L'esempio `search-flow-templates` seguente cerca tutti i flussi (di lavoro) che contengono il modello di dispositivo Camera.

```
aws iotthingsgraph search-flow-templates \
    --filters name="DEVICE_MODEL_ID",value="urn:tdm:aws/examples:DeviceModel:Camera"
```

Output:

```
{
    "summaries": [
        {
            "id": "urn:tdm:us-west-2/123456789012/default:Workflow:MyFlow",
            "revisionNumber": 1,
            "createdAt": 1559247540.292
        },
```

```
        {
            "id": "urn:tdm:us-west-2/123456789012/default:Workflow:SecurityFlow",
            "revisionNumber": 3,
            "createdAt": 1548283099.27
        }
    ]
}
```

Per ulteriori informazioni, consulta Utilizzo dei flussi nella Guida per l'utente di AWS IoT Things Graph.

- Per i dettagli sull'API, consulta SearchFlowTemplates AWS CLICommand Reference.

`search-system-instances`

Il seguente esempio di codice mostra come utilizzare`search-system-instances`.

AWS CLI

Come cercare le istanze di sistema

L'esempio `search-system-instances` seguente cerca tutte le istanze di sistema che contengono il sistema specificato.

```
aws iotthingsgraph search-system-instances \
    --filters name="SYSTEM_TEMPLATE_ID",value="urn:tdm:us-west-2/123456789012/
default:System:SecurityFlow"
```

Output:

```
{
    "summaries": [
        {
            "id": "urn:tdm:us-west-2/123456789012/
default:Deployment:DeploymentForSample",
            "arn": "arn:aws:iotthingsgraph:us-west-2:123456789012:Deployment/
default/DeploymentForSample",
            "status": "NOT_DEPLOYED",
            "target": "GREENGRASS",
            "greengrassGroupName": "ThingsGraphGrnGr",
            "createdAt": 1555716314.707,
            "updatedAt": 1555716314.707
        },
```

```
        {
            "id": "urn:tdm:us-west-2/123456789012/
default:Deployment:MockDeployment",
            "arn": "arn:aws:iotthingsgraph:us-west-2:123456789012:Deployment/
default/MockDeployment",
            "status": "DELETED_IN_TARGET",
            "target": "GREENGRASS",
            "greengrassGroupName": "ThingsGraphGrnGr",
            "createdAt": 1549416462.049,
            "updatedAt": 1549416722.361,
            "greengrassGroupId": "01d04b07-2a51-467f-9d03-0c90b3cdcaaf",
            "greengrassGroupVersionId": "7365aed7-2d3e-4d13-aad8-75443d45eb05"
        },
        {
            "id": "urn:tdm:us-west-2/123456789012/
default:Deployment:MockDeployment2",
            "arn": "arn:aws:iotthingsgraph:us-west-2:123456789012:Deployment/
default/MockDeployment2",
            "status": "DEPLOYED_IN_TARGET",
            "target": "GREENGRASS",
            "greengrassGroupName": "ThingsGraphGrnGr",
            "createdAt": 1549572385.774,
            "updatedAt": 1549572418.408,
            "greengrassGroupId": "01d04b07-2a51-467f-9d03-0c90b3cdcaaf",
            "greengrassGroupVersionId": "bfa70ab3-2bf7-409c-a4d4-bc8328ae5b86"
        },
        {
            "id": "urn:tdm:us-west-2/123456789012/default:Deployment:Room215",
            "arn": "arn:aws:iotthingsgraph:us-west-2:123456789012:Deployment/
default/Room215",
            "status": "NOT_DEPLOYED",
            "target": "GREENGRASS",
            "greengrassGroupName": "ThingsGraphGG",
            "createdAt": 1547056918.413,
            "updatedAt": 1547056918.413
        },
        {
            "id": "urn:tdm:us-west-2/123456789012/default:Deployment:Room218",
            "arn": "arn:aws:iotthingsgraph:us-west-2:123456789012:Deployment/
default/Room218",
            "status": "NOT_DEPLOYED",
            "target": "CLOUD",
            "createdAt": 1559249315.208,
            "updatedAt": 1559249315.208
```

```
        }
    ]
}
```

Per ulteriori informazioni, consulta Utilizzo di configurazioni di sistemi e flussi nella Guida per l'utente di AWS IoT Things Graph.

- Per i dettagli sull'API, consulta SearchSystemInstances AWS CLICommand Reference.

`search-system-templates`

Il seguente esempio di codice mostra come utilizzare`search-system-templates`.

AWS CLI

Come cercare il sistema

L'esempio `search-system-templates` seguente cerca tutti i sistemi che contengono il flusso specificato.

```
aws iotthingsgraph search-system-templates \
    --filters name="FLOW_TEMPLATE_ID",value="urn:tdm:us-west-2/123456789012/
default:Workflow:SecurityFlow"
```

Output:

```
{
    "summaries": [
        {
            "id": "urn:tdm:us-west-2/123456789012/default:System:SecurityFlow",
            "arn": "arn:aws:iotthingsgraph:us-west-2:123456789012:System/default/
SecurityFlow",
            "revisionNumber": 1,
            "createdAt": 1548283099.433
        }
    ]
}
```

Per ulteriori informazioni, consulta Utilizzo dei flussi nella Guida per l'utente di AWS IoT Things Graph.

- Per i dettagli sull'API, consulta SearchSystemTemplates AWS CLICommand Reference.

`search-things`

Il seguente esempio di codice mostra come utilizzare`search-things`.

AWS CLI

Come cercare gli elementi associati a dispositivi e modelli di dispositivi

L'`search-things`esempio seguente cerca tutti gli elementi associati al MotionSensor dispositivo HCSR5 01.

```
aws iotthingsgraph search-things \
    --entity-id "urn:tdm:aws/examples:Device:HCSR501MotionSensor"
```

Output:

```
{
    "things": [
        {
            "thingArn": "arn:aws:iot:us-west-2:123456789012:thing/MotionSensor1",
            "thingName": "MotionSensor1"
        },
        {
            "thingArn": "arn:aws:iot:us-west-2:123456789012:thing/TG_MS",
            "thingName": "TG_MS"
        }
    ]
}
```

Per ulteriori informazioni, consulta Creazione e caricamento di modelli nella Guida per l'utente di AWS IoT Things Graph.

- Per i dettagli sull'API, vedere SearchThingsin AWS CLI Command Reference.

`tag-resource`

Il seguente esempio di codice mostra come utilizzare`tag-resource`.

AWS CLI

Come creare un tag per una risorsa

L'esempio `tag-resource` seguente rimuove un tag dalla risorsa specificata.

```
aws iotthingsgraph tag-resource \
    --resource-arn "arn:aws:iotthingsgraph:us-west-2:123456789012:Deployment/
default/Room218" \
    --tags key="Type",value="Residential"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Tagging Your AWS IoT Things Graph Resources nella AWS IoT Things Graph User Guide.

- Per i dettagli sull'API, consulta TagResource AWS CLICommand Reference.

`undeploy-system-instance`

Il seguente esempio di codice mostra come utilizzare`undeploy-system-instance`.

AWS CLI

Come annullare l'implementazione di un'istanza di sistema nella relativa destinazione

L'esempio `undeploy-system-instance` seguente rimuove un'istanza di sistema dalla relativa destinazione.

```
aws iotthingsgraph undeploy-system-instance \
    --id "urn:tdm:us-west-2/123456789012/default:Deployment:Room215"
```

Output:

```
{
    "summary": {
        "id": "urn:tdm:us-west-2/123456789012/default:Deployment:Room215",
        "arn": "arn:aws:iotthingsgraph:us-west-2:123456789012:Deployment/default/
Room215",
        "status": "PENDING_DELETE",
        "target": "GREENGRASS",
        "greengrassGroupName": "ThingsGraphGrnGr",
        "createdAt": 1553189694.255,
        "updatedAt": 1559344549.601,
        "greengrassGroupId": "01d04b07-2a51-467f-9d03-0c90b3cdcaaf",
        "greengrassGroupVersionId": "731b371d-d644-4b67-ac64-3934e99b75d7"
    }
}
```

Per ulteriori informazioni, consulta Lifecycle Management for AWS IoT Things Graph Entities, Flows, Systems and Deployments nella AWS IoT Things Graph User Guide.

- Per i dettagli sull'API, consulta UndeploySystemInstanceCommand Reference.AWS CLI

`untag-resource`

Il seguente esempio di codice mostra come utilizzare`untag-resource`.

AWS CLI

Come rimuovere un tag da una risorsa

L'esempio `untag-resource` seguente rimuove un tag dalla risorsa specificata.

```
aws iotthingsgraph untag-resource \
    --resource-arn "arn:aws:iotthingsgraph:us-west-2:123456789012:Deployment/
default/Room218" \
    --tag-keys "Type"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Tagging Your AWS IoT Things Graph Resources nella AWS IoT Things Graph User Guide.

- Per i dettagli sull'API, consulta UntagResource AWS CLICommand Reference.

`update-flow-template`

Il seguente esempio di codice mostra come utilizzare`update-flow-template`.

AWS CLI

Come aggiornare un flusso

L'esempio `update-flow-template` seguente aggiorna un flusso (di lavoro). Il valore di `MyFlowDefinition` è il GraphQL che modella il flusso.

```
aws iotthingsgraph update-flow-template \
    --id "urn:tdm:us-west-2/123456789012/default:Workflow:MyFlow" \
    --definition language=GRAPHQL,text="MyFlowDefinition"
```

Output:

```
{
    "summary": {
        "createdAt": 1559248067.545,
        "id": "urn:tdm:us-west-2/123456789012/default:Workflow:MyFlow",
        "revisionNumber": 2
    }
}
```

Per ulteriori informazioni, consulta Utilizzo dei flussi nella Guida per l'utente di AWS IoT Things Graph.

- Per i dettagli sull'API, consulta UpdateFlowTemplate AWS CLICommand Reference.

`update-system-template`

Il seguente esempio di codice mostra come utilizzare`update-system-template`.

AWS CLI

Come aggiornare un sistema

L'esempio `update-system-template` seguente aggiorna un sistema. Il valore di `MySystemDefinition` è il GraphQL che modella il sistema.

```
aws iotthingsgraph update-system-template \
    --id "urn:tdm:us-west-2/123456789012/default:System:MySystem" \
    --definition language=GRAPHQL,text="MySystemDefinition"
```

Output:

```
{
    "summary": {
        "createdAt": 1559249776.254,
        "id": "urn:tdm:us-west-2/123456789012/default:System:MySystem",
        "arn": "arn:aws:iotthingsgraph:us-west-2:123456789012:System/default/
MySystem",
        "revisionNumber": 2
    }
}
```

Per ulteriori informazioni, consulta Creazione dei sistemi nella Guida per l'utente di AWS IoT Things Graph.

- Per i dettagli sull'API, consulta UpdateSystemTemplate AWS CLICommand Reference.

`upload-entity-definitions`

Il seguente esempio di codice mostra come utilizzare`upload-entity-definitions`.

AWS CLI

Come caricare le definizioni delle entità

L'esempio `upload-entity-definitions` seguente carica le definizioni di entità nel tuo namespace. Il valore di `MyEntityDefinitions` è il GraphQL che modella le entità.

```
aws iotthingsgraph upload-entity-definitions \
    --document language=GRAPHQL,text="MyEntityDefinitions"
```

Output:

```
{
    "uploadId": "f6294f1e-b109-4bbe-9073-f451a2dda2da"
}
```

Per ulteriori informazioni, consulta Modellazione delle entità nella Guida per l'utente di AWS IoT Things Graph.

- Per i dettagli sull'API, consulta UploadEntityDefinitions AWS CLICommand Reference.

Wireless AWS IoT esempi utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with Wireless AWS IoT.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`associate-aws-account-with-partner-account`

Il seguente esempio di codice mostra come utilizzare`associate-aws-account-with-partner-account`.

AWS CLI

Per associare un account partner al tuo AWS account

L'`associate-aws-account-with-partner-account`esempio seguente associa le seguenti credenziali dell'account Sidewalk al tuo account. AWS

```
aws iotwireless associate-aws-account-with-partner-account \
    --sidewalk
 AmazonId="12345678901234",AppServerPrivateKey="a123b45c6d78e9f012a34cd5e6a7890b12c3d45e6f78
```

Output:

```
{
    "Sidewalk": {
        "AmazonId": "12345678901234",
        "AppServerPrivateKey":
 "a123b45c6d78e9f012a34cd5e6a7890b12c3d45e6f78a1b234c56d7e890a1234"
    }
}
```

Per ulteriori informazioni, consulta Amazon Sidewalk Integration for AWS IoT Core nella IoT Developers Guide.AWS

- Per i dettagli sull'API, consulta AssociateAwsAccountWithPartnerAccount AWS CLICommand Reference.

`associate-wireless-device-with-thing`

Il seguente esempio di codice mostra come utilizzare`associate-wireless-device-with-thing`.

AWS CLI

Come associare un oggetto a un dispositivo wireless

L'esempio `associate-wireless-device-with-thing` seguente associa un oggetto al dispositivo wireless con l'ID specificato.

```
aws iotwireless associate-wireless-device-with-thing \
    --id "12345678-a1b2-3c45-67d8-e90fa1b2c34d" \
    --thing-arn "arn:aws:iot:us-east-1:123456789012:thing/MyIoTWirelessThing"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Aggiungere i gateway e i dispositivi wireless a AWS IoT Core for LoRa WAN nella Guida per sviluppatori AWS IoT.

- Per i dettagli sull'API, consulta AssociateWirelessDeviceWithThing AWS CLICommand Reference.

`associate-wireless-gateway-with-certificate`

Il seguente esempio di codice mostra come utilizzare`associate-wireless-gateway-with-certificate`.

AWS CLI

Come associare il certificato al gateway wireless

L'esempio `associate-wireless-gateway-with-certificate` seguente associa un gateway wireless a un certificato.

```
aws iotwireless associate-wireless-gateway-with-certificate \
    --id "12345678-a1b2-3c45-67d8-e90fa1b2c34d" \
    --iot-certificate-
id "a123b45c6d78e9f012a34cd5e6a7890b12c3d45e6f78a1b234c56d7e890a1234"
```

Output:

```
{
    "IotCertificateId":
 "a123b45c6d78e9f012a34cd5e6a7890b12c3d45e6f78a1b234c56d7e890a1234"
}
```

Per ulteriori informazioni, consulta Aggiungere i gateway e i dispositivi wireless a AWS IoT Core for LoRa WAN nella Guida per sviluppatori AWS IoT.

- Per i dettagli sull'API, consulta AssociateWirelessGatewayWithCertificate AWS CLICommand Reference.

`associate-wireless-gateway-with-thing`

Il seguente esempio di codice mostra come utilizzare`associate-wireless-gateway-with-thing`.

AWS CLI

Come associare un oggetto a un gateway wireless

L'esempio `associate-wireless-gateway-with-thing` seguente associa un oggetto a un gateway wireless.

```
aws iotwireless associate-wireless-gateway-with-thing \
    --id "12345678-a1b2-3c45-67d8-e90fa1b2c34d" \
    --thing-arn "arn:aws:iot:us-east-1:123456789012:thing/MyIoTWirelessThing"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Aggiungere i gateway e i dispositivi wireless a AWS IoT Core for LoRa WAN nella Guida per sviluppatori AWS IoT.

- Per i dettagli sull'API, consulta AssociateWirelessGatewayWithThing AWS CLICommand Reference.

`create-destination`

Il seguente esempio di codice mostra come utilizzare`create-destination`.

AWS CLI

Come creare una destinazione wireless IoT

L'`create-destination`esempio seguente crea una destinazione per mappare un messaggio del dispositivo a una regola AWS IoT. Prima di eseguire questo comando, devi aver creato un ruolo IAM che fornisca a AWS IoT Core for LoRa WAN le autorizzazioni necessarie per inviare dati alla regola AWS IoT.

```
aws iotwireless create-destination \
```

```
    --name IoTWirelessDestination \
    --expression-type RuleName \
    --expression IoTWirelessRule \
    --role-arn arn:aws:iam::123456789012:role/IoTWirelessDestinationRole
```

Output:

```
{
    "Arn": "arn:aws:iotwireless:us-east-1:123456789012:Destination/
IoTWirelessDestination",
    "Name": "IoTWirelessDestination"
}
```

Per ulteriori informazioni, consulta Aggiungere destinazioni a AWS IoT Core for LoRa WAN nella AWS IoT Developers Guide.

- Per i dettagli sull'API, consulta CreateDestination AWS CLICommand Reference.

`create-device-profile`

Il seguente esempio di codice mostra come utilizzare`create-device-profile`.

AWS CLI

Come creare un nuovo profilo del dispositivo

L'esempio `create-device-profile` seguente crea un nuovo profilo del dispositivo wireless IoT.

```
aws iotwireless create-device-profile
```

Output:

```
{
    "Arn": "arn:aws:iotwireless:us-east-1:123456789012:DeviceProfile/12345678-
a1b2-3c45-67d8-e90fa1b2c34d",
    "Id": "12345678-a1b2-3c45-67d8-e90fa1b2c34d"
}
```

Per ulteriori informazioni, consulta Aggiungere profili a AWS IoT Core for LoRa WAN nella AWS IoT Developers Guide.

- Per i dettagli sull'API, consulta CreateDeviceProfile AWS CLICommand Reference.

`create-service-profile`

Il seguente esempio di codice mostra come utilizzare`create-service-profile`.

AWS CLI

Come creare un nuovo profilo del servizio

L'esempio `create-service-profile` seguente crea un nuovo profilo del servizio wireless IoT.

```
aws iotwireless create-service-profile
```

Output:

```
{
    "Arn": "arn:aws:iotwireless:us-east-1:123456789012:ServiceProfile/12345678-
a1b2-3c45-67d8-e90fa1b2c34d",
    "Id": "12345678-a1b2-3c45-67d8-e90fa1b2c34d"
}
```

Per ulteriori informazioni, consulta Aggiungere profili a AWS IoT Core for LoRa WAN nella AWS IoT Developers Guide.

- Per i dettagli sull'API, consulta CreateServiceProfile AWS CLICommand Reference.

`create-wireless-device`

Il seguente esempio di codice mostra come utilizzare`create-wireless-device`.

AWS CLI

Come creare un dispositivo wireless IoT

L'`create-wireless-device`esempio seguente crea una risorsa di dispositivo wireless di tipo LoRa WAN.

```
aws iotwireless create-wireless-device \
    --cli-input-json file://input.json
```

Contenuto di `input.json`:

```
{
    "Description": "My LoRaWAN wireless device"
    "DestinationName": "IoTWirelessDestination"
    "LoRaWAN": {
        "DeviceProfileId": "ab0c23d3-b001-45ef-6a01-2bc3de4f5333",
        "ServiceProfileId": "fe98dc76-cd12-001e-2d34-5550432da100",
        "OtaaV1_1": {
            "AppKey": "3f4ca100e2fc675ea123f4eb12c4a012",
            "JoinEui": "b4c231a359bc2e3d",
            "NwkKey": "01c3f004a2d6efffe32c4eda14bcd2b4"
        },
        "DevEui": "ac12efc654d23fc2"
    },
    "Name": "SampleIoTWirelessThing"
    "Type": LoRaWAN
}
```

Output:

```
{
    "Arn": "arn:aws:iotwireless:us-
east-1:123456789012:WirelessDevice/1ffd32c8-8130-4194-96df-622f072a315f",
    "Id": "1ffd32c8-8130-4194-96df-622f072a315f"
}
```

Per ulteriori informazioni, consulta Connessione di dispositivi e gateway a AWS IoT Core for LoRa WAN nella AWS IoT Developers Guide.

- Per i dettagli sull'API, consulta CreateWirelessDevice AWS CLICommand Reference.

`create-wireless-gateway-task-definition`

Il seguente esempio di codice mostra come utilizzare`create-wireless-gateway-task-definition`.

AWS CLI

Come creare una definizione di attività del gateway wireless

L'esempio `create-wireless-gateway-task-definition` seguente crea automaticamente attività utilizzando questa definizione di attività per tutti i gateway con la versione corrente specificata.

```
aws iotwireless create-wireless-gateway-task-definition \
    --cli-input-json file://input.json
```

Contenuto di `input.json`:

```
{
    "AutoCreateTasks": true,
    "Name": "TestAutoUpdate",
    "Update":{
        "UpdateDataSource" : "s3://cupsalphagafirmwarebin/station",
        "UpdateDataRole" : "arn:aws:iam::001234567890:role/SDK_Test_Role",
        "LoRaWAN" :{
            "CurrentVersion" :{
                "PackageVersion" : "1.0.0",
                "Station" : "2.0.5",
                "Model" : "linux"
            },
            "UpdateVersion" :{
                "PackageVersion" : "1.0.1",
                "Station" : "2.0.5",
                "Model" : "minihub"
            }
        }
    }
}
```

Output:

```
{
    "Id": "b7d3baad-25c7-35e7-a4e1-1683a0d61da9"
}
```

Per ulteriori informazioni, consulta Connessione di dispositivi e gateway a AWS IoT Core for LoRa WAN nella AWS IoT Developers Guide.

- Per i dettagli sull'API, consulta CreateWirelessGatewayTaskDefinition AWS CLICommand Reference.

`create-wireless-gateway-task`

Il seguente esempio di codice mostra come utilizzare`create-wireless-gateway-task`.

AWS CLI

Come creare l'attività per un gateway wireless

L'esempio `create-wireless-gateway-task` seguente crea un'attività per un gateway wireless.

```
aws iotwireless create-wireless-gateway-task \
    --id "12345678-a1b2-3c45-67d8-e90fa1b2c34d" \
    --wireless-gateway-task-definition-id "aa000102-0304-b0cd-ef56-a1b23cde456a"
```

Output:

```
{
    "WirelessGatewayTaskDefinitionId": "aa204003-0604-30fb-ac82-a4f95aaf450a",
    "Status": "Success"
}
```

Per ulteriori informazioni, consulta Connessione di dispositivi e gateway a AWS IoT Core for LoRa WAN nella AWS IoT Developers Guide.

- Per i dettagli sull'API, consulta CreateWirelessGatewayTask AWS CLICommand Reference.

`create-wireless-gateway`

Il seguente esempio di codice mostra come utilizzare`create-wireless-gateway`.

AWS CLI

Come creare un gateway wireless

L'`create-wireless-gateway`esempio seguente crea un gateway per dispositivi LoRa WAN wireless.

```
aws iotwireless create-wireless-gateway \
    --lorawan GatewayEui="a1b2c3d4567890ab",RfRegion="US915" \
    --name "myFirstLoRaWANGateway" \
    --description "Using my first LoRaWAN gateway"
```

Output:

```
{
    "Arn": "arn:aws:iotwireless:us-east-1:123456789012:WirelessGateway/12345678-
a1b2-3c45-67d8-e90fa1b2c34d",
    "Id": "12345678-a1b2-3c45-67d8-e90fa1b2c34d"
}
```

Per ulteriori informazioni, consulta Connessione di dispositivi e gateway a AWS IoT Core for LoRa WAN nella AWS IoT Developers Guide.

- Per i dettagli sull'API, consulta CreateWirelessGateway AWS CLICommand Reference.

`delete-destination`

Il seguente esempio di codice mostra come utilizzare`delete-destination`.

AWS CLI

Come eliminare una destinazione wireless IoT

L'esempio `delete-destination` seguente elimina la risorsa di destinazione wireless creata con il nome `IoTWirelessDestination`.

```
aws iotwireless delete-destination \
    --name "IoTWirelessDestination"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Aggiungere destinazioni a AWS IoT Core for LoRa WAN nella AWS IoT Developers Guide.

- Per i dettagli sull'API, consulta DeleteDestination AWS CLICommand Reference.

`delete-device-profile`

Il seguente esempio di codice mostra come utilizzare`delete-device-profile`.

AWS CLI

Come eliminare un profilo del dispositivo

L'esempio `delete-device-profile` seguente elimina un profilo del dispositivo creato con l'ID specificato.

```
aws iotwireless delete-device-profile \
    --id "12345678-a1b2-3c45-67d8-e90fa1b2c34d"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Aggiungere profili a AWS IoT Core for LoRa WAN nella AWS IoT Developers Guide.

- Per i dettagli sull'API, consulta DeleteDeviceProfile AWS CLICommand Reference.

`delete-service-profile`

Il seguente esempio di codice mostra come utilizzare`delete-service-profile`.

AWS CLI

Come eliminare un profilo del servizio

L'esempio `delete-service-profile` seguente elimina un profilo del servizio creato con l'ID specificato.

```
aws iotwireless delete-service-profile \
    --id "12345678-a1b2-3c45-67d8-e90fa1b2c34d"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Aggiungere profili a AWS IoT Core for LoRa WAN nella AWS IoT Developers Guide.

- Per i dettagli sull'API, consulta DeleteServiceProfile AWS CLICommand Reference.

`delete-wireless-device`

Il seguente esempio di codice mostra come utilizzare`delete-wireless-device`.

AWS CLI

Come eliminare un dispositivo wireless

L'esempio `delete-wireless-device` seguente elimina un dispositivo wireless con l'ID specificato.

```
aws iotwireless delete-wireless-device \
    --id "12345678-a1b2-3c45-67d8-e90fa1b2c34d"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Connessione di dispositivi e gateway a AWS IoT Core for LoRaWAN nella AWS IoT Developers Guide.

- Per i dettagli sull'API, consulta DeleteWirelessDevice AWS CLICommand Reference.

`delete-wireless-gateway-task-definition`

Il seguente esempio di codice mostra come utilizzare`delete-wireless-gateway-task-definition`.

AWS CLI

Come eliminare una definizione di attività del gateway wireless

L'esempio `delete-wireless-gateway-task-definition` seguente elimina la definizione dell'attività del gateway wireless creata con il seguente ID.

```
aws iotwireless delete-wireless-gateway-task-definition \
    --id "12345678-a1b2-3c45-67d8-e90fa1b2c34d"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Connessione di dispositivi e gateway a AWS IoT Core for LoRaWAN nella AWS IoT Developers Guide.

- Per i dettagli sull'API, consulta DeleteWirelessGatewayTaskDefinition AWS CLICommand Reference.

`delete-wireless-gateway-task`

Il seguente esempio di codice mostra come utilizzare`delete-wireless-gateway-task`.

AWS CLI

Come eliminare un'attività del gateway wireless

L'esempio `delete-wireless-gateway-task` seguente elimina l'attività del gateway wireless con l'ID specificato.

```
aws iotwireless delete-wireless-gateway-task \
    --id "12345678-a1b2-3c45-67d8-e90fa1b2c34d"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Connessione di dispositivi e gateway a AWS IoT Core for LoRaWAN nella AWS IoT Developers Guide.

- Per i dettagli sull'API, consulta DeleteWirelessGatewayTask AWS CLICommand Reference.

`delete-wireless-gateway`

Il seguente esempio di codice mostra come utilizzare`delete-wireless-gateway`.

AWS CLI

Come eliminare un gateway wireless

L'esempio `delete-wireless-gateway` seguente elimina un gateway wireless con l'ID specificato.

```
aws iotwireless delete-wireless-gateway \
    --id "12345678-a1b2-3c45-67d8-e90fa1b2c34d"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Connessione di dispositivi e gateway a AWS IoT Core for LoRaWAN nella AWS IoT Developers Guide.

- Per i dettagli sull'API, consulta DeleteWirelessGateway AWS CLICommand Reference.

`disassociate-aws-account-from-partner-account`

Il seguente esempio di codice mostra come utilizzare`disassociate-aws-account-from-partner-account`.

AWS CLI

Per dissociare l'account partner dall' AWS account

L'`disassociate-aws-account-from-partner-account`esempio seguente dissocia un account partner dall'account attualmente associato. AWS

```
aws iotwireless disassociate-aws-account-from-partner-account \
    --partner-account-id "12345678901234" \
    --partner-type "Sidewalk"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Aggiungere i gateway e i dispositivi wireless a AWS IoT Core for LoRa WAN nella Guida per sviluppatori AWS IoT.

- Per i dettagli sull'API, consulta DisassociateAwsAccountFromPartnerAccount AWS CLICommand Reference.

`disassociate-wireless-device-from-thing`

Il seguente esempio di codice mostra come utilizzare`disassociate-wireless-device-from-thing`.

AWS CLI

Come annullare l'associazione dell'oggetto al dispositivo wireless

L'esempio `disassociate-wireless-device-from-thing` seguente annulla l'associazione di un dispositivo wireless all'oggetto a cui è attualmente associato.

```
aws iotwireless disassociate-wireless-device-from-thing \
    --id "12345678-a1b2-3c45-67d8-e90fa1b2c34d"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Aggiungere i gateway e i dispositivi wireless a AWS IoT Core for LoRa WAN nella Guida per sviluppatori AWS IoT.

- Per i dettagli sull'API, consulta DisassociateWirelessDeviceFromThing AWS CLICommand Reference.

`disassociate-wireless-gateway-from-certificate`

Il seguente esempio di codice mostra come utilizzare`disassociate-wireless-gateway-from-certificate`.

AWS CLI

Come annullare l'associazione del certificato al gateway wireless

L'esempio `disassociate-wireless-gateway-from-certificate` seguente annulla l'associazione di un gateway wireless al certificato a cui è attualmente associato.

```
aws iotwireless disassociate-wireless-gateway-from-certificate \
    --id "12345678-a1b2-3c45-67d8-e90fa1b2c34d"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Aggiungere i gateway e i dispositivi wireless a AWS IoT Core for LoRa WAN nella Guida per sviluppatori AWS IoT.

- Per i dettagli sull'API, consulta DisassociateWirelessGatewayFromCertificate AWS CLICommand Reference.

`disassociate-wireless-gateway-from-thing`

Il seguente esempio di codice mostra come utilizzare`disassociate-wireless-gateway-from-thing`.

AWS CLI

Come annullare l'associazione dell'oggetto al gateway wireless

L'esempio `disassociate-wireless-gateway-from-thing` seguente annulla l'associazione di un gateway wireless all'oggetto a cui è attualmente associato.

```
aws iotwireless disassociate-wireless-gateway-from-thing \
    --id "12345678-a1b2-3c45-67d8-e90fa1b2c34d"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Aggiungere i gateway e i dispositivi wireless a AWS IoT Core for LoRa WAN nella Guida per sviluppatori AWS IoT.

- Per i dettagli sull'API, consulta DisassociateWirelessGatewayFromThing AWS CLICommand Reference.

`get-destination`

Il seguente esempio di codice mostra come utilizzare`get-destination`.

AWS CLI

Come ottenere informazioni su una destinazione wireless IoT

L'esempio `get-destination` seguente ottiene informazioni sulla risorsa di destinazione creata con il nome `IoTWirelessDestination`.

```
aws iotwireless get-destination \
    --name "IoTWirelessDestination"
```

Output:

```
{
    "Arn": "arn:aws:iotwireless:us-east-1:123456789012:Destination/
IoTWirelessDestination",
    "Name": "IoTWirelessDestination",
    "Expression": "IoTWirelessRule",
    "ExpressionType": "RuleName",
    "RoleArn": "arn:aws:iam::123456789012:role/IoTWirelessDestinationRole"
}
```

Per ulteriori informazioni, consulta Aggiungere destinazioni a AWS IoT Core for LoRa WAN nella AWS IoT Developers Guide.

- Per i dettagli sull'API, consulta GetDestination AWS CLICommand Reference.

`get-device-profile`

Il seguente esempio di codice mostra come utilizzare`get-device-profile`.

AWS CLI

Come ottenere informazioni su un profilo del dispositivo

L'esempio `get-device-profile` seguente ottiene informazioni sul profilo del dispositivo creato con l'ID specificato.

```
aws iotwireless get-device-profile \
```

```
    --id "12345678-a1b2-3c45-67d8-e90fa1b2c34d"
```

Output:

```
{
    "Arn": "arn:aws:iotwireless:us-east-1:123456789012:DeviceProfile/12345678-
a1b2-3c45-67d8-e90fa1b2c34d",
    "Id": "12345678-a1b2-3c45-67d8-e90fa1b2c34d",
    "LoRaWAN": {
    "MacVersion": "1.0.3",
    "MaxDutyCycle": 10,
    "Supports32BitFCnt": false,
    "RegParamsRevision": "RP002-1.0.1",
    "SupportsJoin": true,
    "RfRegion": "US915",
    "MaxEirp": 13,
    "SupportsClassB": false,
    "SupportsClassC": false
    }
}
```

Per ulteriori informazioni, consulta Aggiungere profili a AWS IoT Core for LoRa WAN nella AWS IoT Developers Guide.

- Per i dettagli sull'API, consulta GetDeviceProfile AWS CLICommand Reference.

`get-partner-account`

Il seguente esempio di codice mostra come utilizzare`get-partner-account`.

AWS CLI

Come ottenere informazioni sull'account partner

L'esempio `get-partner-account` seguente ottiene informazioni sull'account Sidewalk con il seguente ID.

```
aws iotwireless get-partner-account \
    --partner-account-id "12345678901234" \
    --partner-type "Sidewalk"
```

Output:

```
{
    "Sidewalk": {
        "AmazonId": "12345678901234",
        "Fingerprint":
 "a123b45c6d78e9f012a34cd5e6a7890b12c3d45e6f78a1b234c56d7e890a1234"
    },
    "AccountLinked": false
}
```

Per ulteriori informazioni, consulta Amazon Sidewalk Integration for AWS IoT Core nella IoT Developers Guide.AWS

- Per i dettagli sull'API, consulta GetPartnerAccount AWS CLICommand Reference.

`get-service-endpoint`

Il seguente esempio di codice mostra come utilizzare`get-service-endpoint`.

AWS CLI

Come ottenere l'endpoint del servizio

L'esempio `get-service-endpoint` seguente ottiene l'endpoint specifico dell'account per il protocollo CUPS.

```
aws iotwireless get-service-endpoint
```

Output:

```
{
    "ServiceType": "CUPS",
    "ServiceEndpoint": "https://A1RMKZ37ACAGOT.cups.lorawan.us-
east-1.amazonaws.com:443",
    "ServerTrust": "-----BEGIN CERTIFICATE-----\n
    MIIESTCCAzGgAwIBAgITBn+UV4WH6Kx33rJTMlu8mYtWDTANBgkqhkiG9w0BAQsF\n
    ADA5MQswCQYDVQQGEwJVUzEPMA0GA1UEChMGQW1hem9uMRkwFwYDVQQDExBBbWF6\n
    b24gUm9vdCBDQSAxMB4XDTE1MTAyMjAwMDAwMFoXDTI1MTAxOTAwMDAwMFowRjEL\n
    MAkGA1UEBhMCVVMxDzANBgNVBAoTBkFtYXpvbjEVMBMGA1UECxMMU2VydmVyIENB\n
    IDFCMQ8wDQYDVQQDEwZBbWF6b24wggEiMA0GCSqGSIb3DQEBAQUAA4IBDwAwggEK\n
    AoIBAQDCThZn3c68asg3Wuw6MLAd5tES6BIoSMzoKcG5blPVo+sDORrMd4f2AbnZ\n
    cMzPa43j4wNxhplty6aUKk4T1qe9BOwKFjwK6zmxxLVYo7bHViXsPlJ6qOMpFge5\n
    blDP+18x+B26A0piiQOuPkfyDyeR4xQghfj66Yo19V+emU3nazfvpFA+ROz6WoVm\n
```

```
    B5x+F2pV8xeKNR7u6azDdU5YVX1TawprmxRC1+WsAYmz6qP+z8ArDITC2FMVy2fw\n
    0IjKOtEXc/VfmtTFch5+AfGYMGMqqvJ6LcXiAhqG5TI+Dr0RtM88k+8XUBCeQ8IG\n
    KuANaL7TiItKZYxK1MMuTJtV9IblAgMBAAGjggE7MIIBNzASBgNVHRMBAf8ECDAG\n
    AQH/AgEAMA4GA1UdDwEB/wQEAwIBhjAdBgNVHQ4EFgQUWaRmBlKge5WSPKOUByeW\n
    dFv5PdAwHwYDVR0jBBgwFoAUhBjMhTTsvAyUlC4IWZzHshBOCggwewYIKwYBBQUH\n
    AQEEbzBtMC8GCCsGAQUFBzABhiNodHRwOi8vb2NzcC5yb290Y2ExLmFtYXpvbnRy\n
    dXN0LmNvbTA6BggrBgEFBQcwAoYuaHR0cDovL2NydC5yb290Y2ExLmFtYXpvbnRy\n
    dXN0LmNvbS9yb290Y2ExLmNlcjA/BgNVHR8EODA2MDSgMqAwhi5odHRwOi8vY3Js\n
    LnJvb3RjYTEuYW1hem9udHJ1c3QuY29tL3Jvb3RjYTEuY3JsMBMGA1UdIAQMMAow\n
    CAYGZ4EMAQIBMA0GCSqGSIb3DQEBCwUAA4IBAQCFkr41u3nPo4FCHOTjY3NTOVI1\n
    59Gt/a6ZiqyJEi+752+a1U5y6iAwYfmXss2lJwJFqMp2PphKg5625kXg8kP2CN5t\n
    6G7bMQcT8C8xDZNtYTd7WPD8UZiRKAJPBXa30/AbwuZe0GaFEQ8ugcYQgSn+IGBI\n
    8/LwhBNTZTUVEWuCUUBVV18YtbAiPq3yXqMB48Oz+ctBWuZSkbvkNodPLamkB2g1\n
    upRyzQ7qDn1X8nn8N8V7YJ6y68AtkHcNSRAnpTitxBKjtKPISLMVCx7i4hncxHZS\n
    yLyKQXhw2W2Xs0qLeC1etA+jTGDK4UfLeC0SF7FSi8o5LL21L8IzApar2pR/\n
    -----END CERTIFICATE-----\n"
}
```

Per ulteriori informazioni, consulta Connessione di dispositivi e gateway a AWS IoT Core for LoRaWAN nella AWS IoT Developers Guide.

- Per i dettagli sull'API, consulta GetServiceEndpoint AWS CLICommand Reference.

`get-service-profile`

Il seguente esempio di codice mostra come utilizzare`get-service-profile`.

AWS CLI

Come ottenere informazioni su un profilo del servizio

L'esempio `get-service-profile` seguente ottiene informazioni sul profilo del servizio creato con l'ID specificato.

```
aws iotwireless get-service-profile \
    --id "12345678-a1b2-3c45-67d8-e90fa1b2c34d"
```

Output:

```
{
    "Arn": "arn:aws:iotwireless:us-east-1:651419225604:ServiceProfile/538185bb-
d7e7-4b95-96a0-c51aa4a5b9a0",
    "Id": "12345678-a1b2-3c45-67d8-e90fa1b2c34d",
```

```
    "LoRaWAN": {
        "HrAllowed": false,
        "NwkGeoLoc": false,
        "DrMax": 15,
        "UlBucketSize": 4096,
        "PrAllowed": false,
        "ReportDevStatusBattery": false,
        "DrMin": 0,
        "DlRate": 60,
        "AddGwMetadata": false,
        "ReportDevStatusMargin": false,
        "MinGwDiversity": 1,
        "RaAllowed": false,
        "DlBucketSize": 4096,
        "DevStatusReqFreq": 24,
        "TargetPer": 5,
        "UlRate": 60
    }
}
```

Per ulteriori informazioni, consulta Aggiungere profili a AWS IoT Core for LoRa WAN nella AWS IoT Developers Guide.

- Per i dettagli sull'API, consulta GetServiceProfile AWS CLICommand Reference.

`get-wireless-device-statistics`

Il seguente esempio di codice mostra come utilizzare`get-wireless-device-statistics`.

AWS CLI

Come ottenere informazioni operative su un dispositivo wireless

L'esempio `get-wireless-device-statistics` seguente ottiene informazioni operative su un dispositivo wireless.

```
aws iotwireless get-wireless-device-statistics \
    --wireless-device-id "1ffd32c8-8130-4194-96df-622f072a315f"
```

Output:

```
{
    "WirelessDeviceId": "1ffd32c8-8130-4194-96df-622f072a315f"
```

```
}
```

Per ulteriori informazioni, consulta Connessione di dispositivi e gateway a AWS IoT Core for LoRaWAN nella AWS IoT Developers Guide.

- Per i dettagli sull'API, consulta GetWirelessDeviceStatistics AWS CLICommand Reference.

`get-wireless-device`

Il seguente esempio di codice mostra come utilizzare`get-wireless-device`.

AWS CLI

Come ottenere informazioni sul dispositivo wireless

L'`get-wireless-device`esempio seguente elenca i widget disponibili nel tuo AWS account.

```
aws iotwireless get-wireless-device \
    --identifier "1ffd32c8-8130-4194-96df-622f072a315f" \
    --identifier-type WirelessDeviceID
```

Output:

```
{
    "Name": "myLoRaWANDevice",
    "ThingArn": "arn:aws:iot:us-east-1:123456789012:thing/44b87eb4-9bce-423d-
b5fc-973f5ecc358b",
    "DestinationName": "IoTWirelessDestination",
    "Id": "1ffd32c8-8130-4194-96df-622f072a315f",
    "ThingName": "44b87eb4-9bce-423d-b5fc-973f5ecc358b",
    "Type": "LoRaWAN",
    "LoRaWAN": {
        "DeviceProfileId": "ab0c23d3-b001-45ef-6a01-2bc3de4f5333",
        "ServiceProfileId": "fe98dc76-cd12-001e-2d34-5550432da100",
        "OtaaV1_1": {
            "AppKey": "3f4ca100e2fc675ea123f4eb12c4a012",
            "JoinEui": "b4c231a359bc2e3d",
            "NwkKey": "01c3f004a2d6efffe32c4eda14bcd2b4"
        },
        "DevEui": "ac12efc654d23fc2"
    },
    "Arn": "arn:aws:iotwireless:us-
east-1:123456789012:WirelessDevice/1ffd32c8-8130-4194-96df-622f072a315f",
```

```
    "Description": "My LoRaWAN wireless device"
}
```

Per ulteriori informazioni, consulta Connessione di dispositivi e gateway a AWS IoT Core for LoRaWAN nella AWS IoT Developers Guide.

- Per i dettagli sull'API, consulta GetWirelessDevice AWS CLICommand Reference.

`get-wireless-gateway-certificate`

Il seguente esempio di codice mostra come utilizzare`get-wireless-gateway-certificate`.

AWS CLI

Come ottenere l'ID di un certificato associato a un gateway wireless

L'esempio `get-wireless-gateway-certificate` seguente ottiene l'ID del certificato associato a un gateway wireless con l'ID specificato.

```
aws iotwireless get-wireless-gateway-certificate \
    --id "6c44ab31-8b4d-407a-bed3-19b6c7cda551"
```

Output:

```
{
    "IotCertificateId":
 "8ea4aeae3db34c78cce75d9abd830356869ead6972997e0603e5fd032c804b6f"
}
```

Per ulteriori informazioni, consulta Connessione di dispositivi e gateway a AWS IoT Core for LoRaWAN nella AWS IoT Developers Guide.

- Per i dettagli sull'API, consulta GetWirelessGatewayCertificate AWS CLICommand Reference.

`get-wireless-gateway-firmware-information`

Il seguente esempio di codice mostra come utilizzare`get-wireless-gateway-firmware-information`.

AWS CLI

Come ottenere informazioni sul firmware di un gateway wireless

L'esempio `get-wireless-gateway-firmware-information` seguente ottiene la versione del firmware e altre informazioni su un gateway wireless.

```
aws iotwireless get-wireless-gateway-firmware-information \
    --id "3039b406-5cc9-4307-925b-9948c63da25b"
```

Output:

```
{
    "LoRaWAN" :{
        "CurrentVersion" :{
            "PackageVersion" : "1.0.0",
            "Station" : "2.0.5",
            "Model" : "linux"
        }
    }
}
```

Per ulteriori informazioni, consulta Connessione di dispositivi e gateway a AWS IoT Core for LoRa WAN nella AWS IoT Developers Guide.

- Per i dettagli sull'API, consulta GetWirelessGatewayFirmwareInformation AWS CLICommand Reference.

`get-wireless-gateway-statistics`

Il seguente esempio di codice mostra come utilizzare`get-wireless-gateway-statistics`.

AWS CLI

Come ottenere informazioni operative su un gateway wireless

L'esempio `get-wireless-gateway-statistics` seguente ottiene informazioni operative su un gateway wireless.

```
aws iotwireless get-wireless-gateway-statistics \
    --wireless-gateway-id "3039b406-5cc9-4307-925b-9948c63da25b"
```

Output:

```
{
```

```
    "WirelessGatewayId": "3039b406-5cc9-4307-925b-9948c63da25b"
}
```

Per ulteriori informazioni, consulta Connessione di dispositivi e gateway a AWS IoT Core for LoRaWAN nella AWS IoT Developers Guide.

- Per i dettagli sull'API, consulta GetWirelessGatewayStatistics AWS CLICommand Reference.

`get-wireless-gateway-task-definition`

Il seguente esempio di codice mostra come utilizzare`get-wireless-gateway-task-definition`.

AWS CLI

Come ottenere informazioni sulla definizione dell'attività di un gateway wireless

L'esempio `get-wireless-gateway-task-definition` seguente ottiene informazioni sulla definizione dell'attività wireless con l'ID specificato.

```
aws iotwireless get-wireless-gateway-task-definition \
    --id "b7d3baad-25c7-35e7-a4e1-1683a0d61da9"
```

Output:

```
{
    "AutoCreateTasks": true,
    "Name": "TestAutoUpdate",
    "Update":{
        "UpdateDataSource" : "s3://cupsalphagafirmwarebin/station",
        "UpdateDataRole" : "arn:aws:iam::001234567890:role/SDK_Test_Role",
        "LoRaWAN" :{
            "CurrentVersion" :{
                "PackageVersion" : "1.0.0",
                "Station" : "2.0.5",
                "Model" : "linux"
            },
            "UpdateVersion" :{
                "PackageVersion" : "1.0.1",
                "Station" : "2.0.5",
                "Model" : "minihub"
            }
```

```
        }
    }
}
```

Per ulteriori informazioni, consulta Connessione di dispositivi e gateway a AWS IoT Core for LoRaWAN nella AWS IoT Developers Guide.

- Per i dettagli sull'API, consulta GetWirelessGatewayTaskDefinition AWS CLICommand Reference.

`get-wireless-gateway-task`

Il seguente esempio di codice mostra come utilizzare`get-wireless-gateway-task`.

AWS CLI

Come ottenere informazioni sull'attività del gateway wireless

L'esempio `get-wireless-gateway-task` seguente ottiene informazioni sull'attività del gateway wireless con l'ID specificato.

```
aws iotwireless get-wireless-gateway-task \
    --id "11693a46-6866-47c3-a031-c9a616e7644b"
```

Output:

```
{
    "WirelessGatewayId": "6c44ab31-8b4d-407a-bed3-19b6c7cda551",
    "WirelessGatewayTaskDefinitionId": "b7d3baad-25c7-35e7-a4e1-1683a0d61da9",
    "Status": "Success"
}
```

Per ulteriori informazioni, consulta Connessione di dispositivi e gateway a AWS IoT Core for LoRaWAN nella AWS IoT Developers Guide.

- Per i dettagli sull'API, consulta GetWirelessGatewayTask AWS CLICommand Reference.

`get-wireless-gateway`

Il seguente esempio di codice mostra come utilizzare`get-wireless-gateway`.

AWS CLI

Come ottenere informazioni su un gateway wireless

L'esempio `get-wireless-gateway` seguente ottiene informazioni sul gateway wireless `myFirstLoRaWANGateway`.

```
aws iotwireless get-wireless-gateway \
    --identifier "12345678-a1b2-3c45-67d8-e90fa1b2c34d" \
    --identifier-type WirelessGatewayId
```

Output:

```
{
    "Description": "My first LoRaWAN gateway",
    "ThingArn": "arn:aws:iot:us-east-1:123456789012:thing/a1b2c3d4-5678-90ab-
cdef-12ab345c67de",
    "LoRaWAN": {
        "RfRegion": "US915",
        "GatewayEui": "a1b2c3d4567890ab"
    },
    "ThingName": "a1b2c3d4-5678-90ab-cdef-12ab345c67de",
    "Id": "12345678-a1b2-3c45-67d8-e90fa1b2c34d",
    "Arn": "arn:aws:iotwireless:us-
east-1:123456789012:WirelessGateway/6c44ab31-8b4d-407a-bed3-19b6c7cda551",
    "Name": "myFirstLoRaWANGateway"
}
```

Per ulteriori informazioni, consulta Connessione di dispositivi e gateway a AWS IoT Core for LoRa WAN nella AWS IoT Developers Guide.

- Per i dettagli sull'API, consulta GetWirelessGateway AWS CLICommand Reference.

`list-destinations`

Il seguente esempio di codice mostra come utilizzare`list-destinations`.

AWS CLI

Come elencare le destinazioni wireless

L'`list-destinations`esempio seguente elenca le destinazioni disponibili registrate nel tuo AWS account.

```
aws iotwireless list-destinations
```

Output:

```
{
    "DestinationList": [
        {
            "Arn": "arn:aws:iotwireless:us-east-1:123456789012:Destination/
IoTWirelessDestination",
            "Name": "IoTWirelessDestination",
            "Expression": "IoTWirelessRule",
            "Description": "Destination for messages processed using
 IoTWirelessRule",
            "RoleArn": "arn:aws:iam::123456789012:role/IoTWirelessDestinationRole"
        },
        {
            "Arn": "arn:aws:iotwireless:us-east-1:123456789012:Destination/
IoTWirelessDestination2",
            "Name": "IoTWirelessDestination2",
            "Expression": "IoTWirelessRule2",
            "RoleArn": "arn:aws:iam::123456789012:role/IoTWirelessDestinationRole"
        }
    ]
}
```

Per ulteriori informazioni, consulta Aggiungere destinazioni a AWS IoT Core for LoRa WAN nella AWS IoT Developers Guide.

- Per i dettagli sull'API, consulta ListDestinations AWS CLICommand Reference.

`list-device-profiles`

Il seguente esempio di codice mostra come utilizzare`list-device-profiles`.

AWS CLI

Come elencare i profili di dispositivo

L'`list-device-profiles`esempio seguente elenca i profili di dispositivo disponibili registrati nell' AWS account.

```
aws iotwireless list-device-profiles
```

Output:

```
{
    "DeviceProfileList": [
        {
            "Id": "12345678-a1b2-3c45-67d8-e90fa1b2c34d",
            "Arn": "arn:aws:iotwireless:us-
east-1:123456789012:DeviceProfile/12345678-a1b2-3c45-67d8-e90fa1b2c34d"
        },
        {
            "Id": "a1b2c3d4-5678-90ab-cdef-12ab345c67de",
            "Arn": "arn:aws:iotwireless:us-east-1:123456789012:DeviceProfile/
a1b2c3d4-5678-90ab-cdef-12ab345c67de"
        }
    ]
}
```

Per ulteriori informazioni, consulta Aggiungere profili a AWS IoT Core for LoRa WAN nella AWS IoT Developers Guide.

- Per i dettagli sull'API, consulta ListDeviceProfiles AWS CLICommand Reference.

`list-partner-accounts`

Il seguente esempio di codice mostra come utilizzare`list-partner-accounts`.

AWS CLI

Come elencare gli account partner

L'`list-partner-accounts`esempio seguente elenca gli account partner disponibili associati al tuo AWS account.

```
aws iotwireless list-partner-accounts
```

Output:

```
{
    "Sidewalk": [
        {
            "AmazonId": "78965678771228",
```

```
            "Fingerprint":
 "bd96d8ef66dbfd2160eb60e156849e82ad7018b8b73c1ba0b4fc65c32498ee35"
        },
        {
            "AmazonId": "89656787651228",
            "Fingerprint":
 "bc5e99e151c07be14be7e6603e4489c53f858b271213a36ebe3370777ba06e9b"
        }
    ]
}
```

Per ulteriori informazioni, consulta Amazon Sidewalk Integration for AWS IoT Core nella IoT Developers Guide.AWS

- Per i dettagli sull'API, consulta ListPartnerAccounts AWS CLICommand Reference.

`list-service-profiles`

Il seguente esempio di codice mostra come utilizzare`list-service-profiles`.

AWS CLI

Come elencare i profili di servizio

L'`list-service-profiles`esempio seguente elenca i profili di servizio disponibili registrati nell' AWS account dell'utente.

```
aws iotwireless list-service-profiles
```

Output:

```
{
    "ServiceProfileList": [
        {
            "Id": "12345678-a1b2-3c45-67d8-e90fa1b2c34d",
            "Arn": "arn:aws:iotwireless:us-
east-1:123456789012:ServiceProfile/538185bb-d7e7-4b95-96a0-c51aa4a5b9a0"
        },
        {
            "Id": "a1b2c3d4-5678-90ab-cdef-12ab345c67de",
            "Arn": "arn:aws:iotwireless:us-east-1:123456789012:ServiceProfile/
ea8bc823-5d13-472e-8d26-9550737d8100"
        }
```

```
    ]
}
```

Per ulteriori informazioni, consulta Aggiungere profili a AWS IoT Core for LoRa WAN nella AWS IoT Developers Guide.

- Per i dettagli sull'API, consulta ListServiceProfiles AWS CLICommand Reference.

`list-tags-for-resource`

Il seguente esempio di codice mostra come utilizzare`list-tags-for-resource`.

AWS CLI

Come elencare i tag assegnati alla risorsa

L'esempio `list-tags-for-resource` seguente elenca i tag assegnati a una risorsa di destinazione wireless.

```
aws iotwireless list-tags-for-resource \
    --resource-arn "arn:aws:iotwireless:us-east-1:123456789012:Destination/
IoTWirelessDestination"
```

Output:

```
{
    "Tags": [
        {
            "Value": "MyValue",
            "Key": "MyTag"
        }
    ]
}
```

Per ulteriori informazioni, consulta Descrivi le tue risorse AWS IoT Core for LoRa WAN nella AWS IoT Developers Guide.

- Per i dettagli sull'API, consulta ListTagsForResource AWS CLICommand Reference.

`list-wireless-devices`

Il seguente esempio di codice mostra come utilizzare`list-wireless-devices`.

AWS CLI

Come elencare i dispositivi wireless disponibili

L'`list-wireless-devices`esempio seguente elenca i dispositivi wireless disponibili registrati nell' AWS account.

```
aws iotwireless list-wireless-devices
```

Output:

```
{
    "WirelessDeviceList": [
        {
            "Name": "myLoRaWANDevice",
            "DestinationName": "IoTWirelessDestination",
            "Id": "1ffd32c8-8130-4194-96df-622f072a315f",
            "Type": "LoRaWAN",
            "LoRaWAN": {
                "DevEui": "ac12efc654d23fc2"
            },
            "Arn": "arn:aws:iotwireless:us-
east-1:123456789012:WirelessDevice/1ffd32c8-8130-4194-96df-622f072a315f"
        }
    ]
}
```

Per ulteriori informazioni, consulta Connessione di dispositivi e gateway a AWS IoT Core for LoRa WAN nella AWS IoT Developers Guide.

- Per i dettagli sull'API, consulta ListWirelessDevices AWS CLICommand Reference.

`list-wireless-gateway-task-definitions`

Il seguente esempio di codice mostra come utilizzare`list-wireless-gateway-task-definitions`.

AWS CLI

Come elencare le definizioni di attività del gateway wireless

L'`list-wireless-gateway-task-definitions`esempio seguente elenca le definizioni delle attività del gateway wireless disponibili registrate nell' AWS account.

```
aws iotwireless list-wireless-gateway-task-definitions
```

Output:

```
{
    "TaskDefinitions": [
        {
            "Id": "b7d3baad-25c7-35e7-a4e1-1683a0d61da9",
            "LoRaWAN" :
                {
                "CurrentVersion" :{
                    "PackageVersion" : "1.0.0",
                    "Station" : "2.0.5",
                    "Model" : "linux"
                },
                "UpdateVersion" :{
                    "PackageVersion" : "1.0.1",
                    "Station" : "2.0.5",
                    "Model" : "minihub"
                }
            }
        }
    ]
}
```

Per ulteriori informazioni, consulta Connessione di dispositivi e gateway a AWS IoT Core for LoRa WAN nella AWS IoT Developers Guide.

- Per i dettagli sull'API, consulta ListWirelessGatewayTaskDefinitions AWS CLICommand Reference.

`list-wireless-gateways`

Il seguente esempio di codice mostra come utilizzare`list-wireless-gateways`.

AWS CLI

Come elencare i gateway wireless

L'`list-wireless-gateways`esempio seguente elenca i gateway wireless disponibili nell' AWS account.

```
aws iotwireless list-wireless-gateways
```

Output:

```
{
    "WirelessGatewayList": [
        {
            "Description": "My first LoRaWAN gateway",
            "LoRaWAN": {
                "RfRegion": "US915",
                "GatewayEui": "dac632ebc01d23e4"
            },
            "Id": "3039b406-5cc9-4307-925b-9948c63da25b",
            "Arn": "arn:aws:iotwireless:us-
east-1:123456789012:WirelessGateway/3039b406-5cc9-4307-925b-9948c63da25b",
            "Name": "myFirstLoRaWANGateway"
        },
        {
            "Description": "My second LoRaWAN gateway",
            "LoRaWAN": {
                "RfRegion": "US915",
                "GatewayEui": "cda123fffe92ecd2"
            },
            "Id": "3285bdc7-5a12-4991-84ed-dadca65e342e",
            "Arn": "arn:aws:iotwireless:us-
east-1:123456789012:WirelessGateway/3285bdc7-5a12-4991-84ed-dadca65e342e",
            "Name": "mySecondLoRaWANGateway"
        }
    ]
}
```

Per ulteriori informazioni, consulta Connessione di dispositivi e gateway a AWS IoT Core for LoRa WAN nella AWS IoT Developers Guide.

- Per i dettagli sull'API, consulta ListWirelessGateways AWS CLICommand Reference.

`send-data-to-wireless-device`

Il seguente esempio di codice mostra come utilizzare`send-data-to-wireless-device`.

AWS CLI

Come inviare dati al dispositivo wireless

L'esempio `send-data-to-wireless-device` seguente invia un frame di dati dell'applicazione decrittografato al dispositivo wireless.

```
aws iotwireless send-data-to-wireless-device \
    --id "11aa5eae-2f56-4b8e-a023-b28d98494e49" \
    --transmit-mode "1" \
    --payload-data "SGVsbG8gVG8gRGV2c2lt" \
    --wireless-metadata LoRaWAN={FPort=1}
```

Output:

```
{
    MessageId: "6011dd36-0043d6eb-0072-0008"
}
```

Per ulteriori informazioni, consulta Connessione di dispositivi e gateway a AWS IoT Core for LoRa WAN nella AWS IoT Developers Guide.

- Per i dettagli sull'API, consulta SendDataToWirelessDevice AWS CLICommand Reference.

`tag-resource`

Il seguente esempio di codice mostra come utilizzare`tag-resource`.

AWS CLI

Come specificare una chiave e un valore di tag per una risorsa

L'esempio `tag-resource` seguente assegna alla destinazione wireless `IoTWirelessDestination` il tag con la chiave `MyTag` e il valore `MyValue`.

```
aws iotwireless tag-resource \
    --resource-arn "arn:aws:iotwireless:us-east-1:651419225604:Destination/
IoTWirelessDestination" \
    --tags Key="MyTag",Value="MyValue"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Descrivi le tue risorse AWS IoT Core for LoRa WAN nella AWS IoT Developers Guide.

- Per i dettagli sull'API, consulta TagResource AWS CLICommand Reference.

`test-wireless-device`

Il seguente esempio di codice mostra come utilizzare`test-wireless-device`.

AWS CLI

Come testare il dispositivo wireless

L'esempio `test-wireless-device` seguente invia i dati di uplink `Hello` a un dispositivo con l'ID specificato.

```
aws iotwireless test-wireless-device \
    --id "11aa5eae-2f56-4b8e-a023-b28d98494e49"
```

Output:

```
{
    Result: "Test succeeded. one message is sent with payload: hello"
}
```

Per ulteriori informazioni, consulta Connessione di dispositivi e gateway a AWS IoT Core for LoRa WAN nella AWS IoT Developers Guide.

- Per i dettagli sull'API, consulta TestWirelessDevice AWS CLICommand Reference.

`untag-resource`

Il seguente esempio di codice mostra come utilizzare`untag-resource`.

AWS CLI

Come rimuovere uno o più tag da una risorsa

L'esempio `untag-resource` seguente rimuove il tag `MyTag` e il relativo valore dalla destinazione wireless `IoTWirelessDestination`.

```
aws iotwireless untag-resource \
    --resource-arn "arn:aws:iotwireless:us-east-1:123456789012:Destination/
IoTWirelessDestination" \
    --tag-keys "MyTag"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Descrivi le tue risorse AWS IoT Core for LoRa WAN nella AWS IoT Developers Guide.

- Per i dettagli sull'API, consulta UntagResource AWS CLICommand Reference.

`update-destination`

Il seguente esempio di codice mostra come utilizzare`update-destination`.

AWS CLI

Come aggiornare le proprietà di una destinazione

L'esempio `update-destination` seguente aggiorna la proprietà di descrizione di una destinazione wireless.

```
aws iotwireless update-destination \
    --name "IoTWirelessDestination" \
    --description "Destination for messages processed using IoTWirelessRule"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Aggiungere destinazioni a AWS IoT Core for LoRa WAN nella AWS IoT Developers Guide.

- Per i dettagli sull'API, consulta UpdateDestination AWS CLICommand Reference.

`update-partner-account`

Il seguente esempio di codice mostra come utilizzare`update-partner-account`.

AWS CLI

Come aggiornare le proprietà di un account partner

L'esempio `update-partner-account` seguente aggiorna `AppServerPrivateKey` per l'account con l'ID specificato.

```
aws iotwireless update-partner-account \
    --partner-account-id "78965678771228" \
    --partner-type "Sidewalk" \
    --sidewalk
 AppServerPrivateKey="f798ab4899346a88599180fee9e14fa1ada7b6df989425b7c6d2146dd6c815bb"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Amazon Sidewalk Integration for AWS IoT Core nella IoT Developers Guide.AWS

- Per i dettagli sull'API, consulta UpdatePartnerAccount AWS CLICommand Reference.

`update-wireless-device`

Il seguente esempio di codice mostra come utilizzare`update-wireless-device`.

AWS CLI

Come aggiornare le proprietà di un dispositivo wireless

L'`update-wireless-device`esempio seguente aggiorna le proprietà di un dispositivo wireless registrato nell' AWS account dell'utente.

```
aws iotwireless update-wireless-device \
    --id "1ffd32c8-8130-4194-96df-622f072a315f" \
    --destination-name IoTWirelessDestination2 \
    --description "Using my first LoRaWAN device"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Connessione di dispositivi e gateway a AWS IoT Core for LoRa WAN nella AWS IoT Developers Guide.

- Per i dettagli sull'API, consulta UpdateWirelessDevice AWS CLICommand Reference.

`update-wireless-gateway`

Il seguente esempio di codice mostra come utilizzare`update-wireless-gateway`.

AWS CLI

Come aggiornare il gateway wireless

L'esempio `update-wireless-gateway` seguente aggiorna la descrizione del gateway wireless.

```
aws iotwireless update-wireless-gateway \
    --id "3285bdc7-5a12-4991-84ed-dadca65e342e" \
    --description "Using my LoRaWAN gateway"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Connessione di dispositivi e gateway a AWS IoT Core for LoRaWAN nella AWS IoT Developers Guide.

- Per i dettagli sull'API, consulta UpdateWirelessGateway AWS CLICommand Reference.

Esempi di utilizzo di Amazon IVS AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando Amazon IVS. AWS Command Line Interface

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`batch-get-channel`

Il seguente esempio di codice mostra come usare`batch-get-channel`.

AWS CLI

Come ottenere informazioni sulla configurazione di canale per più canali

L'esempio `batch-get-channel` seguente elenca le informazioni sui canali specificati.

```
aws ivs batch-get-channel \
    --arns arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh \
        arn:aws:ivs:us-west-2:123456789012:channel/efghEFGHijkl
```

Output:

```
{
    "channels": [
        {
            "arn": "arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh",
            "authorized": false,
            "containerFormat": "TS",
            "ingestEndpoint": "a1b2c3d4e5f6.global-contribute.live-video.net",
            "insecureIngest": false,
            "latencyMode": "LOW",
            "multitrackInputConfiguration": {
                "enabled": false,
                "maximumResolution": "FULL_HD",
                "policy": "ALLOW"
            },
            "name": "channel-1",
            "playbackUrl": "https://a1b2c3d4e5f6.us-west-2.playback.live-video.net/
api/video/v1/us-west-2.123456789012.channel-1.abcdEFGH.m3u8",
            "preset": "",
            "playbackRestrictionPolicyArn": "",
            "recordingConfigurationArn": "arn:aws:ivs:us-
west-2:123456789012:recording-configuration/ABCD12cdEFgh",
            "srt": {
                "endpoint": "a1b2c3d4e5f6.srt.live-video.net",
                "passphrase":
 "AB1C2defGHijkLMNo3PqQRstUvwxyzaBCDEfghh4ijklMN5opqrStuVWxyzAbCDEfghIJ"
            },
            "tags": {},
            "type": "STANDARD"
        },
        {
            "arn": "arn:aws:ivs:us-west-2:123456789012:channel/efghEFGHijkl",
            "authorized": false,
            "containerFormat": "FRAGMENTED_MP4",
            "ingestEndpoint": "a1b2c3d4e5f6.global-contribute.live-video.net",
            "insecureIngest": false,
```

```
            "latencyMode": "LOW",
            "multitrackInputConfiguration": {
                "enabled": true,
                "maximumResolution": "FULL_HD",
                "policy": "ALLOW"
            },
            "name": "channel-2",
            "playbackUrl": "https://a1b2c3d4e5f6.us-west-2.playback.live-video.net/
api/video/v1/us-west-2.123456789012.channel-2.abcdEFGH.m3u8",
            "preset": "",
            "playbackRestrictionPolicyArn": "arn:aws:ivs:us-
west-2:123456789012:playback-restriction-policy/ABcdef34ghIJ"",
            "recordingConfigurationArn": "",
            "srt": {
                "endpoint": "a1b2c3d4e5f6.srt.live-video.net",
                "passphrase":
 "BA1C2defGHijkLMNo3PqQRstUvwxyzaBCDEfghh4ijklMN5opqrStuVWxyzAbCDEfghIJ"
            },
            "tags": {},
            "type": "STANDARD"
        }
    ]
}
```

Per ulteriori informazioni, consulta Creare un canale nella Guida per l'utente dello Streaming a bassa latenza di IVS.

- Per i dettagli sull'API, consulta BatchGetChannel AWS CLICommand Reference.

`batch-get-stream-key`

Il seguente esempio di codice mostra come utilizzare`batch-get-stream-key`.

AWS CLI

Come ottenere informazioni su più chiavi di flusso

L'esempio `batch-get-stream-key` seguente ottiene le informazioni sulle chiavi di flusso specificate.

```
aws ivs batch-get-stream-key \
    --arns arn:aws:ivs:us-west-2:123456789012:stream-key/skSKABCDefgh \
       arn:aws:ivs:us-west-2:123456789012:stream-key/skSKIJKLmnop
```

Output:

```
{
    "streamKeys": [
        {
            "arn": "arn:aws:ivs:us-west-2:123456789012:stream-key/skSKABCDefgh",
            "value": "sk_us-west-2_abcdABCDefgh_567890abcdef",
            "channelArn": "arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh",
            "tags": {}
        },
        {
            "arn": "arn:aws:ivs:us-west-2:123456789012:stream-key/skSKIJKLmnop",
            "value": "sk_us-west-2_abcdABCDefgh_567890ghijkl",
            "channelArn": "arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh",
            "tags": {}
        }
     ]
}
```

Per ulteriori informazioni, consulta Creare un canale nella Guida per l'utente dello Streaming a bassa latenza di IVS.

- Per i dettagli sull'API, consulta BatchGetStreamKey AWS CLICommand Reference.

`batch-start-viewer-session-revocation`

Il seguente esempio di codice mostra come utilizzare`batch-start-viewer-session-revocation`.

AWS CLI

Come revocare le sessioni di visualizzazione per più coppie ARN del canale e ID visualizzazione

L'esempio `batch-start-viewer-session-revocation` seguente esegue la revoca della sessione su più coppie ARN del canale e ID visualizzazione contemporaneamente. La richiesta può essere completata normalmente, ma restituisce valori nel campo degli errori se il chiamante non è autorizzato a revocare la sessione specificata.

```
aws ivs batch-start-viewer-session-revocation \
    --viewer-sessions '[{"channelArn":"arn:aws:ivs:us-west-2:123456789012:channel/
abcdABCDefgh1","viewerId":"abcdefg1","viewerSessionVersionsLessThanOrEqualTo":1234567890},
 \
```

```
    {"channelArn":"arn:aws:ivs:us-west-2:123456789012:channel/
abcdABCDefgh2","viewerId":"abcdefg2","viewerSessionVersionsLessThanOrEqualTo":1234567890}]'
```

Output:

```
{
    "errors": [
        {
            "channelArn": "arn:aws:ivs:us-west-2:123456789012:channel/
abcdABCDefgh1",
            "viewerId": "abcdefg1",
            "code": "403",
            "message": "not authorized",
        },
        {
            "channelArn": "arn:aws:ivs:us-west-2:123456789012:channel/
abcdABCDefgh2",
            "viewerId": "abcdefg2",
            "code": "403",
            "message": "not authorized",
        }
    ]
}
```

Per ulteriori informazioni, consulta Impostazione dei canali privati nella Guida per l'utente di Amazon Interactive Video Service.

- Per i dettagli sull'API, consulta BatchStartViewerSessionRevocation AWS CLICommand Reference.

`create-channel`

Il seguente esempio di codice mostra come utilizzare`create-channel`.

AWS CLI

Esempio 1: come creare un canale senza registrazione

L'esempio `create-channel` seguente crea un nuovo canale e una chiave di flusso associata per avviare la trasmissione.

```
aws ivs create-channel \
```

```
    --name 'test-channel' \
    --no-insecure-ingest
```

Output:

```
{
    "channel": {
        "arn": "arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh",
        "authorized": false,
        "containerFormat": "TS",
        "name": "test-channel",
        "latencyMode": "LOW",
        "multitrackInputConfiguration": {
            "enabled": false,
            "maximumResolution": "FULL_HD",
            "policy": "ALLOW"
        },
        "playbackRestrictionPolicyArn": "",
        "recordingConfigurationArn": "",
        "srt": {
            "endpoint": "a1b2c3d4e5f6.srt.live-video.net",
            "passphrase":
 "AB1C2defGHijkLMNo3PqQRstUvwxyzaBCDEfghh4ijklMN5opqrStuVWxyzAbCDEfghIJ"
        },
        "ingestEndpoint": "a1b2c3d4e5f6.global-contribute.live-video.net",
        "insecureIngest": false,
        "playbackUrl": "https://a1b2c3d4e5f6.us-west-2.playback.live-video.net/api/
video/v1/us-west-2.123456789012.channel.abcdEFGH.m3u8",
        "preset": "",
        "tags": {},
        "type": "STANDARD"
    },
    "streamKey": {
        "arn": "arn:aws:ivs:us-west-2:123456789012:stream-key/g1H2I3j4k5L6",
        "value": "sk_us-west-2_abcdABCDefgh_567890abcdef",
        "channelArn": "arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh",
        "tags": {}
    }
}
```

Per ulteriori informazioni, consulta Creare un canale nella Guida per l'utente dello Streaming a bassa latenza di IVS.

Esempio 2: creare un canale con la registrazione abilitata, utilizzando la RecordingConfiguration risorsa specificata dal relativo ARN

L'esempio `create-channel` seguente crea un nuovo canale e una chiave di flusso associata per avviare la trasmissione, nonché imposta la registrazione per il canale:

```
aws ivs create-channel \
    --name test-channel-with-recording \
    --insecure-ingest \
    --recording-configuration-arn 'arn:aws:ivs:us-west-2:123456789012:recording-configuration/ABCD12cdEFgh'
```

Output:

```
{
    "channel": {
        "arn": "arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh",
        "containerFormat": "TS",
        "name": "test-channel-with-recording",
        "latencyMode": "LOW",
        "multitrackInputConfiguration": {
            "enabled": false,
            "maximumResolution": "FULL_HD",
            "policy": "ALLOW"
        },
        "type": "STANDARD",
        "playbackRestrictionPolicyArn": "",
        "recordingConfigurationArn": "arn:aws:ivs:us-west-2:123456789012:recording-configuration/ABCD12cdEFgh",
        "srt": {
            "endpoint": "a1b2c3d4e5f6.srt.live-video.net",
            "passphrase":
 "BA1C2defGHijkLMNo3PqQRstUvwxyzaBCDEfghh4ijklMN5opqrStuVWxyzAbCDEfghIJ"
        },
        "ingestEndpoint": "a1b2c3d4e5f6.global-contribute.live-video.net",
        "insecureIngest": true,
        "playbackUrl": "https://a1b2c3d4e5f6.us-west-2.playback.live-video.net/api/video/v1/us-west-2.123456789012.channel.abcdEFGH.m3u8",
        "preset": "",
        "authorized": false,
        "tags": {},
        "type": "STANDARD"
    },
```

```
    "streamKey": {
        "arn": "arn:aws:ivs:us-west-2:123456789012:stream-key/abcdABCDefgh",
        "value": "sk_us-west-2_abcdABCDefgh_567890abcdef",
        "channelArn": "arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh",
        "tags": {}
    }
}
```

Per ulteriori informazioni, consulta Registrazione su Amazon S3 nella Guida per l'utente dello Streaming a bassa latenza di IVS.

Esempio 3: come creare un canale con una policy di restrizione della riproduzione specificata dal relativo nome della risorsa Amazon (ARN)

L'esempio `create-channel` seguente crea un nuovo canale e una chiave di flusso associata per avviare la trasmissione, nonché imposta la policy di restrizione della riproduzione per il canale:

```
aws ivs create-channel \
    --name test-channel-with-playback-restriction-policy\
    --insecure-ingest \
    --playback-restriction-policy-arn 'arn:aws:ivs:us-west-2:123456789012:playback-restriction-policy/ABcdef34ghIJ'
```

Output:

```
{
    "channel": {
        "arn": "arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh",
        "containerFormat": "TS",
        "name": "test-channel-with-playback-restriction-policy",
        "latencyMode": "LOW",
        "multitrackInputConfiguration": {
            "enabled": false,
            "maximumResolution": "FULL_HD",
            "policy": "ALLOW"
        },
        "type": "STANDARD",
        "playbackRestrictionPolicyArn": "arn:aws:ivs:us-
west-2:123456789012:playback-restriction-policy/ABcdef34ghIJ",
        "recordingConfigurationArn": "",
        "srt": {
            "endpoint": "a1b2c3d4e5f6.srt.live-video.net",
```

```
            "passphrase":
 "AB1C2edfGHijkLMNo3PqQRstUvwxyzaBCDEfghh4ijklMN5opqrStuVWxyzAbCDEfghIJ"
        },
        "ingestEndpoint": "a1b2c3d4e5f6.global-contribute.live-video.net",
        "insecureIngest": true,
        "playbackUrl": "https://a1b2c3d4e5f6.us-west-2.playback.live-video.net/api/
video/v1/us-west-2.123456789012.channel.abcdEFGH.m3u8",
        "preset": "",
        "authorized": false,
        "tags": {},
        "type": "STANDARD"
    },
    "streamKey": {
        "arn": "arn:aws:ivs:us-west-2:123456789012:stream-key/abcdABCDefgh",
        "value": "sk_us-west-2_abcdABCDefgh_567890abcdef",
        "channelArn": "arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh",
        "tags": {}
    }
}
```

Per ulteriori informazioni, consulta Contenuti e spettatori indesiderati in IVS nella Guida per l'utente dello Streaming a bassa latenza di IVS.

Esempio 4: come creare un canale con multitraccia abilitato

L'esempio `create-channel` seguente crea un nuovo canale e una chiave di flusso associata per avviare la trasmissione, nonché abilita il multitraccia.

```
aws ivs create-channel \
    --name 'test-channel' \
    --no-insecure-ingest \
    --container-format 'FRAGMENTED_MP4' \
    --multitrack-input-configuration '{"enabled": true,"maximumResolution":
 "FULL_HD","policy": "ALLOW"}'
```

Output:

```
{
    "channel": {
        "arn": "arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh",
        "authorized": false,
        "containerFormat": "FRAGMENTED_MP4",
```

```
        "name": "test-channel",
        "latencyMode": "LOW",
        "multitrackInputConfiguration": {
            "enabled": true,
            "maximumResolution": "FULL_HD",
            "policy": "ALLOW"
        },
        "playbackRestrictionPolicyArn": "",
        "recordingConfigurationArn": "",
        "srt": {
            "endpoint": "a1b2c3d4e5f6.srt.live-video.net",
            "passphrase":
 "AB1C2defGHijkLMNo3PqQRstUvwxyzaBCDEfghh4ijklMN5opqrStuVWxyzAbCDEfghIJ"
        },
        "ingestEndpoint": "a1b2c3d4e5f6.global-contribute.live-video.net",
        "insecureIngest": false,
        "playbackUrl": "https://a1b2c3d4e5f6.us-west-2.playback.live-video.net/api/
video/v1/us-west-2.123456789012.channel.abcdEFGH.m3u8",
        "preset": "",
        "tags": {},
        "type": "STANDARD"
    },
    "streamKey": {
        "arn": "arn:aws:ivs:us-west-2:123456789012:stream-key/g1H2I3j4k5L6",
        "value": "sk_us-west-2_abcdABCDefgh_567890abcdef",
        "channelArn": "arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh",
        "tags": {}
    }
}
```

Per ulteriori informazioni, consulta Creare un canale nella Guida per l'utente dello Streaming a bassa latenza di IVS.

- Per i dettagli sull'API, consulta CreateChannel AWS CLICommand Reference.

`create-playback-restriction-policy`

Il seguente esempio di codice mostra come utilizzare`create-playback-restriction-policy`.

AWS CLI

Come creare una policy di restrizione della riproduzione

L'esempio `create-playback-restriction-policy` seguente crea una nuova policy di restrizione della riproduzione.

```
aws ivs create-playback-restriction-policy \
    --name "test-playback-restriction-policy" \
    --enable-strict-origin-enforcement \
    --tags "key1=value1, key2=value2" \
    --allowed-countries US MX \
    --allowed-origins https://www.website1.com https://www.website2.com
```

Output:

```
{
    "playbackRestrictionPolicy": {
        "arn": "arn:aws:ivs:us-west-2:123456789012:playback-restriction-policy/
ABcdef34ghIJ",
        "allowedCountries": [
            "US",
            "MX"
        ],
        "allowedOrigins": [
            "https://www.website1.com",
            "https://www.website2.com"
        ],
        "enableStrictOriginEnforcement": true,
        "name": "test-playback-restriction-policy",
        "tags": {
            "key1": "value1",
            "key2": "value2"
        }
    }
}
```

Per ulteriori informazioni, consulta Contenuti e spettatori indesiderati in IVS nella Guida per l'utente dello Streaming a bassa latenza di IVS.

- Per i dettagli sull'API, consulta CreatePlaybackRestrictionPolicy AWS CLICommand Reference.

`create-recording-configuration`

Il seguente esempio di codice mostra come utilizzare`create-recording-configuration`.

AWS CLI

Per creare una RecordingConfiguration risorsa

L'`create-recording-configuration`esempio seguente crea una RecordingConfiguration risorsa per abilitare la registrazione su Amazon S3.

```
aws ivs create-recording-configuration \
    --name "test-recording-config" \
    --recording-reconnect-window-seconds 60 \
    --tags "key1=value1, key2=value2" \
    --rendition-configuration renditionSelection="CUSTOM",renditions="HD" \
    --thumbnail-configuration
 recordingMode="INTERVAL",targetIntervalSeconds=1,storage="LATEST",resolution="LOWEST_RESOLU
 \
    --destination-configuration s3={bucketName=demo-recording-bucket}
```

Output:

```
{
    "recordingConfiguration": {
        "arn": "arn:aws:ivs:us-west-2:123456789012:recording-configuration/
ABcdef34ghIJ",
        "name": "test-recording-config",
        "destinationConfiguration": {
            "s3": {
                "bucketName": "demo-recording-bucket"
            }
        },
        "state": "CREATING",
        "tags": {
            "key1": "value1",
            "key2": "value2"
        },
        "thumbnailConfiguration": {
            "recordingMode": "INTERVAL",
            "targetIntervalSeconds": 1,
            "resolution": "LOWEST_RESOLUTION",
            "storage": [
                "LATEST"
            ]
        },
        "recordingReconnectWindowSeconds": 60,
```

```
        "renditionConfiguration": {
            "renditionSelection": "CUSTOM",
            "renditions": [
                "HD"
            ]
        }
    }
}
```

Per ulteriori informazioni, consulta Registrazione su Amazon S3 nella Guida per l'utente del Servizio video interattivo Amazon.

- Per i dettagli sull'API, consulta CreateRecordingConfiguration AWS CLICommand Reference.

`create-stream-key`

Il seguente esempio di codice mostra come utilizzare`create-stream-key`.

AWS CLI

Come creare una chiave di flusso

L'esempio `create-stream-key` seguente crea una chiave di flusso per un nome della risorsa Amazon (ARN) specificato.

```
aws ivs create-stream-key \
    --channel-arn arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh
```

Output:

```
{
    "streamKey": {
        "arn": "arn:aws:ivs:us-west-2:123456789012:stream-key/abcdABCDefgh",
        "value": "sk_us-west-2_abcdABCDefgh_567890abcdef",
        "channelArn": "arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh",
        "tags": {}
    }
}
```

Per ulteriori informazioni, consulta Creare un canale nella Guida per l'utente dello Streaming a bassa latenza di IVS.

- Per i dettagli sull'API, consulta CreateStreamKey AWS CLICommand Reference.

`delete-channel`

Il seguente esempio di codice mostra come utilizzare`delete-channel`.

AWS CLI

Come eliminare un canale e le relative chiavi di flusso associate

L'esempio `delete-channel` seguente elimina il canale con il nome della risorsa Amazon (ARN) specificato.

```
aws ivs delete-channel \
    --arn arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Creare un canale nella Guida per l'utente dello Streaming a bassa latenza di IVS.

- Per i dettagli sull'API, consulta DeleteChannel AWS CLICommand Reference.

`delete-playback-key-pair`

Il seguente esempio di codice mostra come utilizzare`delete-playback-key-pair`.

AWS CLI

Come eliminare una coppia di chiavi di riproduzione specificata

L'esempio `delete-playback-key-pair` seguente restituisce l'impronta digitale della coppia di chiavi specificata.

```
aws ivs delete-playback-key-pair \
    --arn arn:aws:ivs:us-west-2:123456789012:playback-key/abcd1234efgh
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Impostazione dei canali privati nella Guida per l'utente di Amazon Interactive Video Service.

- Per i dettagli sull'API, consulta DeletePlaybackKeyPair AWS CLICommand Reference.

`delete-playback-restriction-policy`

Il seguente esempio di codice mostra come utilizzare`delete-playback-restriction-policy`.

AWS CLI

Come eliminare una policy di restrizione della riproduzione

L'esempio `delete-playback-restriction-policy` seguente elimina la policy di restrizione della riproduzione con il nome della risorsa Amazon (ARN) specificato.

```
aws ivs delete-playback-restriction-policy \
    --arn "arn:aws:ivs:us-west-2:123456789012:playback-restriction-policy/
ABcdef34ghIJ"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Contenuti e spettatori indesiderati in IVS nella Guida per l'utente dello Streaming a bassa latenza di IVS.

- Per i dettagli sull'API, consulta DeletePlaybackRestrictionPolicy AWS CLICommand Reference.

`delete-recording-configuration`

Il seguente esempio di codice mostra come utilizzare`delete-recording-configuration`.

AWS CLI

Per eliminare la RecordingConfiguration risorsa specificata dal relativo ARN

L'`delete-recording-configuration`esempio seguente elimina la RecordingConfiguration risorsa con l'ARN specificato.

```
aws ivs delete-recording-configuration \
    --arn "arn:aws:ivs:us-west-2:123456789012:recording-configuration/ABcdef34ghIJ"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Registrazione su Amazon S3 nella Guida per l'utente del Servizio video interattivo Amazon.

- Per i dettagli sull'API, vedere DeleteRecordingConfigurationin AWS CLI Command Reference.

`delete-stream-key`

Il seguente esempio di codice mostra come utilizzare`delete-stream-key`.

AWS CLI

Come eliminare una chiave di flusso

L'esempio `delete-stream-key` seguente elimina la chiave di flusso per un nome della risorsa Amazon (ARN) specificato, in modo che non possa più essere utilizzata per il flusso.

```
aws ivs delete-stream-key \
    --arn arn:aws:ivs:us-west-2:123456789012:stream-key/g1H2I3j4k5L6
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Creare un canale nella Guida per l'utente dello Streaming a bassa latenza di IVS.

- Per i dettagli sull'API, consulta DeleteStreamKey AWS CLICommand Reference.

`get-channel`

Il seguente esempio di codice mostra come utilizzare`get-channel`.

AWS CLI

Come ottenere informazioni sulla configurazione di un canale

L'esempio `get-channel` seguente ottiene la configurazione del canale per un nome della risorsa Amazon (ARN) specificato.

```
aws ivs get-channel \
    --arn 'arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh'
```

Output:

```
{
    "channel": {
        "arn": "arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh",
        "authorized": false,
        "containerFormat": "TS",
        "ingestEndpoint": "a1b2c3d4e5f6.global-contribute.live-video.net",
```

```
        "insecureIngest": false,
        "latencyMode": "LOW",
        "multitrackInputConfiguration": {
            "enabled": false,
            "maximumResolution": "FULL_HD",
            "policy": "ALLOW"
        },
        "name": "channel-1",
        "playbackRestrictionPolicyArn": "",
        "playbackUrl": "https://a1b2c3d4e5f6.us-west-2.playback.live-video.net/api/
video/v1/us-west-2.123456789012.channel.abcdEFGH.m3u8",
        "preset": "",
        "recordingConfigurationArn": "",
        "srt": {
            "endpoint": "a1b2c3d4e5f6.srt.live-video.net",
            "passphrase":
 "AB1C2defGHijkLMNo3PqQRstUvwxyzaBCDEfghh4ijklMN5opqrStuVWxyzAbCDEfghIJ"
        },
        "tags": {}
        "type": "STANDARD",
    }
}
```

Per ulteriori informazioni, consulta Creare un canale nella Guida per l'utente dello Streaming a bassa latenza di IVS.

- Per i dettagli sull'API, consulta GetChannel AWS CLICommand Reference.

`get-playback-key-pair`

Il seguente esempio di codice mostra come utilizzare`get-playback-key-pair`.

AWS CLI

Come ottenere una coppia di chiavi di riproduzione specificata

L'esempio `get-playback-key-pair` seguente restituisce l'impronta digitale della coppia di chiavi specificata.

```
aws ivs get-playback-key-pair \
    --arn arn:aws:ivs:us-west-2:123456789012:playback-key/abcd1234efgh
```

Output:

```
{
    "keyPair": {
        "arn": "arn:aws:ivs:us-west-2:123456789012:playback-key/abcd1234efgh",
        "name": "my-playback-key",
        "fingerprint": "0a:1b:2c:ab:cd:ef:34:56:70:b1:b2:71:01:2a:a3:72",
        "tags": {}
    }
}
```

Per ulteriori informazioni, consulta Impostazione dei canali privati nella Guida per l'utente di Amazon Interactive Video Service.

- Per i dettagli sull'API, consulta GetPlaybackKeyPair AWS CLICommand Reference.

`get-playback-restriction-policy`

Il seguente esempio di codice mostra come utilizzare`get-playback-restriction-policy`.

AWS CLI

Come ottenere le informazioni sulla configurazione di una policy di restrizione della riproduzione

L'esempio `get-playback-restriction-policy` seguente ottiene la configurazione della policy di restrizione della riproduzione con il nome della risorsa Amazon (ARN) specificato.

```
aws ivs get-playback-restriction-policy \
    --arn "arn:aws:ivs:us-west-2:123456789012:playback-restriction-policy/
ABcdef34ghIJ"
```

Output:

```
{
    "playbackRestrictionPolicy": {
        "arn": "arn:aws:ivs:us-west-2:123456789012:playback-restriction-policy/
ABcdef34ghIJ",
        "allowedCountries": [
            "US",
            "MX"
        ],
        "allowedOrigins": [
            "https://www.website1.com",
            "https://www.website2.com"
```

```
        ],
        "enableStrictOriginEnforcement": true,
        "name": "test-playback-restriction-policy",
        "tags": {
            "key1": "value1",
            "key2": "value2"
        }
    }
}
```

Per ulteriori informazioni, consulta Contenuti e spettatori indesiderati in IVS nella Guida per l'utente dello Streaming a bassa latenza di IVS.

- Per i dettagli sull'API, consulta GetPlaybackRestrictionPolicy AWS CLICommand Reference.

`get-recording-configuration`

Il seguente esempio di codice mostra come utilizzare`get-recording-configuration`.

AWS CLI

Per ottenere informazioni su una RecordingConfiguration risorsa

L'`get-recording-configuration`esempio seguente ottiene informazioni sulla RecordingConfiguration risorsa per l'ARN specificato.

```
aws ivs get-recording-configuration \
    --arn "arn:aws:ivs:us-west-2:123456789012:recording-configuration/ABcdef34ghIJ"
```

Output:

```
{
    "recordingConfiguration": {
        "arn": "arn:aws:ivs:us-west-2:123456789012:recording-configuration/
ABcdef34ghIJ",
        "destinationConfiguration": {
            "s3": {
                "bucketName": "demo-recording-bucket"
            }
        },
        "name": "test-recording-config",
        "recordingReconnectWindowSeconds": 60,
        "state": "ACTIVE",
```

```
        "tags": {
            "key1" : "value1",
            "key2" : "value2"
        },
        "thumbnailConfiguration": {
            "recordingMode": "INTERVAL",
            "targetIntervalSeconds": 1,
            "resolution": "LOWEST_RESOLUTION",
            "storage": [
                "LATEST"
            ]
        },
        "renditionConfiguration": {
            "renditionSelection": "CUSTOM",
            "renditions": [
                "HD"
            ]
        }
    }
}
```

Per ulteriori informazioni, consulta Registrazione su Amazon S3 nella Guida per l'utente del Servizio video interattivo Amazon.

- Per i dettagli sull'API, vedere GetRecordingConfigurationin AWS CLI Command Reference.

`get-stream-key`

Il seguente esempio di codice mostra come utilizzare`get-stream-key`.

AWS CLI

Come ottenere informazioni su un flusso

L'esempio `get-stream-key` seguente ottiene informazioni sulla chiave di flusso specificata.

```
aws ivs get-stream-key \
    --arn arn:aws:ivs:us-west-2:123456789012:stream-key/skSKABCDefgh --region=us-
west-2
```

Output:

```
{
```

```
    "streamKey": {
        "arn": "arn:aws:ivs:us-west-2:123456789012:stream-key/skSKABCDefgh",
        "value": "sk_us-west-2_abcdABCDefgh_567890abcdef",
        "channelArn": "arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh",
        "tags": {}
    }
}
```

Per ulteriori informazioni, consulta Creare un canale nella Guida per l'utente dello Streaming a bassa latenza di IVS.

- Per i dettagli sull'API, consulta GetStreamKey AWS CLICommand Reference.

`get-stream-session`

Il seguente esempio di codice mostra come utilizzare`get-stream-session`.

AWS CLI

Come ottenere i metadati per un flusso specificato

L'esempio `get-stream-session` seguente ottiene la configurazione dei metadati per il nome della risorsa Amazon (ARN) specificato per il canale e il flusso specificato; se `streamId` non viene fornito, viene selezionato il flusso più recente del canale.

```
aws ivs get-stream-session \
    --channel-arn 'arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh' \
    --stream-id 'mystream'
```

Output:

```
{
    "streamSession": {
        "streamId": "mystream1",
        "startTime": "2023-06-26T19:09:28+00:00",
        "channel": {
            "arn": "arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh",
            "name": "mychannel",
            "latencyMode": "LOW",
            "type": "STANDARD",
            "recordingConfigurationArn": "arn:aws:ivs:us-
west-2:123456789012:recording-configuration/ABcdef34ghIJ",
            "ingestEndpoint": "a1b2c3d4e5f6.global-contribute.live-video.net",
```

```
            "playbackUrl": "url-string",
            "authorized": false,
            "insecureIngest": false,
            "preset": ""
        },
        "ingestConfiguration": {
            "audio": {
                "channels": 2,
                "codec": "mp4a.40.2",
                "sampleRate": 8000,
                "targetBitrate": 46875,
                "track": "Track0"
            },
            "video": {
                "avcProfile": "Baseline",
                "avcLevel": "4.2",
                "codec": "avc1.42C02A",
                "encoder": "Lavf58.45.100",
                "level": "4.2",
                "profile": "Baseline",
                "targetBitrate": 8789062,
                "targetFramerate": 60,
                "track": "Track0",
                "videoHeight": 1080,
                "videoWidth": 1920
            }
        },
        "ingestConfigurations": {
            "audioConfigurations": [
                {
                    "channels": 2,
                    "codec": "mp4a.40.2",
                    "sampleRate": 8000,
                    "targetBitrate": 46875,
                    "track": "Track0"
                }
            ],
            "videoConfigurations": [
                {
                    "codec": "avc1.42C02A",
                    "encoder": "Lavf58.45.100",
                    "level": "4.2",
                    "profile": "Baseline",
                    "targetBitrate": 8789062,
```

```
                    "targetFramerate": 60,
                    "track": "Track0",
                    "videoHeight": 1080,
                    "videoWidth": 1920
                }
            ]
        },
        "recordingConfiguration": {
            "arn": "arn:aws:ivs:us-west-2:123456789012:recording-configuration/
ABcdef34ghIJ",
            "name": "test-recording-config",
            "destinationConfiguration": {
                "s3": {
                    "bucketName": "demo-recording-bucket"
                }
            },
            "state": "ACTIVE",
            "tags": {
                "key1": "value1",
                "key2": "value2"
            },
            "thumbnailConfiguration": {
                "recordingMode": "INTERVAL",
                "targetIntervalSeconds": 1,
                "resolution": "LOWEST_RESOLUTION",
                "storage": [
                    "LATEST"
                ]
            },
            "recordingReconnectWindowSeconds": 60,
            "renditionConfiguration": {
                "renditionSelection": "CUSTOM",
                "renditions": [
                    "HD"
                ]
            }
        },
        "truncatedEvents": [
            {
                "code": "StreamTakeoverInvalidPriority",
                "name": "Stream Takeover Failure",
                "type": "IVS Stream State Change",
                "eventTime": "2023-06-26T19:09:48+00:00"
            },
```

```
            {
                "name": "Stream Takeover",
                "type": "IVS Stream State Change",
                "eventTime": "2023-06-26T19:09:47+00:00"
            },
            {
                "name": "Recording Start",
                "type": "IVS Recording State Change",
                "eventTime": "2023-06-26T19:09:35+00:00"
            },
            {
                "name": "Stream Start",
                "type": "IVS Stream State Change",
                "eventTime": "2023-06-26T19:09:34+00:00"
            },
            {
                "name": "Session Created",
                "type": "IVS Stream State Change",
                "eventTime": "2023-06-26T19:09:28+00:00"
            }
        ]
    }
}
```

Per ulteriori informazioni, consulta Creare un canale nella Guida per l'utente dello Streaming a bassa latenza di IVS.

- Per i dettagli sull'API, consulta GetStreamSession AWS CLICommand Reference.

`get-stream`

Il seguente esempio di codice mostra come utilizzare`get-stream`.

AWS CLI

Come ottenere informazioni su un flusso

L'esempio `get-stream` seguente ottiene informazioni sul flusso per il canale specificato.

```
aws ivs get-stream \
    --channel-arn arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh
```

Output:

```
{
    "stream": {
        "channelArn": "arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh",
        "playbackUrl": "https://a1b2c3d4e5f6.us-west-2.playback.live-video.net/api/
video/v1/us-west-2.123456789012.channel.abcdEFGH.m3u8",
        "startTime": "2020-05-05T21:55:38Z",
        "state": "LIVE",
        "health": "HEALTHY",
        "streamId": "st-ABCDEfghij01234KLMN5678",
        "viewerCount": 1
    }
}
```

Per ulteriori informazioni, consulta Creare un canale nella Guida per l'utente dello Streaming a bassa latenza di IVS.

- Per i dettagli sull'API, consulta GetStream AWS CLICommand Reference.

`import-playback-key-pair`

Il seguente esempio di codice mostra come utilizzare`import-playback-key-pair`.

AWS CLI

Come importare la parte pubblica di una nuova coppia di chiavi

L'esempio `import-playback-key-pair` seguente importa la chiave pubblica specificata (come stringa in formato PEM) e restituisce il nome della risorsa Amazon (ARN) e l'impronta digitale della nuova coppia di chiavi.

```
aws ivs import-playback-key-pair \
    --name "my-playback-key" \
    --public-key-material "G1lbnQxOTA3BgNVBAMMMFdoeSBhcmUgeW91IGRl..."
```

Output:

```
{
    "keyPair": {
        "arn": "arn:aws:ivs:us-west-2:123456789012:playback-key/abcd1234efgh",
        "name": "my-playback-key",
        "fingerprint": "0a:1b:2c:ab:cd:ef:34:56:70:b1:b2:71:01:2a:a3:72",
```

```
        "tags": {}
    }
}
```

Per ulteriori informazioni, consulta Impostazione dei canali privati nella Guida per l'utente di Amazon Interactive Video Service.

- Per i dettagli sull'API, consulta ImportPlaybackKeyPair AWS CLICommand Reference.

`list-channels`

Il seguente esempio di codice mostra come utilizzare`list-channels`.

AWS CLI

Esempio 1: come ottenere informazioni di riepilogo su tutti i canali

L'`list-channels`esempio seguente elenca tutti i canali del tuo AWS account.

```
aws ivs list-channels
```

Output:

```
{
    "channels": [
        {
            "arn": "arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh",
            "name": "channel-1",
            "latencyMode": "LOW",
            "authorized": false,
            "insecureIngest": false,
            "preset": "",
            "playbackRestrictionPolicyArn": "",
            "recordingConfigurationArn": "arn:aws:ivs:us-
west-2:123456789012:recording-configuration/ABCD12cdEFgh",
            "tags": {},
            "type": "STANDARD"
        },
        {
            "arn": "arn:aws:ivs:us-west-2:123456789012:channel/efghEFGHijkl",
            "name": "channel-2",
            "latencyMode": "LOW",
```

```
            "authorized": false,
            "preset": "",
            "playbackRestrictionPolicyArn": "arn:aws:ivs:us-
west-2:123456789012:playback-restriction-policy/ABcdef34ghIJ",
            "recordingConfigurationArn": "",
            "tags": {},
            "type": "STANDARD"
        }
    ]
}
```

Per ulteriori informazioni, consulta Creare un canale nella Guida per l'utente dello Streaming a bassa latenza di IVS.

Esempio 2: per ottenere informazioni di riepilogo su tutti i canali, filtrate in base all'ARN specificato RecordingConfiguration

L'`list-channels`esempio seguente elenca tutti i canali AWS dell'account associati all' RecordingConfiguration ARN specificato.

```
aws ivs list-channels \
    --filter-by-recording-configuration-arn "arn:aws:ivs:us-
west-2:123456789012:recording-configuration/ABCD12cdEFgh"
```

Output:

```
{
    "channels": [
        {
            "arn": "arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh",
            "name": "channel-1",
            "latencyMode": "LOW",
            "authorized": false,
            "insecureIngest": false,
            "preset": "",
            "playbackRestrictionPolicyArn": "",
            "recordingConfigurationArn": "arn:aws:ivs:us-
west-2:123456789012:recording-configuration/ABCD12cdEFgh",
            "tags": {},
            "type": "STANDARD"
        }
    ]
```

```
}
```

Per ulteriori informazioni, consulta Registrazione su Amazon S3 nella Guida per l'utente dello Streaming a bassa latenza di IVS.

Esempio 3: per ottenere informazioni di riepilogo su tutti i canali, filtrate in base all'ARN specificato PlaybackRestrictionPolicy

L'`list-channels`esempio seguente elenca tutti i canali AWS dell'account associati all' PlaybackRestrictionPolicy ARN specificato.

```
aws ivs list-channels \
    --filter-by-playback-restriction-policy-arn "arn:aws:ivs:us-
west-2:123456789012:playback-restriction-policy/ABcdef34ghIJ"
```

Output:

```
{
    "channels": [
        {
            "arn": "arn:aws:ivs:us-west-2:123456789012:channel/efghEFGHijkl",
            "name": "channel-2",
            "latencyMode": "LOW",
            "authorized": false,
            "preset": "",
            "playbackRestrictionPolicyArn": "arn:aws:ivs:us-
west-2:123456789012:playback-restriction-policy/ABcdef34ghIJ",
            "recordingConfigurationArn": "",
            "tags": {},
            "type": "STANDARD"
        }
    ]
}
```

Per ulteriori informazioni, consulta Contenuti e spettatori indesiderati in IVS nella Guida per l'utente dello Streaming a bassa latenza di IVS.

- Per i dettagli sull'API, vedere ListChannelsin AWS CLI Command Reference.

`list-playback-key-pairs`

Il seguente esempio di codice mostra come utilizzare`list-playback-key-pairs`.

AWS CLI

Come ottenere informazioni di riepilogo su tutte le coppie di chiavi di riproduzione

L'esempio `list-playback-key-pairs` seguente restituisce informazioni su tutte le coppie di chiavi.

```
aws ivs list-playback-key-pairs
```

Output:

```
{
    "keyPairs": [
        {
            "arn": "arn:aws:ivs:us-west-2:123456789012:playback-key/abcd1234efgh",
            "name": "test-key-0",
            "tags": {}
        },
        {
            "arn": "arn:aws:ivs:us-west-2:123456789012:playback-key/ijkl5678mnop",
            "name": "test-key-1",
            "tags": {}
        }
    ]
}
```

Per ulteriori informazioni, consulta Impostazione dei canali privati nella Guida per l'utente di Amazon Interactive Video Service.

- Per i dettagli sull'API, consulta ListPlaybackKeyPairs AWS CLICommand Reference.

`list-playback-restriction-policies`

Il seguente esempio di codice mostra come utilizzare`list-playback-restriction-policies`.

AWS CLI

Come ottenere informazioni di riepilogo su tutte le policy di restrizione della riproduzione

L'`list-playback-restriction-policies`esempio seguente elenca tutte le politiche di restrizione della riproduzione per il tuo AWS account.

```
aws ivs list-playback-restriction-policies
```

Output:

```
{
    "playbackRestrictionPolicies": [
        {
            "arn": "arn:aws:ivs:us-west-2:123456789012:playback-restriction-policy/
ABcdef34ghIJ",
            "allowedCountries": [
                "US",
                "MX"
            ],
            "allowedOrigins": [
                "https://www.website1.com",
                "https://www.website2.com"
            ],
            "enableStrictOriginEnforcement": true,
            "name": "test-playback-restriction-policy",
            "tags": {
                "key1": "value1",
                "key2": "value2"
            }
        }
    ]
}
```

Per ulteriori informazioni, consulta Contenuti e spettatori indesiderati in IVS nella Guida per l'utente dello Streaming a bassa latenza di IVS.

- Per i dettagli sull'API, consulta AWS CLI Command ListPlaybackRestrictionPoliciesReference.

`list-recording-configurations`

Il seguente esempio di codice mostra come utilizzare`list-recording-configurations`.

AWS CLI

Per elencare tutte le RecordingConfiguration risorse create in questo account

L'`list-recording-configurations`esempio seguente ottiene informazioni su tutte le RecordingConfiguration risorse del tuo account.

```
aws ivs list-recording-configurations
```

Output:

```
{
    "recordingConfigurations": [
        {
            "arn": "arn:aws:ivs:us-west-2:123456789012:recording-configuration/
ABcdef34ghIJ",
            "name": "test-recording-config-1",
            "destinationConfiguration": {
                "s3": {
                    "bucketName": "demo-recording-bucket-1"
                }
            },
            "state": "ACTIVE",
            "tags": {}
        },
        {
            "arn": "arn:aws:ivs:us-west-2:123456789012:recording-configuration/
CD12abcdGHIJ",
            "name": "test-recording-config-2",
            "destinationConfiguration": {
                "s3": {
                    "bucketName": "demo-recording-bucket-2"
                }
            },
            "state": "ACTIVE",
            "tags": {}
        }
    ]
}
```

Per ulteriori informazioni, consulta Registrazione su Amazon S3 nella Guida per l'utente del Servizio video interattivo Amazon.

- Per i dettagli sull'API, consulta ListRecordingConfigurations AWS CLICommand Reference.

`list-stream-keys`

Il seguente esempio di codice mostra come utilizzare`list-stream-keys`.

AWS CLI

Come ottenere un elenco di chiavi di flusso

L'esempio `list-stream-keys` seguente elenca tutte le chiavi di flusso per un nome della risorsa Amazon (ARN) specificato.

```
aws ivs list-stream-keys \
    --channel-arn arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh
```

Output:

```
{
    "streamKeys": [
        {
            "arn": "arn:aws:ivs:us-west-2:123456789012:stream-key/abcdABCDefgh",
            "channelArn": "arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh",
            "tags": {}
        }
    ]
}
```

FFor per ulteriori informazioni, vedere Create a Channel nella IVS Low-Latency User Guide.

- Per i dettagli sull'API, consulta Command ListStreamKeysReference AWS CLI .

`list-stream-sessions`

Il seguente esempio di codice mostra come utilizzare`list-stream-sessions`.

AWS CLI

Per ottenere un riepilogo degli stream correnti e precedenti per un canale specificato nella regione corrente AWS

L'esempio `list-stream-sessions` seguente ottiene informazioni di riepilogo per i flussi di un nome della risorsa Amazon (ARN) specificato per il canale.

```
aws ivs list-stream-sessions \
    --channel-arn arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh \
```

```
    --max-results 25 \
    --next-token ""
```

Output:

```
{
    "nextToken": "set-2",
    "streamSessions": [
        {
            "startTime": 1641578182,
            "endTime": 1641579982,
            "hasErrorEvent": false,
            "streamId": "mystream"
        }
        ...
    ]
}
```

Per ulteriori informazioni, consulta Creare un canale nella Guida per l'utente dello Streaming a bassa latenza di IVS.

- Per i dettagli sull'API, consulta ListStreamSessions AWS CLICommand Reference.

`list-streams`

Il seguente esempio di codice mostra come utilizzare`list-streams`.

AWS CLI

Come ottenere un elenco dei flussi in tempo reale con il relativo stato

L'`list-streams`esempio seguente elenca tutti i live streaming del tuo AWS account.

```
aws ivs list-streams
```

Output:

```
{
   "streams": [
        {
            "channelArn": "arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh",
```

```
            "state": "LIVE",
            "health": "HEALTHY",
            "streamId": "st-ABCDEfghij01234KLMN5678",
            "viewerCount": 1
        }
    ]
}
```

Per ulteriori informazioni, consulta Creare un canale nella Guida per l'utente dello Streaming a bassa latenza di IVS.

- Per i dettagli sull'API, consulta ListStreams AWS CLICommand Reference.

`list-tags-for-resource`

Il seguente esempio di codice mostra come utilizzare`list-tags-for-resource`.

AWS CLI

Per elencare tutti i tag di una AWS risorsa (ad esempio: canale, chiave stream)

L'esempio `list-tags-for-resource` seguente elenca tutti i tag per un nome della risorsa Amazon (ARN) specificato per la risorsa.

```
aws ivs list-tags-for-resource \
    --resource-arn arn:aws:ivs:us-west-2:12345689012:channel/abcdABCDefgh
```

Output:

```
{
    "tags":
    {
        "key1": "value1",
        "key2": "value2"
    }
}
```

Per ulteriori informazioni, consulta Tagging nella Guida di riferimento all'API del Servizio video interattivo Amazon (Amazon IVS).

- Per i dettagli sull'API, consulta ListTagsForResource AWS CLICommand Reference.

`put-metadata`

Il seguente esempio di codice mostra come utilizzare`put-metadata`.

AWS CLI

Come inserire i metadati nel flusso attivo per un canale specificato

L'esempio `put-metadata` seguente inserisce i metadati indicati nel flusso per il canale specificato.

```
aws ivs put-metadata \
    --channel-arn arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh \
    --metadata '{"my": "metadata"}'
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Creare un canale nella Guida per l'utente dello Streaming a bassa latenza di IVS.

- Per i dettagli sull'API, consulta PutMetadata AWS CLICommand Reference.

`start-viewer-session-revocation`

Il seguente esempio di codice mostra come utilizzare`start-viewer-session-revocation`.

AWS CLI

Come revocare una sessione di visualizzazione per una coppia ARN del canale e ID visualizzazione

L'esempio `start-viewer-session-revocation` seguente avvia il processo di revoca della sessione di visualizzazione associata a un nome della risorsa Amazon (ARN) del canale specificato e a un ID visualizzazione, fino al numero di versione della sessione specificato incluso. Se la versione non viene fornita, il valore predefinito è 0.

```
aws ivs batch-start-viewer-session-revocation \
    --channel-arn arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh \
    --viewer-id abcdefg \
    --viewer-session-versions-less-than-or-equal-to 1234567890
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Impostazione dei canali privati nella Guida per l'utente di Amazon Interactive Video Service.

- Per i dettagli sull'API, consulta StartViewerSessionRevocation AWS CLICommand Reference.

`stop-stream`

Il seguente esempio di codice mostra come utilizzare`stop-stream`.

AWS CLI

Come arrestare un flusso specificato

L'esempio `stop-stream` seguente arresta il flusso sul canale specificato.

```
aws ivs stop-stream \
    --channel-arn arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Creare un canale nella Guida per l'utente dello Streaming a bassa latenza di IVS.

- Per i dettagli sull'API, consulta StopStream AWS CLICommand Reference.

`tag-resource`

Il seguente esempio di codice mostra come utilizzare`tag-resource`.

AWS CLI

Per aggiungere o aggiornare tag per una AWS risorsa (ad esempio: canale, chiave stream)

L'esempio `tag-resource` seguente aggiunge o aggiorna i tag per un nome della risorsa Amazon (ARN) specificato per la risorsa.

```
aws ivs tag-resource \
    --resource-arn arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh \
    --tags "tagkey1=tagvalue1, tagkey2=tagvalue2"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Tagging nella Guida di riferimento all'API del Servizio video interattivo Amazon (Amazon IVS).

- Per i dettagli sull'API, consulta TagResource AWS CLICommand Reference.

`untag-resource`

Il seguente esempio di codice mostra come utilizzare`untag-resource`.

AWS CLI

Per rimuovere i tag da una AWS risorsa (ad esempio: canale, chiave stream)

L'esempio `untag-resource` seguente rimuove i tag per un nome della risorsa Amazon (ARN) specificato per la risorsa.

```
aws ivs untag-resource \
    --resource-arn arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh \
    --tag-keys "tagkey1, tagkey2"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Tagging nella Guida di riferimento all'API del Servizio video interattivo Amazon (Amazon IVS).

- Per i dettagli sull'API, consulta UntagResource AWS CLICommand Reference.

`update-channel`

Il seguente esempio di codice mostra come utilizzare`update-channel`.

AWS CLI

Esempio 1: come aggiornare le informazioni di configurazione di un canale

L'esempio `update-channel` seguente aggiorna la configurazione del canale con il nome della risorsa Amazon (ARN) specificato per modificare il nome del canale. Ciò non influisce sul flusso in corso di questo canale; è necessario interrompere e riavviare il flusso per rendere effettive le modifiche.

```
aws ivs update-channel \
    --arn 'arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh' \
```

```
    --name 'channel-1' \
    --insecure-ingest
```

Output:

```
{
    "channel": {
        "arn": "arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh",
        "name": "channel-1",
        "latencyMode": "LOW",
        "containerFormat": "TS",
        "multitrackInputConfiguration": {
            "enabled": false,
            "maximumResolution": "FULL_HD",
            "policy": "ALLOW"
        },
        "type": "STANDARD",
        "playbackRestrictionPolicyArn": "",
        "recordingConfigurationArn": "",
        "srt": {
            "endpoint": "a1b2c3d4e5f6.srt.live-video.net",
            "passphrase":
 "AB1C2defGHijkLMNo3PqQRstUvwxyzaBCDEfghh4ijklMN5opqrStuVWxyzAbCDEfghIJ"
        },
        "ingestEndpoint": "a1b2c3d4e5f6.global-contribute.live-video.net",
        "insecureIngest": true,
        "playbackUrl": "https://a1b2c3d4e5f6.us-west-2.playback.live-video.net/api/
video/v1/us-west-2.123456789012.channel.abcdEFGH.m3u8",
        "preset": "",
        "authorized": false,
        "tags": {}
}
```

Per ulteriori informazioni, consulta Creare un canale nella Guida per l'utente dello Streaming a bassa latenza di IVS.

Esempio 2: come aggiornare la configurazione di un canale per abilitare la registrazione

L'esempio `update-channel` seguente aggiorna la configurazione del canale con il nome della risorsa Amazon (ARN) specificato per abilitare la registrazione. Ciò non influisce sul flusso in corso di questo canale; è necessario interrompere e riavviare il flusso per rendere effettive le modifiche.

```
aws ivs update-channel \
    --arn 'arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh' \
    --no-insecure-ingest \
    --recording-configuration-arn 'arn:aws:ivs:us-west-2:123456789012:recording-
configuration/ABCD12cdEFgh'
```

Output:

```
{
    "channel": {
        "arn": "arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh",
        "name": "test-channel-with-recording",
        "latencyMode": "LOW",
        "containerFormat": "TS",
        "multitrackInputConfiguration": {
            "enabled": false,
            "maximumResolution": "FULL_HD",
            "policy": "ALLOW"
        },
        "type": "STANDARD",
        "playbackRestrictionPolicyArn": "",
        "recordingConfigurationArn": "arn:aws:ivs:us-west-2:123456789012:recording-
configuration/ABCD12cdEFgh",
        "srt": {
            "endpoint": "a1b2c3d4e5f6.srt.live-video.net",
            "passphrase":
 "BA1C2defGHijkLMNo3PqQRstUvwxyzaBCDEfghh4ijklMN5opqrStuVWxyzAbCDEfghIJ"
        },
        "ingestEndpoint": "a1b2c3d4e5f6.global-contribute.live-video.net",
        "insecureIngest": false,
        "playbackUrl": "https://a1b2c3d4e5f6.us-west-2.playback.live-video.net/api/
video/v1/us-west-2.123456789012.channel.abcdEFGH.m3u8",
        "preset": "",
        "authorized": false,
        "tags": {}
    }
}
```

Per ulteriori informazioni, consulta Registrazione su Amazon S3 nella Guida per l'utente dello Streaming a bassa latenza di IVS.

Esempio 3: come aggiornare la configurazione di un canale per disabilitare la registrazione

L'esempio `update-channel` seguente aggiorna la configurazione del canale con il nome della risorsa Amazon (ARN) specificato per disabilitare la registrazione. Ciò non influisce sul flusso in corso di questo canale; è necessario interrompere e riavviare il flusso per rendere effettive le modifiche.

```
aws ivs update-channel \
    --arn 'arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh' \
    --recording-configuration-arn ''
```

Output:

```
{
    "channel": {
        "arn": "arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh",
        "name": "test-channel-with-recording",
        "latencyMode": "LOW",
        "containerFormat": "TS",
        "multitrackInputConfiguration": {
            "enabled": false,
            "maximumResolution": "FULL_HD",
            "policy": "ALLOW"
        },
        "type": "STANDARD",
        "playbackRestrictionPolicyArn": "",
        "recordingConfigurationArn": "",
        "srt": {
            "endpoint": "a1b2c3d4e5f6.srt.live-video.net",
            "passphrase":
 "AB1C2edfGHijkLMNo3PqQRstUvwxyzaBCDEfghh4ijklMN5opqrStuVWxyzAbCDEfghIJ"
        },
        "ingestEndpoint": "a1b2c3d4e5f6.global-contribute.live-video.net",
        "insecureIngest": false,
        "playbackUrl": "https://a1b2c3d4e5f6.us-west-2.playback.live-video.net/api/
video/v1/us-west-2.123456789012.channel.abcdEFGH.m3u8",
        "preset": "",
        "authorized": false,
        "tags": {}
    }
}
```

Per ulteriori informazioni, consulta Registrazione su Amazon S3 nella Guida per l'utente dello Streaming a bassa latenza di IVS.

Esempio 4: come aggiornare la configurazione di un canale per abilitare la restrizione della riproduzione

L'esempio `update-channel` seguente aggiorna la configurazione del canale con il nome della risorsa Amazon (ARN) specificato per applicare una policy di restrizione della riproduzione. Ciò non influisce sul flusso in corso di questo canale; è necessario interrompere e riavviare il flusso per rendere effettive le modifiche.

```
aws ivs update-channel \
    --arn 'arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh' \
    --no-insecure-ingest \
    --playback-restriction-policy-arn 'arn:aws:ivs:us-west-2:123456789012:playback-restriction-policy/ABcdef34ghIJ'
```

Output:

```
{
    "channel": {
        "arn": "arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh",
        "name": "test-channel-with-playback-restriction-policy",
        "latencyMode": "LOW",
        "containerFormat": "TS",
        "multitrackInputConfiguration": {
            "enabled": false,
            "maximumResolution": "FULL_HD",
            "policy": "ALLOW"
        },
        "type": "STANDARD",
        "playbackRestrictionPolicyArn": "arn:aws:ivs:us-
west-2:123456789012:playback-restriction-policy/ABcdef34ghIJ",
        "recordingConfigurationArn": "",
        "srt": {
            "endpoint": "a1b2c3d4e5f6.srt.live-video.net",
            "passphrase":
 "AB1C2defGHijkLMNo3PqQRstUvwxyzaCBDEfghh4ijklMN5opqrStuVWxyzAbCDEfghIJ"
        },
        "ingestEndpoint": "a1b2c3d4e5f6.global-contribute.live-video.net",
        "insecureIngest": false,
        "playbackUrl": "https://a1b2c3d4e5f6.us-west-2.playback.live-video.net/api/
video/v1/us-west-2.123456789012.channel.abcdEFGH.m3u8",
        "preset": "",
        "authorized": false,
```

```
        "tags": {}
    }
}
```

Per ulteriori informazioni, consulta Contenuti e spettatori indesiderati in IVS nella Guida per l'utente dello Streaming a bassa latenza di IVS.

Esempio 5: come aggiornare la configurazione di un canale per disabilitare la restrizione della riproduzione

L'esempio `update-channel` seguente aggiorna la configurazione del canale con il nome della risorsa Amazon (ARN) specificato per disabilitare la restrizione della riproduzione. Ciò non influisce sul flusso in corso di questo canale; è necessario interrompere e riavviare il flusso per rendere effettive le modifiche.

```
aws ivs update-channel \
    --arn 'arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh' \
    --playback-restriction-policy-arn ''
```

Output:

```
{
    "channel": {
        "arn": "arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh",
        "name": "test-channel-with-playback-restriction-policy",
        "latencyMode": "LOW",
        "containerFormat": "TS",
        "multitrackInputConfiguration": {
            "enabled": false,
            "maximumResolution": "FULL_HD",
            "policy": "ALLOW"
        },
        "type": "STANDARD",
        "playbackRestrictionPolicyArn": "",
        "recordingConfigurationArn": "",
        "srt": {
            "endpoint": "a1b2c3d4e5f6.srt.live-video.net",
            "passphrase":
 "AB1C2defGHijkLMNo3PqQRstUvwxyzaBCDeFghh4ijklMN5opqrStuVWxyzAbCDEfghIJ"
        },
        "ingestEndpoint": "a1b2c3d4e5f6.global-contribute.live-video.net",
```

```
        "insecureIngest": false,
        "playbackUrl": "https://a1b2c3d4e5f6.us-west-2.playback.live-video.net/api/
video/v1/us-west-2.123456789012.channel.abcdEFGH.m3u8",
        "preset": "",
        "authorized": false,
        "tags": {}
    }
}
```

Per ulteriori informazioni, consulta Contenuti e spettatori indesiderati in IVS nella Guida per l'utente dello Streaming a bassa latenza di IVS.

Esempio 6: come aggiornare la configurazione di un canale per abilitare il multitraccia

L'esempio `update-channel` seguente aggiorna la configurazione del canale con il nome della risorsa Amazon (ARN) specificato per abilitare il multitraccia. Ciò non influisce sul flusso in corso di questo canale; è necessario interrompere e riavviare il flusso per rendere effettive le modifiche.

```
aws ivs update-channel \
    --arn 'arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh' \
    --container-format 'FRAGMENTED_MP4' \
    --multitrack-input-configuration '{"enabled": true,"maximumResolution":
 "FULL_HD","policy": "ALLOW"}'
```

Output:

```
{
    "channel": {
        "arn": "arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh",
        "containerFormat": "FRAGMENTED_MP4",
        "name": "test-channel-with-multitrack",
        "latencyMode": "LOW",
        "multitrackInputConfiguration": {
            "enabled": true,
            "maximumResolution": "FULL_HD",
            "policy": "ALLOW"
        },
        "type": "STANDARD",
        "playbackRestrictionPolicyArn": "",
        "recordingConfigurationArn": "",
        "srt": {
            "endpoint": "a1b2c3d4e5f6.srt.live-video.net",
```

```
            "passphrase":
 "AB1C2defGHijkLMNo3PqQRstUvwxyzaCBDEfghh4ijklMN5opqrStuVWxyzAbCDEfghIJ"
        },
        "ingestEndpoint": "a1b2c3d4e5f6.global-contribute.live-video.net",
        "insecureIngest": false,
        "playbackUrl": "https://a1b2c3d4e5f6.us-west-2.playback.live-video.net/api/
video/v1/us-west-2.123456789012.channel.abcdEFGH.m3u8",
        "preset": "",
        "authorized": false,
        "tags": {}
    }
}
```

Per ulteriori informazioni, consulta Contenuti e spettatori indesiderati in IVS nella Guida per l'utente dello Streaming a bassa latenza di IVS.

Esempio 7: come aggiornare la configurazione di un canale per disabilitare la restrizione della riproduzione

L'esempio `update-channel` seguente aggiorna la configurazione del canale con il nome della risorsa Amazon (ARN) specificato per disabilitare il multitraccia. Ciò non influisce sul flusso in corso di questo canale; è necessario interrompere e riavviare il flusso per rendere effettive le modifiche.

```
aws ivs update-channel \
    --arn 'arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh' \
    --container-format 'TS' \
    --multitrack-input-configuration '{"enabled": false}'
```

Output:

```
{
    "channel": {
        ""arn": "arn:aws:ivs:us-west-2:123456789012:channel/abcdABCDefgh",
        "containerFormat": "TS",
        "name": "test-channel-with-multitrack",
        "latencyMode": "LOW",
        "multitrackInputConfiguration": {
            "enabled": false,
            "maximumResolution": "FULL_HD",
            "policy": "ALLOW"
        },
```

```
        "type": "STANDARD",
        "playbackRestrictionPolicyArn": "",
        "recordingConfigurationArn": "",
        "srt": {
            "endpoint": "a1b2c3d4e5f6.srt.live-video.net",
            "passphrase":
 "AB1C2defGHijkLMNo3PqQRstUvwxyzaCBDEfghh4ijklMN5opqrStuVWxyzAbCDEfghIJ"
        },
        "ingestEndpoint": "a1b2c3d4e5f6.global-contribute.live-video.net",
        "insecureIngest": false,
        "playbackUrl": "https://a1b2c3d4e5f6.us-west-2.playback.live-video.net/api/
video/v1/us-west-2.123456789012.channel.abcdEFGH.m3u8",
        "preset": "",
        "authorized": false,
        "tags": {}
    }
}
```

Per ulteriori informazioni, consulta Contenuti e spettatori indesiderati in IVS nella Guida per l'utente dello Streaming a bassa latenza di IVS.

- Per i dettagli sull'API, consulta UpdateChannel AWS CLICommand Reference.

`update-playback-restriction-policy`

Il seguente esempio di codice mostra come utilizzare`update-playback-restriction-policy`.

AWS CLI

Come aggiornare una policy di restrizione della riproduzione

L'esempio `update-playback-restriction-policy` seguente aggiorna la policy di restrizione della riproduzione con il nome della risorsa Amazon (ARN) specificato per disabilitare l'applicazione rigorosa dell'origine. Ciò non influisce sul flusso in corso del canale associato; è necessario interrompere e riavviare il flusso per rendere effettive le modifiche.

```
aws ivs update-playback-restriction-policy \
    --arn "arn:aws:ivs:us-west-2:123456789012:playback-restriction-policy/
ABcdef34ghIJ" \
    --no-enable-strict-origin-enforcement
```

Output:

```
{
    "playbackRestrictionPolicy": {
        "arn": "arn:aws:ivs:us-west-2:123456789012:playback-restriction-policy/
ABcdef34ghIJ",
        "allowedCountries": [
            "US",
            "MX"
        ],
        "allowedOrigins": [
            "https://www.website1.com",
            "https://www.website2.com"
        ],
        "enableStrictOriginEnforcement": false,
        "name": "test-playback-restriction-policy",
        "tags": {
            "key1": "value1",
            "key2": "value2"
        }
    }
}
```

Per ulteriori informazioni, consulta Contenuti e spettatori indesiderati in IVS nella Guida per l'utente dello Streaming a bassa latenza di IVS.

- Per i dettagli sull'API, consulta UpdatePlaybackRestrictionPolicy AWS CLICommand Reference.

Esempi di Amazon IVS Chat con AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando la chat AWS Command Line Interface con Amazon IVS.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`create-chat-token`

Il seguente esempio di codice mostra come usare`create-chat-token`.

AWS CLI

Come creare un token di chat

L'`create-chat-token`esempio seguente crea un token di chat crittografato che viene utilizzato per stabilire una WebSocket connessione individuale a una stanza. Il token è valido per un minuto e una connessione (sessione) stabilita con il token è valida per la durata specificata.

```
aws ivschat create-chat-token \
    --roomIdentifier "arn:aws:ivschat:us-west-2:12345689012:room/g1H2I3j4k5L6", \
    --userId" "11231234" \
    --capabilities "SEND_MESSAGE", \
    --sessionDurationInMinutes" 30
```

Output:

```
{
    "token": "ACEGmnoq#1rstu2...BDFH3vxwy!4hlm!#5",
    "sessionExpirationTime": "2022-03-16T04:44:09+00:00"
    "state": "CREATING",
    "tokenExpirationTime": "2022-03-16T03:45:09+00:00"
}
```

Per ulteriori informazioni, consulta Passaggio 3: autenticare e autorizzare i client di chat nella Guida per l'utente del Servizio video interattivo Amazon (Amazon IVS).

- Per i dettagli sull'API, consulta CreateChatToken AWS CLICommand Reference.

`create-logging-configuration`

Il seguente esempio di codice mostra come utilizzare`create-logging-configuration`.

AWS CLI

Per creare una LoggingConfiguration risorsa di chat

L'`create-logging-configuration`esempio seguente crea una LoggingConfiguration risorsa che consente ai client di archiviare e registrare i messaggi inviati.

```
aws ivschat create-logging-configuration \
    --destination-configuration s3={bucketName=demo-logging-bucket} \
    --name "test-logging-config" \
    --tags "key1=value1, key2=value2"
```

Output:

```
{
    "arn": "arn:aws:ivschat:us-west-2:123456789012:logging-configuration/
ABcdef34ghIJ",
    "createTime": "2022-09-14T17:48:00.653000+00:00",
    "destinationConfiguration": {
        "s3": {
            "bucketName": "demo-logging-bucket"
        }
    },
    "id": "ABcdef34ghIJ",
    "name": "test-logging-config",
    "state": "ACTIVE",
    "tags": { "key1" : "value1", "key2" : "value2" },
    "updateTime": "2022-09-14T17:48:01.104000+00:00"
}
```

Per ulteriori informazioni, consulta Guida introduttiva alla Chat Amazon IVS nella Guida per l'utente del Servizio video interattivo Amazon (Amazon IVS).

- Per i dettagli sull'API, vedere CreateLoggingConfigurationin AWS CLI Command Reference.

`create-room`

Il seguente esempio di codice mostra come utilizzare`create-room`.

AWS CLI

Come creare una stanza

L'esempio `create-room` seguente crea una nuova stanza.

```
aws ivschat create-room \
```

```
    --name "test-room-1" \
    --logging-configuration-identifiers "arn:aws:ivschat:us-
west-2:123456789012:logging-configuration/ABcdef34ghIJ" \
    --maximum-message-length 256 \
    --maximum-message-rate-per-second 5
```

Output:

```
{
    "arn": "arn:aws:ivschat:us-west-2:12345689012:room/g1H2I3j4k5L6",
    "id": "g1H2I3j4k5L6",
    "createTime": "2022-03-16T04:44:09+00:00",
    "loggingConfigurationIdentifiers": ["arn:aws:ivschat:us-
west-2:123456789012:logging-configuration/ABcdef34ghIJ"],
    "maximumMessageLength": 256,
    "maximumMessageRatePerSecond": 5,
    "name": "test-room-1",
    "tags": {}
    "updateTime": "2022-03-16T07:22:09+00:00"
}
```

Per ulteriori informazioni, consulta Passaggio 2: creare una chat room nella Guida per l'utente del Servizio video interattivo Amazon (Amazon IVS).

- Per i dettagli sull'API, consulta CreateRoom AWS CLICommand Reference.

`delete-logging-configuration`

Il seguente esempio di codice mostra come utilizzare`delete-logging-configuration`.

AWS CLI

Per eliminare una LoggingConfiguration risorsa di chat

L'`delete-logging-configuration`esempio seguente elimina la LoggingConfiguration risorsa per l'ARN specificato.

```
aws ivschat delete-logging-configuration \
    --identifier "arn:aws:ivschat:us-west-2:123456789012:logging-configuration/
ABcdef34ghIJ"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Guida introduttiva alla Chat Amazon IVS nella Guida per l'utente del Servizio video interattivo Amazon (Amazon IVS).

- Per i dettagli sull'API, vedere DeleteLoggingConfigurationin AWS CLI Command Reference.

`delete-message`

Il seguente esempio di codice mostra come utilizzare`delete-message`.

AWS CLI

Come eliminare messaggi da una stanza specificata

L'esempio `delete-message` seguente invia un evento alla stanza specificata, che ordina ai client di eliminare il messaggio specificato, ovvero di annullarne la visualizzazione ed eliminarlo dalla cronologia chat del client.

```
aws ivschat delete-message \
    --roomIdentifier "arn:aws:ivschat:us-west-2:12345689012:room/g1H2I3j4k5L6" \
    --id "ABC123def456" \
    --reason "Message contains profanity"
```

Output:

```
{
    "id": "12345689012"
}
```

Per ulteriori informazioni, consulta Guida introduttiva alla Chat Amazon IVS nella Guida per l'utente del Servizio video interattivo Amazon (Amazon IVS).

- Per i dettagli sull'API, consulta DeleteMessage AWS CLICommand Reference.

`delete-room`

Il seguente esempio di codice mostra come utilizzare`delete-room`.

AWS CLI

Come eliminare una stanza

L'esempio `delete-room` seguente elimina la stanza specificata. I client connessi vengono disconnessi. In caso di successo, restituisce HTTP 204 con un corpo di risposta vuoto.

```
aws ivschat delete-room \
    --identifier "arn:aws:ivschat:us-west-2:12345689012:room/g1H2I3j4k5L6"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Guida introduttiva alla Chat Amazon IVS nella Guida per l'utente del Servizio video interattivo Amazon (Amazon IVS).

- Per i dettagli sull'API, consulta DeleteRoom AWS CLICommand Reference.

`disconnect-user`

Il seguente esempio di codice mostra come utilizzare`disconnect-user`.

AWS CLI

Come disconnettere un utente da una stanza

L'esempio `disconnect-user` seguente disconnette tutte le connessioni dell'utente specificato dalla stanza specificata. In caso di successo, restituisce HTTP 200 con un corpo di risposta vuoto.

```
aws ivschat disconnect-user \
    --roomIdentifier "arn:aws:ivschat:us-west-2:12345689012:room/g1H2I3j4k5L6" \
    --userId "ABC123def456" \
    --reason "Violated terms of service"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Guida introduttiva alla Chat Amazon IVS nella Guida per l'utente del Servizio video interattivo Amazon (Amazon IVS).

- Per i dettagli sull'API, consulta DisconnectUser AWS CLICommand Reference.

`get-logging-configuration`

Il seguente esempio di codice mostra come utilizzare`get-logging-configuration`.

AWS CLI

Per ottenere informazioni su una LoggingConfiguration risorsa

L'`get-logging-configuration`esempio seguente ottiene informazioni sulla LoggingConfiguration risorsa per l'ARN specificato.

```
aws ivschat get-logging-configuration \
    --identifier "arn:aws:ivschat:us-west-2:123456789012:logging-configuration/
ABcdef34ghIJ"
```

Output:

```
{
    "arn": "arn:aws:ivschat:us-west-2:123456789012:logging-configuration/
ABcdef34ghIJ",
    "createTime": "2022-09-14T17:48:00.653000+00:00",
    "destinationConfiguration": {
        "s3": {
            "bucketName": "demo-logging-bucket"
        }
    },
    "id": "ABcdef34ghIJ",
    "name": "test-logging-config",
    "state": "ACTIVE",
    "tags": { "key1" : "value1", "key2" : "value2" },
    "updateTime": "2022-09-14T17:48:01.104000+00:00"
}
```

Per ulteriori informazioni, consulta Guida introduttiva alla Chat Amazon IVS nella Guida per l'utente del Servizio video interattivo Amazon (Amazon IVS).

- Per i dettagli sull'API, vedere GetLoggingConfigurationin AWS CLI Command Reference.

`get-room`

Il seguente esempio di codice mostra come utilizzare`get-room`.

AWS CLI

Come ottenere la stanza specificata

L'esempio `get-room` seguente ottiene informazioni sulla stanza specificata.

```
aws ivschat get-room \
    --identifier "arn:aws:ivschat:us-west-2:12345689012:room/g1H2I3j4k5L6"
```

Output:

```
{
    "arn": "arn:aws:ivschat:us-west-2:12345689012:room/g1H2I3j4k5L6",
    "createTime": "2022-03-16T04:44:09+00:00",
    "id": "g1H2I3j4k5L6",
    "loggingConfigurationIdentifiers": ["arn:aws:ivschat:us-
west-2:123456789012:logging-configuration/ABcdef34ghIJ"],
    "maximumMessageLength": 256,
    "maximumMessageRatePerSecond": 5,
    "name": "test-room-1",
    "tags": {},
    "updateTime": "2022-03-16T07:22:09+00:00"
}
```

Per ulteriori informazioni, consulta Guida introduttiva alla Chat Amazon IVS nella Guida per l'utente del Servizio video interattivo Amazon (Amazon IVS).

- Per i dettagli sull'API, consulta GetRoom AWS CLICommand Reference.

`list-logging-configurations`

Il seguente esempio di codice mostra come utilizzare`list-logging-configurations`.

AWS CLI

Per ottenere informazioni di riepilogo su tutte le configurazioni di registrazione per l'utente nella AWS regione in cui viene elaborata la richiesta API

L'`list-logging-configurations`esempio seguente elenca le informazioni su tutte le LoggingConfiguration risorse per l'utente nella AWS regione in cui viene elaborata la richiesta API.

```
aws ivschat list-logging-configurations \
    --max-results 2 \
    --next-token ""
```

Output:

```
{
    "nextToken": "set-2",
    "loggingConfigurations": [
        {
            "arn": "arn:aws:ivschat:us-west-2:123456789012:logging-configuration/
ABcdef34ghIJ",
            "createTime": "2022-09-14T17:48:00.653000+00:00",
            "destinationConfiguration": {
                "s3": {
                    "bucketName": "demo-logging-bucket"
                }
            },
            "id": "ABcdef34ghIJ",
            "name": "test-logging-config",
            "state": "ACTIVE",
            "tags": { "key1" : "value1", "key2" : "value2" },
            "updateTime": "2022-09-14T17:48:01.104000+00:00"
        }
        ...
    ]
}
```

Per ulteriori informazioni, consulta Guida introduttiva alla Chat Amazon IVS nella Guida per l'utente del Servizio video interattivo Amazon (Amazon IVS).

- Per i dettagli sull'API, consulta ListLoggingConfigurations AWS CLICommand Reference.

`list-rooms`

Il seguente esempio di codice mostra come utilizzare`list-rooms`.

AWS CLI

Come ottenere informazioni di riepilogo su tutte le tue stanze nella Regione corrente

L'`list-rooms`esempio seguente ottiene informazioni di riepilogo su tutte le sale della AWS regione in cui viene elaborata la richiesta. I risultati sono ordinati in ordine decrescente di updateTime.

```
aws ivschat list-rooms \
```

```
    --logging-configuration-identifier "arn:aws:ivschat:us-
west-2:123456789012:logging-configuration/ABcdef34ghIJ" \
    --max-results 10 \
    --next-token ""
```

Output:

```
{
    "nextToken": "page3",
    "rooms": [
        {
            "arn:aws:ivschat:us-west-2:12345689012:room/g1H2I3j4k5L6",
            "createTime": "2022-03-16T04:44:09+00:00",
            "id": "g1H2I3j4k5L6",
            "loggingConfigurationIdentifiers": ["arn:aws:ivschat:us-
west-2:123456789012:logging-configuration/ABcdef34ghIJ"],
            "name": "test-room-1",
            "tags": {},
            "updateTime": "2022-03-16T07:22:09+00:00"
        }
    ]
}
```

Per ulteriori informazioni, consulta Guida introduttiva alla Chat Amazon IVS nella Guida per l'utente del Servizio video interattivo Amazon (Amazon IVS).

- Per i dettagli sull'API, consulta ListRooms AWS CLICommand Reference.

`list-tags-for-resource`

Il seguente esempio di codice mostra come utilizzare`list-tags-for-resource`.

AWS CLI

Per elencare tutti i tag di una AWS risorsa (ad esempio: Room)

L'esempio `list-tags-for-resource` seguente elenca tutti i tag per un nome della risorsa Amazon (ARN) specificato per la risorsa.

```
aws ivschat list-tags-for-resource \
    --resource-arn arn:aws:ivschat:us-west-2:12345689012:room/g1H2I3j4k5L6
```

Output:

```
{
    "tags":
    {
        "key1": "value1",
        "key2": "value2"
    }
}
```

Per ulteriori informazioni, consulta Tagging nella Guida di riferimento all'API del Servizio video interattivo Amazon (Amazon IVS).

- Per i dettagli sull'API, consulta ListTagsForResource AWS CLICommand Reference.

`send-event`

Il seguente esempio di codice mostra come utilizzare`send-event`.

AWS CLI

Come inviare un evento a una stanza

L'esempio `send-event` seguente invia l'evento specificato alla stanza specificata.

```
aws ivschat send-event \
    --roomIdentifier "arn:aws:ivschat:us-west-2:12345689012:room/g1H2I3j4k5L6" \
    --eventName "SystemMessage" \
    --attributes \
        "msgType"="user-notification", \
        "msgText"="This chat room will close in 15 minutes."
```

Output:

```
{
    "id": "12345689012"
}
```

Per ulteriori informazioni, consulta Guida introduttiva alla Chat Amazon IVS nella Guida per l'utente del Servizio video interattivo Amazon (Amazon IVS).

- Per i dettagli sull'API, consulta SendEvent AWS CLICommand Reference.

`tag-resource`

Il seguente esempio di codice mostra come utilizzare`tag-resource`.

AWS CLI

Per aggiungere o aggiornare i tag per una AWS risorsa (ad esempio: Room)

L'esempio `tag-resource` seguente aggiunge o aggiorna i tag per un nome della risorsa Amazon (ARN) specificato per la risorsa. In caso di successo, restituisce HTTP 200 con un corpo di risposta vuoto.

```
aws ivschat tag-resource \
    --resource-arn arn:aws:ivschat:us-west-2:12345689012:room/g1H2I3j4k5L6 \
    --tags "tagkey1=tagkeyvalue1, tagkey2=tagkeyvalue2"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Tagging nella Guida di riferimento all'API del Servizio video interattivo Amazon (Amazon IVS).

- Per i dettagli sull'API, consulta TagResource AWS CLICommand Reference.

`untag-resource`

Il seguente esempio di codice mostra come utilizzare`untag-resource`.

AWS CLI

Per rimuovere i tag da una AWS risorsa (ad esempio: Room)

L'esempio `untag-resource` seguente rimuove i tag per un nome della risorsa Amazon (ARN) specificato per la risorsa. In caso di successo, restituisce HTTP 200 con un corpo di risposta vuoto.

```
aws ivschat untag-resource \
    --resource-arn arn:aws:ivschat:us-west-2:12345689012:room/g1H2I3j4k5L6 \
    --tag-keys "tagkey1, tagkey2"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Tagging nella Guida di riferimento all'API del Servizio video interattivo Amazon (Amazon IVS).

- Per i dettagli sull'API, consulta UntagResource AWS CLICommand Reference.

`update-logging-configuration`

Il seguente esempio di codice mostra come utilizzare`update-logging-configuration`.

AWS CLI

Come aggiornare una configurazione di registrazione di log

L'`update-logging-configuration`esempio seguente aggiorna una LoggingConfiguration risorsa con i dati forniti.

```
aws ivschat update-logging-configuration \
    --destination-configuration s3={bucketName=demo-logging-bucket} \
    --identifier "arn:aws:ivschat:us-west-2:123456789012:logging-configuration/
ABcdef34ghIJ" \
    --name "test-logging-config"
```

Output:

```
{
    "arn": "arn:aws:ivschat:us-west-2:123456789012:logging-configuration/
ABcdef34ghIJ",
    "createTime": "2022-09-14T17:48:00.653000+00:00",
    "destinationConfiguration": {
        "s3": {
            "bucketName": "demo-logging-bucket"
        }
    },
    "id": "ABcdef34ghIJ",
    "name": "test-logging-config",
    "state": "ACTIVE",
    "tags": { "key1" : "value1", "key2" : "value2" },
    "updateTime": "2022-09-14T17:48:01.104000+00:00"
}
```

Per ulteriori informazioni, consulta Guida introduttiva alla Chat Amazon IVS nella Guida per l'utente del Servizio video interattivo Amazon (Amazon IVS).

- Per i dettagli sull'API, vedere UpdateLoggingConfigurationin AWS CLI Command Reference.

`update-room`

Il seguente esempio di codice mostra come utilizzareupdate-room.

AWS CLI

Come aggiornare una configurazione di sincronizzazione

L'esempio update-room seguente aggiorna la configurazione della stanza specificata con i dati forniti.

```
aws ivschat update-room \
    --identifier "arn:aws:ivschat:us-west-2:12345689012:room/g1H2I3j4k5L6" \
    --logging-configuration-identifiers "arn:aws:ivschat:us-
west-2:123456789012:logging-configuration/ABcdef34ghIJ" \
    --name "chat-room-a" \
    --maximum-message-length 256 \
    --maximum-message-rate-per-second 5
```

Output:

```
{
    "arn": "arn:aws:ivschat:us-west-2:12345689012:room/g1H2I3j4k5L6",
    "createTime": "2022-03-16T04:44:09+00:00",
    "id": "g1H2I3j4k5L6",
    "loggingConfigurationIdentifiers": ["arn:aws:ivschat:us-
west-2:123456789012:logging-configuration/ABcdef34ghIJ"],
    "maximumMessageLength": 256,
    "maximumMessageRatePerSecond": 5,
    "name": "chat-room-a",
    "tags": {},
    "updateTime": "2022-03-16T07:22:09+00:00"
}
```

Per ulteriori informazioni, consulta Guida introduttiva alla Chat Amazon IVS nella Guida per l'utente del Servizio video interattivo Amazon (Amazon IVS).

- Per i dettagli sull'API, consulta UpdateRoom AWS CLICommand Reference.

Esempi di streaming in tempo reale di Amazon IVS utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando Amazon IVS Real-Time Streaming. AWS Command Line Interface

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`create-encoder-configuration`

Il seguente esempio di codice mostra come usare`create-encoder-configuration`.

AWS CLI

Come creare una configurazione del codificatore di composizione

L'esempio `create-encoder-configuration` seguente crea una configurazione del codificatore di composizione con le proprietà specificate.

```
aws ivs-realtime create-encoder-configuration \
    --name test-ec --video bitrate=3500000,framerate=30.0,height=1080,width=1920
```

Output:

```
{
    "encoderConfiguration": {
        "arn": "arn:aws:ivs:ap-northeast-1:123456789012:encoder-configuration/
ABabCDcdEFef",
        "name": "test-ec",
        "tags": {},
        "video": {
            "bitrate": 3500000,
```

```
            "framerate": 30,
            "height": 1080,
            "width": 1920
        }
    }
}
```

Per ulteriori informazioni, consulta Abilitazione di host multipli su un flusso Amazon IVS nella Guida per l'utente del Servizio video interattivo Amazon.

- Per i dettagli sull'API, consulta CreateEncoderConfiguration AWS CLICommand Reference.

`create-ingest-configuration`

Il seguente esempio di codice mostra come utilizzare`create-ingest-configuration`.

AWS CLI

Come creare una configurazione di importazione

L'esempio `create-ingest-configuration` seguente crea una configurazione di importazione utilizzando il protocollo RTMPS.

```
aws ivs-realtime create-ingest-configuration \
    --name ingest1 \
    --ingest-protocol rtmps
```

Output:

```
{
    "ingestConfiguration": {
        "name": "ingest1",
        "arn": "arn:aws:ivs:us-west-2:123456789012:ingest-configuration/
AbCdEfGh1234",
        "ingestProtocol": "RTMPS",
        "streamKey": "rt_123456789012_us-
west-2_AbCdEfGh1234_abcd1234efgh5678ijkl9012MNOP34",
        "stageArn": "",
        "participantId": "xyZ654abC321",
        "state": "INACTIVE",
        "userId": "",
        "tags": {}
```

```
    }
}
```

Per ulteriori informazioni, consulta Acquisizione dei flussi in IVS | Streaming in tempo reale nella Guida per l'utente di Servizio video interattivo Amazon.

- Per i dettagli sull'API, consulta CreateIngestConfiguration AWS CLICommand Reference.

`create-participant-token`

Il seguente esempio di codice mostra come utilizzare`create-participant-token`.

AWS CLI

Come creare un token di partecipazione di fase

L'esempio `create-participant-token` seguente crea un token di partecipazione per la fase specificata.

```
aws ivs-realtime create-participant-token \
    --stage-arn arn:aws:ivs:us-west-2:123456789012:stage/abcdABCDefgh \
    --user-id bob
```

Output:

```
{
    "participantToken": {
        "expirationTime": "2023-03-07T09:47:43+00:00",
        "participantId": "ABCDEfghij01234KLMN6789",
        "token": "abcd1234defg5678"
    }
}
```

Per ulteriori informazioni, consulta Abilitazione di host multipli su un flusso Amazon IVS nella Guida per l'utente del Servizio video interattivo Amazon.

- Per i dettagli sull'API, consulta CreateParticipantToken AWS CLICommand Reference.

`create-stage`

Il seguente esempio di codice mostra come utilizzare`create-stage`.

AWS CLI

Esempio 1: come creare una fase

L'esempio `create-stage` seguente crea una fase e un token di partecipazione di fase per un utente specificato.

```
aws ivs-realtime create-stage \
    --name stage1 \
    --participant-token-configurations userId=alice
```

Output:

```
{
    "participantTokens": [
        {
            "participantId": "ABCDEfghij01234KLMN5678",
            "token": "a1b2c3d4567890ab",
            "userId": "alice"
        }
    ],
    "stage": {
        "activeSessionId": "st-a1b2c3d4e5f6g",
        "arn": "arn:aws:ivs:us-west-2:123456789012:stage/abcdABCDefgh",
        "autoParticipantRecordingConfiguration": {
            "storageConfigurationArn": "",
            "mediaTypes": [
                "AUDIO_VIDEO"
            ],
            "thumbnailConfiguration": {
                "targetIntervalSeconds": 60,
                "storage": [
                    "SEQUENTIAL"
                ],
                "recordingMode": "DISABLED"
            },
            "recordingReconnectWindowSeconds": 0,
            "hlsConfiguration": {
                "targetSegmentDurationSeconds": 6
            },
            "recordParticipantReplicas": true
        },
        "endpoints": {
```

```
            "events": "wss://global.events.live-video.net",
            "rtmp": "rtmp://9x0y8z7s6t5u.global-contribute-staging.live-video.net/
app/",
            "rtmps": "rtmps://9x0y8z7s6t5u.global-contribute-staging.live-
video.net:443/app/",
            "whip": "https://9x0y8z7s6t5u.global-bm.whip.live-video.net"
        },
        "name": "stage1",
        "tags": {}
    }
}
```

Per ulteriori informazioni, consulta Abilitazione di host multipli su un flusso Amazon IVS nella Guida per l'utente dello Streaming a bassa latenza di Amazon IVS.

Esempio 2: come creare una fase e configurare la registrazione dei singoli partecipanti

L'esempio `create-stage` seguente crea una fase e configura la registrazione dei singoli partecipanti.

```
aws ivs-realtime create-stage \
    --name stage1 \
    --auto-participant-recording-configuration '{"mediaTypes":
 ["AUDIO_VIDEO"],"storageConfigurationArn": "arn:aws:ivs:us-
west-2:123456789012:storage-configuration/abcdABCDefgh",
 "recordingReconnectWindowSeconds": 100, \
        "hlsConfiguration": {"targetSegmentDurationSeconds": 5}}'
```

Output:

```
{
    "stage": {
        "activeSessionId": "st-a1b2c3d4e5f6g",
        "arn": "arn:aws:ivs:us-west-2:123456789012:stage/abcdABCDefgh",
        "autoParticipantRecordingConfiguration": {
            "storageConfigurationArn": "arn:aws:ivs:us-west-2:123456789012:storage-
configuration/abcdABCDefgh"
            "mediaTypes": [
                "AUDIO_VIDEO"
            ],
            "thumbnailConfiguration": {
                "targetIntervalSeconds": 60,
                "storage": [
```

```
                    "SEQUENTIAL"
                ],
                "recordingMode": "DISABLED"
            },
            "recordingReconnectWindowSeconds": 100,
            "hlsConfiguration": {
                "targetSegmentDurationSeconds": 5
            },
            "recordParticipantReplicas": true
        },
        "endpoints": {
            "events": "wss://global.events.live-video.net",
            "rtmp": "rtmp://9x0y8z7s6t5u.global-contribute-staging.live-video.net/
app/",
            "rtmps": "rtmps://9x0y8z7s6t5u.global-contribute-staging.live-
video.net:443/app/",
            "whip": "https://9x0y8z7s6t5u.global-bm.whip.live-video.net"
        },
        "name": "stage1",
        "tags": {}
    }
}
```

Per ulteriori informazioni, consulta Abilitazione di host multipli su un flusso Amazon IVS nella Guida per l'utente dello Streaming a bassa latenza di Amazon IVS.

Esempio 3: come creare una fase e configurare la registrazione dei singoli partecipanti con la registrazione delle anteprime abilitata

L'esempio `create-stage` seguente crea una fase e configura la registrazione dei singoli partecipanti con la registrazione delle anteprime abilitata.

```
aws ivs-realtime create-stage \
    --name stage1 \
    --auto-participant-recording-configuration '{"mediaTypes":
 ["AUDIO_VIDEO"],"storageConfigurationArn": "arn:aws:ivs:us-
west-2:123456789012:storage-configuration/abcdABCDefgh", \
        "thumbnailConfiguration": {"recordingMode": "INTERVAL","storage":
 ["SEQUENTIAL"],"targetIntervalSeconds": 60}}'
```

Output:

```
{
```

```
    "stage": {
        "activeSessionId": "st-a1b2c3d4e5f6g",
        "arn": "arn:aws:ivs:us-west-2:123456789012:stage/abcdABCDefgh",
        "autoParticipantRecordingConfiguration": {
            "storageConfigurationArn": "arn:aws:ivs:us-west-2:123456789012:storage-
configuration/abcdABCDefgh",
            "mediaTypes": [
                "AUDIO_VIDEO"
            ],
            "thumbnailConfiguration": {
                "targetIntervalSeconds": 60,
                "storage": [
                    "SEQUENTIAL"
                ],
                "recordingMode": "INTERVAL"
            },
            "recordingReconnectWindowSeconds": 0,
            "hlsConfiguration": {
                "targetSegmentDurationSeconds": 6
            },
            "recordParticipantReplicas": true
        },
        "endpoints": {
            "events": "wss://global.events.live-video.net",
            "rtmp": "rtmp://9x0y8z7s6t5u.global-contribute-staging.live-video.net/
app/",
            "rtmps": "rtmps://9x0y8z7s6t5u.global-contribute-staging.live-
video.net:443/app/",
            "whip": "https://9x0y8z7s6t5u.global-bm.whip.live-video.net"
        },
        "name": "stage1",
        "tags": {}
    }
}
```

Per ulteriori informazioni, consulta Abilitazione di host multipli su un flusso Amazon IVS nella Guida per l'utente dello Streaming a bassa latenza di Amazon IVS.

- Per i dettagli sull'API, consulta CreateStage AWS CLICommand Reference.

`create-storage-configuration`

Il seguente esempio di codice mostra come utilizzare`create-storage-configuration`.

AWS CLI

Come creare una configurazione dell'archiviazione della composizione

L'esempio `create-storage-configuration` seguente crea una configurazione dell'archiviazione della composizione con le proprietà specificate.

```
aws ivs-realtime create-storage-configuration \
    --name "test-sc" --s3 "bucketName=amzn-s3-demo-bucket"
```

Output:

```
{
    "storageConfiguration": {
        "arn": "arn:aws:ivs:ap-northeast-1:123456789012:storage-configuration/
ABabCDcdEFef",
        "name": "test-sc",
        "s3": {
            "bucketName": "amzn-s3-demo-bucket"
        },
        "tags": {}
    }
}
```

Per ulteriori informazioni, consulta Abilitazione di host multipli su un flusso Amazon IVS nella Guida per l'utente del Servizio video interattivo Amazon.

- Per i dettagli sull'API, consulta CreateStorageConfiguration AWS CLICommand Reference.

`delete-encoder-configuration`

Il seguente esempio di codice mostra come utilizzare`delete-encoder-configuration`.

AWS CLI

Come eliminare una configurazione del codificatore di composizione

L'esempio `delete-encoder-configuration` seguente elimina la configurazione del codificatore di composizione specificata dal nome della risorsa Amazon (ARN) indicato.

```
aws ivs-realtime delete-encoder-configuration \
```

```
    --arn "arn:aws:ivs:ap-northeast-1:123456789012:encoder-configuration/
ABabCDcdEFef"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Abilitazione di host multipli su un flusso Amazon IVS nella Guida per l'utente del Servizio video interattivo Amazon.

- Per i dettagli sull'API, consulta DeleteEncoderConfiguration AWS CLICommand Reference.

`delete-ingest-configuration`

Il seguente esempio di codice mostra come utilizzare`delete-ingest-configuration`.

AWS CLI

Esempio 1: come eliminare una configurazione di importazione inattiva

L'esempio `delete-ingest-configuration` seguente elimina la configurazione di importazione inattiva per un nome della risorsa Amazon (ARN) di configurazione di importazione specificato.

```
aws ivs-realtime delete-ingest-configuration \
    --arn arn:aws:ivs:us-west-2:123456789012:ingest-configuration/AbCdEfGh1234
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Acquisizione dei flussi in IVS | Streaming in tempo reale nella Guida per l'utente di Servizio video interattivo Amazon.

Esempio 2: come forzare l'eliminazione di una configurazione di importazione attiva

L'esempio `delete-ingest-configuration` seguente forza l'eliminazione della configurazione di importazione attiva per un nome della risorsa Amazon (ARN) di configurazione di importazione specificato.

```
aws ivs-realtime delete-ingest-configuration \
    --arn arn:aws:ivs:us-west-2:123456789012:ingest-configuration/AbCdEfGh1234 \
    --force
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Acquisizione dei flussi in IVS | Streaming in tempo reale nella Guida per l'utente di Servizio video interattivo Amazon.

- Per i dettagli sull'API, consulta DeleteIngestConfiguration AWS CLICommand Reference.

`delete-public-key`

Il seguente esempio di codice mostra come utilizzare`delete-public-key`.

AWS CLI

Come eliminare una chiave pubblica

L'esempio `delete-public-key` seguente elimina la chiave pubblica specificata.

```
aws ivs-realtime delete-public-key \
    --arn arn:aws:ivs:us-west-2:123456789012:public-key/abcdABC1efg2
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Distribuzione dei token di partecipazione nella Guida per l'utente dello Streaming in tempo reale di Amazon IVS.

- Per i dettagli sull'API, consulta DeletePublicKey AWS CLICommand Reference.

`delete-stage`

Il seguente esempio di codice mostra come utilizzare`delete-stage`.

AWS CLI

Come eliminare una fase

L'esempio `delete-stage` seguente elimina la fase specificata.

```
aws ivs-realtime delete-stage \
    --arn arn:aws:ivs:us-west-2:123456789012:stage/abcdABCDefgh
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Abilitazione di host multipli su un flusso Amazon IVS nella Guida per l'utente del Servizio video interattivo Amazon.

- Per i dettagli sull'API, consulta DeleteStage AWS CLICommand Reference.

`delete-storage-configuration`

Il seguente esempio di codice mostra come utilizzare`delete-storage-configuration`.

AWS CLI

Come eliminare una configurazione dell'archiviazione della composizione

L'esempio `delete-storage-configuration` seguente elimina la configurazione dell'archiviazione della composizione specificata dal nome della risorsa Amazon (ARN) indicato.

```
aws ivs-realtime delete-storage-configuration \
    --arn "arn:aws:ivs:ap-northeast-1:123456789012:storage-configuration/
ABabCDcdEFef"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Abilitazione di host multipli su un flusso Amazon IVS nella Guida per l'utente del Servizio video interattivo Amazon.

- Per i dettagli sull'API, consulta DeleteStorageConfiguration AWS CLICommand Reference.

`disconnect-participant`

Il seguente esempio di codice mostra come utilizzare`disconnect-participant`.

AWS CLI

Come disconnettere un partecipante dalla fase

L'esempio `disconnect-participant` seguente disconnette il partecipante specificato dalla fase specificata.

```
aws ivs-realtime disconnect-participant \
    --stage-arn arn:aws:ivs:us-west-2:123456789012:stage/abcdABCDefgh \
    --participant-id ABCDEfghij01234KLMN5678
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Abilitazione di host multipli su un flusso Amazon IVS nella Guida per l'utente del Servizio video interattivo Amazon.

- Per i dettagli sull'API, consulta DisconnectParticipant AWS CLICommand Reference.

`get-composition`

Il seguente esempio di codice mostra come utilizzare`get-composition`.

AWS CLI

Esempio 1: come ottenere una composizione con le impostazioni di layout predefinite

L'esempio `get-composition` seguente ottiene la composizione per il nome della risorsa Amazon (ARN) specificato.

```
aws ivs-realtime get-composition \
    --arn "arn:aws:ivs:ap-northeast-1:123456789012:composition/abcdABCDefgh"
```

Output:

```
{
    "composition": {
        "arn": "arn:aws:ivs:ap-northeast-1:123456789012:composition/abcdABCDefgh",
        "destinations": [
            {
                "configuration": {
                    "channel": {
                        "channelArn": "arn:aws:ivs:ap-
northeast-1:123456789012:channel/abcABCdefDEg",
                        "encoderConfigurationArn": "arn:aws:ivs:ap-
northeast-1:123456789012:encoder-configuration/ABabCDcdEFef"
                    },
                    "name": ""
                },
                "id": "AabBCcdDEefF",
                "startTime": "2023-10-16T23:26:00+00:00",
                "state": "ACTIVE"
            },
            {
                "configuration": {
                    "name": "",
                    "s3": {
```

```
                        "encoderConfigurationArns": [
                            "arn:aws:ivs:arn:aws:ivs:ap-
northeast-1:123456789012:encoder-configuration/ABabCDcdEFef"
                        ],
                        "recordingConfiguration": {
                            "format": "HLS",
                            "hlsConfiguration": {
                                "targetSegmentDurationSeconds": 2
                            }
                        },
                        "storageConfigurationArn": "arn:arn:aws:ivs:ap-
northeast-1:123456789012:storage-configuration/FefABabCDcdE",
                    }
                },
                "detail": {
                    "s3": {
                        "recordingPrefix": "aBcDeFgHhGfE/AbCdEfGhHgFe/GHFabcgefABC/
composite"
                    }
                },
                "id": "GHFabcgefABC",
                "startTime": "2023-10-16T23:26:00+00:00",
                "state": "STARTING"
            }
        ],
        "layout": {
            "grid": {
                "featuredParticipantAttribute": "",
                "gridGap": 2,
                "omitStoppedVideo": false,
                "participantOrderAttribute": "",
                "videoAspectRatio": "VIDEO",
                "videoFillMode": ""
            }
        },
        "stageArn": "arn:aws:ivs:ap-northeast-1:123456789012:stage/defgABCDabcd",
        "startTime": "2023-10-16T23:24:00+00:00",
        "state": "ACTIVE",
        "tags": {}
    }
}
```

Per ulteriori informazioni, consulta IVS Composite Recording | Real-Time Streaming nella Amazon Interactive Video Service User Guide.

Esempio 2: come ottenere una composizione con layout PiP

L'esempio `get-composition` seguente ottiene la composizione per il nome della risorsa Amazon (ARN) specificato che utilizza il layout PiP.

```
aws ivs-realtime get-composition \
    --arn "arn:aws:ivs:ap-northeast-1:123456789012:composition/wxyzWXYZpqrs"
```

Output:

```
{
    "composition": {
        "arn": "arn:aws:ivs:ap-northeast-1:123456789012:composition/wxyzWXYZpqrs",
        "destinations": [
            {
                "configuration": {
                    "channel": {
                        "channelArn": "arn:aws:ivs:ap-
northeast-1:123456789012:channel/abcABCdefDEg",
                        "encoderConfigurationArn": "arn:aws:ivs:ap-
northeast-1:123456789012:encoder-configuration/ABabCDcdEFef"
                    },
                    "name": ""
                },
                "id": "AabBCcdDEefF",
                "startTime": "2023-10-16T23:26:00+00:00",
                "state": "ACTIVE"
            },
            {
                "configuration": {
                    "name": "",
                    "s3": {
                        "encoderConfigurationArns": [
                            "arn:aws:ivs:arn:aws:ivs:ap-
northeast-1:123456789012:encoder-configuration/ABabCDcdEFef"
                        ],
                        "recordingConfiguration": {
                            "format": "HLS",
                            "hlsConfiguration": {
                                "targetSegmentDurationSeconds": 2
```

```
                            }
                        },
                        "storageConfigurationArn": "arn:arn:aws:ivs:ap-
northeast-1:123456789012:storage-configuration/FefABabCDcdE"
                    }
                },
                "detail": {
                    "s3": {
                        "recordingPrefix": "aBcDeFgHhGfE/AbCdEfGhHgFe/GHFabcgefABC/
composite"
                    }
                },
                "id": "GHFabcgefABC",
                "startTime": "2023-10-16T23:26:00+00:00",
                "state": "STARTING"
            }
        ],
        "layout": {
            "pip": {
                "featuredParticipantAttribute": "abcdefg",
                "gridGap": 0,
                "omitStoppedVideo": false,
                "participantOrderAttribute": "",
                "pipBehavior": "STATIC",
                "pipOffset": 0,
                "pipParticipantAttribute": "",
                "pipPosition": "BOTTOM_RIGHT",
                "videoFillMode": "COVER"
            }
        },
        "stageArn": "arn:aws:ivs:ap-northeast-1:123456789012:stage/defgABCDabcd",
        "startTime": "2023-10-16T23:24:00+00:00",
        "state": "ACTIVE",
        "tags": {}
    }
}
```

Per ulteriori informazioni, consulta IVS Composite Recording | Real-Time Streaming nella Amazon Interactive Video Service User Guide.

Esempio 3: come ottenere una composizione con la registrazione delle anteprime abilitata

L'esempio `get-composition` seguente ottiene la composizione per il nome della risorsa Amazon (ARN) specificato, che ha la registrazione delle anteprime abilitata con le impostazioni predefinite.

```
aws ivs-realtime get-composition \
    --arn "arn:aws:ivs:ap-northeast-1:123456789012:composition/abcdABCDefgh"
```

Output:

```
{
    "composition": {
        "arn": "arn:aws:ivs:ap-northeast-1:123456789012:composition/abcdABCDefgh",
        "destinations": [
            {
                "configuration": {
                    "channel": {
                        "channelArn": "arn:aws:ivs:ap-
northeast-1:123456789012:channel/abcABCdefDEg",
                        "encoderConfigurationArn": "arn:aws:ivs:ap-
northeast-1:123456789012:encoder-configuration/ABabCDcdEFef"
                    },
                    "name": ""
                },
                "id": "AabBCcdDEefF",
                "startTime": "2023-10-16T23:26:00+00:00",
                "state": "ACTIVE"
            },
            {
                "configuration": {
                    "name": "",
                    "s3": {
                        "encoderConfigurationArns": [
                            "arn:aws:ivs:arn:aws:ivs:ap-
northeast-1:123456789012:encoder-configuration/ABabCDcdEFef"
                        ],
                        "recordingConfiguration": {
                            "format": "HLS",
                            "hlsConfiguration": {
                                "targetSegmentDurationSeconds": 2
                            }
                        },
                        "storageConfigurationArn": "arn:arn:aws:ivs:ap-
northeast-1:123456789012:storage-configuration/FefABabCDcdE",
```

```
                        "thumbnailConfigurations": [
                           {
                              "targetIntervalSeconds": 60,
                              "storage": [
                                  "SEQUENTIAL"
                              ],
                           }
                        ]
                    }
                },
                "detail": {
                    "s3": {
                        "recordingPrefix": "aBcDeFgHhGfE/AbCdEfGhHgFe/GHFabcgefABC/
composite"
                    }
                },
                "id": "GHFabcgefABC",
                "startTime": "2023-10-16T23:26:00+00:00",
                "state": "STARTING"
            }
        ],
        "layout": {
            "grid": {
                "featuredParticipantAttribute": ""
                "gridGap": 2,
                "omitStoppedVideo": false,
                "participantOrderAttribute": "",
                "videoAspectRatio": "VIDEO",
                "videoFillMode": ""                }
        },
        "stageArn": "arn:aws:ivs:ap-northeast-1:123456789012:stage/defgABCDabcd",
        "startTime": "2023-10-16T23:24:00+00:00",
        "state": "ACTIVE",
        "tags": {}
    }
}
```

Per ulteriori informazioni, consulta IVS Composite Recording | Real-Time Streaming nella Amazon Interactive Video Service User Guide.

- Per i dettagli sull'API, consulta GetComposition AWS CLICommand Reference.

`get-encoder-configuration`

Il seguente esempio di codice mostra come utilizzare`get-encoder-configuration`.

AWS CLI

Come ottenere una configurazione del codificatore di composizione

L'esempio `get-encoder-configuration` seguente ottiene la configurazione del codificatore di composizione specificata dal nome della risorsa Amazon (ARN) indicato.

```
aws ivs-realtime get-encoder-configuration \
    --arn "arn:aws:ivs:ap-northeast-1:123456789012:encoder-configuration/
abcdABCDefgh"
```

Output:

```
{
    "encoderConfiguration": {
        "arn": "arn:aws:ivs:ap-northeast-1:123456789012:encoder-configuration/
abcdABCDefgh",
        "name": "test-ec",
        "tags": {},
        "video": {
            "bitrate": 3500000,
            "framerate": 30,
            "height": 1080,
            "width": 1920
        }
    }
}
```

Per ulteriori informazioni, consulta Abilitazione di host multipli su un flusso Amazon IVS nella Guida per l'utente del Servizio video interattivo Amazon.

- Per i dettagli sull'API, consulta GetEncoderConfiguration AWS CLICommand Reference.

`get-ingest-configuration`

Il seguente esempio di codice mostra come utilizzare`get-ingest-configuration`.

AWS CLI

Come ottenere informazioni sulla configurazione di importazione

L'esempio `get-ingest-configuration` seguente ottiene la configurazione di importazione per un nome della risorsa Amazon (ARN) di configurazione di importazione specificato.

```
aws ivs-realtime get-ingest-configuration \
    --arn arn:aws:ivs:us-west-2:123456789012:ingest-configuration/AbCdEfGh1234
```

Output:

```
{
    "ingestConfiguration": {
        "name": "ingest1",
        "arn": "arn:aws:ivs:us-west-2:123456789012:ingest-configuration/
AbCdEfGh1234",
        "ingestProtocol": "RTMPS",
        "streamKey": "rt_123456789012_us-
west-2_AbCdEfGh1234_abcd1234efgh5678ijkl9012MNOP34",
        "stageArn": "",
        "participantId": "xyZ654abC321",
        "state": "INACTIVE",
        "userId": "",
        "tags": {}
    }
}
```

Per ulteriori informazioni, consulta Acquisizione dei flussi in IVS | Streaming in tempo reale nella Guida per l'utente di Servizio video interattivo Amazon.

- Per i dettagli sull'API, consulta GetIngestConfiguration AWS CLICommand Reference.

`get-participant`

Il seguente esempio di codice mostra come utilizzare`get-participant`.

AWS CLI

Esempio 1: come ottenere un partecipante alla fase

L'esempio `get-participant` seguente ottiene il partecipante alla fase per un ID partecipante e un ID sessione specificati nel nome della risorsa Amazon (ARN) della fase specificato.

```
aws ivs-realtime get-participant \
    --stage-arn arn:aws:ivs:us-west-2:123456789012:stage/abcdABCDefgh \
    --session-id st-a1b2c3d4e5f6g \
    --participant-id abCDEf12GHIj
```

Output:

```
{
    "participant": {
        "browserName": "Google Chrome",
        "browserVersion": "116",
        "firstJoinTime": "2023-04-26T20:30:34+00:00",
        "ispName": "Comcast",
        "osName": "Microsoft Windows 10 Pro",
        "osVersion": "10.0.19044",
        "participantId": "abCDEf12GHIj",
        "published": true,
        "recordingS3BucketName": "bucket-name",
        "recordingS3Prefix": "abcdABCDefgh/st-a1b2c3d4e5f6g/
abCDEf12GHIj/1234567890",
        "recordingState": "ACTIVE",
        "sdkVersion": "",
        "state": "CONNECTED",
        "userId": ""
    }
}
```

Per ulteriori informazioni, consulta Abilitazione di host multipli su un flusso Amazon IVS nella Guida per l'utente dello Streaming a bassa latenza di Amazon IVS.

Esempio 2: come ottenere un partecipante alla fase che è stato replicato in un'altra fase

L'esempio `get-participant` seguente ottiene il partecipante alla fase per un ID partecipante e un ID sessione specificati nel nome della risorsa Amazon (ARN) della fase specificato, quando il partecipante è stato replicato anche in un'altra fase.

```
aws ivs-realtime get-participant \
    --stage-arn arn:aws:ivs:us-west-2:123456789012:stage/abcdABCDefgh \
    --session-id st-a1b2c3d4e5f6g \
    --participant-id abCDEf12GHIj
```

Output:

```
{
    "participant": {
        "browserName": "Google Chrome",
        "browserVersion": "116",
        "firstJoinTime": "2023-04-26T20:30:34+00:00",
        "ispName": "Comcast",
        "osName": "Microsoft Windows 10 Pro",
        "osVersion": "10.0.19044",
        "participantId": "abCDEf12GHIj",
        "published": true,
        "recordingS3BucketName": "bucket-name",
        "recordingS3Prefix": "abcdABCDefgh/st-a1b2c3d4e5f6g/
abCDEf12GHIj/1234567890",
        "recordingState": "ACTIVE",
        "replicationState": "ACTIVE",
        "replicationType": "SOURCE",
        "sdkVersion": "",
        "state": "CONNECTED",
        "userId": ""
    }
}
```

Per ulteriori informazioni, consulta Abilitazione di host multipli su un flusso Amazon IVS nella Guida per l'utente dello Streaming a bassa latenza di Amazon IVS.

Esempio 3: come ottenere un partecipante alla fase che è stato replicato da un'altra fase

L'esempio `get-participant` seguente ottiene il partecipante alla fase per un ID partecipante e un ID sessione specificati nel nome della risorsa Amazon (ARN) della fase specificato, quando il partecipante è stato replicato da un'altra fase.

```
aws ivs-realtime get-participant \
    --stage-arn arn:aws:ivs:us-west-2:123456789012:stage/abcdABCDefgh \
    --session-id st-a1b2c3d4e5f6g \
    --participant-id abCDEf12GHIj
```

Output:

```
{
    "participant": {
        "browserName": "Google Chrome",
        "browserVersion": "116",
```

```
        "firstJoinTime": "2023-04-26T20:30:34+00:00",
        "ispName": "Comcast",
        "osName": "Microsoft Windows 10 Pro",
        "osVersion": "10.0.19044",
        "participantId": "abCDEf12GHIj",
        "published": true,
        "recordingS3BucketName": "bucket-name",
        "recordingS3Prefix": "abcdABCDefgh/st-a1b2c3d4e5f6g/
abCDEf12GHIj/1234567890",
        "recordingState": "ACTIVE",
        "replicationState": "ACTIVE",
        "replicationType": "REPLICA",
        "sdkVersion": "",
        "state": "CONNECTED",
        "userId": ""
    }
}
```

Per ulteriori informazioni, consulta Abilitazione di host multipli su un flusso Amazon IVS nella Guida per l'utente dello Streaming a bassa latenza di Amazon IVS.

- Per i dettagli sull'API, consulta GetParticipant AWS CLICommand Reference.

`get-public-key`

Il seguente esempio di codice mostra come utilizzare`get-public-key`.

AWS CLI

Come ottenere una chiave pubblica esistente utilizzata per firmare i token di partecipazione di fase

L'esempio `get-public-key` seguente ottiene una chiave pubblica specificata dall'ARN fornito, per firmare i token di partecipazione di fase.

```
aws ivs-realtime get-public-key \
    --arn arn:aws:ivs:us-west-2:123456789012:public-key/abcdABC1efg2
```

Output:

```
{
    "publicKey": {
        "arn": "arn:aws:ivs:us-west-2:123456789012:public-key/abcdABC1efg2",
```

```
        "name": "",
        "publicKeyMaterial": "-----BEGIN PUBLIC KEY-----
\nMHYwEAYHKoZIzj0CAQYFK4EEACIDYgAEqVWUtqs6EktQMR1sCYmEzGvRwtaycI16\n9pmzcpiWu/
uhNStGlteJ5odRfRwVkoQUMnSZXTCcbn9bBTTmiWo4mJcFOOAzsthH
\n0UAb8NdD4tUE0At4a9hYP9IETEXAMPLE\n-----END PUBLIC KEY-----",
        "fingerprint": "12:a3:44:56:bc:7d:e8:9f:10:2g:34:hi:56:78:90:12",
        "tags": {}
    }
}
```

Per ulteriori informazioni, consulta Distribuzione dei token di partecipazione nella Guida per l'utente dello Streaming in tempo reale di Amazon IVS.

- Per i dettagli sull'API, consulta GetPublicKey AWS CLICommand Reference.

`get-stage-session`

Il seguente esempio di codice mostra come utilizzare`get-stage-session`.

AWS CLI

Come ottenere una sessione di fase

L'esempio `get-stage-session` seguente ottiene la sessione di fase per un ID di sessione specificato di un nome della risorsa Amazon (ARN) della fase specificato.

```
aws ivs-realtime get-stage-session \
    --stage-arn arn:aws:ivs:us-west-2:123456789012:stage/abcdABCDefgh \
    --session-id st-a1b2c3d4e5f6g
```

Output:

```
{
    "stageSession": {
        "endTime": "2023-04-26T20:36:29+00:00",
        "sessionId": "st-a1b2c3d4e5f6g",
        "startTime": "2023-04-26T20:30:29.602000+00:00"
    }
}
```

Per ulteriori informazioni, consulta Abilitazione di host multipli su un flusso Amazon IVS nella Guida per l'utente del Servizio video interattivo Amazon.

- Per i dettagli sull'API, consulta GetStageSession AWS CLICommand Reference.

`get-stage`

Il seguente esempio di codice mostra come utilizzare`get-stage`.

AWS CLI

Come ottenere informazioni sulla configurazione di una fase

L'esempio `get-stage` seguente ottiene la configurazione della fase per un nome della risorsa Amazon (ARN) specificato.

```
aws ivs-realtime get-stage \
    --arn arn:aws:ivs:us-west-2:123456789012:stage/abcdABCDefgh
```

Output:

```
{
    "stage": {
        "activeSessionId": "st-a1b2c3d4e5f6g",
        "arn": "arn:aws:ivs:us-west-2:123456789012:stage/abcdABCDefgh",
        "autoParticipantRecordingConfiguration": {
            "storageConfigurationArn": "",
            "mediaTypes": [
                "AUDIO_VIDEO"
            ],
            "thumbnailConfiguration": {
                "targetIntervalSeconds": 60,
                "storage": [
                    "SEQUENTIAL"
                ],
                "recordingMode": "DISABLED"
            },
            "recordingReconnectWindowSeconds": 0,
            "hlsConfiguration": {
                "targetSegmentDurationSeconds": 6
            },
            "recordParticipantReplicas": true
        },
        "endpoints": {
            "events": "wss://global.events.live-video.net",
```

```
            "rtmp": "rtmp://9x0y8z7s6t5u.global-contribute-staging.live-video.net/
app/",
            "rtmps": "rtmps://9x0y8z7s6t5u.global-contribute-staging.live-
video.net:443/app/",
            "whip": "https://9x0y8z7s6t5u.global-bm.whip.live-video.net"
        },
        "name": "test",
        "tags": {}
    }
}
```

Per ulteriori informazioni, consulta Abilitazione di host multipli su un flusso Amazon IVS nella Guida per l'utente dello Streaming a bassa latenza di Amazon IVS.

- Per i dettagli sull'API, consulta GetStage AWS CLICommand Reference.

`get-storage-configuration`

Il seguente esempio di codice mostra come utilizzare`get-storage-configuration`.

AWS CLI

Come ottenere la configurazione di un'archiviazione della composizione

L'esempio `get-storage-configuration` seguente ottiene la configurazione dell'archiviazione della composizione specificata dal nome della risorsa Amazon (ARN) indicato.

```
aws ivs-realtime get-storage-configuration \
    --name arn "arn:aws:ivs:ap-northeast-1:123456789012:storage-configuration/
abcdABCDefgh"
```

Output:

```
{
    "storageConfiguration": {
        "arn": "arn:aws:ivs:ap-northeast-1:123456789012:storage-configuration/
abcdABCDefgh",
        "name": "test-sc",
        "s3": {
            "bucketName": "amzn-s3-demo-bucket"
        },
        "tags": {}
```

```
    }
}
```

Per ulteriori informazioni, consulta Abilitazione di host multipli su un flusso Amazon IVS nella Guida per l'utente del Servizio video interattivo Amazon.

- Per i dettagli sull'API, consulta GetStorageConfiguration AWS CLICommand Reference.

`import-public-key`

Il seguente esempio di codice mostra come utilizzare`import-public-key`.

AWS CLI

Come importare una chiave pubblica esistente da utilizzare per firmare i token di partecipazione di fase

L'esempio `import-public-key` seguente importa una chiave pubblica da un file di materiale, da utilizzare per firmare i token di partecipazione di fase.

```
aws ivs-realtime import-public-key \
    --public-key-material="`cat public.pem`"
```

Output:

```
{
    "publicKey": {
        "arn": "arn:aws:ivs:us-west-2:123456789012:public-key/abcdABC1efg2",
        "name": "",
        "publicKeyMaterial": "-----BEGIN PUBLIC KEY-----
\nMHYwEAYHKoZIzj0CAQYFK4EEACIDYgAEqVWUtqs6EktQMR1sCYmEzGvRwtaycI16\n9pmzcpiWu/
uhNStGlteJ5odRfRwVkoQUMnSZXTCcbn9bBTTmiWo4mJcFOOAzsthH
\n0UAb8NdD4tUE0At4a9hYP9IETEXAMPLE\n-----END PUBLIC KEY-----",
        "fingerprint": "12:a3:44:56:bc:7d:e8:9f:10:2g:34:hi:56:78:90:12",
        "tags": {}
    }
}
```

Per ulteriori informazioni, consulta Distribuzione dei token di partecipazione nella Guida per l'utente dello Streaming in tempo reale di Amazon IVS.

- Per i dettagli sull'API, consulta ImportPublicKey AWS CLICommand Reference.

`list-compositions`

Il seguente esempio di codice mostra come utilizzare`list-compositions`.

AWS CLI

Come ottenere un elenco di composizioni

Di seguito `list-compositions` sono elencate tutte le composizioni per il tuo AWS account, nella AWS regione in cui viene elaborata la richiesta API.

```
aws ivs-realtime list-compositions
```

Output:

```
{
    "compositions": [
        {
            "arn": "arn:aws:ivs:ap-northeast-1:123456789012:composition/
abcdABCDefgh",
            "destinations": [
                {
                    "id": "AabBCcdDEefF",
                    "startTime": "2023-10-16T23:25:23+00:00",
                    "state": "ACTIVE"
                }
            ],
            "stageArn": "arn:aws:ivs:ap-northeast-1:123456789012:stage/
defgABCDabcd",
            "startTime": "2023-10-16T23:25:21+00:00",
            "state": "ACTIVE",
            "tags": {}
        },
        {
            "arn": "arn:aws:ivs:ap-northeast-1:123456789012:composition/
ABcdabCDefgh",
            "destinations": [
                {
                    "endTime": "2023-10-16T23:25:00.786512+00:00",
                    "id": "aABbcCDdeEFf",
                    "startTime": "2023-10-16T23:24:01+00:00",
                    "state": "STOPPED"
                },
```

```
                {
                    "endTime": "2023-10-16T23:25:00.786512+00:00",
                    "id": "deEFfaABbcCD",
                    "startTime": "2023-10-16T23:24:01+00:00",
                    "state": "STOPPED"
                }
            ],
            "endTime": "2023-10-16T23:25:00+00:00",
            "stageArn": "arn:aws:ivs:ap-northeast-1:123456789012:stage/
efghabcdABCD",
            "startTime": "2023-10-16T23:24:00+00:00",
            "state": "STOPPED",
            "tags": {}
        }
    ]
}
```

Per ulteriori informazioni, consulta Abilitazione di host multipli su un flusso Amazon IVS nella Guida per l'utente del Servizio video interattivo Amazon.

- Per i dettagli sull'API, consulta ListCompositions AWS CLICommand Reference.

`list-encoder-configurations`

Il seguente esempio di codice mostra come utilizzare`list-encoder-configurations`.

AWS CLI

Come elencare le configurazioni del codificatore di composizione

Di seguito `list-encoder-configurations` sono elencate tutte le configurazioni del codificatore di composizione per il tuo AWS account, nella AWS regione in cui viene elaborata la richiesta API.

```
aws ivs-realtime list-encoder-configurations
```

Output:

```
{
    "encoderConfigurations": [
        {
            "arn": "arn:aws:ivs:ap-northeast-1:123456789012:encoder-configuration/
abcdABCDefgh",
```

```
            "name": "test-ec-1",
            "tags": {}
        },
        {
            "arn": "arn:aws:ivs:ap-northeast-1:123456789012:encoder-configuration/
ABCefgEFGabc",
            "name": "test-ec-2",
            "tags": {}
        }
    ]
}
```

Per ulteriori informazioni, consulta Abilitazione di host multipli su un flusso Amazon IVS nella Guida per l'utente del Servizio video interattivo Amazon.

- Per i dettagli sull'API, consulta AWS CLI Command ListEncoderConfigurationsReference.

`list-ingest-configurations`

Il seguente esempio di codice mostra come utilizzare`list-ingest-configurations`.

AWS CLI

Come ottenere informazioni di riepilogo su tutte le configurazioni di importazione

L'`list-ingest-configurations`esempio seguente elenca tutte le configurazioni di acquisizione per il tuo AWS account, nella AWS regione in cui viene elaborata la richiesta API.

```
aws ivs-realtime list-ingest-configurations
```

Output:

```
{
    "ingestConfigurations": [
        {
            "name": "",
            "arn": "arn:aws:ivs:us-west-2:123456789012:ingest-configuration/
XYZuvwSt4567",
            "ingestProtocol": "RTMPS",
            "stageArn": "arn:aws:ivs:us-west-2:123456789012:stage/abcdABCDefgh",
            "participnatId": "abC789Xyz456",
            "state": "INACTIVE"
```

```
            "userId": "",
        }
    ]
}
```

Per ulteriori informazioni, consulta Acquisizione dei flussi in IVS | Streaming in tempo reale nella Guida per l'utente di Servizio video interattivo Amazon.

- Per i dettagli sull'API, consulta AWS CLI Command ListIngestConfigurationsReference.

`list-participant-events`

Il seguente esempio di codice mostra come utilizzare`list-participant-events`.

AWS CLI

Esempio 1: come ottenere un elenco degli eventi di partecipazione di fase

L'esempio `list-participant-events` seguente elenca tutti gli eventi di partecipazione per un ID partecipante e un ID sessione specificati di un nome della risorsa Amazon (ARN) della fase specificato.

```
aws ivs-realtime list-participant-events \
    --stage-arn arn:aws:ivs:us-west-2:123456789012:stage/abcdABCDefgh \
    --session-id st-a1b2c3d4e5f6g \
    --participant-id abCDEf12GHIj
```

Output:

```
{
    "events": [
        {
            "eventTime": "2023-04-26T20:36:28+00:00",
            "name": "LEFT",
            "participantId": "abCDEf12GHIj"
        },
        {
            "eventTime": "2023-04-26T20:36:28+00:00",
            "name": "PUBLISH_STOPPED",
            "participantId": "abCDEf12GHIj"
        },
        {
```

```
            "eventTime": "2023-04-26T20:30:34+00:00",
            "name": "JOINED",
            "participantId": "abCDEf12GHIj"
        },
        {
            "eventTime": "2023-04-26T20:30:34+00:00",
            "name": "PUBLISH_STARTED",
            "participantId": "abCDEf12GHIj"
        }
    ]
}
```

Per ulteriori informazioni, consulta Abilitazione di host multipli su un flusso Amazon IVS nella Guida per l'utente dello Streaming a bassa latenza di Amazon IVS.

Esempio 2: come ottenere un elenco di eventi di partecipazione di fase, inclusi l'arresto e l'avvio della replica del partecipante

L'esempio `list-participant-events` seguente elenca tutti gli eventi di partecipazione per un ID sessione specificato di un nome della risorsa Amazon (ARN) della fase specificato, in cui un partecipante viene replicato in un'altra fase.

```
aws ivs-realtime list-participant-events \
    --stage-arn arn:aws:ivs:us-west-2:123456789012:stage/abcdABCDefgh \
    --session-id st-a1b2c3d4e5f6g \
    --participant-id abCDEf12GHIj
```

Output:

```
{
    "events": [
        {
            "eventTime": "2025-04-26T20:36:28+00:00",
            "name": "LEFT",
            "participantId": "abCDEf12GHIj"
        },
        {
            "eventTime": "2025-04-26T20:36:28+00:00",
            "name": "PUBLISH_STOPPED",
            "participantId": "abCDEf12GHIj"
        },
        {
```

```
            "eventTime": "2025-04-26T20:30:34+00:00",
            "name": "JOINED",
            "participantId": "abCDEf12GHIj"
        },
        {
            "eventTime": "2025-04-26T20:30:34+00:00",
            "name": "PUBLISH_STARTED",
            "participantId": "abCDEf12GHIj"
        },
        {
            "name": "REPLICATION_STARTED",
            "participantId": "abCDEf12GHIj",
            "eventTime": "2025-04-26T20:30:34+00:00",
            "destinationStageArn": "arn:aws:ivs:us-west-2:12345678901:stage/
ABCDabcdefgh",
            "destinationSessionId": "st-b1c2d3e4f5g6a"
        },
        {
            "name": "REPLICATION_STOPPED",
            "participantId": "abCDEf12GHIj",
            "eventTime": "2025-04-26T20:32:34+00:00",
            "destinationStageArn": "arn:aws:ivs:us-west-2:12345678901:stage/
ABCDabcdefgh",
            "destinationSessionId": "st-b1c2d3e4f5g6a"
        }
    ]
}
```

Per ulteriori informazioni, consulta Abilitazione di host multipli su un flusso Amazon IVS nella Guida per l'utente dello Streaming a bassa latenza di Amazon IVS.

- Per i dettagli sull'API, consulta ListParticipantEvents AWS CLICommand Reference.

`list-participant-replicas`

Il seguente esempio di codice mostra come utilizzare`list-participant-replicas`.

AWS CLI

Come ottenere un elenco delle repliche dei partecipanti alla fase

L'esempio `list-participant-replicas` seguente elenca tutti i partecipanti alla fase replicati dal nome della risorsa Amazon (ARN) della fase di origine specificato in un'altra fase.

```
aws ivs-realtime list-participant-replicas \
    --source-stage-arn arn:aws:ivs:us-west-2:123456789012:stage/abcdABCDefgh \
    --participant-id abCDEf12GHIj
```

Output:

```
{
    "replicas": [
        {
            "sourceStageArn": "arn:aws:ivs:us-west-2:123456789012:stage/
abcdABCDefgh",
            "participantId": "abCDEf12GHIj",
            "sourceSessionId": "st-a1b2c3d4e5f6g",
            "destinationStageArn": "arn:aws:ivs:us-west-2:012345678901:stage/
ABCDabcdefgh",
            "destinationSessionId": "st-b1c2d3e4f5g6a",
            "replicationState": "ACTIVE"
        }
    ]
}
```

Per ulteriori informazioni, consulta Replica dei partecipanti IVS I Streaming in tempo reale nella Guida per l'utente dello Streaming in tempo reale di Amazon IVS.

- Per i dettagli sull'API, consulta ListParticipantReplicas AWS CLICommand Reference.

`list-participants`

Il seguente esempio di codice mostra come utilizzare`list-participants`.

AWS CLI

Esempio 1: come ottenere un elenco dei partecipanti alla fase

L'esempio `list-participants` seguente elenca tutti i partecipanti per un ID sessione specificato di un nome della risorsa Amazon (ARN) della fase specificato.

```
aws ivs-realtime list-participants \
    --stage-arn arn:aws:ivs:us-west-2:123456789012:stage/abcdABCDefgh \
    --session-id st-a1b2c3d4e5f6g
```

Output:

```
{
    "participants": [
        {
            "firstJoinTime": "2023-04-26T20:30:34+00:00",
            "participantId": "abCDEf12GHIj",
            "published": true,
            "recordingState": "STOPPED",
            "state": "DISCONNECTED",
            "userId": ""
        }
    ]
}
```

Per ulteriori informazioni, consulta Abilitazione di host multipli su un flusso Amazon IVS nella Guida per l'utente dello Streaming a bassa latenza di Amazon IVS.

Esempio 2: come ottenere un elenco dei partecipanti alla fase, quando un partecipante è stato replicato in un'altra fase

L'esempio `list-participants` seguente elenca tutti i partecipanti per un ID sessione specificato di un nome della risorsa Amazon (ARN) della fase specificato, quando un partecipante è stato replicato in un'altra fase.

```
aws ivs-realtime list-participants \
    --stage-arn arn:aws:ivs:us-west-2:123456789012:stage/abcdABCDefgh \
    --session-id st-a1b2c3d4e5f6g
```

Output:

```
{
    "participants": [
        {
            "firstJoinTime": "2023-04-26T20:30:34+00:00",
            "participantId": "abCDEf12GHIj",
            "published": true,
            "recordingState": "STOPPED",
            "state": "DISCONNECTED",
            "userId": "",
            "replicationState": "ACTIVE",
            "replicationType": "SOURCE",
            "sourceStageArn": "",
```

```
            "sourceSessionId": ""
        }
    ]
}
```

Per ulteriori informazioni, consulta Abilitazione di host multipli su un flusso Amazon IVS nella Guida per l'utente dello Streaming a bassa latenza di Amazon IVS.

Esempio 3: come ottenere un elenco dei partecipanti alla fase, quando un partecipante è stato replicato da un'altra fase

L'esempio `list-participants` seguente elenca tutti i partecipanti per un ID sessione specificato di un nome della risorsa Amazon (ARN) della fase specificato, quando un partecipante è stato replicato da un'altra fase.

```
aws ivs-realtime list-participants \
    --stage-arn arn:aws:ivs:us-west-2:123456789012:stage/abcdABCDefgh \
    --session-id st-a1b2c3d4e5f6g
```

Output:

```
{
    "participants": [
        {
            "firstJoinTime": "2023-04-26T20:30:34+00:00",
            "participantId": "abCDEf12GHIj",
            "published": true,
            "recordingState": "STOPPED",
            "state": "DISCONNECTED",
            "userId": "",
            "replicationState": "ACTIVE",
            "replicationType": "REPLICA",
            "sourceStageArn": "arn:aws:ivs:us-west-2:123456789012:stage/
abcdABCDefgh",
            "sourceSessionId": "st-a1b2c3d4e5f6g"
        }
    ]
}
```

Per ulteriori informazioni, consulta Abilitazione di host multipli su un flusso Amazon IVS nella Guida per l'utente dello Streaming a bassa latenza di Amazon IVS.

- Per i dettagli sull'API, consulta ListParticipants AWS CLICommand Reference.

`list-public-keys`

Il seguente esempio di codice mostra come utilizzare`list-public-keys`.

AWS CLI

Come elencare le chiavi pubbliche esistenti disponibili per firmare i token di partecipazione alla fase

L'`list-public-keys`esempio seguente elenca tutte le chiavi pubbliche disponibili per la firma dei token dei partecipanti allo stage, nella AWS regione in cui viene elaborata la richiesta API.

```
aws ivs-realtime list-public-keys
```

Output:

```
{
    "publicKeys": [
        {
            "arn": "arn:aws:ivs:us-west-2:123456789012:public-key/abcdABC1efg2",
            "name": "",
            "tags": {}
        },
        {
            "arn": "arn:aws:ivs:us-west-2:123456789012:public-key/3bcdABCDefg4",
            "name": "",
            "tags": {}
        }
    ]
}
```

Per ulteriori informazioni, consulta Distribuzione dei token di partecipazione nella Guida per l'utente dello Streaming in tempo reale di Amazon IVS.

- Per i dettagli sull'API, consulta Command ListPublicKeysReference AWS CLI .

`list-stage-sessions`

Il seguente esempio di codice mostra come utilizzare`list-stage-sessions`.

AWS CLI

Come ottenere un elenco delle sessioni di fase

L'esempio `list-stage-sessions` seguente elenca tutte le sessioni per un nome della risorsa Amazon (ARN) della fase specificato.

```
aws ivs-realtime list-stage-sessions \
    --stage-arn arn:aws:ivs:us-west-2:123456789012:stage/abcdABCDefgh
```

Output:

```
{
    "stageSessions": [
        {
            "endTime": "2023-04-26T20:36:29+00:00",
            "sessionId": "st-a1b2c3d4e5f6g",
            "startTime": "2023-04-26T20:30:29.602000+00:00"
        }
    ]
}
```

Per ulteriori informazioni, consulta Abilitazione di host multipli su un flusso Amazon IVS nella Guida per l'utente del Servizio video interattivo Amazon.

- Per i dettagli sull'API, consulta ListStageSessions AWS CLICommand Reference.

`list-stages`

Il seguente esempio di codice mostra come utilizzare`list-stages`.

AWS CLI

Come ottenere informazioni di riepilogo su tutte le fasi

L'`list-stages`esempio seguente elenca tutte le fasi del tuo AWS account, nella AWS regione in cui viene elaborata la richiesta API.

```
aws ivs-realtime list-stages
```

Output:

```
{
    "stages": [
        {
            "activeSessionId": "st-a1b2c3d4e5f6g",
            "arn": "arn:aws:ivs:us-west-2:123456789012:stage/abcdABCDefgh",
            "name": "stage1",
            "tags": {}
        },
        {
            "activeSessionId": "st-a123bcd456efg",
            "arn": "arn:aws:ivs:us-west-2:123456789012:stage/abcd1234ABCD",
            "name": "stage2",
            "tags": {}
        },
        {
            "activeSessionId": "st-abcDEF1234ghi",
            "arn": "arn:aws:ivs:us-west-2:123456789012:stage/ABCD1234efgh",
            "name": "stage3",
            "tags": {}
        }
    ]
}
```

Per ulteriori informazioni, consulta Abilitazione di host multipli su un flusso Amazon IVS nella Guida per l'utente del Servizio video interattivo Amazon.

- Per i dettagli sull'API, consulta ListStages AWS CLICommand Reference.

`list-storage-configurations`

Il seguente esempio di codice mostra come utilizzare`list-storage-configurations`.

AWS CLI

Come elencare le configurazioni dell'archiviazione della composizione

Di seguito `list-storage-configurations` sono elencate tutte le configurazioni di archiviazione della composizione per il tuo AWS account, nella AWS regione in cui viene elaborata la richiesta API.

```
aws ivs-realtime list-storage-configurations
```

Output:

```
{
    "storageConfigurations": [
        {
            "arn": "arn:aws:ivs:ap-northeast-1:123456789012:storage-configuration/
abcdABCDefgh",
            "name": "test-sc-1",
            "s3": {
                "bucketName": "amzn-s3-demo-bucket-1"
            },
            "tags": {}
        },
        {
            "arn": "arn:aws:ivs:ap-northeast-1:123456789012:storage-configuration/
ABCefgEFGabc",
            "name": "test-sc-2",
            "s3": {
                "bucketName": "amzn-s3-demo-bucket-2"
            },
            "tags": {}
        }
    ]
}
```

Per ulteriori informazioni, consulta Abilitazione di host multipli su un flusso Amazon IVS nella Guida per l'utente del Servizio video interattivo Amazon.

- Per i dettagli sull'API, consulta ListStorageConfigurations AWS CLICommand Reference.

`start-composition`

Il seguente esempio di codice mostra come utilizzare`start-composition`.

AWS CLI

Esempio 1: come avviare una composizione con le impostazioni di layout predefinite

L'esempio `start-composition` seguente avvia una composizione per la fase specificata da trasmettere nelle posizioni specificate.

```
aws ivs-realtime start-composition \
```

```
    --stage-arn arn:aws:ivs:ap-northeast-1:123456789012:stage/defgABCDabcd \
    --destinations '[{"channel": {"channelArn": "arn:aws:ivs:ap-
northeast-1:123456789012:channel/abcABCdefDEg", \
        "encoderConfigurationArn": "arn:aws:ivs:ap-northeast-1:123456789012:encoder-
configuration/ABabCDcdEFef"}}, \
        {"s3":{"encoderConfigurationArns":["arn:aws:ivs:ap-
northeast-1:123456789012:encoder-configuration/ABabCDcdEFef"], \
        "recordingConfiguration": {"hlsConfiguration":
 {"targetSegmentDurationSeconds": 5}}, \
        "storageConfigurationArn":"arn:aws:ivs:ap-northeast-1:123456789012:storage-
configuration/FefABabCDcdE"}}]'
```

Output:

```
{
    "composition": {
        "arn": "arn:aws:ivs:ap-northeast-1:123456789012:composition/abcdABCDefgh",
        "destinations": [
            {
                "configuration": {
                    "channel": {
                        "channelArn": "arn:aws:ivs:ap-
northeast-1:123456789012:channel/abcABCdefDEg",
                        "encoderConfigurationArn": "arn:aws:ivs:ap-
northeast-1:123456789012:encoder-configuration/ABabCDcdEFef"
                    },
                    "name": ""
                },
                "id": "AabBCcdDEefF",
                "state": "STARTING"
            },
            {
                "configuration": {
                    "name": "",
                    "s3": {
                        "encoderConfigurationArns": [
                            "arn:aws:ivs:arn:aws:ivs:ap-
northeast-1:123456789012:encoder-configuration/ABabCDcdEFef"
                        ],
                        "recordingConfiguration": {
                            "format": "HLS",
                            "hlsConfiguration": {
                                "targetSegmentDurationSeconds": 5
```

```
                            }
                        },
                        "storageConfigurationArn": "arn:arn:aws:ivs:ap-
northeast-1:123456789012:storage-configuration/FefABabCDcdE"
                    }
                },
                "detail": {
                    "s3": {
                        "recordingPrefix": "aBcDeFgHhGfE/AbCdEfGhHgFe/GHFabcgefABC/
composite"
                    }
                },
                "id": "GHFabcgefABC",
                "state": "STARTING"
            }
        ],
        "layout": {
            "grid": {
                "featuredParticipantAttribute": ""
                "gridGap": 2,
                "omitStoppedVideo": false,
                "participantOrderAttribute": "",
                "videoAspectRatio": "VIDEO",
                "videoFillMode": ""
            }
        },
        "stageArn": "arn:aws:ivs:ap-northeast-1:123456789012:stage/defgABCDabcd",
        "startTime": "2023-10-16T23:24:00+00:00",
        "state": "STARTING",
        "tags": {}
    }
}
```

Per ulteriori informazioni, consulta IVS Composite Recording | Real-Time Streaming nella Amazon Interactive Video Service User Guide.

Esempio 2: come avviare una composizione con il layout PiP

L'esempio `start-composition` seguente avvia una composizione per la fase specificata da trasmettere nelle posizioni specificate utilizzando il layout PiP.

```
aws ivs-realtime start-composition \
    --stage-arn arn:aws:ivs:ap-northeast-1:123456789012:stage/defgABCDabcd \
```

```
    --destinations '[{"channel": {"channelArn": "arn:aws:ivs:ap-
northeast-1:123456789012:channel/abcABCdefDEg", \
        "encoderConfigurationArn": "arn:aws:ivs:ap-northeast-1:123456789012:encoder-
configuration/ABabCDcdEFef"}}, \
        {"s3":{"encoderConfigurationArns":["arn:aws:ivs:ap-
northeast-1:123456789012:encoder-configuration/ABabCDcdEFef"], \
        "storageConfigurationArn":"arn:aws:ivs:ap-northeast-1:123456789012:storage-
configuration/FefABabCDcdE"}}]' \
    --layout pip='{featuredParticipantAttribute="abcdefg"}'
```

Output:

```
{
    "composition": {
        "arn": "arn:aws:ivs:ap-northeast-1:123456789012:composition/wxyzWXYZpqrs",
        "destinations": [
            {
                "configuration": {
                    "channel": {
                        "channelArn": "arn:aws:ivs:ap-
northeast-1:123456789012:channel/abcABCdefDEg",
                        "encoderConfigurationArn": "arn:aws:ivs:ap-
northeast-1:123456789012:encoder-configuration/ABabCDcdEFef"
                    },
                    "name": ""
                },
                "id": "AabBCcdDEefF",
                "state": "STARTING"
            },
            {
                "configuration": {
                    "name": "",
                    "s3": {
                        "encoderConfigurationArns": [
                            "arn:aws:ivs:arn:aws:ivs:ap-
northeast-1:123456789012:encoder-configuration/ABabCDcdEFef"
                        ],
                        "recordingConfiguration": {
                            "format": "HLS",
                            "hlsConfiguration": {
                                "targetSegmentDurationSeconds": 2
                            }
                        },
```

```
                        "storageConfigurationArn": "arn:arn:aws:ivs:ap-
northeast-1:123456789012:storage-configuration/FefABabCDcdE"
                    }
                },
                "detail": {
                    "s3": {
                        "recordingPrefix": "aBcDeFgHhGfE/AbCdEfGhHgFe/GHFabcgefABC/
composite"
                    }
                },
                "id": "GHFabcgefABC",
                "state": "STARTING"
            }
        ],
        "layout": {
            "pip": {
                "featuredParticipantAttribute": "abcdefg",
                "gridGap": 0,
                "omitStoppedVideo": false,
                "participantOrderAttribute": "",
                "pipBehavior": "STATIC",
                "pipOffset": 0,
                "pipParticipantAttribute": "",
                "pipPosition": "BOTTOM_RIGHT",
                "videoFillMode": "COVER"
            }
        },
        "stageArn": "arn:aws:ivs:ap-northeast-1:123456789012:stage/defgABCDabcd",
        "startTime": "2023-10-16T23:24:00+00:00",
        "state": "STARTING",
        "tags": {}
    }
}
```

Per ulteriori informazioni, consulta IVS Composite Recording | Real-Time Streaming nella Amazon Interactive Video Service User Guide.

Esempio 3: come avviare una composizione con la registrazione delle anteprime abilitata

L'esempio `start-composition` seguente avvia una composizione per la fase specificata da trasmettere nelle posizioni specificate con la registrazione delle anteprime abilitata.

```
aws ivs-realtime start-composition \
```

```
    --stage-arn arn:aws:ivs:ap-northeast-1:123456789012:stage/defgABCDabcd \
    --destinations '[{"channel": {"channelArn": "arn:aws:ivs:ap-
northeast-1:123456789012:channel/abcABCdefDEg", \
        "encoderConfigurationArn": "arn:aws:ivs:ap-northeast-1:123456789012:encoder-
configuration/ABabCDcdEFef"}}, \
        {"s3": {"encoderConfigurationArns": ["arn:aws:ivs:ap-
northeast-1:123456789012:encoder-configuration/ABabCDcdEFef"], \
        "storageConfigurationArn": "arn:aws:ivs:ap-northeast-1:123456789012:storage-
configuration/FefABabCDcdE", \
        "thumbnailConfigurations": [{"storage":
 ["SEQUENTIAL"],"targetIntervalSeconds": 60}]}}]'
```

Output:

```
{
    "composition": {
        "arn": "arn:aws:ivs:ap-northeast-1:123456789012:composition/abcdABCDefgh",
        "destinations": [
            {
                "configuration": {
                    "channel": {
                        "channelArn": "arn:aws:ivs:ap-
northeast-1:123456789012:channel/abcABCdefDEg",
                        "encoderConfigurationArn": "arn:aws:ivs:ap-
northeast-1:123456789012:encoder-configuration/ABabCDcdEFef"
                    },
                    "name": ""
                },
                "id": "AabBCcdDEefF",
                "state": "STARTING"
            },
            {
                "configuration": {
                    "name": "",
                    "s3": {
                        "encoderConfigurationArns": [
                            "arn:aws:ivs:arn:aws:ivs:ap-
northeast-1:123456789012:encoder-configuration/ABabCDcdEFef"
                        ],
                        "recordingConfiguration": {
                            "format": "HLS",
                            "hlsConfiguration": {
                                "targetSegmentDurationSeconds": 2
```

```
                            }
                        },
                        "storageConfigurationArn": "arn:arn:aws:ivs:ap-
northeast-1:123456789012:storage-configuration/FefABabCDcdE",
                        "thumbnailConfigurations": [
                           {
                              "targetIntervalSeconds": 60,
                              "storage": [
                                  "SEQUENTIAL"
                              ]
                           }
                        ]
                    }
                },
                "detail": {
                    "s3": {
                        "recordingPrefix": "aBcDeFgHhGfE/AbCdEfGhHgFe/GHFabcgefABC/
composite"
                    }
                },
                "id": "GHFabcgefABC",
                "state": "STARTING"
            }
        ],
        "layout": {
            "grid": {
                "featuredParticipantAttribute": ""
                "gridGap": 2,
                "omitStoppedVideo": false,
                "participantOrderAttribute": "",
                "videoAspectRatio": "VIDEO",
                "videoFillMode": ""
            }
        },
        "stageArn": "arn:aws:ivs:ap-northeast-1:123456789012:stage/defgABCDabcd",
        "startTime": "2023-10-16T23:24:00+00:00",
        "state": "STARTING",
        "tags": {}
    }
}
```

Per ulteriori informazioni, consulta Composite Recording (Real-Time Streaming) nella Guida per l'utente di Amazon Interactive Video Service.

Esempio 4: iniziare una composizione utilizzando un layout a griglia con un ordine personalizzato dei partecipanti

L'`start-composition`esempio seguente avvia una composizione per la fase specificata da trasmettere in streaming alle posizioni specificate utilizzando un layout a griglia con un ordine personalizzato dei partecipanti.

```
aws ivs-realtime start-composition \
    --stage-arn arn:aws:ivs:ap-northeast-1:123456789012:stage/defgABCDabcd \
    --destinations '[{"channel": {"channelArn": "arn:aws:ivs:ap-
northeast-1:123456789012:channel/abcABCdefDEg", \
        "encoderConfigurationArn": "arn:aws:ivs:ap-northeast-1:123456789012:encoder-
configuration/ABabCDcdEFef"}}, \
        {"s3": {"encoderConfigurationArns": ["arn:aws:ivs:ap-
northeast-1:123456789012:encoder-configuration/ABabCDcdEFef"], \
        "storageConfigurationArn": "arn:aws:ivs:ap-northeast-1:123456789012:storage-
configuration/FefABabCDcdE", \
        "thumbnailConfigurations": [{"storage":
 ["SEQUENTIAL"],"targetIntervalSeconds": 60}]}}]' \
    --layout grid='{participantOrderAttribute="abcdefg"}'
```

Output:

```
{
    "composition": {
        "arn": "arn:aws:ivs:ap-northeast-1:123456789012:composition/abcdABCDefgh",
        "destinations": [
            {
                "configuration": {
                    "channel": {
                        "channelArn": "arn:aws:ivs:ap-
northeast-1:123456789012:channel/abcABCdefDEg",
                        "encoderConfigurationArn": "arn:aws:ivs:ap-
northeast-1:123456789012:encoder-configuration/ABabCDcdEFef"
                    },
                    "name": ""
                },
                "id": "AabBCcdDEefF",
                "state": "STARTING"
            },
            {
                "configuration": {
                    "name": "",
```

```
                    "s3": {
                        "encoderConfigurationArns": [
                            "arn:aws:ivs:arn:aws:ivs:ap-
northeast-1:123456789012:encoder-configuration/ABabCDcdEFef"
                        ],
                        "recordingConfiguration": {
                            "format": "HLS",
                            "hlsConfiguration": {
                                "targetSegmentDurationSeconds": 2
                            }
                        },
                        "storageConfigurationArn": "arn:arn:aws:ivs:ap-
northeast-1:123456789012:storage-configuration/FefABabCDcdE",
                        "thumbnailConfigurations": [
                           {
                              "targetIntervalSeconds": 60,
                              "storage": [
                                  "SEQUENTIAL"
                              ]
                           }
                        ]
                    }
                },
                "detail": {
                    "s3": {
                        "recordingPrefix": "aBcDeFgHhGfE/AbCdEfGhHgFe/GHFabcgefABC/
composite"
                    }
                },
                "id": "GHFabcgefABC",
                "state": "STARTING"
            }
        ],
        "layout": {
            "grid": {
                "featuredParticipantAttribute": ""
                "gridGap": 2,
                "omitStoppedVideo": false,
                "participantOrderAttribute": "abcdefg",
                "videoAspectRatio": "VIDEO",
                "videoFillMode": ""
            }
        },
        "stageArn": "arn:aws:ivs:ap-northeast-1:123456789012:stage/defgABCDabcd",
```

```
        "startTime": "2023-10-16T23:24:00+00:00",
        "state": "STARTING",
        "tags": {}
    }
}
```

Per ulteriori informazioni, consulta IVS Composite Recording | Real-Time Streaming nella Amazon Interactive Video Service User Guide.

- Per i dettagli sull'API, consulta StartComposition AWS CLICommand Reference.

`start-participant-replication`

Il seguente esempio di codice mostra come utilizzare`start-participant-replication`.

AWS CLI

Come avviare la replica di un partecipante da una fase all'altra

L'esempio `start-participant-replication` seguente replica un partecipante da una fase di origine a una fase di destinazione, con ogni fase specificata dal relativo nome della risorsa Amazon (ARN).

```
aws ivs-realtime start-participant-replication \
    --source-stage-arn arn:aws:ivs:us-west-2:123456789012:stage/abcdABCDefgh \
    --destination-stage-arn arn:aws:ivs:us-west-2:234567890123:stage/bcdABCDefghi \
    --participant-id abCDEf12GHIj
```

Output:

```
{
    "accessControlAllowOrigin": "*",
    "accessControlExposeHeaders": "Access-Control-Allow-Origin,Access-Control-
Expose-Headers,Cache-Control,Content-Length, \
    Content-Security-Policy,Content-Type,date,Strict-Transport-Security,x-amz-apigw-
id,x-amzn-errormessage,x-amzn-errortype, \
    x-amzn-requestid,x-amzn-trace-id,X-Content-Type-Options,X-Frame-Options",
    "cacheControl": "no-store, no-cache",
    "contentSecurityPolicy": "default-src 'self'; upgrade-insecure-requests;",
    "strictTransportSecurity": "max-age:47304000; includeSubDomains",
    "xContentTypeOptions": "nosniff",
    "xFrameOptions": "DENY"
```

```
}
```

Per ulteriori informazioni, consulta Replica dei partecipanti IVS I Streaming in tempo reale nella Guida per l'utente dello Streaming in tempo reale di Amazon IVS.

- Per i dettagli sull'API, consulta StartParticipantReplication AWS CLICommand Reference.

`stop-composition`

Il seguente esempio di codice mostra come utilizzare`stop-composition`.

AWS CLI

Come arrestare una composizione

L'esempio `stop-composition` seguente arresta la composizione specificata dal nome della risorsa Amazon (ARN) indicato.

```
aws ivs-realtime stop-composition \
    --arn "arn:aws:ivs:ap-northeast-1:123456789012:composition/abcdABCDefgh"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Abilitazione di host multipli su un flusso Amazon IVS nella Guida per l'utente del Servizio video interattivo Amazon.

- Per i dettagli sull'API, consulta StopComposition AWS CLICommand Reference.

`stop-participant-replication`

Il seguente esempio di codice mostra come utilizzare`stop-participant-replication`.

AWS CLI

Come arrestare la replica di un partecipante da una fase all'altra

L'esempio `stop-participant-replication` seguente arresta la replica di un partecipante da una fase di origine a una fase di destinazione, con ogni fase specificata dal relativo nome della risorsa Amazon (ARN).

```
aws ivs-realtime stop-participant-replication \
    --source-stage-arn arn:aws:ivs:us-west-2:123456789012:stage/abcdABCDefgh \
```

```
    --destination-stage-arn arn:aws:ivs:us-west-2:234567890123:stage/bcdABCDefghi \
    --participant-id abCDEf12GHIj
```

Output:

```
{
    "accessControlAllowOrigin": "*",
    "accessControlExposeHeaders": "Access-Control-Allow-Origin,Access-Control-
Expose-Headers,Cache-Control,Content-Length, \
    Content-Security-Policy,Content-Type,date,Strict-Transport-Security,x-amz-apigw-
id,x-amzn-errormessage,x-amzn-errortype, \
    x-amzn-requestid,x-amzn-trace-id,X-Content-Type-Options,X-Frame-Options",
    "cacheControl": "no-store, no-cache",
    "contentSecurityPolicy": "default-src 'self'; upgrade-insecure-requests;",
    "strictTransportSecurity": "max-age:47304000; includeSubDomains",
    "xContentTypeOptions": "nosniff",
    "xFrameOptions": "DENY"
}
```

Per ulteriori informazioni, consulta Replica dei partecipanti IVS I Streaming in tempo reale nella Guida per l'utente dello Streaming in tempo reale di Amazon IVS.

- Per i dettagli sull'API, consulta StopParticipantReplication AWS CLICommand Reference.

`update-ingest-configuration`

Il seguente esempio di codice mostra come utilizzare`update-ingest-configuration`.

AWS CLI

Come aggiornare una configurazione di importazione

L'esempio `update-inegst-configuration` seguente aggiorna una configurazione di importazione per collegarla a una fase.

```
aws ivs-realtime update-ingest-configuration \
    --arn arn:aws:ivs:us-west-2:123456789012:ingest-configuration/AbCdEfGh1234 \
    --stage-arn arn:aws:ivs:us-west-2:123456789012:stage/abcdABCDefgh
```

Output:

```
{
```

```
    "ingestConfiguration": {
        "name": "ingest1",
        "arn": "arn:aws:ivs:us-west-2:123456789012:ingest-configuration/
AbCdEfGh1234",
        "ingestProtocol": "RTMPS",
        "streamKey": "rt_123456789012_us-
west-2_AbCdEfGh1234_abcd1234efgh5678ijkl9012MNOP34",
        "stageArn": "arn:aws:ivs:us-west-2:123456789012:stage/abcdABCDefgh",
        "participantId": "xyZ654abC321",
        "state": "INACTIVE",
        "userId": "",
        "tags": {}
    }
}
```

Per ulteriori informazioni, consulta Acquisizione dei flussi in IVS | Streaming in tempo reale nella Guida per l'utente di Servizio video interattivo Amazon.

- Per i dettagli sull'API, consulta UpdateIngestConfiguration AWS CLICommand Reference.

`update-stage`

Il seguente esempio di codice mostra come utilizzare`update-stage`.

AWS CLI

Esempio 1: come aggiornare la configurazione di una fase

L'esempio `update-stage` seguente aggiorna una fase per un nome della risorsa Amazon (ARN) specificato per aggiornare il nome della fase e configurare la registrazione dei singoli partecipanti con la registrazione delle anteprime abilitata.

```
aws ivs-realtime update-stage \
    --arn arn:aws:ivs:us-west-2:123456789012:stage/abcdABCDefgh \
    --auto-participant-recording-configuration '{"mediaTypes":
 ["AUDIO_VIDEO"],"storageConfigurationArn": "arn:aws:ivs:us-
west-2:123456789012:storage-configuration/abcdABCDefgh",
 "recordingReconnectWindowSeconds": 100, \
        "thumbnailConfiguration": {"recordingMode": "INTERVAL","storage":
 ["SEQUENTIAL"],"targetIntervalSeconds": 60}} \
        "hlsConfiguration": {"targetSegmentDurationSeconds": 5}}' \
    --name stage1a
```

Output:

```
{
    "stage": {
        "arn": "arn:aws:ivs:us-west-2:123456789012:stage/abcdABCDefgh",
        "autoParticipantRecordingConfiguration": {
            "mediaTypes": [
                "AUDIO_VIDEO"
            ],
            "storageConfigurationArn": "arn:aws:ivs:us-west-2:123456789012:storage-
configuration/abcdABCDefgh",
            "thumbnailConfiguration": {
                "targetIntervalSeconds": 60,
                "storage": [
                    "SEQUENTIAL"
                ],
                "recordingMode": "INTERVAL"
            },
            "recordingReconnectWindowSeconds": 100,
            "hlsConfiguration": {
                "targetSegmentDurationSeconds": 5
            },
            "recordParticipantReplicas": true
        },
        "endpoints": {
            "events": "wss://global.events.live-video.net",
            "rtmp": "rtmp://9x0y8z7s6t5u.global-contribute-staging.live-video.net/
app/",
            "rtmps": "rtmps://9x0y8z7s6t5u.global-contribute-staging.live-
video.net:443/app/",
            "whip": "https://1a2b3c4d5e6f.global-bm.whip.live-video.net"
        },
        "name": "stage1a",
        "tags": {}
    }
}
```

Per ulteriori informazioni, consulta Abilitazione di host multipli su un flusso Amazon IVS nella Guida per l'utente dello Streaming a bassa latenza di Amazon IVS.

Esempio 2: come aggiornare la configurazione di una fase, inclusa la disabilitazione della registrazione delle repliche dei partecipanti

L'esempio `update-stage` seguente aggiorna una fase per un nome della risorsa Amazon (ARN) specificato per aggiornare il nome della fase e configurare la registrazione dei singoli partecipanti con la registrazione delle anteprime abilitata e la registrazione delle repliche dei partecipanti disabilitata.

```
aws ivs-realtime update-stage \
    --arn arn:aws:ivs:us-west-2:123456789012:stage/abcdABCDefgh \
    --auto-participant-recording-configuration '{"mediaTypes":
 ["AUDIO_VIDEO"],"storageConfigurationArn": "arn:aws:ivs:us-
west-2:123456789012:storage-configuration/abcdABCDefgh",
 "recordingReconnectWindowSeconds": 100, \
        "thumbnailConfiguration": {"recordingMode": "INTERVAL","storage":
 ["SEQUENTIAL"],"targetIntervalSeconds": 60}, "recordParticipantReplicas":false} \
        "hlsConfiguration": {"targetSegmentDurationSeconds": 5}}' \
    --name stage1a
```

Output:

```
{
    "stage": {
        "arn": "arn:aws:ivs:us-west-2:123456789012:stage/abcdABCDefgh",
        "autoParticipantRecordingConfiguration": {
            "mediaTypes": [
                "AUDIO_VIDEO"
            ],
            "storageConfigurationArn": "arn:aws:ivs:us-west-2:123456789012:storage-
configuration/abcdABCDefgh",
            "thumbnailConfiguration": {
                "targetIntervalSeconds": 60,
                "storage": [
                    "SEQUENTIAL"
                ],
                "recordingMode": "INTERVAL"
            },
            "recordingReconnectWindowSeconds": 100,
            "hlsConfiguration": {
                "targetSegmentDurationSeconds": 5
            },
            "recordParticipantReplicas": false
        },
        "endpoints": {
            "events": "wss://global.events.live-video.net",
```

```
            "rtmp": "rtmp://9x0y8z7s6t5u.global-contribute-staging.live-video.net/
app/",
            "rtmps": "rtmps://9x0y8z7s6t5u.global-contribute-staging.live-
video.net:443/app/",
            "whip": "https://1a2b3c4d5e6f.global-bm.whip.live-video.net"
        },
        "name": "stage1a",
        "tags": {}
    }
}
```

Per ulteriori informazioni, consulta Abilitazione di host multipli su un flusso Amazon IVS nella Guida per l'utente dello Streaming a bassa latenza di Amazon IVS.

- Per i dettagli sull'API, consulta UpdateStage AWS CLICommand Reference.

Esempi di utilizzo di Amazon Kendra AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando Amazon Kendra. AWS Command Line Interface

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`create-data-source`

Il seguente esempio di codice mostra come usare. `create-data-source`

AWS CLI

Come creare un connettore di origine dati Amazon Kendra

L'esempio `create-data-source` seguente crea e configura un connettore di origine dati di Amazon Kendra. Puoi utilizzare `describe-data-source` per visualizzare lo stato di un connettore di origine dati e leggere eventuali messaggi di errore se lo stato indica che la creazione completa di un connettore di origine dati è "FAILED".

```
aws kendra create-data-source \
    --name "example data source 1" \
    --description "Example data source 1 for example index 1 contains the first set
 of example documents" \
    --tags '{"Key": "test resources", "Value": "kendra"}, {"Key": "test resources",
 "Value": "aws"}' \
    --role-arn "arn:aws:iam::my-account-id:role/
KendraRoleForS3TemplateConfigDataSource" \
    --index-id exampleindex1 \
    --language-code "es" \
    --schedule "0 0 18 ? * TUE,MON,WED,THU,FRI,SAT *" \
    --configuration '{"TemplateConfiguration": {"Template": file://
s3schemaconfig.json}}' \
    --type "TEMPLATE" \
    --custom-document-enrichment-configuration '{"PostExtractionHookConfiguration":
 {"LambdaArn": "arn:aws:iam::my-account-id:function/my-function-ocr-docs",
 "S3Bucket": "s3://amzn-s3-demo-bucket/scanned-image-text-example-docs"}, "RoleArn":
 "arn:aws:iam:my-account-id:role/KendraRoleForCDE"}' \
    --vpc-configuration '{"SecurityGroupIds": ["sg-1234567890abcdef0"], "SubnetIds":
 ["subnet-1c234","subnet-2b134"]}'
```

Output:

```
{
    "Id": "exampledatasource1"
}
```

Per ulteriori informazioni, consulta Nozioni di base su un indice e un connettore dell'origine dati di Amazon Kendra nella Guida per gli sviluppatori di Amazon Kendra.

- Per i dettagli sull'API, consulta CreateDataSource AWS CLICommand Reference.

`create-index`

Il seguente esempio di codice mostra come utilizzare`create-index`.

AWS CLI

Come creare un indice Amazon Kendra

Il comando `create-index` seguente crea e configura un indice di Amazon Kendra. Puoi utilizzare `describe-index` per visualizzare lo stato di un indice e leggere eventuali messaggi di errore se lo stato indica che la creazione completa di un indice è "FAILED".

```
aws kendra create-index \
    --name "example index 1" \
    --description "Example index 1 contains the first set of example documents" \
    --tags '{"Key": "test resources", "Value": "kendra"}, {"Key": "test resources",
 "Value": "aws"}' \
    --role-arn "arn:aws:iam::my-account-id:role/KendraRoleForExampleIndex" \
    --edition "DEVELOPER_EDITION" \
    --server-side-encryption-configuration '{"KmsKeyId": "my-kms-key-id"}' \
    --user-context-policy "USER_TOKEN" \
    --user-token-configurations '{"JsonTokenTypeConfiguration":
 {"GroupAttributeField": "groupNameField", "UserNameAttributeField":
 "userNameField"}}'
```

Output:

```
{
   "Id": index1
}
```

Per ulteriori informazioni, consulta Nozioni di base su un indice e un connettore dell'origine dati di Amazon Kendra nella Guida per gli sviluppatori di Amazon Kendra.

- Per i dettagli sull'API, consulta CreateIndex AWS CLICommand Reference.

`describe-data-source`

Il seguente esempio di codice mostra come utilizzare`describe-data-source`.

AWS CLI

Come ottenere informazioni su un connettore di origine dati Amazon Kendra

Il comando `describe-data-source` seguente fornisce informazioni su un connettore di origine dati di Amazon Kendra. Puoi visualizzare la configurazione di un connettore di origine dati e

leggere eventuali messaggi di errore se lo stato indica che la creazione completa di un connettore di origine dati è "FAILED".

```
aws kendra describe-data-source \
    --id exampledatasource1 \
    --index-id exampleindex1
```

Output:

```
{
    "Configuration": {
        "TemplateConfiguration": {
            "Template": {
                "connectionConfiguration": {
                    "repositoryEndpointMetadata": {
                        "BucketName": "amzn-s3-demo-bucket"
                    }
                },
                "repositoryConfigurations": {
                    "document":{
                        "fieldMappings": [
                            {
                                "indexFieldName":"_document_title",
                                "indexFieldType":"STRING",
                                "dataSourceFieldName": "title"
                            },
                            {
                                "indexFieldName":"_last_updated_at",
                                "indexFieldType":"DATE",
                                "dataSourceFieldName": "modified_date"
                            }
                        ]
                    }
                },
                "additionalProperties": {
                    "inclusionPatterns": [
                        "*.txt",
                        "*.doc",
                        "*.docx"
                    ],
                    "exclusionPatterns": [
                        "*.json"
                    ],
```

```
                    "inclusionPrefixes": [
                        "PublicExampleDocsFolder"
                    ],
                    "exclusionPrefixes": [
                        "PrivateDocsFolder/private"
                    ],
                    "aclConfigurationFilePath": "ExampleDocsFolder/AclConfig.json",
                    "metadataFilesPrefix": "metadata"
                },
                "syncMode": "FULL_CRAWL",
                "type" : "S3",
                "version": "1.0.0"
            }
        }
    },
    "CreatedAt": 2024-02-25T13:30:10+00:00,
    "CustomDocumentEnrichmentConfiguration": {
        "PostExtractionHookConfiguration": {
            "LambdaArn": "arn:aws:iam::my-account-id:function/my-function-ocr-docs",
            "S3Bucket": "s3://amzn-s3-demo-bucket/scanned-image-text-example-docs/
function"
        },
        "RoleArn": "arn:aws:iam:my-account-id:role/KendraRoleForCDE"
    }
    "Description": "Example data source 1 for example index 1 contains the first set
 of example documents",
    "Id": exampledatasource1,
    "IndexId": exampleindex1,
    "LanguageCode": "en",
    "Name": "example data source 1",
    "RoleArn": "arn:aws:iam::my-account-id:role/
KendraRoleForS3TemplateConfigDataSource",
    "Schedule": "0 0 18 ? * TUE,MON,WED,THU,FRI,SAT *",
    "Status": "ACTIVE",
    "Type": "TEMPLATE",
    "UpdatedAt": 1709163615,
    "VpcConfiguration": {
        "SecurityGroupIds": ["sg-1234567890abcdef0"],
        "SubnetIds": ["subnet-1c234","subnet-2b134"]
    }
}
```

Per ulteriori informazioni, consulta Nozioni di base su un indice e un connettore dell'origine dati di Amazon Kendra nella Guida per gli sviluppatori di Amazon Kendra.

- Per i dettagli sull'API, consulta DescribeDataSource AWS CLICommand Reference.

`describe-index`

Il seguente esempio di codice mostra come utilizzare`describe-index`.

AWS CLI

Come ottenere informazioni su un indice Amazon Kendra

Il comando `describe-index` seguente fornisce informazioni su un indice Amazon Kendra. Puoi visualizzare la configurazione di un indice e leggere eventuali messaggi di errore se lo stato indica che la creazione completa di un indice è "FAILED".

```
aws kendra describe-index \
    --id exampleindex1
```

Output:

```
{
    "CapacityUnits": {
        "QueryCapacityUnits": 0,
        "StorageCapacityUnits": 0
    },
    "CreatedAt": 2024-02-25T12:30:10+00:00,
    "Description": "Example index 1 contains the first set of example documents",
    "DocumentMetadataConfigurations": [
        {
            "Name": "_document_title",
            "Relevance": {
                "Importance": 8
            },
            "Search": {
                "Displayable": true,
                "Facetable": false,
                "Searchable": true,
                "Sortable": false
            },
            "Type": "STRING_VALUE"
        },
```

```
        {
            "Name": "_document_body",
            "Relevance": {
                "Importance": 5
            },
            "Search": {
                "Displayable": true,
                "Facetable": false,
                "Searchable": true,
                "Sortable": false
            },
            "Type": "STRING_VALUE"
        },
        {
            "Name": "_last_updated_at",
            "Relevance": {
                "Importance": 6,
                "Duration": "2628000s",
                "Freshness": true
            },
            "Search": {
                "Displayable": true,
                "Facetable": false,
                "Searchable": true,
                "Sortable": true
            },
            "Type": "DATE_VALUE"
        },
        {
            "Name": "department_custom_field",
            "Relevance": {
                "Importance": 7,
                "ValueImportanceMap": {
                    "Human Resources" : 4,
                    "Marketing and Sales" : 2,
                    "Research and innvoation" : 3,
                    "Admin" : 1
                }
            },
            "Search": {
                "Displayable": true,
                "Facetable": true,
                "Searchable": true,
                "Sortable": true
```

```
            },
            "Type": "STRING_VALUE"
        }
    ],
    "Edition": "DEVELOPER_EDITION",
    "Id": "index1",
    "IndexStatistics": {
        "FaqStatistics": {
            "IndexedQuestionAnswersCount": 10
        },
        "TextDocumentStatistics": {
            "IndexedTextBytes": 1073741824,
            "IndexedTextDocumentsCount": 1200
        }
    },
    "Name": "example index 1",
    "RoleArn": "arn:aws:iam::my-account-id:role/KendraRoleForExampleIndex",
    "ServerSideEncryptionConfiguration": {
        "KmsKeyId": "my-kms-key-id"
    },
    "Status": "ACTIVE",
    "UpdatedAt": 1709163615,
    "UserContextPolicy": "USER_TOKEN",
    "UserTokenConfigurations": [
        {
            "JsonTokenTypeConfiguration": {
                "GroupAttributeField": "groupNameField",
                "UserNameAttributeField": "userNameField"
            }
        }
    ]
}
```

Per ulteriori informazioni, consulta Nozioni di base su un indice e un connettore dell'origine dati di Amazon Kendra nella Guida per gli sviluppatori di Amazon Kendra.

- Per i dettagli sull'API, consulta DescribeIndex AWS CLICommand Reference.

`update-data-source`

Il seguente esempio di codice mostra come utilizzare`update-data-source`.

AWS CLI

Come aggiornare un connettore di origine dati Amazon Kendra

Il comando `update-data-source` seguente aggiorna la configurazione di un connettore di origine dati di Amazon Kendra. Se l'azione ha esito positivo, il servizio non restituisce alcun output, il codice di stato HTTP 200 o il codice restituito AWS CLI 0. È possibile utilizzare `describe-data-source` per visualizzare la configurazione e lo stato di un connettore di origine dati.

```
aws kendra update-data-source \
    --id exampledatasource1 \
    --index-id exampleindex1 \
    --name "new name for example data source 1" \
    --description "new description for example data source 1" \
    --role-arn arn:aws:iam::my-account-id:role/KendraNewRoleForExampleDataSource \
    --configuration '{"TemplateConfiguration": {"Template": file://
s3schemanewconfig.json}}' \
    --custom-document-enrichment-configuration '{"PostExtractionHookConfiguration":
 {"LambdaArn": "arn:aws:iam::my-account-id:function/my-function-ocr-docs",
 "S3Bucket": "s3://amzn-s3-demo-bucket/scanned-image-text-example-docs"}, "RoleArn":
 "arn:aws:iam:my-account-id:role/KendraNewRoleForCDE"}' \
    --language-code "es" \
    --schedule "0 0 18 ? * MON,WED,FRI *" \
    --vpc-configuration '{"SecurityGroupIds": ["sg-1234567890abcdef0"], "SubnetIds":
 ["subnet-1c234","subnet-2b134"]}'
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Nozioni di base su un indice e un connettore dell'origine dati di Amazon Kendra nella Guida per gli sviluppatori di Amazon Kendra.

- Per i dettagli sull'API, consulta UpdateDataSource AWS CLICommand Reference.

`update-index`

Il seguente esempio di codice mostra come utilizzare`update-index`.

AWS CLI

Come aggiornare un indice Amazon Kendra

Il comando `update-index` seguente aggiorna la configurazione di un indice Amazon Kendra. Se l'azione ha esito positivo, il servizio non restituisce alcun output, il codice di stato HTTP 200 o il codice restituito AWS CLI 0. È possibile utilizzare `describe-index` per visualizzare la configurazione e lo stato di un indice.

```
aws kendra update-index \
    --id enterpriseindex1 \
    --name "new name for Enterprise Edition index 1" \
    --description "new description for Enterprise Edition index 1" \
    --role-arn arn:aws:iam::my-account-id:role/KendraNewRoleForEnterpriseIndex \
    --capacity-units '{"QueryCapacityUnits": 2, "StorageCapacityUnits": 1}' \
    --document-metadata-configuration-updates '{"Name": "_document_title",
 "Relevance": {"Importance": 6}}, {"Name": "_last_updated_at", "Relevance":
 {"Importance": 8}}' \
    --user-context-policy "USER_TOKEN" \
    --user-token-configurations '{"JsonTokenTypeConfiguration":
 {"GroupAttributeField": "groupNameField", "UserNameAttributeField":
 "userNameField"}}'
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Nozioni di base su un indice e un connettore dell'origine dati di Amazon Kendra nella Guida per gli sviluppatori di Amazon Kendra.

- Per i dettagli sull'API, consulta UpdateIndex AWS CLICommand Reference.

Esempi di Kinesis che utilizzano AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface con Kinesis.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`add-tags-to-stream`

Il seguente esempio di codice mostra come utilizzare`add-tags-to-stream`.

AWS CLI

Come aggiungere tag a un flusso di dati

L'esempio `add-tags-to-stream` seguente assegna un tag con la chiave `samplekey` e il valore `example` al flusso specificato.

```
aws kinesis add-tags-to-stream \
    --stream-name samplestream \
    --tags samplekey=example
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Tagging dei flussi nella Guida per gli sviluppatori di Flusso di dati Amazon Kinesis.

- Per i dettagli sull'API, consulta AddTagsToStream AWS CLICommand Reference.

`create-stream`

Il seguente esempio di codice mostra come utilizzare`create-stream`.

AWS CLI

Come creare un flusso di dati

L'esempio `create-stream` seguente crea un flusso di dati denominato samplestream con 3 shard.

```
aws kinesis create-stream \
    --stream-name samplestream \
    --shard-count 3
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Creazione di un flusso nella Guida per gli sviluppatori di Flusso di dati Amazon Kinesis.

- Per i dettagli sull'API, consulta CreateStream AWS CLICommand Reference.

`decrease-stream-retention-period`

Il seguente esempio di codice mostra come utilizzare`decrease-stream-retention-period`.

AWS CLI

Come ridurre il periodo di conservazione del flusso di dati

L'esempio `decrease-stream-retention-period` seguente riduce il periodo di conservazione (tempo durante il quale i record di dati sono accessibili dopo essere stati aggiunti al flusso) di un flusso denominato samplestream a 48 ore.

```
aws kinesis decrease-stream-retention-period \
    --stream-name samplestream \
    --retention-period-hours 48
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Modifica del periodo di conservazione dei dati nella Guida per gli sviluppatori di Flusso di dati Amazon Kinesis.

- Per i dettagli sull'API, consulta DecreaseStreamRetentionPeriod AWS CLICommand Reference.

`delete-stream`

Il seguente esempio di codice mostra come utilizzare`delete-stream`.

AWS CLI

Come eliminare un flusso di dati

L'esempio `delete-stream` seguente elimina il flusso di dati specificato.

```
aws kinesis delete-stream \
```

```
    --stream-name samplestream
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Eliminazione di un flusso nella Guida per gli sviluppatori di Flusso di dati Amazon Kinesis.

- Per i dettagli sull'API, consulta DeleteStream AWS CLICommand Reference.

`deregister-stream-consumer`

Il seguente esempio di codice mostra come utilizzare`deregister-stream-consumer`.

AWS CLI

Come annullare la registrazione di un consumer del flusso di dati

L'esempio `deregister-stream-consumer` seguente annulla la registrazione del consumer specificato dal flusso di dati specificato.

```
aws kinesis deregister-stream-consumer \
    --stream-arn arn:aws:kinesis:us-west-2:123456789012:stream/samplestream \
    --consumer-name KinesisConsumerApplication
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Sviluppo di applicazioni consumer con fan-out avanzato con l'API di Flusso dati Kinesis nella Guida per gli sviluppatori di Flusso di dati Amazon Kinesis.

- Per i dettagli sull'API, consulta DeregisterStreamConsumer AWS CLICommand Reference.

`describe-limits`

Il seguente esempio di codice mostra come utilizzare`describe-limits`.

AWS CLI

Come descrivere i limiti degli shard

L'`describe-limits`esempio seguente mostra i limiti e l'utilizzo degli shard per l' AWS account corrente.

```
aws kinesis describe-limits
```

Output:

```
{
    "ShardLimit": 500,
    "OpenShardCount": 29
}
```

Per ulteriori informazioni, consulta Resharding di un flusso nella Guida per gli sviluppatori di Flusso di dati Amazon Kinesis.

- Per i dettagli sull'API, consulta DescribeLimits AWS CLICommand Reference.

`describe-stream-consumer`

Il seguente esempio di codice mostra come utilizzare`describe-stream-consumer`.

AWS CLI

Come descrivere un consumer del flusso di dati

L'esempio `describe-stream-consumer` seguente restituisce la descrizione del consumer specificato, registrato con il flusso di dati specificato.

```
aws kinesis describe-stream-consumer \
    --stream-arn arn:aws:kinesis:us-west-2:012345678912:stream/samplestream \
    --consumer-name KinesisConsumerApplication
```

Output:

```
{
    "ConsumerDescription": {
        "ConsumerName": "KinesisConsumerApplication",
        "ConsumerARN": "arn:aws:kinesis:us-west-2:123456789012:stream/samplestream/
consumer/KinesisConsumerApplication:1572383852",
        "ConsumerStatus": "ACTIVE",
        "ConsumerCreationTimestamp": 1572383852.0,
        "StreamARN": "arn:aws:kinesis:us-west-2:123456789012:stream/samplestream"
    }
}
```

Per ulteriori informazioni, consulta Lettura di dati dal flusso di dati Amazon Kinesis nella Guida per gli sviluppatori di Amazon Kinesis Data Streams.

- Per i dettagli sull'API, consulta DescribeStreamConsumer AWS CLICommand Reference.

`describe-stream-summary`

Il seguente esempio di codice mostra come utilizzare`describe-stream-summary`.

AWS CLI

Come descrivere un riepilogo del flusso di dati

L'esempio `describe-stream-summary` seguente fornisce un riepilogo della descrizione (senza l'elenco degli shard) del flusso di dati specificato.

```
aws kinesis describe-stream-summary \
    --stream-name samplestream
```

Output:

```
{
    "StreamDescriptionSummary": {
        "StreamName": "samplestream",
        "StreamARN": "arn:aws:kinesis:us-west-2:123456789012:stream/samplestream",
        "StreamStatus": "ACTIVE",
        "RetentionPeriodHours": 48,
        "StreamCreationTimestamp": 1572297168.0,
        "EnhancedMonitoring": [
            {
                "ShardLevelMetrics": []
            }
        ],
        "EncryptionType": "NONE",
        "OpenShardCount": 3,
        "ConsumerCount": 0
    }
}
```

Per ulteriori informazioni, consulta Creazione e gestione dei flussi nella Guida per gli sviluppatori di Flusso di dati Amazon Kinesis.

- Per i dettagli sull'API, consulta DescribeStreamSummary AWS CLICommand Reference.

`describe-stream`

Il seguente esempio di codice mostra come utilizzare`describe-stream`.

AWS CLI

Come descrivere un flusso di dati

L'esempio `describe-stream` seguente restituisce i dettagli del flusso di dati specificato.

```
aws kinesis describe-stream \
    --stream-name samplestream
```

Output:

```
{
    "StreamDescription": {
        "Shards": [
            {
                "ShardId": "shardId-000000000000",
                "HashKeyRange": {
                    "StartingHashKey": "0",
                    "EndingHashKey": "113427455640312821154458202477256070484"
                },
                "SequenceNumberRange": {
                    "StartingSequenceNumber":
 "496008716829570364423650249261910734372510605801286533 14"
                }
            },
            {
                "ShardId": "shardId-000000000001",
                "HashKeyRange": {
                    "StartingHashKey": "113427455640312821154458202477256070485",
                    "EndingHashKey": "226854911280625642308916404954512140969"
                },
                "SequenceNumberRange": {
                    "StartingSequenceNumber":
 "496008716829793371875635555493326091555237089416346337 46"
                }
            },
            {
                "ShardId": "shardId-000000000002",
                "HashKeyRange": {
```

```
                    "StartingHashKey": "22685491128062564230891640495451214097O",
                    "EndingHashKey": "340282366920938463463374607431768211455"
                },
                "SequenceNumberRange": {
                    "StartingSequenceNumber":
 "49600871683001637932762086172474144873796357303140614178"
                }
            }
        ],
        "StreamARN": "arn:aws:kinesis:us-west-2:123456789012:stream/samplestream",
        "StreamName": "samplestream",
        "StreamStatus": "ACTIVE",
        "RetentionPeriodHours": 24,
        "EnhancedMonitoring": [
            {
                "ShardLevelMetrics": []
            }
        ],
        "EncryptionType": "NONE",
        "KeyId": null,
        "StreamCreationTimestamp": 1572297168.0
    }
}
```

Per ulteriori informazioni, consulta Creazione e gestione dei flussi nella Guida per gli sviluppatori di Flusso di dati Amazon Kinesis.

- Per i dettagli sull'API, consulta DescribeStream AWS CLICommand Reference.

`disable-enhanced-monitoring`

Il seguente esempio di codice mostra come utilizzare`disable-enhanced-monitoring`.

AWS CLI

Come disabilitare il monitoraggio avanzato delle metriche a livello di shard

L'esempio `disable-enhanced-monitoring` seguente disabilita il monitoraggio avanzato delle metriche a livello di shard per il flusso di dati Kinesis.

```
aws kinesis disable-enhanced-monitoring \
    --stream-name samplestream --shard-level-metrics ALL
```

Output:

```
{
    "StreamName": "samplestream",
    "CurrentShardLevelMetrics": [
        "IncomingBytes",
        "OutgoingRecords",
        "IteratorAgeMilliseconds",
        "IncomingRecords",
        "ReadProvisionedThroughputExceeded",
        "WriteProvisionedThroughputExceeded",
        "OutgoingBytes"
    ],
    "DesiredShardLevelMetrics": []
}
```

Per ulteriori informazioni, consulta Monitoraggio dei flussi nel flusso di dati Amazon Kinesis nella Guida per gli sviluppatori di Flusso di dati Amazon Kinesis.

- Per i dettagli sull'API, consulta DisableEnhancedMonitoring AWS CLICommand Reference.

`enable-enhanced-monitoring`

Il seguente esempio di codice mostra come utilizzare`enable-enhanced-monitoring`.

AWS CLI

Come abilitare il monitoraggio avanzato delle metriche a livello di shard

L'esempio `enable-enhanced-monitoring` seguente abilita il monitoraggio avanzato delle metriche a livello di shard per il flusso di dati Kinesis.

```
aws kinesis enable-enhanced-monitoring \
    --stream-name samplestream \
    --shard-level-metrics ALL
```

Output:

```
{
    "StreamName": "samplestream",
    "CurrentShardLevelMetrics": [],
    "DesiredShardLevelMetrics": [
```

```
        "IncomingBytes",
        "OutgoingRecords",
        "IteratorAgeMilliseconds",
        "IncomingRecords",
        "ReadProvisionedThroughputExceeded",
        "WriteProvisionedThroughputExceeded",
        "OutgoingBytes"
    ]
}
```

Per ulteriori informazioni, consulta Monitoraggio dei flussi nel flusso di dati Amazon Kinesis nella Guida per gli sviluppatori di Flusso di dati Amazon Kinesis.

- Per i dettagli sull'API, consulta EnableEnhancedMonitoring AWS CLICommand Reference.

`get-records`

Il seguente esempio di codice mostra come utilizzare`get-records`.

AWS CLI

Come ottenere record da uno shard

L'esempio `get-records` seguente ottiene i record di dati dallo shard di un flusso di dati Kinesis mediante l'iterazione dei shard specificata.

```
aws kinesis get-records \
    --shard-iterator AAAAAAAAAAF7/0mWD7IuHj1yGv/
TKuNgx2ukD5xipCY4cy4gU96orWwZwcSXh3K9tAmGYeOZyLZrvzzeOFVf9iN99hUPw/w/
b0YWYeehfNvnf1DYt5XpDJghLKr3DzgznkTmMymDP3R+3wRKeuEw6/kdxY2yKJH0veaiekaVc4N2VwK/
GvaGP2Hh9Fg7N++q0Adg6fIDQPt4p8RpavDbk+A4sL9SWGE1
```

Output:

```
{
    "Records": [],
    "MillisBehindLatest": 80742000
}
```

Per ulteriori informazioni, consulta Developing Consumers Using the Kinesis Data Streams API AWS with the SDK for Java nella Amazon Kinesis Data Streams Developer Guide.

- Per i dettagli sull'API, consulta Command Reference. GetRecordsAWS CLI

`get-shard-iterator`

Il seguente esempio di codice mostra come utilizzare`get-shard-iterator`.

AWS CLI

Come ottenere un iteratore di shard

L'esempio `get-shard-iterator` seguente utilizza il tipo di iteratore dello shard `AT_SEQUENCE_NUMBER` e genera un iteratore dello shard per avviare la lettura dei record di dati esattamente dalla posizione indicata dal numero di sequenza specificato.

```
aws kinesis get-shard-iterator \
    --stream-name samplestream \
    --shard-id shardId-000000000001 \
    --shard-iterator-type LATEST
```

Output:

```
{
    "ShardIterator": "AAAAAAAAAAFEvJjIYI+3jw/4aqgH9FifJ+n48XWTh/
IFIsbILP6o5eDueD39NXNBfpZ10WL5K6ADXk8w+5H+Qhd9cFA9k268CPXCz/kebq1TGYI7Vy
+lUkA9BuN3xvATxMBGxRY3zYK05gqgvaIRn94O8SqeEqwhigwZxNWxID3Ej7YYYcxQi8Q/fIrCjGAy/
n2r5Z9G864YpWDfN9upNNQAR/iiOWKs"
}
```

Per ulteriori informazioni, consulta Developing Consumers Using the Kinesis Data Streams API AWS with the SDK for Java nella Amazon Kinesis Data Streams Developer Guide.

- Per i dettagli sull'API, consulta Command Reference. GetShardIteratorAWS CLI

`increase-stream-retention-period`

Il seguente esempio di codice mostra come utilizzare`increase-stream-retention-period`.

AWS CLI

Come aumentare il periodo di conservazione del flusso di dati

L'esempio `increase-stream-retention-period` seguente aumenta il periodo di conservazione (tempo durante il quale i record di dati sono accessibili dopo essere stati aggiunti al flusso) del flusso specificato a 168 ore.

```
aws kinesis increase-stream-retention-period \
    --stream-name samplestream \
    --retention-period-hours 168
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Modifica del periodo di conservazione dei dati nella Guida per gli sviluppatori di Flusso di dati Amazon Kinesis.

- Per i dettagli sull'API, consulta IncreaseStreamRetentionPeriod AWS CLICommand Reference.

`list-shards`

Il seguente esempio di codice mostra come utilizzare`list-shards`.

AWS CLI

Come elencare gli shard in un flusso di dati

L'esempio `list-shards` seguente elenca tutti gli shard nel flusso specificato, a partire dallo shard il cui ID segue immediatamente il parametro `exclusive-start-shard-id` specificato di `shardId-000000000000`.

```
aws kinesis list-shards \
    --stream-name samplestream \
    --exclusive-start-shard-id shardId-000000000000
```

Output:

```
{
    "Shards": [
        {
            "ShardId": "shardId-000000000001",
            "HashKeyRange": {
                "StartingHashKey": "113427455640312821154458202477256070485",
                "EndingHashKey": "226854911280625642308916404954512140969"
            },
            "SequenceNumberRange": {
                "StartingSequenceNumber":
 "496008716829793371875635555493326091555237089416346337746"
            }
```

```
        },
        {
            "ShardId": "shardId-000000000002",
            "HashKeyRange": {
                "StartingHashKey": "22685491128062564230891640495451214097O",
                "EndingHashKey": "340282366920938463463374607431768211455"
            },
            "SequenceNumberRange": {
                "StartingSequenceNumber":
 "49600871683001637932762086172474144873796357303140614178"
            }
        }
    ]
}
```

Per ulteriori informazioni, consulta Elenco degli shard nella Guida per gli sviluppatori di Flusso di dati Amazon Kinesis.

- Per i dettagli sull'API, consulta ListShards AWS CLICommand Reference.

`list-streams`

Il seguente esempio di codice mostra come utilizzare`list-streams`.

AWS CLI

Come elencare i flussi di dati

L'esempio `list-streams` seguente elenca tutti i flussi di dati attivi nell'account e nella Regione correnti.

```
aws kinesis list-streams
```

Output:

```
{
    "StreamNames": [
        "samplestream",
        "samplestream1"
    ]
}
```

Per ulteriori informazioni, consulta Elenco di flussi nella Guida per gli sviluppatori di Flusso di dati Amazon Kinesis.

- Per i dettagli sull'API, consulta ListStreams AWS CLICommand Reference.

`list-tags-for-stream`

Il seguente esempio di codice mostra come utilizzare`list-tags-for-stream`.

AWS CLI

Come elencare i tag di un flusso di dati

L'esempio `list-tags-for-stream` seguente elenca i tag associati al flusso di dati specificato.

```
aws kinesis list-tags-for-stream \
    --stream-name samplestream
```

Output:

```
{
    "Tags": [
        {
            "Key": "samplekey",
            "Value": "example"
        }
    ],
    "HasMoreTags": false
}
```

Per ulteriori informazioni, consulta Tagging dei flussi nella Guida per gli sviluppatori di Flusso di dati Amazon Kinesis.

- Per i dettagli sull'API, consulta ListTagsForStream AWS CLICommand Reference.

`merge-shards`

Il seguente esempio di codice mostra come utilizzare`merge-shards`.

AWS CLI

Come unire shard

L'`merge-shards`esempio seguente unisce due shard adiacenti con IDs shardId-000000000000 e shardId-000000000001 nel flusso di dati specificato e li combina in un unico shard.

```
aws kinesis merge-shards \
    --stream-name samplestream \
    --shard-to-merge shardId-000000000000 \
    --adjacent-shard-to-merge shardId-000000000001
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Unione di due shard nella Guida per gli sviluppatori di Flusso di dati Amazon Kinesis.

- Per i MergeShardsdettagli AWS CLI sull'API, consulta Command Reference.

`put-record`

Il seguente esempio di codice mostra come utilizzare`put-record`.

AWS CLI

Come scrivere un record in un flusso di dati

L'esempio `put-record` seguente scrive un unico record di dati nel flusso di dati specificato utilizzando la chiave di partizione specificata.

```
aws kinesis put-record \
    --stream-name samplestream \
    --data sampledatarecord \
    --partition-key samplepartitionkey
```

Output:

```
{
    "ShardId": "shardId-000000000009",
    "SequenceNumber": "49600902273357540915989931256901506243878407835297513618",
    "EncryptionType": "KMS"
}
```

Per ulteriori informazioni, consulta Developing Producers Using the Amazon Kinesis Data Streams API AWS with the SDK for Java nella Amazon Kinesis Data Streams Developer Guide.

- Per i dettagli sull'API, consulta Command Reference. PutRecordAWS CLI

`put-records`

Il seguente esempio di codice mostra come utilizzare`put-records`.

AWS CLI

Come scrivere più record in un flusso di dati

L'esempio `put-records` seguente scrive un record di dati utilizzando la chiave di partizione specificata e un altro record di dati utilizzando una chiave di partizione diversa in un'unica chiamata.

```
aws kinesis put-records \
    --stream-name samplestream \
    --
records Data=blob1,PartitionKey=partitionkey1 Data=blob2,PartitionKey=partitionkey2
```

Output:

```
{
    "FailedRecordCount": 0,
    "Records": [
        {
            "SequenceNumber":
 "49600883331171471519674795588238531498465399900093808706",
            "ShardId": "shardId-000000000004"
        },
        {
            "SequenceNumber":
 "49600902273357540915989931256902715169698037101720764562",
            "ShardId": "shardId-000000000009"
        }
    ],
    "EncryptionType": "KMS"
}
```

Per ulteriori informazioni, consulta Developing Producers Using the Amazon Kinesis Data Streams API AWS with the SDK for Java nella Amazon Kinesis Data Streams Developer Guide.

- Per i dettagli sull'API, consulta Command Reference. PutRecordsAWS CLI

`register-stream-consumer`

Il seguente esempio di codice mostra come utilizzare`register-stream-consumer`.

AWS CLI

Come registrare un consumer del flusso di dati

L'esempio `register-stream-consumer` seguente registra un consumer denominato `KinesisConsumerApplication` con il flusso di dati specificato.

```
aws kinesis register-stream-consumer \
    --stream-arn arn:aws:kinesis:us-west-2:012345678912:stream/samplestream \
    --consumer-name KinesisConsumerApplication
```

Output:

```
{
    "Consumer": {
        "ConsumerName": "KinesisConsumerApplication",
        "ConsumerARN": "arn:aws:kinesis:us-west-2: 123456789012:stream/samplestream/
consumer/KinesisConsumerApplication:1572383852",
        "ConsumerStatus": "CREATING",
        "ConsumerCreationTimestamp": 1572383852.0
    }
}
```

Per ulteriori informazioni, consulta Sviluppo di applicazioni consumer con fan-out avanzato con l'API di Flusso dati Kinesis nella Guida per gli sviluppatori di Flusso di dati Amazon Kinesis.

- Per i dettagli sull'API, consulta RegisterStreamConsumer AWS CLICommand Reference.

`remove-tags-from-stream`

Il seguente esempio di codice mostra come utilizzare`remove-tags-from-stream`.

AWS CLI

Come rimuovere i tag da un flusso di dati

L'esempio `remove-tags-from-stream` seguente rimuove il tag con la chiave specificata dal flusso di dati specificato.

```
aws kinesis remove-tags-from-stream \
    --stream-name samplestream \
    --tag-keys samplekey
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Tagging dei flussi nella Guida per gli sviluppatori di Flusso di dati Amazon Kinesis.

- Per i dettagli sull'API, consulta RemoveTagsFromStream AWS CLICommand Reference.

`split-shard`

Il seguente esempio di codice mostra come utilizzare`split-shard`.

AWS CLI

Come dividere gli shard

L'esempio `split-shard` seguente divide lo shard specificato in due nuovi shard utilizzando una nuova chiave hash iniziale pari a 10.

```
aws kinesis split-shard \
    --stream-name samplestream \
    --shard-to-split shardId-000000000000 \
    --new-starting-hash-key 10
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Suddivisione di uno shard nella Guida per gli sviluppatori di Flusso di dati Amazon Kinesis.

- Per i dettagli sull'API, consulta SplitShard AWS CLICommand Reference.

`start-stream-encryption`

Il seguente esempio di codice mostra come utilizzare`start-stream-encryption`.

AWS CLI

Come abilitare la crittografia del flusso di dati

L'`start-stream-encryption`esempio seguente abilita la crittografia lato server per lo stream specificato, utilizzando la chiave AWS KMS specificata.

```
aws kinesis start-stream-encryption \
    --encryption-type KMS \
    --key-id arn:aws:kms:us-west-2:012345678912:key/a3c4a7cd-728b-45dd-
b334-4d3eb496e452 \
    --stream-name samplestream
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Protezione dei dati in Flusso di dati Amazon Kinesis nella Guida per gli sviluppatori di Flusso di dati Amazon Kinesis.

- Per i dettagli sull'API, vedere StartStreamEncryptionin AWS CLI Command Reference.

`stop-stream-encryption`

Il seguente esempio di codice mostra come utilizzare`stop-stream-encryption`.

AWS CLI

Come disabilitare la crittografia del flusso di dati

L'`stop-stream-encryption`esempio seguente disabilita la crittografia lato server per lo stream specificato, utilizzando la chiave KMS specificata AWS .

```
aws kinesis start-stream-encryption \
    --encryption-type KMS \
    --key-id arn:aws:kms:us-west-2:012345678912:key/a3c4a7cd-728b-45dd-
b334-4d3eb496e452 \
    --stream-name samplestream
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Protezione dei dati in Flusso di dati Amazon Kinesis nella Guida per gli sviluppatori di Flusso di dati Amazon Kinesis.

- Per i dettagli sull'API, vedere StopStreamEncryptionin Command Reference.AWS CLI

`update-shard-count`

Il seguente esempio di codice mostra come utilizzare`update-shard-count`.

AWS CLI

Come aggiornare il numero di shard in un flusso di dati

L'esempio `update-shard-count` seguente aggiorna il numero di shard del flusso di dati specificato a 6. Questo esempio utilizza un dimensionamento uniforme, che crea shard di dimensioni uguali.

```
aws kinesis update-shard-count \
    --stream-name samplestream \
    --scaling-type UNIFORM_SCALING \
    --target-shard-count 6
```

Output:

```
{
    "StreamName": "samplestream",
    "CurrentShardCount": 3,
    "TargetShardCount": 6
}
```

Per ulteriori informazioni, consulta Resharding di un flusso nella Guida per gli sviluppatori di Flusso di dati Amazon Kinesis.

- Per i dettagli sull'API, consulta UpdateShardCount AWS CLICommand Reference.

AWS KMS esempi utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with AWS KMS.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`cancel-key-deletion`

Il seguente esempio di codice mostra come utilizzare`cancel-key-deletion`.

AWS CLI

Come annullare l'eliminazione pianificata di una chiave KMS gestita dal cliente

L'esempio `cancel-key-deletion` seguente annulla l'eliminazione pianificata di una chiave KMS gestita dal cliente.

```
aws kms cancel-key-deletion \
    --key-id 1234abcd-12ab-34cd-56ef-1234567890ab
```

Output:

```
{
    "KeyId": "arn:aws:kms:us-
west-2:123456789012:key/1234abcd-12ab-34cd-56ef-1234567890ab"
}
```

Quando il comando `cancel-key-deletion` ha esito positivo, l'eliminazione pianificata viene annullata. Tuttavia, lo stato della chiave KMS è `Disabled`, quindi non è possibile utilizzare la chiave KMS nelle operazioni di crittografia. Per ripristinarne la funzionalità, utilizza il comando `enable-key`.

Per ulteriori informazioni, consulta Pianificazione e annullamento dell'eliminazione di chiavi nella Guida per gli sviluppatori del Servizio AWS di gestione delle chiavi.

- Per i dettagli sull'API, consulta CancelKeyDeletion AWS CLICommand Reference.

`connect-custom-key-store`

Il seguente esempio di codice mostra come utilizzare`connect-custom-key-store`.

AWS CLI

Come connettere un archivio chiavi personalizzato

L'esempio `connect-custom-key-store` seguente riconnette l'archivio chiavi personalizzato specificato. È possibile utilizzare un comando come questo per connettere un archivio chiavi personalizzato per la prima volta o per riconnetterne uno disconnesso.

È possibile utilizzare questo comando per connettere un key store AWS CloudHSM o un key store esterno.

```
aws kms connect-custom-key-store \
    --custom-key-store-id cks-1234567890abcdef0
```

Questo comando non restituisce alcun output. Per verificare l'efficacia del comando, utilizza il comando `describe-custom-key-stores`.

Per informazioni sulla connessione di un key store AWS CloudHSM, consulta Connessione e disconnessione di un key store AWS CloudHSM nella Key Management Service Developer Guide.AWS

Per informazioni sulla connessione di un archivio chiavi esterno, consulta Connessione e disconnessione di un archivio delle chiavi esterne nella Guida per gli sviluppatori del Servizio AWS di gestione delle chiavi.

- Per i dettagli sull'API, consulta Command Reference. ConnectCustomKeyStoreAWS CLI

`create-alias`

Il seguente esempio di codice mostra come utilizzare`create-alias`.

AWS CLI

Come creare un alias per una chiave KMS

Il comando `create-alias` seguente crea un alias denominato `example-alias` per la chiave KMS identificata dall'ID chiave `1234abcd-12ab-34cd-56ef-1234567890ab`.

Un nome di alias deve iniziare con `alias/`. Non utilizzare alias che iniziano con`alias/aws`. Questi sono riservati all'uso di AWS.

```
aws kms create-alias \
```

```
    --alias-name alias/example-alias \
    --target-key-id 1234abcd-12ab-34cd-56ef-1234567890ab
```

Questo comando non restituisce output. Per visualizzare il nuovo alias, utilizza il comando `list-aliases`.

Per ulteriori informazioni, consulta Utilizzo di alias nella Guida per gli sviluppatori di Servizio AWS di gestione delle chiavi.

- Per i dettagli sull'API, consulta CreateAlias AWS CLICommand Reference.

`create-custom-key-store`

Il seguente esempio di codice mostra come utilizzare`create-custom-key-store`.

AWS CLI

Esempio 1: creare un key store AWS CloudHSM

L'`create-custom-key-store`esempio seguente crea un archivio di chiavi AWS CloudHSM supportato da un cluster AWS CloudHSM utilizzando i parametri richiesti. Puoi anche utilizzare `custom-key-store-type``parameter with the default value: ``AWS_CLOUDHSM`.

Per specificare l'input del file per il `trust-anchor-certificate` comando nella AWS CLI, è necessario il `file://` prefisso.

```
aws kms create-custom-key-store \
    --custom-key-store-name ExampleCloudHSMKeyStore \
    --cloud-hsm-cluster-id cluster-1a23b4cdefg \
    --key-store-password kmsPswd \
    --trust-anchor-certificate file://customerCA.crt
```

Output:

```
{
    "CustomKeyStoreId": cks-1234567890abcdef0
}
```

Per ulteriori informazioni, consulta Creating an AWS CloudHSM key store nella Key Management Service AWS Developer Guide.

Esempio 2: come creare un archivio chiavi esterno con connettività dell'endpoint pubblico

L'`create-custom-key-store`esempio seguente crea un archivio di chiavi esterno (XKS) che comunica con AWS KMS tramite Internet.

In questo esempio, `XksProxyUriPath` utilizza un prefisso opzionale di `example-prefix`.

NOTA: se utilizzi la versione AWS CLI 1.0, esegui il comando seguente prima di specificare un parametro con un valore HTTP o HTTPS, come il parametro. XksProxyUriEndpoint

```
aws configure set cli_follow_urlparam false
```

Altrimenti, la versione AWS CLI 1.0 sostituisce il valore del parametro con il contenuto trovato in quell'indirizzo URI.

```
aws kms create-custom-key-store \
    --custom-key-store-name ExamplePublicEndpointXKS \
    --custom-key-store-type EXTERNAL_KEY_STORE \
    --xks-proxy-connectivity PUBLIC_ENDPOINT \
    --xks-proxy-uri-endpoint "https://myproxy.xks.example.com" \
    --xks-proxy-uri-path "/example-prefix/kms/xks/v1" \
    --xks-proxy-authentication-credential "AccessKeyId=ABCDE12345670EXAMPLE,
 RawSecretAccessKey=DXjSUawnel2fr6SKC7G25CNxTyWKE5PF9XX6H/u9pSo="
```

Output:

```
{
    "CustomKeyStoreId": cks-2234567890abcdef0
}
```

Per ulteriori informazioni, consulta Creazione di un archivio chiavi esterno nella Guida per gli sviluppatori del Servizio AWS di gestione delle chiavi.

Esempio 3: come creare un archivio chiavi esterno con connettività di servizio dell'endpoint VPC

L'`create-custom-key-store`esempio seguente crea un archivio di chiavi esterno (XKS) che utilizza un servizio endpoint Amazon VPC per comunicare con KMS. AWS

NOTA: se utilizzi la versione AWS CLI 1.0, esegui il comando seguente prima di specificare un parametro con un valore HTTP o HTTPS, come il parametro. XksProxyUriEndpoint

```
aws configure set cli_follow_urlparam false
```

Altrimenti, la versione AWS CLI 1.0 sostituisce il valore del parametro con il contenuto trovato in quell'indirizzo URI.

```
aws kms create-custom-key-store \
    --custom-key-store-name ExampleVPCEndpointXKS \
    --custom-key-store-type EXTERNAL_KEY_STORE \
    --xks-proxy-connectivity VPC_ENDPOINT_SERVICE \
    --xks-proxy-uri-endpoint "https://myproxy-private.xks.example.com" \
    --xks-proxy-uri-path "/kms/xks/v1" \
    --xks-proxy-vpc-endpoint-service-name "com.amazonaws.vpce.us-east-1.vpce-svc-example1" \
    --xks-proxy-authentication-credential "AccessKeyId=ABCDE12345670EXAMPLE,RawSecretAccessKey=DXjSUawnel2fr6SKC7G25CNxTyWKE5PF9XX6H/u9pSo="
```

Output:

```
{
    "CustomKeyStoreId": cks-3234567890abcdef0
}
```

Per ulteriori informazioni, consulta Creazione di un archivio chiavi esterno nella Guida per gli sviluppatori del Servizio AWS di gestione delle chiavi.

- Per i dettagli sull'API, consulta AWS CLI Command CreateCustomKeyStoreReference.

`create-grant`

Il seguente esempio di codice mostra come utilizzare`create-grant`.

AWS CLI

Come creare una concessione

L'esempio `create-grant` seguente crea una concessione che consente all'utente `exampleUser` di utilizzare il comando `decrypt` sulla chiave KMS di esempio `1234abcd-12ab-34cd-56ef-1234567890ab`. Il principale in fase di ritiro è il ruolo `adminRole`. La concessione utilizza il vincolo di concessione `EncryptionContextSubset` per consentire questa autorizzazione solo quando il contesto di crittografia nella richiesta `decrypt` include la coppia chiave-valore `"Department": "IT"`.

```
aws kms create-grant \
```

```
    --key-id 1234abcd-12ab-34cd-56ef-1234567890ab \
    --grantee-principal arn:aws:iam::123456789012:user/exampleUser \
    --operations Decrypt \
    --constraints EncryptionContextSubset={Department=IT} \
    --retiring-principal arn:aws:iam::123456789012:role/adminRole
```

Output:

```
{
    "GrantId": "1a2b3c4d2f5e69f440bae30eaec9570bb1fb7358824f9ddfa1aa5a0dab1a59b2",
    "GrantToken": "<grant token here>"
}
```

Per visualizzare informazioni dettagliate sulla concessione, utilizza il comando `list-grants`.

Per ulteriori informazioni, consulta Grants in AWS KMS nella AWS Key Management Service Developer Guide.

- Per i dettagli sull'API, consulta AWS CLI Command CreateGrantReference.

`create-key`

Il seguente esempio di codice mostra come utilizzare`create-key`.

AWS CLI

Esempio 1: creare una chiave KMS gestita dal cliente in AWS KMS

L'esempio `create-key` seguente crea una chiave KMS di crittografia simmetrica.

Per creare la chiave KMS di base, una chiave di crittografia simmetrica, non è necessario specificare alcun parametro. I valori predefiniti per questi parametri creano una chiave di crittografia simmetrica.

Poiché questo comando non specifica una policy della chiave, la chiave KMS ottiene la policy di chiave predefinita per le chiavi KMS create a livello di programmazione. Per visualizzare la policy della chiave, utilizza il comando `get-key-policy`. Per modificare la policy della chiave, utilizza il comando `put-key-policy`.

```
aws kms create-key
```

Il comando `create-key` restituisce i metadati della chiave, inclusi l'ID chiave e l'ARN della nuova chiave KMS. Puoi utilizzare questi valori per identificare la chiave KMS in altre AWS operazioni KMS. L'output non include i tag. Per visualizzare i tag di una chiave KMS, utilizza `list-resource-tags command`.

Output:

```
{
    "KeyMetadata": {
        "AWSAccountId": "111122223333",
        "Arn": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
        "CreationDate": "2017-07-05T14:04:55-07:00",
        "CurrentKeyMaterialId":
 "0b7fd7ddbac6eef27907413567cad8c810e2883dc8a7534067a82ee1142fc1e6",
        "CustomerMasterKeySpec": "SYMMETRIC_DEFAULT",
        "Description": "",
        "Enabled": true,
        "KeyId": "1234abcd-12ab-34cd-56ef-1234567890ab",
        "KeyManager": "CUSTOMER",
        "KeySpec": "SYMMETRIC_DEFAULT",
        "KeyState": "Enabled",
        "KeyUsage": "ENCRYPT_DECRYPT",
        "MultiRegion": false,
        "Origin": "AWS_KMS"
        "EncryptionAlgorithms": [
            "SYMMETRIC_DEFAULT"
        ]
    }
}
```

Nota: il comando `create-key` non consente di specificare un alias. Per creare un alias per la nuova chiave KMS, utilizza il comando `create-alias`.

Per ulteriori informazioni, consulta Creazione di chiavi nella Guida per gli sviluppatori del Servizio AWS di gestione delle chiavi.

Esempio 2: come creare una chiave KMS RSA asimmetrica per la crittografia e la decrittografia

L'esempio `create-key` seguente crea una chiave KMS contenente una coppia di chiavi RSA asimmetriche per la crittografia e la decrittografia. La specifica e l'utilizzo della chiave non possono essere modificati dopo che la chiave è stata creata:

```
aws kms create-key \
   --key-spec RSA_4096 \
   --key-usage ENCRYPT_DECRYPT
```

Output:

```
{
    "KeyMetadata": {
        "Arn": "arn:aws:kms:us-
east-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
        "AWSAccountId": "111122223333",
        "CreationDate": "2021-04-05T14:04:55-07:00",
        "CustomerMasterKeySpec": "RSA_4096",
        "Description": "",
        "Enabled": true,
        "EncryptionAlgorithms": [
            "RSAES_OAEP_SHA_1",
            "RSAES_OAEP_SHA_256"
        ],
        "KeyId": "1234abcd-12ab-34cd-56ef-1234567890ab",
        "KeyManager": "CUSTOMER",
        "KeySpec": "RSA_4096",
        "KeyState": "Enabled",
        "KeyUsage": "ENCRYPT_DECRYPT",
        "MultiRegion": false,
        "Origin": "AWS_KMS"
    }
}
```

Per ulteriori informazioni, consulta Chiavi asimmetriche in AWS KMS nella Guida per gli sviluppatori del servizio di gestione delle AWS chiavi.

Esempio 3: come creare una chiave KMS asimmetrica a curva ellittica per la firma e la verifica

Come creare una chiave KMS asimmetrica contenente una coppia di chiavi asimmetriche basate su curva ellittica (ECC) per la firma e la verifica. Il parametro `--key-usage` è obbligatorio anche se `SIGN_VERIFY` è l'unico valore valido per le chiavi KMS ECC. La specifica e l'utilizzo della chiave non possono essere modificati dopo che la chiave è stata creata:

```
aws kms create-key \
    --key-spec ECC_NIST_P521 \
```

```
    --key-usage SIGN_VERIFY
```

Output:

```
{
    "KeyMetadata": {
        "Arn": "arn:aws:kms:us-
east-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
        "AWSAccountId": "111122223333",
        "CreationDate": "2019-12-02T07:48:55-07:00",
        "CustomerMasterKeySpec": "ECC_NIST_P521",
        "Description": "",
        "Enabled": true,
        "KeyId": "1234abcd-12ab-34cd-56ef-1234567890ab",
        "KeyManager": "CUSTOMER",
        "KeySpec": "ECC_NIST_P521",
        "KeyState": "Enabled",
        "KeyUsage": "SIGN_VERIFY",
        "MultiRegion": false,
        "Origin": "AWS_KMS",
        "SigningAlgorithms": [
            "ECDSA_SHA_512"
        ]
    }
}
```

Per ulteriori informazioni, consulta Chiavi asimmetriche in AWS KMS nella Key Management Service Developer Guide.AWS

Esempio 4: come creare una chiave KMS ML-DSA asimmetrica per la firma e la verifica

Questo esempio crea una chiave basata sullo schema ML-DSA (Module-Lattice Digital Signature Algorithm) per la firma e la verifica. Il parametro key-usage è obbligatorio anche se `SIGN_VERIFY` è l'unico valore valido per le chiavi ML-DSA.

```
aws kms create-key \
    --key-spec ML_DSA_65 \
    --key-usage SIGN_VERIFY
```

Output:

```
{
```

```
    "KeyMetadata": {
        "Arn": "arn:aws:kms:us-
east-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
        "AWSAccountId": "111122223333",
        "CreationDate": "2019-12-02T07:48:55-07:00",
        "Description": "",
        "Enabled": true,
        "KeyId": "1234abcd-12ab-34cd-56ef-1234567890ab",
        "KeyManager": "CUSTOMER",
        "KeySpec": "ML_DSA_65",
        "KeyState": "Enabled",
        "KeyUsage": "SIGN_VERIFY",
        "MultiRegion": false,
        "Origin": "AWS_KMS",
        "SigningAlgorithms": [
            "ML_DSA_SHAKE_256"
        ]
    }
}
```

Per ulteriori informazioni, consulta Chiavi asimmetriche in AWS KMS nella Key Management Service Developer Guide.AWS

Esempio 5: come creare una chiave KMS HMAC

L'esempio `create-key` seguente crea una chiave KMS HMAC a 384 bit. Il valore `GENERATE_VERIFY_MAC` del parametro `--key-usage` è obbligatorio anche se è l'unico valore valido per le chiavi KMS HMAC.

```
aws kms create-key \
    --key-spec HMAC_384 \
    --key-usage GENERATE_VERIFY_MAC
```

Output:

```
{
    "KeyMetadata": {
        "Arn": "arn:aws:kms:us-
east-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
        "AWSAccountId": "111122223333",
        "CreationDate": "2022-04-05T14:04:55-07:00",
        "CustomerMasterKeySpec": "HMAC_384",
```

```
        "Description": "",
        "Enabled": true,
        "KeyId": "1234abcd-12ab-34cd-56ef-1234567890ab",
        "KeyManager": "CUSTOMER",
        "KeySpec": "HMAC_384",
        "KeyState": "Enabled",
        "KeyUsage": "GENERATE_VERIFY_MAC",
        "MacAlgorithms": [
            "HMAC_SHA_384"
        ],
        "MultiRegion": false,
        "Origin": "AWS_KMS"
    }
}
```

Per ulteriori informazioni, consulta le chiavi HMAC in AWS KMS nella Key Management Service Developer Guide.AWS

Esempio 6: come creare una chiave KMS primaria multi-Regione

L'esempio `create-key` seguente crea una chiave di crittografia simmetrica primaria multi-Regione. Poiché i valori predefiniti per tutti i parametri creano una chiave di crittografia simmetrica, per questa chiave KMS è richiesto solo il parametro `--multi-region`. Nella AWS CLI, per indicare che un parametro booleano è vero, è sufficiente specificare il nome del parametro.

```
aws kms create-key \
    --multi-region
```

Output:

```
{
    "KeyMetadata": {
        "Arn": "arn:aws:kms:us-west-2:111122223333:key/
mrk-1234abcd12ab34cd56ef12345678990ab",
        "AWSAccountId": "111122223333",
        "CreationDate": "2021-09-02T016:15:21-09:00",
        "CurrentKeyMaterialId":
 "0b7fd7ddbac6eef27907413567cad8c810e2883dc8a7534067a82ee1142fc1e6",
        "CustomerMasterKeySpec": "SYMMETRIC_DEFAULT",
        "Description": "",
        "Enabled": true,
```

```
        "EncryptionAlgorithms": [
          "SYMMETRIC_DEFAULT"
        ],
        "KeyId": "mrk-1234abcd12ab34cd56ef12345678990ab",
        "KeyManager": "CUSTOMER",
        "KeySpec": "SYMMETRIC_DEFAULT",
        "KeyState": "Enabled",
        "KeyUsage": "ENCRYPT_DECRYPT",
        "MultiRegion": true,
        "MultiRegionConfiguration": {
            "MultiRegionKeyType": "PRIMARY",
            "PrimaryKey": {
                "Arn": "arn:aws:kms:us-west-2:111122223333:key/
mrk-1234abcd12ab34cd56ef12345678990ab",
                "Region": "us-west-2"
            },
            "ReplicaKeys": []
        },
        "Origin": "AWS_KMS"
    }
}
```

Per ulteriori informazioni, consulta la sezione Chiavi asimmetriche in AWS KMS nella Key Management Service Developer Guide.AWS

Esempio 7: come creare una chiave KMS per il materiale della chiave importato

L'esempio `create-key` seguente crea una chiave KMS senza materiale di chiave. Quando l'operazione è stata completata, puoi importare il materiale della chiave nella chiave KMS. Per creare questa chiave KMS, imposta il parametro `--origin` su `EXTERNAL`.

```
aws kms create-key \
    --origin EXTERNAL
```

Output:

```
{
     "KeyMetadata": {
         "Arn": "arn:aws:kms:us-
east-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
         "AWSAccountId": "111122223333",
         "CreationDate": "2019-12-02T07:48:55-07:00",
```

```
        "CustomerMasterKeySpec": "SYMMETRIC_DEFAULT",
        "Description": "",
        "Enabled": false,
        "EncryptionAlgorithms": [
            "SYMMETRIC_DEFAULT"
        ],
        "KeyId": "1234abcd-12ab-34cd-56ef-1234567890ab",
        "KeyManager": "CUSTOMER",
        "KeySpec": "SYMMETRIC_DEFAULT",
        "KeyState": "PendingImport",
        "KeyUsage": "ENCRYPT_DECRYPT",
        "MultiRegion": false,
        "Origin": "EXTERNAL"
    }
}
```

Per ulteriori informazioni, consulta Importazione di materiale chiave nelle chiavi AWS KMS nella Key Management Service Developer Guide.AWS

Esempio 6: creare una chiave KMS in un key store AWS CloudHSM

L'`create-key`esempio seguente crea una chiave KMS nell'archivio di chiavi AWS CloudHSM specificato. L'operazione crea la chiave KMS e i relativi metadati in AWS KMS e crea il materiale chiave nel cluster CloudHSM associato all' AWS archivio chiavi personalizzato. I parametri `--custom-key-store-id` e `--origin` sono obbligatori.

```
aws kms create-key \
    --origin AWS_CLOUDHSM \
    --custom-key-store-id cks-1234567890abcdef0
```

Output:

```
{
    "KeyMetadata": {
        "Arn": "arn:aws:kms:us-
east-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
        "AWSAccountId": "111122223333",
        "CloudHsmClusterId": "cluster-1a23b4cdefg",
        "CreationDate": "2019-12-02T07:48:55-07:00",
        "CustomerMasterKeySpec": "SYMMETRIC_DEFAULT",
        "CustomKeyStoreId": "cks-1234567890abcdef0",
        "Description": "",
```

```
        "Enabled": true,
        "EncryptionAlgorithms": [
            "SYMMETRIC_DEFAULT"
        ],
        "KeyId": "1234abcd-12ab-34cd-56ef-1234567890ab",
        "KeyManager": "CUSTOMER",
        "KeySpec": "SYMMETRIC_DEFAULT",
        "KeyState": "Enabled",
        "KeyUsage": "ENCRYPT_DECRYPT",
        "MultiRegion": false,
        "Origin": "AWS_CLOUDHSM"
    }
}
```

Per ulteriori informazioni, consulta Archivio chiavi AWS CloudHSM nella Guida per gli sviluppatori del Servizio AWS di gestione delle chiavi.

Esempio 8: come creare una chiave KMS in un archivio chiavi esterno

L'esempio `create-key` seguente crea una chiave KMS nell'archivio chiavi esterno specificato. I parametri `--custom-key-store-id`, `--origin` e `--xks-key-id` sono tutti obbligatori in questo comando.

Il parametro `--xks-key-id` specifica l'ID di una chiave di crittografia simmetrica esistente nel gestore chiavi esterno. Questa chiave funge da materiale della chiave esterna per la chiave KMS. Il valore del parametro `--origin` deve essere `EXTERNAL_KEY_STORE`. Il parametro `custom-key-store-id` deve identificare un archivio chiavi esterno collegato al proxy dell'archivio chiavi esterno.

```
aws kms create-key \
    --origin EXTERNAL_KEY_STORE \
    --custom-key-store-id cks-9876543210fedcba9 \
    --xks-key-id bb8562717f809024
```

Output:

```
{
    "KeyMetadata": {
        "Arn": "arn:aws:kms:us-
east-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
        "AWSAccountId": "111122223333",
        "CreationDate": "2022-12-02T07:48:55-07:00",
```

```
        "CustomerMasterKeySpec": "SYMMETRIC_DEFAULT",
        "CustomKeyStoreId": "cks-9876543210fedcba9",
        "Description": "",
        "Enabled": true,
        "EncryptionAlgorithms": [
            "SYMMETRIC_DEFAULT"
        ],
        "KeyId": "1234abcd-12ab-34cd-56ef-1234567890ab",
        "KeyManager": "CUSTOMER",
        "KeySpec": "SYMMETRIC_DEFAULT",
        "KeyState": "Enabled",
        "KeyUsage": "ENCRYPT_DECRYPT",
        "MultiRegion": false,
        "Origin": "EXTERNAL_KEY_STORE",
        "XksKeyConfiguration": {
            "Id": "bb8562717f809024"
        }
    }
}
```

Per ulteriori informazioni, consulta Archivi chiavi esterni nella Guida per gli sviluppatori del Servizio AWS di gestione delle chiavi.

- Per i dettagli sull'API, consulta Command Reference. CreateKeyAWS CLI

`decrypt`

Il seguente esempio di codice mostra come utilizzare`decrypt`.

AWS CLI

Esempio 1: come decrittografare un messaggio crittografato con una chiave KMS simmetrica (Linux e macOS)

Il seguente esempio di `decrypt` comando mostra il metodo consigliato per decrittografare i dati con la CLI. AWS Questa versione mostra come decrittografare i dati con una chiave KMS simmetrica.

Fornisci il testo criptato in un file. Nel valore del parametro `--ciphertext-blob` utilizza il prefisso `fileb://` per indicare alla CLI di leggere i dati da un file binario. Se il file non è presente nella directory corrente, digita il percorso completo del file. Per ulteriori informazioni sulla lettura dei valori dei parametri AWS CLI da un file, consulta Loading AWS CLI parameters

from a file < https://docs.aws.amazon.com/cli/ latest/userguide/cli - usage-parameters-file .html> nella Command Line Interface User Guide e Best Practices for Local File Parameters< https:// aws.amazon.com/blogs/ developer/ best-practices-for-local -file-parameters/> nel AWS Command Line Tool Blog .Specificare la chiave KMS per decrittografare il CipherText.Il AWS parametro non è necessario quando si esegue la decrittografia con una chiave KMS simmetrica. `--key-id` AWS KMS può ottenere l'ID della chiave KMS utilizzata per crittografare i dati dai metadati nel testo cifrato. Tuttavia è sempre consigliabile specificare la chiave KMS che stai utilizzando. Questa procedura consente di utilizzare la chiave KMS desiderata e impedire di decrittografare inavvertitamente un testo criptato utilizzando una chiave KMS non attendibile. Richiedi l'output in testo non crittografato come valore di testo. Il parametro `--query` indica alla CLI di recuperare solo il valore del campo `Plaintext` dall'output. Il parametro `--output` restituisce l'output come testo. Procedi a decrittografare in formato Base64 il testo non crittografato e a salvarlo in un file. Nell'esempio seguente il valore del parametro `Plaintext` viene inviato (|) all'utilità Base64, che lo decodifica. Quindi, reindirizza (>) l'output decodificato al file `ExamplePlaintext`.

Prima di eseguire questo comando, sostituisci l'ID della chiave di esempio con un ID chiave valido del tuo account. AWS

```
aws kms decrypt \
    --ciphertext-blob fileb://ExampleEncryptedFile \
    --key-id 1234abcd-12ab-34cd-56ef-1234567890ab \
    --output text \
    --query Plaintext | base64 \
    --decode > ExamplePlaintextFile
```

Questo comando non produce alcun output. L'output del comando `decrypt` viene decrittografato in formato Base64 e salvato in un file.

Per ulteriori informazioni, consulta Decrypt nel riferimento API del Servizio AWS di gestione delle chiavi.

Esempio 2: come decrittografare un messaggio crittografato con una chiave KMS simmetrica (prompt dei comandi di Windows)

L'esempio seguente è lo stesso del precedente, tranne per il fatto che utilizza l'utilità `certutil` per decrittografato in formato Base64 i dati del testo non crittografato. Questa procedura richiede due comandi, come illustrato negli esempi seguenti.

Prima di eseguire questo comando, sostituisci l'ID della chiave di esempio con un ID chiave valido del tuo AWS account.

```
aws kms decrypt ^
    --ciphertext-blob fileb://ExampleEncryptedFile ^
    --key-id 1234abcd-12ab-34cd-56ef-1234567890ab ^
    --output text ^
    --query Plaintext > ExamplePlaintextFile.base64
```

Esegui il comando `certutil`.

```
certutil -decode ExamplePlaintextFile.base64 ExamplePlaintextFile
```

Output:

```
Input Length = 18
Output Length = 12
CertUtil: -decode command completed successfully.
```

Per ulteriori informazioni, consulta Decrypt nel riferimento API del Servizio AWS di gestione delle chiavi.

Esempio 3: come decrittografare un messaggio crittografato con una chiave KMS asimmetrica (Linux e macOS)

L'esempio `decrypt` seguente mostra come decrittografare i dati crittografati con una chiave KMS asimmetrica RSA.

Quando si utilizza una chiave KMS asimmetrica, il parametro `encryption-algorithm` obbligatorio specifica l'algoritmo utilizzato per crittografare il testo non crittografato.

Prima di eseguire questo comando, sostituisci l'ID della chiave di esempio con un ID chiave valido del tuo AWS account.

```
aws kms decrypt \
    --ciphertext-blob fileb://ExampleEncryptedFile \
    --key-id 0987dcba-09fe-87dc-65ba-ab0987654321 \
    --encryption-algorithm RSAES_OAEP_SHA_256 \
    --output text \
    --query Plaintext | base64 \
    --decode > ExamplePlaintextFile
```

Questo comando non produce alcun output. L'output del comando `decrypt` viene decrittografato in formato Base64 e salvato in un file.

Per ulteriori informazioni, consulta Chiavi asimmetriche in AWS KMS nella Guida per gli sviluppatori del servizio di gestione delle AWS chiavi.

- Per informazioni dettagliate sull'API, consulta Decrypt in AWS CLI Command Reference.

`delete-alias`

Il seguente esempio di codice mostra come utilizzare. `delete-alias`

AWS CLI

Per eliminare un AWS alias KMS

L'esempio `delete-alias` seguente elimina l'alias `alias/example-alias`. Il nome di alias deve iniziare con alias/.

```
aws kms delete-alias \
    --alias-name alias/example-alias
```

Questo comando non produce alcun output. Per cercare l'alias, utilizza il comando `list-aliases`.

Per ulteriori informazioni, consulta Eliminazione di un alias nella Guida per gli sviluppatori del Servizio AWS di gestione delle chiavi.

- Per i dettagli sull'API, consulta AWS CLI Command DeleteAliasReference.

`delete-custom-key-store`

Il seguente esempio di codice mostra come utilizzare`delete-custom-key-store`.

AWS CLI

Come eliminare un archivio chiavi personalizzato

L'esempio `delete-custom-key-store` seguente elimina l'archivio chiavi personalizzato specificato.

L'eliminazione di un key store AWS CloudHSM non ha alcun effetto sul cluster CloudHSM associato. L'eliminazione di un archivio chiavi esterno non ha alcun effetto sul relativo proxy, sul gestore chiavi esterno o sulle chiavi esterne associati.

NOTA: prima di poter eliminare un archivio chiavi personalizzato, è necessario pianificare l'eliminazione di tutte le chiavi KMS nell'archivio chiavi personalizzato e quindi attendere che tali chiavi KMS vengano eliminate. Quindi, è necessario disconnettere l'archivio chiavi personalizzato. Per assistenza nella ricerca delle chiavi KMS nell'archivio chiavi personalizzato, consulta Eliminare un AWS archivio di chiavi (API) di CloudHSM nella Key Management Service Developer AWS Guide.

```
delete-custom-key-store \
    --custom-key-store-id cks-1234567890abcdef0
```

Questo comando non restituisce alcun output. Per verificare che l'archivio chiavi personalizzato sia stato eliminato, utilizza il comando `describe-custom-key-stores`.

Per informazioni sull'eliminazione di un archivio di chiavi AWS CloudHSM, consulta Eliminazione di un archivio di chiavi AWS CloudHSM nella Key Management Service Developer Guide.AWS

Per informazioni sull'eliminazione di archivi chiavi esterni, consulta Eliminazione di un archivio chiavi esterno nella Guida per gli sviluppatori del Servizio AWS di gestione delle chiavi.

- Per i dettagli sull'API, consulta Command Reference. DeleteCustomKeyStoreAWS CLI

`delete-imported-key-material`

Il seguente esempio di codice mostra come utilizzare`delete-imported-key-material`.

AWS CLI

Come eliminare il materiale della chiave importato da una chiave KMS

L'esempio `delete-imported-key-material` seguente elimina il materiale della chiave importato in una chiave KMS.

```
aws kms delete-imported-key-material \
   --key-id 1234abcd-12ab-34cd-56ef-1234567890ab
```

Output:

```
{
    "KeyId": "1234abcd-12ab-34cd-56ef-1234567890ab",
    "KeyMaterialId":
 "0b7fd7ddbac6eef27907413567cad8c810e2883dc8a7534067a82ee1142fc1e6"
```

```
}
```

Per ulteriori informazioni, consulta Eliminazione del materiale della chiave importato nella Guida per gli sviluppatori del Servizio AWS di gestione delle chiavi.

- Per i dettagli sull'API, consulta DeleteImportedKeyMaterial AWS CLICommand Reference.

`derive-shared-secret`

Il seguente esempio di codice mostra come utilizzare`derive-shared-secret`.

AWS CLI

Come derivare un segreto condiviso

L'esempio `derive-shared-secret` seguente deriva un segreto condiviso utilizzando un algoritmo di chiave concordata.

È necessario utilizzare una coppia di chiavi KMS asimmetrica a curva ellittica (ECC) o (solo per le SM2 regioni della Cina) consigliata dal NIST con un valore pari a to call. `KeyUsage` `KEY_AGREEMENT` DeriveSharedSecret

```
aws kms derive-shared-secret \
    --key-id 1234abcd-12ab-34cd-56ef-1234567890ab \
    --key-agreement-algorithm ECDH \
    --public-
key "MIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAvH3Yj0wbkLEpUl95Cv1cJVjsVNSjwGq3tCLnzXfhVwV
```

Output:

```
{
    "KeyId": "1234abcd-12ab-34cd-56ef-1234567890ab",
    "SharedSecret": "MEYCIQCKZLWyTk5runarx6XiAkU9gv3lbwPO/pHa
+DXFehzdDwIhANwpsIV2g/9SPWLLsF6p/hiSskuIXMTRwqrMdVKWTMHG",
    "KeyAgreementAlgorithm": "ECDH",
    "KeyOrigin": "AWS_KMS"
}
```

Per ulteriori informazioni, consulta la sezione Key Management Service API Reference. DeriveSharedSecretAWS

- Per i dettagli sull'API, consulta DeriveSharedSecret AWS CLICommand Reference.

`describe-custom-key-stores`

Il seguente esempio di codice mostra come utilizzare`describe-custom-key-stores`.

AWS CLI

Esempio 1: Per ottenere dettagli su un key store AWS CloudHSM

L'`describe-custom-key-store`esempio seguente visualizza i dettagli sull'archivio di chiavi AWS CloudHSM specificato. Il comando è lo stesso per tutti i tipi di archivi chiavi personalizzati, ma l'output differisce a seconda del tipo di archivio chiavi e, per un archivio chiavi esterno, della relativa opzione di connettività.

Per impostazione predefinita, questo comando visualizza informazioni su tutti gli archivi chiavi personalizzati nell'account e nella Regione. Per visualizzare informazioni su un determinato archivio chiavi personalizzato, utilizza il parametro `custom-key-store-name` o `custom-key-store-id`.

```
aws kms describe-custom-key-stores \
    --custom-key-store-name ExampleCloudHSMKeyStore
```

L'output di questo comando include dettagli utili sull'archivio di chiavi AWS CloudHSM, incluso lo stato della connessione (). `ConnectionState` Se lo stato della connessione è `FAILED`, l'output include un campo `ConnectionErrorCode` che descrive il problema.

Output:

```
{
    "CustomKeyStores": [
        {
            "CloudHsmClusterId": "cluster-1a23b4cdefg",
            "ConnectionState": "CONNECTED",
            "CreationDate": "2022-04-05T14:04:55-07:00",
            "CustomKeyStoreId": "cks-1234567890abcdef0",
            "CustomKeyStoreName": "ExampleExternalKeyStore",
            "TrustAnchorCertificate": "<certificate appears here>"
        }
    ]
```

```
}
```

Per ulteriori informazioni, consulta Viewing an AWS CloudHSM key store nella Key Management Service AWS Developer Guide.

Esempio 2: come ottenere dettagli su un archivio chiavi esterno con connettività dell'endpoint pubblico

L'esempio `describe-custom-key-store` seguente visualizza i dettagli sull'archivio chiavi esterno specificato. Il comando è lo stesso per tutti i tipi di archivi chiavi personalizzati, ma l'output differisce a seconda del tipo di archivio chiavi e, per un archivio chiavi esterno, della relativa opzione di connettività.

Per impostazione predefinita, questo comando visualizza informazioni su tutti gli archivi chiavi personalizzati nell'account e nella Regione. Per visualizzare informazioni su un determinato archivio chiavi personalizzato, utilizza il parametro `custom-key-store-name` o `custom-key-store-id`.

```
aws kms describe-custom-key-stores \
    --custom-key-store-id cks-9876543210fedcba9
```

L'output di questo comando include dettagli utili sull'archivio chiavi esterno, incluso lo stato della connessione (`ConnectionState`). Se lo stato della connessione è `FAILED`, l'output include un campo `ConnectionErrorCode` che descrive il problema.

Output:

```
{
    "CustomKeyStores": [
        {
            "CustomKeyStoreId": "cks-9876543210fedcba9",
            "CustomKeyStoreName": "ExampleXKS",
            "ConnectionState": "CONNECTED",
            "CreationDate": "2022-12-02T07:48:55-07:00",
            "CustomKeyStoreType": "EXTERNAL_KEY_STORE",
            "XksProxyConfiguration": {
                "AccessKeyId": "ABCDE12345670EXAMPLE",
                "Connectivity": "PUBLIC_ENDPOINT",
                "UriEndpoint": "https://myproxy.xks.example.com",
                "UriPath": "/example-prefix/kms/xks/v1"
            }
```

```
        }
    ]
}
```

Per ulteriori informazioni, consulta Visualizzazione di un archivio chiavi esterno nella Guida per gli sviluppatori del Servizio AWS di gestione delle chiavi.

Esempio 3: come ottenere dettagli su un archivio chiavi esterno con connettività di servizio dell'endpoint VPC

L'esempio `describe-custom-key-store` seguente visualizza i dettagli sull'archivio chiavi esterno specificato. Il comando è lo stesso per tutti i tipi di archivi chiavi personalizzati, ma l'output differisce a seconda del tipo di archivio chiavi e, per un archivio chiavi esterno, della relativa opzione di connettività.

Per impostazione predefinita, questo comando visualizza informazioni su tutti gli archivi chiavi personalizzati nell'account e nella Regione. Per visualizzare informazioni su un determinato archivio chiavi personalizzato, utilizza il parametro `custom-key-store-name` o `custom-key-store-id`.

```
aws kms describe-custom-key-stores \
    --custom-key-store-id cks-2234567890abcdef0
```

L'output di questo comando include dettagli utili sull'archivio chiavi esterno, incluso lo stato della connessione (`ConnectionState`). Se lo stato della connessione è `FAILED`, l'output include un campo `ConnectionErrorCode` che descrive il problema.

Output:

```
{
    "CustomKeyStores": [
        {
            "CustomKeyStoreId": "cks-3234567890abcdef0",
            "CustomKeyStoreName": "ExampleVPCExternalKeyStore",
            "ConnectionState": "CONNECTED",
            "CreationDate": "2022-12-22T07:48:55-07:00",
            "CustomKeyStoreType": "EXTERNAL_KEY_STORE",
            "XksProxyConfiguration": {
                "AccessKeyId": "ABCDE12345670EXAMPLE",
                "Connectivity": "VPC_ENDPOINT_SERVICE",
                "UriEndpoint": "https://myproxy-private.xks.example.com",
```

```
                "UriPath": "/kms/xks/v1",
                "VpcEndpointServiceName": "com.amazonaws.vpce.us-east-1.vpce-svc-
example1"
            }
        }
    ]
}
```

Per ulteriori informazioni, consulta Visualizzazione di un archivio chiavi esterno nella Guida per gli sviluppatori del Servizio AWS di gestione delle chiavi.

- Per i dettagli sull'API, consulta AWS CLI Command DescribeCustomKeyStoresReference.

`describe-key`

Il seguente esempio di codice mostra come utilizzare`describe-key`.

AWS CLI

Esempio 1: come trovare informazioni dettagliate su una chiave KMS

L'`describe-key`esempio seguente ottiene informazioni dettagliate sulla chiave AWS gestita per Amazon S3 nell'account e nella regione di esempio. Puoi utilizzare questo comando per trovare dettagli sulle chiavi gestite e sulle chiavi AWS gestite dai clienti.

Per specificare la chiave KMS, utilizza il parametro `key-id`. Questo esempio utilizza un valore per il nome di alias, ma in questo comando puoi utilizzare un ID di chiave, un ARN di chiave o un ARN di alias.

```
aws kms describe-key \
    --key-id alias/aws/s3
```

Output:

```
{
    "KeyMetadata": {
        "AWSAccountId": "846764612917",
        "KeyId": "b8a9477d-836c-491f-857e-07937918959b",
        "Arn": "arn:aws:kms:us-west-2:846764612917:key/
b8a9477d-836c-491f-857e-07937918959b",
        "CurrentKeyMaterialId":
 "0b7fd7ddbac6eef27907413567cad8c810e2883dc8a7534067a82ee1142fc1e6",
```

```
        "CreationDate": 2017-06-30T21:44:32.140000+00:00,
        "Enabled": true,
        "Description": "Default KMS key that protects my S3 objects when no other
 key is defined",
        "KeyUsage": "ENCRYPT_DECRYPT",
        "KeyState": "Enabled",
        "Origin": "AWS_KMS",
        "KeyManager": "AWS",
        "CustomerMasterKeySpec": "SYMMETRIC_DEFAULT",
        "EncryptionAlgorithms": [
            "SYMMETRIC_DEFAULT"
        ]
    }
}
```

Per ulteriori informazioni, consulta Visualizzazione di chiavi nella Guida per gli sviluppatori del Servizio AWS di gestione delle chiavi.

Esempio 2: come ottenere dettagli su una chiave KMS RSA asimmetrica

L'esempio `describe-key` seguente ottiene informazioni dettagliate su una chiave KMS RSA asimmetrica utilizzata per la firma e la verifica.

```
aws kms describe-key \
    --key-id 1234abcd-12ab-34cd-56ef-1234567890ab
```

Output:

```
{
    "KeyMetadata": {
        "AWSAccountId": "111122223333",
        "KeyId": "1234abcd-12ab-34cd-56ef-1234567890ab",
        "Arn": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
        "CreationDate": "2019-12-02T19:47:14.861000+00:00",
        "CustomerMasterKeySpec": "RSA_2048",
        "Enabled": false,
        "Description": "",
        "KeyState": "Disabled",
        "Origin": "AWS_KMS",
        "MultiRegion": false,
        "KeyManager": "CUSTOMER",
```

```
        "KeySpec": "RSA_2048",
        "KeyUsage": "SIGN_VERIFY",
        "SigningAlgorithms": [
            "RSASSA_PKCS1_V1_5_SHA_256",
            "RSASSA_PKCS1_V1_5_SHA_384",
            "RSASSA_PKCS1_V1_5_SHA_512",
            "RSASSA_PSS_SHA_256",
            "RSASSA_PSS_SHA_384",
            "RSASSA_PSS_SHA_512"
        ]
    }
}
```

Esempio 3: come ottenere dettagli su una chiave di replica multi-Regione

L'esempio `describe-key` seguente ottiene i metadati per una chiave di replica multi-Regione. Questa chiave multi-Regione è una chiave di crittografia simmetrica. L'output di un comando `describe-key` per qualsiasi chiave multi-Regione restituisce informazioni sulla chiave primaria e su tutte le relative repliche.

```
aws kms describe-key \
    --key-id arn:aws:kms:ap-northeast-1:111122223333:key/
mrk-1234abcd12ab34cd56ef1234567890ab
```

Output:

```
{
    "KeyMetadata": {
        "MultiRegion": true,
        "AWSAccountId": "111122223333",
        "Arn": "arn:aws:kms:ap-northeast-1:111122223333:key/
mrk-1234abcd12ab34cd56ef1234567890ab",
        "CreationDate": "2021-06-28T21:09:16.114000+00:00",
        "CurrentKeyMaterialId":
 "0b7fd7ddbac6eef27907413567cad8c810e2883dc8a7534067a82ee1142fc1e6",
        "Description": "",
        "Enabled": true,
        "KeyId": "mrk-1234abcd12ab34cd56ef1234567890ab",
        "KeyManager": "CUSTOMER",
        "KeyState": "Enabled",
        "KeyUsage": "ENCRYPT_DECRYPT",
        "Origin": "AWS_KMS",
```

```
        "CustomerMasterKeySpec": "SYMMETRIC_DEFAULT",
        "EncryptionAlgorithms": [
            "SYMMETRIC_DEFAULT"
        ],
        "MultiRegionConfiguration": {
            "MultiRegionKeyType": "PRIMARY",
            "PrimaryKey": {
                "Arn": "arn:aws:kms:us-west-2:111122223333:key/
mrk-1234abcd12ab34cd56ef1234567890ab",
                "Region": "us-west-2"
            },
            "ReplicaKeys": [
                {
                    "Arn": "arn:aws:kms:eu-west-1:111122223333:key/
mrk-1234abcd12ab34cd56ef1234567890ab",
                    "Region": "eu-west-1"
                },
                {
                    "Arn": "arn:aws:kms:ap-northeast-1:111122223333:key/
mrk-1234abcd12ab34cd56ef1234567890ab",
                    "Region": "ap-northeast-1"
                },
                {
                    "Arn": "arn:aws:kms:sa-east-1:111122223333:key/
mrk-1234abcd12ab34cd56ef1234567890ab",
                    "Region": "sa-east-1"
                }
            ]
        }
    }
}
```

Esempio 4: come ottenere dettagli su una chiave KMS HMAC

L'esempio `describe-key` seguente ottiene informazioni dettagliate su una chiave KMS HMAC.

```
aws kms describe-key \
    --key-id 1234abcd-12ab-34cd-56ef-1234567890ab
```

Output:

```
{
    "KeyMetadata": {
```

```
        "AWSAccountId": "123456789012",
        "KeyId": "1234abcd-12ab-34cd-56ef-1234567890ab",
        "Arn": "arn:aws:kms:us-
west-2:123456789012:key/1234abcd-12ab-34cd-56ef-1234567890ab",
        "CreationDate": "2022-04-03T22:23:10.194000+00:00",
        "Enabled": true,
        "Description": "Test key",
        "KeyUsage": "GENERATE_VERIFY_MAC",
        "KeyState": "Enabled",
        "Origin": "AWS_KMS",
        "KeyManager": "CUSTOMER",
        "CustomerMasterKeySpec": "HMAC_256",
        "MacAlgorithms": [
            "HMAC_SHA_256"
        ],
        "MultiRegion": false
    }
}
```

- Per i dettagli sull'API, consulta DescribeKey AWS CLICommand Reference.

`disable-key-rotation`

Il seguente esempio di codice mostra come utilizzare`disable-key-rotation`.

AWS CLI

Come disabilitare la rotazione automatica di una chiave KMS

L'esempio `disable-key-rotation` seguente disabilita la rotazione automatica di una chiave KMS gestita dal cliente. Per riabilitare la rotazione automatica, utilizza il comando `enable-key-rotation`.

```
aws kms disable-key-rotation \
    --key-id arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab
```

Questo comando non produce alcun output. Per verificare che la rotazione automatica sia disabilitata per la chiave KMS, utilizza il comando `get-key-rotation-status`.

Per ulteriori informazioni, consulta Rotazione delle chiavi nella Guida per gli sviluppatori del Servizio AWS di gestione delle chiavi.

- Per i dettagli sull'API, consulta DisableKeyRotation AWS CLICommand Reference.

`disable-key`

Il seguente esempio di codice mostra come utilizzare`disable-key`.

AWS CLI

Come disabilitare temporaneamente una chiave KMS

Il comando `disable-key` seguente disabilita una chiave KMS gestita dal cliente. Per riabilitare la chiave KMS. utilizza il comando `enable-key`.

```
aws kms disable-key \
    --key-id 1234abcd-12ab-34cd-56ef-1234567890ab
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Abilitazione e disabilitazione delle chiavi nella Guida per gli sviluppatori del Servizio AWS di gestione delle chiavi.

- Per i dettagli sull'API, consulta DisableKey AWS CLICommand Reference.

`disconnect-custom-key-store`

Il seguente esempio di codice mostra come utilizzare`disconnect-custom-key-store`.

AWS CLI

Come disconnettere un archivio chiavi personalizzato

L'`disconnect-custom-key-store`esempio seguente disconnette un key store personalizzato dal relativo cluster AWS CloudHSM. È possibile disconnettere un archivio chiavi per risolvere un problema, aggiornarne le impostazioni o impedire l'utilizzo delle chiavi KMS nell'archivio chiavi nelle operazioni crittografiche.

Questo comando è lo stesso per tutti gli archivi di chiavi personalizzati, inclusi gli archivi di chiavi AWS CloudHSM e gli archivi di chiavi esterni.

Prima di eseguire questo comando, sostituisci l'ID store chiavi personalizzate di esempio con uno valido.

```
$ aws kms disconnect-custom-key-store \
    --custom-key-store-id cks-1234567890abcdef0
```

Questo comando non produce alcun output. Verifica che il comando sia stato eseguito correttamente utilizzando il comando `describe-custom-key-stores`.

Per ulteriori informazioni sulla disconnessione di un key store AWS CloudHSM, consulta Connecting and disconnecting an AWS CloudHSM key store nella Key Management Service Developer Guide.AWS

Per ulteriori informazioni sulla disconnessione di un archivio chiavi esterno, consulta Connessione e disconnessione di un archivio delle chiavi esterne nella Guida per gli sviluppatori del Servizio AWS di gestione delle chiavi.

- Per i dettagli sull'API, consulta Command Reference. DisconnectCustomKeyStoreAWS CLI

`enable-key-rotation`

Il seguente esempio di codice mostra come utilizzare`enable-key-rotation`.

AWS CLI

Come abilitare la rotazione automatica di una chiave KMS

L'esempio `enable-key-rotation` seguente abilita la rotazione automatica di una chiave KMS gestita dal cliente con un periodo di rotazione di 180 giorni. La rotazione della chiave KMS avviene dopo un anno (circa 365 giorni) dalla data di completamento di questo comando e successivamente ogni anno.

Il parametro `--key-id` identifica la chiave KMS. Questo esempio utilizza un valore ARN della chiave, ma è possibile utilizzare l'ID chiave o l'ARN della chiave KMS. Il parametro `--rotation-period-in-days` specifica il numero di giorni tra ogni data di rotazione. Specifica un valore compreso tra 90 e 2.560 giorni. Se non viene specificato alcun valore, il valore predefinito è 365 giorni.

```
aws kms enable-key-rotation \
    --key-id arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab \
    --rotation-period-in-days 180
```

Questo comando non produce alcun output. Per verificare se la chiave KMS è abilitata, utilizza il comando `get-key-rotation-status`.

Per ulteriori informazioni, consulta Rotazione delle chiavi nella Guida per gli sviluppatori del Servizio AWS di gestione delle chiavi.

- Per i dettagli sull'API, consulta EnableKeyRotation AWS CLICommand Reference.

`enable-key`

Il seguente esempio di codice mostra come utilizzare`enable-key`.

AWS CLI

Come abilitare una chiave KMS

L'esempio `enable-key` seguente abilita una chiave gestita dal cliente. Puoi utilizzare un comando come il seguente per abilitare una chiave KMS temporaneamente disabilitata utilizzando il comando `disable-key`. Puoi anche usarlo per abilitare una chiave KMS disabilitata perché era stata pianificata per l'eliminazione e l'eliminazione è stata annullata.

Per specificare la chiave KMS, utilizza il parametro `key-id`. Questo esempio utilizza un valore per l'ID chiave, ma in questo comando puoi utilizzare un ID o un ARN di chiave.

Prima di eseguire questo comando, sostituisci l'ID della chiave di esempio con un ID valido.

```
aws kms enable-key \
    --key-id 1234abcd-12ab-34cd-56ef-1234567890ab
```

Questo comando non produce alcun output. Per verificare se la chiave KMS è abilitata, utilizza il comando `describe-key`. Verifica i valori dei campi `KeyState` e `Enabled` nell'output di `describe-key`.

Per ulteriori informazioni, consulta Abilitazione e disabilitazione delle chiavi nella Guida per gli sviluppatori del Servizio AWS di gestione delle chiavi.

- Per i dettagli sull'API, consulta EnableKey AWS CLICommand Reference.

`encrypt`

Il seguente esempio di codice mostra come utilizzare`encrypt`.

AWS CLI

Esempio 1: come crittografare il contenuto di un file in Linux o macOS

Il `encrypt` comando seguente illustra il metodo consigliato per crittografare i dati con la AWS CLI.

```
aws kms encrypt \
    --key-id 1234abcd-12ab-34cd-56ef-1234567890ab \
    --plaintext fileb://ExamplePlaintextFile \
    --output text \
    --query CiphertextBlob | base64 \
    --decode > ExampleEncryptedFile
```

Il comando esegue diverse operazioni:

Utilizza il parametro `--plaintext` per indicare i dati da crittografare. Il valore di questo parametro deve essere codificato in base 64. Il valore del `plaintext` parametro deve essere codificato in base 64 oppure è necessario utilizzare il prefisso`fileb://`, che indica alla CLI di leggere i dati binari dal file. Se il AWS file non si trova nella directory corrente, digitare il percorso completo del file. Ad esempio: `fileb:///var/tmp/ExamplePlaintextFile` o `fileb://C:\Temp\ExamplePlaintextFile`. Per ulteriori informazioni sulla lettura dei valori dei parametri AWS CLI da un file, vedere Loading Parameters from a File nella AWS Command Line Interface User Guide e Best Practices for Local File Parameters nel blog AWS Command Line Tool. Utilizza `--query` i parametri `--output` and per controllare l'output del comando. Questi parametri estraggono i dati crittografati, chiamati ciphertext, dall'output del comando.Per ulteriori informazioni sul controllo dell'output, vedere Controllo del comando Output nella Guida per l'utente dell'interfaccia a riga di AWS comando. Utilizza l'`base64`utilità per decodificare l'output estratto in dati binari. Il testo cifrato restituito da un `encrypt` comando riuscito è testo con codifica in base64. È necessario decodificare questo testo prima di poter utilizzare la AWS CLI per decrittografarlo. Salva il testo cifrato binario in un file.La parte finale del comando > `ExampleEncryptedFile` () salva il testo cifrato binario in un file per semplificare la decrittografia. Per un comando di esempio che utilizza la AWS CLI per decrittografare i dati, consulta gli esempi di decrittografia.

Esempio 2: utilizzo della AWS CLI per crittografare i dati su Windows

Questo esempio è uguale a quello precedente, tranne per il fatto che utilizza lo strumento `certutil` al posto di `base64`. Questa procedura richiede due comandi, come illustrato nell'esempio seguente.

```
aws kms encrypt \
    --key-id 1234abcd-12ab-34cd-56ef-1234567890ab \
    --plaintext fileb://ExamplePlaintextFile \
    --output text \
    --query CiphertextBlob > C:\Temp\ExampleEncryptedFile.base64

certutil -decode C:\Temp\ExampleEncryptedFile.base64 C:\Temp\ExampleEncryptedFile
```

Esempio 3: crittografia con una chiave KMS asimmetrica

Il comando `encrypt` seguente mostra come crittografare il testo non crittografato con una chiave KMS asimmetrica. Il parametro `--encryption-algorithm` è obbligatorio. Come in tutti i comandi `encrypt` CLI, il `plaintext` parametro deve essere codificato in base 64 oppure è necessario utilizzare il prefisso`fileb://`, che indica alla CLI di leggere i dati binari dal file AWS .

```
aws kms encrypt \
    --key-id 1234abcd-12ab-34cd-56ef-1234567890ab \
    --encryption-algorithm RSAES_OAEP_SHA_256 \
    --plaintext fileb://ExamplePlaintextFile \
    --output text \
    --query CiphertextBlob | base64 \
    --decode > ExampleEncryptedFile
```

Questo comando non produce alcun output.

- Per informazioni dettagliate sull'API, consulta Encrypt in AWS CLI Command Reference.

generate-data-key-pair-without-plaintext

Il seguente esempio di codice mostra come utilizzare. `generate-data-key-pair-without-plaintext`

AWS CLI

Come generare una coppia di chiavi dati asimmetriche ECC NIST P384

L'`generate-data-key-pair-without-plaintext`esempio seguente richiede una coppia di chiavi ECC NIST P384 da utilizzare all'esterno di. AWS

Il comando restituisce una chiave pubblica in testo normale e una copia della chiave privata crittografata con la chiave KMS specificata. Non restituisce una chiave privata in testo normale. È

possibile archiviare in modo sicuro la chiave privata crittografata con i dati crittografati e chiamare AWS KMS per decrittografare la chiave privata quando è necessario utilizzarla.

Per richiedere una coppia di chiavi dati asimmetriche ECC NIST P384, utilizza il parametro `key-pair-spec` con il valore `ECC_NIST_P384`.

La chiave KMS specificata deve essere una chiave KMS di crittografia simmetrica, ovvero una chiave KMS con un valore di `KeySpec` pari a `SYMMETRIC_DEFAULT`.

NOTA: i valori nell'output di questo esempio vengono troncati ai fini della visualizzazione.

```
aws kms generate-data-key-pair-without-plaintext \
    --key-id 1234abcd-12ab-34cd-56ef-1234567890ab \
    --key-pair-spec ECC_NIST_P384
```

Output:

```
{
    "PrivateKeyCiphertextBlob": "AQIDAHi6LtupRpdKl2aJTzkK6FbhOtQkMlQJJH3PdtHvS/y
+hAFFxmiD134doUDzMGmfCEtcAAAHaTCCB2UGCSqGSIb3DQEHBqCCB1...",
    "PublicKey":
 "MIIBojANBgkqhkiG9w0BAQEFAAOCAY8AMIIBigKCAYEA3A3eGMyPrvSn7+LdlJE1oUoQV5HpEuHAVbdOyND
+NmYDH/mL1OSIEuLrcdZ5hrMH4pk83r40l...",
    "KeyId": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
    "KeyMaterialId":
 "0b7fd7ddbac6eef27907413567cad8c810e2883dc8a7534067a82ee1142fc1e6",
    "KeyPairSpec": "ECC_NIST_P384"
}
```

`PublicKey` e `PrivateKeyCiphertextBlob` vengono restituiti crittografati in formato Base64.

Per ulteriori informazioni, consulta Coppia di chiavi di dati nella Guida per gli sviluppatori del Servizio AWS di gestione delle chiavi.

- Per i dettagli sull'API, consulta Command GenerateDataKeyPairWithoutPlaintextReference AWS CLI .

`generate-data-key-pair`

Il seguente esempio di codice mostra come utilizzare`generate-data-key-pair`.

AWS CLI

Come generare una coppia di chiavi dati asimmetriche RSA a 2.048 bit

L'`generate-data-key-pair`esempio seguente richiede una coppia di chiavi dati asimmetriche RSA a 2048 bit da utilizzare all'esterno di. AWS Il comando restituisce una chiave pubblica in testo normale e una chiave privata in testo normale per l'utilizzo e l'eliminazione immediati, nonché una copia della chiave privata crittografata sotto la chiave KMS specificata. Puoi archiviare la chiave privata crittografata in modo sicuro con i dati crittografati.

Per richiedere una coppia di chiavi dati asimmetriche RSA a 2.048 bit, utilizza il parametro `key-pair-spec` con il valore `RSA_2048`.

La chiave KMS specificata deve essere una chiave KMS di crittografia simmetrica, ovvero una chiave KMS con un valore di `KeySpec` pari a `SYMMETRIC_DEFAULT`.

NOTA: i valori nell'output di questo esempio vengono troncati ai fini della visualizzazione.

```
aws kms generate-data-key-pair \
    --key-id 1234abcd-12ab-34cd-56ef-1234567890ab \
    --key-pair-spec RSA_2048
```

Output:

```
{
    "PrivateKeyCiphertextBlob": "AQIDAHi6LtupRpdKl2aJTzkK6FbhOtQkMlQJJH3PdtHvS/y
+hAFFxmiD134doUDzMGmfCEtcAAAHaTCCB2UGCSqGSIb3DQEHBqCCB1...",
    "PrivateKeyPlaintext": "MIIG/
QIBADANBgkqhkiG9w0BAQEFAASCBucwggbjAgEAAoIBgQDcDd4YzI
+u9Kfv4t2UkTWhShBXkekS4cBVt07I0P42ZgMf+YvU5IgS4ut...",
    "PublicKey":
 "MIIBojANBgkqhkiG9w0BAQEFAAOCAY8AMIIBigKCAYEA3A3eGMyPrvSn7+LdlJE1oUoQV5HpEuHAVbdOyND
+NmYDH/mL1OSIEuLrcdZ5hrMH4pk83r40l...",
    "KeyId": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
    "KeyMaterialId":
 "0b7fd7ddbac6eef27907413567cad8c810e2883dc8a7534067a82ee1142fc1e6"
    "KeyPairSpec": "RSA_2048"
}
```

`PublicKey`, `PrivateKeyPlaintext` e `PrivateKeyCiphertextBlob` vengono restituiti crittografati in formato Base64.

Per ulteriori informazioni, consulta Coppia di chiavi di dati nella Guida per gli sviluppatori del Servizio AWS di gestione delle chiavi.

- Per i dettagli sull'API, consulta Command Reference. GenerateDataKeyPairAWS CLI

`generate-data-key-without-plaintext`

Il seguente esempio di codice mostra come utilizzare`generate-data-key-without-plaintext`.

AWS CLI

Come generare una chiave di dati simmetrica a 256 bit senza una chiave di testo non crittografato

L'esempio `generate-data-key-without-plaintext` seguente richiede una copia crittografata di una chiave dati simmetrica a 256 bit da utilizzare all'esterno di AWS. Puoi chiamare AWS KMS per decrittografare la chiave dati quando sei pronto per usarla.

Per richiedere una chiave di dati a 256 bit, utilizza il parametro `key-spec` con il valore `AES_256`. Per richiedere una chiave di dati a 128 bit, utilizza il parametro `key-spec` con il valore `AES_128`. Per tutte le altre lunghezze di chiavi dati, utilizza il parametro `number-of-bytes`.

La chiave KMS specificata deve essere una chiave KMS di crittografia simmetrica, ovvero una chiave KMS con un valore di KeySpec pari a SYMMETRIC_DEFAULT.

```
aws kms generate-data-key-without-plaintext \
    --key-id "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab" \
    --key-spec AES_256
```

Output:

```
{
    "CiphertextBlob":
 "AQEDAHjRYf5WytIc0C857tFSnBaPn2F8DgfmThbJlGfR8P3WlwAAAH4wfAYJKoZIhvcNAQcGoG8wbQIBADBoBgkqhk
    "KeyId": "arn:aws:kms:us-
east-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
    "KeyMaterialId":
 "0b7fd7ddbac6eef27907413567cad8c810e2883dc8a7534067a82ee1142fc1e6"
}
```

Il valore `CiphertextBlob` (chiave dati crittografati) viene restituito crittografato in formato Base64.

Per ulteriori informazioni, consulta Chiavi di dati nella Guida per gli sviluppatori del Servizio AWS di gestione delle chiavi.

- Per i dettagli sull'API, consulta Command GenerateDataKeyWithoutPlaintextReference AWS CLI .

`generate-data-key`

Il seguente esempio di codice mostra come utilizzare`generate-data-key`.

AWS CLI

Esempio 1: come generare una chiave dati simmetrica a 256 bit

L'`generate-data-key`esempio seguente richiede una chiave dati simmetrica a 256 bit da utilizzare all'esterno di. AWS Il comando restituisce una chiave dati in testo non crittografato per l'utilizzo e l'eliminazione immediati e una copia di tale chiave dati crittografata sotto la chiave KMS specificata. Puoi archiviare la chiave di dati crittografata in modo sicuro con i dati crittografati.

Per richiedere una chiave di dati a 256 bit, utilizza il parametro `key-spec` con il valore `AES_256`. Per richiedere una chiave di dati a 128 bit, utilizza il parametro `key-spec` con il valore `AES_128`. Per tutte le altre lunghezze di chiavi dati, utilizza il parametro `number-of-bytes`.

La chiave KMS specificata deve essere una chiave KMS di crittografia simmetrica, ovvero una chiave KMS con un valore di KeySpec pari a SYMMETRIC_DEFAULT.

```
aws kms generate-data-key \
    --key-id alias/ExampleAlias \
    --key-spec AES_256
```

Output:

```
{
    "Plaintext": "VdzKNHGzUAzJeRBVY+uUmofUGGiDzyB3+i9fVkh3piw=",
    "KeyId": "arn:aws:kms:us-
east-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
    "KeyMaterialId":
 "0b7fd7ddbac6eef27907413567cad8c810e2883dc8a7534067a82ee1142fc1e6",
    "CiphertextBlob":
 "AQEDAHjRYf5WytIc0C857tFSnBaPn2F8DgfmThbJlGfR8P3WlwAAAH4wfAYJKoZIhvcNAQcGoG8wbQIBADBoBgkqhk
+YdhV8MrkBQPeac0ReRVNDt9qleAt+SHgIRF8P0H+7U="
```

```
}
```

I valori `Plaintext` (chiave dati in testo normale) `CiphertextBlob` e (chiave dati crittografata) vengono restituiti crittografati in formato Base64.

Per ulteriori informazioni, consulta Chiavi di dati nella Guida per gli sviluppatori del Servizio AWS di gestione delle chiavi. Esempio 2: come generare una chiave dati simmetrica a 512 bit

L'esempio `generate-data-key` seguente richiede una chiave dati simmetrica a 512 bit per la crittografia e la decrittografia. Il comando restituisce una chiave dati in testo non crittografato per l'utilizzo e l'eliminazione immediati e una copia di tale chiave dati crittografata sotto la chiave KMS specificata. Puoi archiviare la chiave di dati crittografata in modo sicuro con i dati crittografati.

Per richiedere una lunghezza di chiave diversa da 128 o 256 bit, utilizza il parametro `number-of-bytes`. Per richiedere una chiave dati a 512 bit, l'esempio seguente utilizza il parametro `number-of-bytes` con il valore 64 (byte).

La chiave KMS specificata deve essere una chiave KMS di crittografia simmetrica, ovvero una chiave KMS con un valore di KeySpec pari a SYMMETRIC_DEFAULT.

NOTA: i valori nell'output di questo esempio vengono troncati ai fini della visualizzazione.

```
aws kms generate-data-key \
    --key-id 1234abcd-12ab-34cd-56ef-1234567890ab \
    --number-of-bytes 64
```

Output:

```
{
    "CiphertextBlob": "AQIBAHi6LtupRpdKl2aJTzkK6FbhOtQkMlQJJH3PdtHvS/y+hAEnX/
QQNmMwDfg2korNMEc8AAACaDCCAmQGCSqGSIb3DQEHBqCCAlUwggJRAgEAMIICSgYJKoZ...",
    "Plaintext": "ty8Lr0Bk6OF07M2BWt6qbFdNB
+G00ZLtf5MSEb4al3R2UKWGOp06njAwy2n72VRm2m7z/
Pm9Wpbvttz6a4lSo9hgPvKhZ5y6RTm4OovEXiVfBveyX3DQxDzRSwbKDPk/...",
    "KeyId": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
    "KeyMaterialId":
 "0b7fd7ddbac6eef27907413567cad8c810e2883dc8a7534067a82ee1142fc1e6"
}
```

I valori `Plaintext` (chiave di dati non crittografati) e `CiphertextBlob` (chiave di dati crittografati) vengono restituiti crittografati in formato Base64.

Per ulteriori informazioni, consulta Chiavi di dati nella Guida per gli sviluppatori del Servizio AWS di gestione delle chiavi.

- Per i dettagli sull'API, consulta GenerateDataKeyCommand Reference.AWS CLI

`generate-mac`

Il seguente esempio di codice mostra come utilizzare`generate-mac`.

AWS CLI

Esempio 1: come generare un codice HMAC per un messaggio

Il comando `generate-mac` seguente genera un codice HMAC per un messaggio, una chiave KMS HMAC e un algoritmo MAC. L'algoritmo deve essere supportato dalla chiave KMS HMAC specificata.

In AWS CLI v2, il valore del `message` parametro deve essere codificato in Base64. In alternativa, puoi salvare il messaggio in un file e utilizzare il `fileb://` prefisso, che indica alla AWS CLI di leggere i dati binari dal file.

Prima di eseguire questo comando, sostituisci l'ID della chiave di esempio con un ID chiave valido del tuo AWS account. L'ID chiave deve rappresentare una chiave KMS HMAC con un utilizzo chiave di `GENERATE_VERIFY_MAC`.

```
msg=(echo 'Hello World' | base64)

aws kms generate-mac \
    --key-id 1234abcd-12ab-34cd-56ef-1234567890ab \
    --message fileb://Message \
    --mac-algorithm HMAC_SHA_384
```

Output:

```
{
    "KeyId": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
    "Mac": "<HMAC_TAG>",
```

```
    "MacAlgorithm": "HMAC_SHA_384"
}
```

Per ulteriori informazioni sull'utilizzo delle chiavi HMAC KMS in AWS KMS, consulta le chiavi HMAC in KMS nella AWSAWS Key Management Service Developer Guide.

Esempio 2: come salvare un codice HMAC in un file (Linux e macOS)

L'esempio `generate-mac` seguente genera un codice HMAC per un messaggio breve archiviato in un file locale. Il comando ottiene anche la `Mac` proprietà dalla risposta, Base64 la decodifica e la salva nel file. ExampleMac Puoi utilizzare il file MAC in un comando `verify-mac` che verifica il codice MAC.

Il comando `generate-mac` richiede un messaggio crittografato in formato Base64 e un algoritmo MAC supportato dalla chiave KMS HMAC asimmetrica. Per ottenere gli algoritmi MAC supportati dalla chiave KMS, utilizza il comando `describe-key`.

Prima di eseguire questo comando, sostituite l'ID della chiave di esempio con un ID chiave valido del vostro account. AWS L'ID chiave deve rappresentare una chiave KMS asimmetrica il cui utilizzo è definito da GENERATE_VERIFY_MAC.

```
echo 'hello world' | base64 > EncodedMessage

aws kms generate-mac \
    --key-id 1234abcd-12ab-34cd-56ef-1234567890ab \
    --message fileb://EncodedMessage \
    --mac-algorithm HMAC_SHA_384 \
    --output text \
    --query Mac | base64 --decode > ExampleMac
```

Questo comando non produce alcun output. Questo esempio estrae la proprietà `Mac` dell'output e la salva in un file.

Per ulteriori informazioni sull'utilizzo delle chiavi HMAC KMS in AWS KMS, consulta le chiavi HMAC in KMS nella AWSAWS Key Management Service Developer Guide.

- Per i dettagli sull'API, consulta Command Reference. GenerateMacAWS CLI

`generate-random`

Il seguente esempio di codice mostra come utilizzare`generate-random`.

AWS CLI

Esempio 1: come generare una stringa di byte casuali a 256 bit (Linux o macOS)

L'esempio `generate-random` seguente genera una stringa di byte casuali a 256 bit (32 byte) crittografata in formato Base64. L'esempio decodifica la stringa di byte e la salva in un file casuale.

Quando esegui questo comando, devi utilizzare il parametro `number-of-bytes` per specificare la lunghezza del valore casuale in byte.

Non viene specificata una chiave KMS quando si esegue questo comando. La stringa di byte casuali non è correlata ad alcuna chiave KMS.

Per impostazione predefinita, AWS KMS genera il numero casuale. Tuttavia, se si specifica un key store personalizzato, la stringa di byte casuale viene generata nel cluster AWS CloudHSM associato all'archivio chiavi personalizzato.

Questo esempio utilizza i seguenti parametri e valori:

Utilizza il `--number-of-bytes` parametro richiesto con un valore di `32` per richiedere una stringa da 32 byte (256 bit). Utilizza il `--output` parametro con un valore di per `text` indirizzare la AWS CLI a restituire l'output come testo, anziché JSON.Utilizza il per estrarre il valore della `Plaintext` proprietà da Response.it invia (|) l'output del comando `--query parameter` all'`base64`utilità, che decodifica l'output estratto.Utilizza l'operatore di reindirizzamento (**>**) per salvare la decodifica stringa di byte aggiunta al file.Utilizza l'operatore di reindirizzamento (**>**) `ExampleRandom` per salvare il testo cifrato binario in un file.

```
aws kms generate-random \
    --number-of-bytes 32 \
    --output text \
    --query Plaintext | base64 --decode > ExampleRandom
```

Questo comando non produce alcun output.

Per ulteriori informazioni, vedere GenerateRandomnel AWS Key Management Service API Reference.

Esempio 2: come generare un numero casuale a 256 bit (prompt dei comandi di Windows)

L'esempio seguente utilizza il comando `generate-random` per generare una stringa di byte casuali a 256 bit (32 byte) crittografata in formato Base64. L'esempio decodifica la stringa di byte

e la salva in un file casuale. Questo esempio è uguale all'esempio precedente tranne per il fatto che utilizza l'utilità `certutil` in Windows per decodificare in formato Base64 una stringa di byte casuali prima di salvarla in un file.

Genera innanzitutto una stringa di byte casuali crittografata in formato Base64 e la salva nel file temporaneo `ExampleRandom.base64`.

```
aws kms generate-random \
    --number-of-bytes 32 \
    --output text \
    --query Plaintext > ExampleRandom.base64
```

Poiché l'output del comando `generate-random` viene salvato in un file, questo esempio non produce alcun output.

Ora utilizza il comando `certutil -decode` per decrittografare la stringa di byte crittografata in formato Base64 nel file `ExampleRandom.base64`. Quindi, salva la stringa di byte decodificata nel file `ExampleRandom`.

```
certutil -decode ExampleRandom.base64 ExampleRandom
```

Output:

```
Input Length = 18
Output Length = 12
CertUtil: -decode command completed successfully.
```

Per ulteriori informazioni, consulta la GenerateRandomsezione AWS Key Management Service API Reference.

- Per i dettagli sull'API, consulta GenerateRandom AWS CLICommand Reference.

`get-key-policy`

Il seguente esempio di codice mostra come utilizzare`get-key-policy`.

AWS CLI

Come copiare una policy della chiave da una chiave KMS a un'altra chiave KMS

L'esempio `get-key-policy` seguente ottiene la policy della chiave da una chiave KMS e la salva in un file di testo. Sostituisce quindi la policy di una chiave KMS diversa utilizzando il file di testo come input della policy.

Poiché il parametro `--policy` di `put-key-policy` richiede una stringa, devi utilizzare l'opzione `--output text` per restituire l'output come stringa di testo anziché JSON.

```
aws kms get-key-policy \
    --policy-name default \
    --key-id 1234abcd-12ab-34cd-56ef-1234567890ab \
    --query Policy \
    --output text > policy.txt

aws kms put-key-policy \
    --policy-name default \
    --key-id 0987dcba-09fe-87dc-65ba-ab0987654321 \
    --policy file://policy.txt
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta la PutKeyPolicysezione AWS KMS API Reference.

- Per i dettagli sull'API, consulta GetKeyPolicy AWS CLICommand Reference.

`get-key-rotation-status`

Il seguente esempio di codice mostra come utilizzare`get-key-rotation-status`.

AWS CLI

Come recuperare lo stato di rotazione di una chiave KMS.

L'esempio `get-key-rotation-status` seguente restituisce informazioni sullo stato di rotazione della chiave KMS specificata, tra cui se la rotazione automatica è abilitata, il periodo di rotazione e la successiva data di rotazione pianificata. È possibile utilizzare questo comando sulle chiavi KMS gestite dal cliente e sulle chiavi KMS AWS gestite. Tuttavia, tutte le chiavi KMS AWS gestite vengono ruotate automaticamente ogni anno.

```
aws kms get-key-rotation-status \
    --key-id 1234abcd-12ab-34cd-56ef-1234567890ab
```

Output:

```
{
    "KeyId": "1234abcd-12ab-34cd-56ef-1234567890ab",
    "KeyRotationEnabled": true,
    "NextRotationDate": "2024-02-14T18:14:33.587000+00:00",
    "RotationPeriodInDays": 365
}
```

Per ulteriori informazioni, consulta Rotazione delle chiavi nella Guida per gli sviluppatori del Servizio AWS di gestione delle chiavi.

- Per i dettagli sull'API, consulta AWS CLI Command GetKeyRotationStatusReference.

`get-parameters-for-import`

Il seguente esempio di codice mostra come utilizzare`get-parameters-for-import`.

AWS CLI

Come ottenere gli elementi necessari per importare il materiale della chiave in una chiave KMS

L'esempio `get-parameters-for-import` seguente ottiene la chiave pubblica e il token di importazione necessari per importare il materiale della chiave in una chiave KMS. Quando utilizzi il comando `import-key-material`, assicurati di utilizzare il token di importazione e il materiale della chiave crittografati dalla chiave pubblica restituiti nello stesso comando `get-parameters-for-import`. Inoltre, l'algoritmo di wrapping specificato in questo comando deve essere utilizzato per crittografare il materiale della chiave con la chiave pubblica.

Per specificare la chiave KMS, utilizza il parametro `key-id`. Questo esempio utilizza un ID chiave, ma in questo comando è possibile utilizzare un ID o un ARN di chiave.

```
aws kms get-parameters-for-import \
    --key-id 1234abcd-12ab-34cd-56ef-1234567890ab \
    --wrapping-algorithm RSAES_OAEP_SHA_256 \
    --wrapping-key-spec RSA_2048
```

Output:

```
{
```

```
    "KeyId": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
    "PublicKey": "<public key base64 encoded data>",
    "ImportToken": "<import token base64 encoded data>",
    "ParametersValidTo": 1593893322.32
}
```

Per ulteriori informazioni, consulta Scaricare la chiave pubblica e il token di importazione nella Guida per gli sviluppatori del Servizio AWS di gestione delle chiavi.

- Per i dettagli sull'API, consulta GetParametersForImport AWS CLICommand Reference.

`get-public-key`

Il seguente esempio di codice mostra come utilizzare`get-public-key`.

AWS CLI

Esempio 1: come scaricare la chiave pubblica di una chiave KMS asimmetrica

L'esempio `get-public-key` seguente scarica la chiave pubblica di una chiave KMS asimmetrica.

Oltre a restituire la chiave pubblica, l'output include le informazioni necessarie per utilizzare la chiave pubblica in modo sicuro al di fuori di AWS KMS, incluso l'utilizzo della chiave e gli algoritmi di crittografia supportati.

```
aws kms get-public-key \
    --key-id 1234abcd-12ab-34cd-56ef-1234567890ab
```

Output:

```
{
    "KeyId": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
    "PublicKey": "jANBgkqhkiG9w0BAQEFAAOCAg8AMIICCgKCAgEAl5epvg1/
QtJhxSi2g9SDEVg8QV/...",
    "CustomerMasterKeySpec": "RSA_4096",
    "KeyUsage": "ENCRYPT_DECRYPT",
    "EncryptionAlgorithms": [
        "RSAES_OAEP_SHA_1",
```

```
        "RSAES_OAEP_SHA_256"
    ]
}
```

Per ulteriori informazioni sull'utilizzo delle chiavi KMS asimmetriche in KMS, consulta la sezione Chiavi asimmetriche in AWS KMS nella Key Management Service Developer Guide. AWS AWS Esempio 2: come convertire una chiave pubblica in formato DER (Linux e macOS)

L'esempio `get-public-key` seguente scarica la chiave pubblica di una chiave KMS asimmetrica e la salva in un file DER.

Quando si utilizza il `get-public-key` comando nella AWS CLI, restituisce una chiave pubblica X.509 con codifica DER e codifica Base64. Questo esempio ottiene il valore della proprietà `PublicKey` come testo. Applica la crittografia in formato Base64 della proprietà `PublicKey` e la salva nel file `public_key.der`. Il parametro `output` restituisce l'output come testo, anziché in formato JSON. Il `--query` parametro ottiene solo la `PublicKey` proprietà, non le proprietà necessarie per utilizzare la chiave pubblica in modo sicuro all'esterno di KMS. AWS

Prima di eseguire questo comando, sostituisci l'ID della chiave di esempio con un ID chiave valido del tuo AWS account.

```
aws kms get-public-key \
    --key-id 1234abcd-12ab-34cd-56ef-1234567890ab \
    --output text \
    --query PublicKey | base64 --decode > public_key.der
```

Questo comando non produce alcun output.

Per ulteriori informazioni sull'utilizzo delle chiavi KMS asimmetriche in KMS, consulta Chiavi asimmetriche in AWS KMS nella AWS Key Management Service Developer Guide.AWS

- Per i dettagli sull'API, consulta Command Reference. GetPublicKeyAWS CLI

`import-key-material`

Il seguente esempio di codice mostra come utilizzare`import-key-material`.

AWS CLI

Come importare il materiale della chiave in una chiave KMS

L'esempio `import-key-material` seguente carica il materiale della chiave in una chiave KMS creata senza il materiale della chiave. Lo stato della chiave KMS deve essere `PendingImport`.

Questo comando utilizza il materiale della chiave crittografato con la chiave pubblica restituita dal comando `get-parameters-for-import`. Utilizza anche il token di importazione dello stesso comando `get-parameters-for-import`.

Il parametro `expiration-model` indica che il materiale della chiave scade automaticamente alla data e all'ora specificate dal parametro `valid-to`. Quando il materiale chiave scade, AWS KMS elimina il materiale chiave, lo stato della chiave KMS cambia `Pending import` e la chiave KMS diventa inutilizzabile. Per ripristinare la chiave KMS, è necessario importare nuovamente lo stesso materiale della chiave. Per utilizzare un materiale della chiave diverso, è necessario creare una nuova chiave KMS.

Prima di eseguire questo comando, sostituisci l'ID chiave di esempio con un ID chiave o un ARN di chiave valido dal tuo AWS account.

```
aws kms import-key-material \
    --key-id 1234abcd-12ab-34cd-56ef-1234567890ab \
    --encrypted-key-material fileb://EncryptedKeyMaterial.bin \
    --import-token fileb://ImportToken.bin \
    --expiration-model KEY_MATERIAL_EXPIRES \
    --valid-to 2021-09-21T19:00:00Z
```

Output:

```
{
    "KeyId": "1234abcd-12ab-34cd-56ef-1234567890ab",
    "KeyMaterialId":
 "0b7fd7ddbac6eef27907413567cad8c810e2883dc8a7534067a82ee1142fc1e6"
}
```

Per ulteriori informazioni sull'importazione del materiale delle chiavi, consulta Importazione del materiale delle chiavi nella Guida per gli sviluppatori del Servizio AWS di gestione delle chiavi.

- Per i dettagli sull'API, consulta ImportKeyMaterial AWS CLICommand Reference.

`list-aliases`

Il seguente esempio di codice mostra come utilizzare`list-aliases`.

AWS CLI

Esempio 1: per elencare tutti gli alias in un AWS account e in una regione

L'esempio seguente utilizza il `list-aliases` comando per elencare tutti gli alias nella regione predefinita dell' AWS account. L'output include alias associati alle chiavi KMS AWS gestite e alle chiavi KMS gestite dal cliente.

```
aws kms list-aliases
```

Output:

```
{
    "Aliases": [
        {
            "AliasArn": "arn:aws:kms:us-west-2:111122223333:alias/testKey",
            "AliasName": "alias/testKey",
            "TargetKeyId": "1234abcd-12ab-34cd-56ef-1234567890ab"
        },
        {
            "AliasArn": "arn:aws:kms:us-west-2:111122223333:alias/FinanceDept",
            "AliasName": "alias/FinanceDept",
            "TargetKeyId": "0987dcba-09fe-87dc-65ba-ab0987654321"
        },
        {
            "AliasArn": "arn:aws:kms:us-west-2:111122223333:alias/aws/dynamodb",
            "AliasName": "alias/aws/dynamodb",
            "TargetKeyId": "1a2b3c4d-5e6f-1a2b-3c4d-5e6f1a2b3c4d"
        },
        {
            "AliasArn": "arn:aws:kms:us-west-2:111122223333:alias/aws/ebs",
            "AliasName": "alias/aws/ebs",
            "TargetKeyId": "0987ab65-43cd-21ef-09ab-87654321cdef"
        },
        ...
    ]
}
```

Esempio 2: come elencare tutti gli alias per una particolare chiave KMS

L'esempio seguente utilizza il comando `list-aliases` e il relativo parametro `key-id` per elencare tutti gli alias associati a una particolare chiave KMS.

Ogni alias è associato a una sola chiave KMS, ma una chiave KMS può avere più alias. Questo comando è molto utile perché la console AWS KMS elenca solo un alias per ogni chiave KMS. Per trovare tutti gli alias di una chiave KMS, devi utilizzare il comando `list-aliases`.

Questo esempio utilizza l'ID della chiave KMS per il parametro `--key-id`, ma in questo comando puoi utilizzare un ID chiave, un ARN di chiave, un nome di alias o un ARN di alias.

```
aws kms list-aliases --key-id 1234abcd-12ab-34cd-56ef-1234567890ab
```

Output:

```
{
    "Aliases": [
        {
            "TargetKeyId": "1234abcd-12ab-34cd-56ef-1234567890ab",
            "AliasArn": "arn:aws:kms:us-west-2:111122223333:alias/oregon-test-key",
            "AliasName": "alias/oregon-test-key"
        },
        {
            "TargetKeyId": "1234abcd-12ab-34cd-56ef-1234567890ab",
            "AliasArn": "arn:aws:kms:us-west-2:111122223333:alias/project121-test",
            "AliasName": "alias/project121-test"
        }
    ]
}
```

Per ulteriori informazioni, consulta Utilizzo degli alias nella Guida per gli sviluppatori del Servizio AWS di gestione delle chiavi.

- Per i dettagli sull'API, consulta Command ListAliasesReference AWS CLI .

`list-grants`

Il seguente esempio di codice mostra come utilizzare`list-grants`.

AWS CLI

Per visualizzare le sovvenzioni su una chiave AWS KMS

L'`list-grants`esempio seguente mostra tutte le concessioni sulla chiave KMS AWS gestita specificata per Amazon DynamoDB nel tuo account. Questa concessione consente a DynamoDB di utilizzare la chiave KMS per tuo conto per crittografare una tabella DynamoDB prima di scriverla

su disco. Puoi utilizzare un comando come questo per visualizzare le concessioni relative alle chiavi KMS gestite e alle chiavi KMS AWS gestite dal cliente nell'account e nella regione. AWS

Questo comando utilizza il parametro `key-id` con un ID chiave per identificare la chiave KMS. Puoi utilizzare un ID o un ARN di chiave per identificare la chiave KMS. Per ottenere l'ID o l'ARN della chiave di una chiave KMS AWS gestita, usa il `list-keys` comando or. `list-aliases`

```
aws kms list-grants \
    --key-id 1234abcd-12ab-34cd-56ef-1234567890ab
```

L'output mostra che la concessione assegna l'autorizzazione Amazon DynamoDB per utilizzare la chiave KMS per le operazioni di crittografia e visualizzare i dettagli sulla chiave KMS (`DescribeKey`) e di ritirare le concessioni (`RetireGrant`). Il vincolo `EncryptionContextSubset` limita queste autorizzazioni alle richieste che includono le coppie di contesti di crittografia specificate. Di conseguenza, le autorizzazioni incluse nella concessione sono valide solo per l'account e la tabella DynamoDB specificati.

```
{
    "Grants": [
        {
            "Constraints": {
                "EncryptionContextSubset": {
                    "aws:dynamodb:subscriberId": "123456789012",
                    "aws:dynamodb:tableName": "Services"
                }
            },
            "IssuingAccount": "arn:aws:iam::123456789012:root",
            "Name": "8276b9a6-6cf0-46f1-b2f0-7993a7f8c89a",
            "Operations": [
                "Decrypt",
                "Encrypt",
                "GenerateDataKey",
                "ReEncryptFrom",
                "ReEncryptTo",
                "RetireGrant",
                "DescribeKey"
            ],
            "GrantId":
 "1667b97d27cf748cf05b487217dd4179526c949d14fb3903858e25193253fe59",
            "KeyId": "arn:aws:kms:us-
west-2:123456789012:key/1234abcd-12ab-34cd-56ef-1234567890ab",
            "RetiringPrincipal": "dynamodb.us-west-2.amazonaws.com",
```

```
            "GranteePrincipal": "dynamodb.us-west-2.amazonaws.com",
            "CreationDate": "2021-05-13T18:32:45.144000+00:00"
        }
    ]
}
```

Per ulteriori informazioni, consulta Grants in AWS KMS nella AWS Key Management Service Developer Guide.

- Per i dettagli sull'API, consulta AWS CLI Command ListGrantsReference.

`list-key-policies`

Il seguente esempio di codice mostra come utilizzare`list-key-policies`.

AWS CLI

Come ottenere i nomi delle policy della chiave per una chiave KMS

L'esempio `list-key-policies` seguente ottiene i nomi delle policy della chiave per una chiave gestita dal cliente nell'account e nella Regione di esempio. È possibile utilizzare questo comando per trovare i nomi delle politiche chiave per le chiavi AWS gestite e le chiavi gestite dai clienti.

Poiché l'unico nome di policy della chiave valido è `default`, questo comando non è utile.

Per specificare la chiave KMS, utilizza il parametro `key-id`. Questo esempio utilizza un valore di ID chiave, ma in questo comando puoi utilizzare un ID o un ARN di chiave.

```
aws kms list-key-policies \
    --key-id 1234abcd-12ab-34cd-56ef-1234567890ab
```

Output:

```
{
    "PolicyNames": [
    "default"
    ]
}
```

Per ulteriori informazioni sulle politiche chiave AWS KMS, consulta Using Key Policies in AWS KMS nella AWS Key Management Service Developer Guide.

- Per i dettagli sulle API, consulta AWS CLI Command ListKeyPoliciesReference.

`list-key-rotations`

Il seguente esempio di codice mostra come utilizzare`list-key-rotations`.

AWS CLI

Come recuperare informazioni su tutte le rotazioni dei materiali della chiave completate

L'esempio `list-key-rotations` seguente elenca le informazioni su tutte le rotazioni del materiale della chiave completate per la chiave KMS specificata.

```
aws kms list-key-rotations \
    --key-id 1234abcd-12ab-34cd-56ef-1234567890ab
```

Output:

```
{
    "Rotations": [
        {
            "KeyId": "1234abcd-12ab-34cd-56ef-1234567890ab",
            "RotationDate": "2024-03-02T10:11:36.564000+00:00",
            "RotationType": "AUTOMATIC"
        },
        {
            "KeyId": "1234abcd-12ab-34cd-56ef-1234567890ab",
            "RotationDate": "2024-04-05T15:14:47.757000+00:00",
            "RotationType": "ON_DEMAND"
        }
    ],
    "Truncated": false
}
```

Per ulteriori informazioni, consulta Rotazione delle chiavi nella Guida per gli sviluppatori del Servizio AWS di gestione delle chiavi.

- Per i dettagli sull'API, consulta ListKeyRotations AWS CLICommand Reference.

`list-keys`

Il seguente esempio di codice mostra come utilizzare`list-keys`.

AWS CLI

Come ottenere le chiavi KMS in un account e in una Regione

L'esempio `list-keys` seguente ottiene le chiavi KMS in un account e in una Regione. Questo comando restituisce sia le chiavi AWS gestite che le chiavi gestite dal cliente.

```
aws kms list-keys
```

Output:

```
{
    "Keys": [
        {
            "KeyArn": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
            "KeyId": "1234abcd-12ab-34cd-56ef-1234567890ab"
        },
        {
            "KeyArn": "arn:aws:kms:us-
west-2:111122223333:key/0987dcba-09fe-87dc-65ba-ab0987654321",
            "KeyId": "0987dcba-09fe-87dc-65ba-ab0987654321"
        },
        {
            "KeyArn": "arn:aws:kms:us-
east-2:111122223333:key/1a2b3c4d-5e6f-1a2b-3c4d-5e6f1a2b3c4d",
            "KeyId": "1a2b3c4d-5e6f-1a2b-3c4d-5e6f1a2b3c4d"
        }
    ]
}
```

Per ulteriori informazioni, consulta Visualizzazione di chiavi nella Guida per gli sviluppatori del Servizio AWS di gestione delle chiavi.

- Per i dettagli sull'API, consulta ListKeys AWS CLICommand Reference.

`list-resource-tags`

Il seguente esempio di codice mostra come utilizzare`list-resource-tags`.

AWS CLI

Come ottenere i tag su una chiave KMS

L'esempio `list-resource-tags` seguente ottiene i tag per una chiave KMS. Per aggiungere o sostituire tag di risorsa sulle chiavi KMS, utilizza il comando `tag-resource`. L'output mostra che questa chiave KMS ha due tag di risorsa, ognuno dei quali ha una chiave e un valore.

Per specificare la chiave KMS, utilizza il parametro `key-id`. Questo esempio utilizza un valore di ID chiave, ma in questo comando puoi utilizzare un ID o un ARN di chiave.

```
aws kms list-resource-tags \
    --key-id 1234abcd-12ab-34cd-56ef-1234567890ab
```

Output:

```
{
    "Tags": [
    {
        "TagKey": "Dept",
        "TagValue": "IT"
    },
    {
        "TagKey": "Purpose",
        "TagValue": "Test"
    }
    ],
    "Truncated": false
}
```

Per ulteriori informazioni sull'utilizzo dei tag in AWS KMS, consulta Tagging keys nella AWS Key Management Service Developer Guide.

- Per i dettagli sull'API, consulta AWS CLI Command ListResourceTagsReference.

`list-retirable-grants`

Il seguente esempio di codice mostra come utilizzare`list-retirable-grants`.

AWS CLI

Come visualizzare le concessioni che un principale può ritirare

L'`list-retirable-grants`esempio seguente mostra tutte le sovvenzioni che l'`ExampleAdmin`utente può ritirare con le chiavi KMS in un AWS account e in una regione. Puoi

usare un comando come questo per visualizzare le concessioni che qualsiasi titolare dell'account può ritirare sulle chiavi KMS nell'account e nella regione. AWS

Il valore del parametro `retiring-principal` obbligatorio deve essere il nome della risorsa Amazon (ARN) di un account, utente o ruolo.

Non è possibile specificare un servizio per il valore di `retiring-principal` in questo comando, anche se un servizio può essere il principale che esegue il ritiro. Per trovare le concessioni in cui un particolare servizio è il principale che esegue il ritiro, utilizza il comando `list-grants`.

L'output mostra che l'utente `ExampleAdmin` dispone dell'autorizzazione a ritirare le concessioni su due diverse chiavi KMS nell'account e nella Regione. Oltre al principale che esegue il ritiro, l'account è autorizzato a ritirare qualsiasi concessione nell'account.

```
aws kms list-retirable-grants \
    --retiring-principal arn:aws:iam::111122223333:user/ExampleAdmin
```

Output:

```
{
    "Grants": [
        {
            "KeyId": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
            "GrantId":
 "156b69c63cb154aa21f59929ff19760717be8d9d82b99df53e18b94a15a5e88e",
            "Name": "",
            "CreationDate": 2021-01-14T20:17:36.419000+00:00,
            "GranteePrincipal": "arn:aws:iam::111122223333:user/ExampleUser",
            "RetiringPrincipal": "arn:aws:iam::111122223333:user/ExampleAdmin",
            "IssuingAccount": "arn:aws:iam::111122223333:root",
            "Operations": [
                "Encrypt"
            ],
            "Constraints": {
                "EncryptionContextSubset": {
                    "Department": "IT"
                }
            }
        },
```

```
        {
            "KeyId": "arn:aws:kms:us-
west-2:111122223333:key/0987dcba-09fe-87dc-65ba-ab0987654321",
            "GrantId":
 "8c94d1f12f5e69f440bae30eaec9570bb1fb7358824f9ddfa1aa5a0dab1a59b2",
            "Name": "",
            "CreationDate": "2021-02-02T19:49:49.638000+00:00",
            "GranteePrincipal": "arn:aws:iam::111122223333:role/ExampleRole",
            "RetiringPrincipal": "arn:aws:iam::111122223333:user/ExampleAdmin",
            "IssuingAccount": "arn:aws:iam::111122223333:root",
            "Operations": [
                "Decrypt"
            ],
            "Constraints": {
                "EncryptionContextSubset": {
                    "Department": "IT"
                }
            }
        }
    ],
    "Truncated": false
}
```

Per ulteriori informazioni, consulta Grants in AWS KMS nella AWS Key Management Service Developer Guide.

- Per i dettagli sull'API, consulta AWS CLI Command ListRetirableGrantsReference.

`put-key-policy`

Il seguente esempio di codice mostra come utilizzare`put-key-policy`.

AWS CLI

Come modificare la policy della chiave per una chiave KMS

L'esempio `put-key-policy` seguente modifica la policy della chiave per una chiave gestita dal cliente.

Per iniziare, crea una policy della chiave e salvala in un file JSON locale. In questo esempio, il file è `key_policy.json`. Puoi anche specificare la policy della chiave come valore di stringa del parametro `policy`.

La prima dichiarazione di questa politica chiave fornisce all' AWS account l'autorizzazione a utilizzare le politiche IAM per controllare l'accesso alla chiave KMS. La seconda istruzione concede all'utente `test-user` l'autorizzazione a eseguire i comandi `describe-key` e `list-keys` sulla chiave KMS.

Contenuto di `key_policy.json`:

```
{
    "Version":"2012-10-17",
    "Id" : "key-default-1",
    "Statement" : [
        {
            "Sid" : "Enable IAM User Permissions",
            "Effect" : "Allow",
            "Principal" : {
                "AWS" : "arn:aws:iam::111122223333:root"
            },
            "Action" : "kms:*",
            "Resource" : "*"
        },
        {
            "Sid" : "Allow Use of Key",
            "Effect" : "Allow",
            "Principal" : {
                "AWS" : "arn:aws:iam::111122223333:user/test-user"
            },
            "Action" : [
                "kms:DescribeKey",
                "kms:ListKeys"
            ],
            "Resource" : "*"
        }
    ]
}
```

Per identificare la chiave KMS, questo esempio utilizza l'ID chiave, ma puoi utilizzare anche un ARN di chiave. Per specificare la policy della chiave, il comando utilizza il parametro `policy`. Per indicare che la policy è contenuta in un file, utilizza il prefisso obbligatorio `file://`. Questo prefisso è necessario per identificare i file in tutti i sistemi operativi supportati. Infine, il comando utilizza il parametro `policy-name` con il valore `default`. Se nessun nome di policy è specificato, il valore predefinito è `default`. L'unico valore valido è `default`.

```
aws kms put-key-policy \
    --policy-name default \
    --key-id 1234abcd-12ab-34cd-56ef-1234567890ab \
    --policy file://key_policy.json
```

Il comando non produce output. Per verificare l'efficacia del comando, utilizza il comando `get-key-policy`. Il comando di esempio seguente ottiene la policy della chiave per la stessa chiave KMS. Il parametro `output` con il valore `text` restituisce un formato di testo facile da leggere.

```
aws kms get-key-policy \
    --policy-name default \
    --key-id 1234abcd-12ab-34cd-56ef-1234567890ab \
    --output text
```

Output:

```
{
    "Version":"2012-10-17",
    "Id" : "key-default-1",
    "Statement" : [
        {
            "Sid" : "Enable IAM User Permissions",
            "Effect" : "Allow",
            "Principal" : {
                "AWS" : "arn:aws:iam::111122223333:root"
            },
            "Action" : "kms:*",
            "Resource" : "*"
            },
            {
            "Sid" : "Allow Use of Key",
            "Effect" : "Allow",
            "Principal" : {
                "AWS" : "arn:aws:iam::111122223333:user/test-user"
            },
            "Action" : [ "kms:Describe", "kms:List" ],
            "Resource" : "*"
        }
    ]
}
```

Per ulteriori informazioni, consulta Modifica di una policy delle chiavi nella Guida per gli sviluppatori del Servizio AWS di gestione delle chiavi.

- Per i dettagli sull'API, consulta PutKeyPolicy AWS CLICommand Reference.

`re-encrypt`

Il seguente esempio di codice mostra come utilizzare`re-encrypt`.

AWS CLI

Esempio 1: come crittografare di nuovo un messaggio crittografato con una chiave KMS simmetrica diversa (Linux e macOS).

L'esempio di `re-encrypt` comando seguente mostra il metodo consigliato per ricrittografare i dati con la CLI. AWS

Fornisci il testo criptato in un file. Nel valore del parametro `--ciphertext-blob` utilizza il prefisso `fileb://` per indicare alla CLI di leggere i dati da un file binario. Se il file non è presente nella directory corrente, digita il percorso completo del file. Per ulteriori informazioni sulla lettura dei valori dei parametri AWS CLI da un file, consulta Caricamento dei parametri AWS CLI da un file nella Guida per l'utente dell'interfaccia a riga di AWS comando e Best Practices for Local File Parameters nel Command Line Tool Blog .Specificare la AWS chiave KMS di origine, che decrittografa il CipherText. Il parametro non è richiesto quando si decrittografa con chiavi KMS di crittografia simmetrica. `--source-key-id` AWS KMS può ottenere la chiave KMS utilizzata per crittografare i dati dai metadati nel blob di testo cifrato. Tuttavia è sempre consigliabile specificare la chiave KMS che stai utilizzando. Questa procedura garantisce di utilizzare la chiave KMS desiderata e impedisce di decrittografare inavvertitamente un testo criptato utilizzando una chiave KMS non attendibile. Specifica la chiave KMS di destinazione, che ripete la crittografia dei dati. Il parametro `--destination-key-id` è sempre obbligatorio. Questo esempio utilizza un ARN di chiave, ma puoi utilizzare qualsiasi identificatore di chiave valido. Richiedi l'output in testo normale come valore di testo. Il parametro `--query` indica alla CLI di recuperare solo il valore del campo `Plaintext` dall'output. Il parametro `--output` restituisce l'output come testo. Procedi a decrittografare in formato Base64 il testo non crittografato e a salvarlo in un file. Nell'esempio seguente il valore del parametro `Plaintext` viene inviato (|) all'utilità Base64, che lo decodifica. Quindi, reindirizza (>) l'output decodificato al file `ExamplePlaintext`.

Prima di eseguire questo comando, sostituisci la chiave di esempio IDs con identificatori di chiave validi del tuo account. AWS

```
aws kms re-encrypt \
    --ciphertext-blob fileb://ExampleEncryptedFile \
    --source-key-id 1234abcd-12ab-34cd-56ef-1234567890ab \
    --destination-key-id 0987dcba-09fe-87dc-65ba-ab0987654321 \
    --query CiphertextBlob \
    --output text | base64 --decode > ExampleReEncryptedFile
```

Questo comando non produce alcun output. L'output del comando `re-encrypt` viene decrittografato in formato Base64 e salvato in un file.

Per ulteriori informazioni, consulta ReEncryptla sezione AWS Key Management Service API Reference.

Esempio 2: come crittografare di nuovo un messaggio crittografato con una chiave KMS simmetrica diversa (prompt dei comandi di Windows).

L'esempio `re-encrypt` seguente è lo stesso del precedente, tranne per il fatto che utilizza l'utilità `certutil` per decrittografare in formato Base64 i dati del testo non crittografato. Questa procedura richiede due comandi, come illustrato negli esempi seguenti.

Prima di eseguire questo comando, sostituisci l'ID della chiave di esempio con un ID chiave valido del tuo AWS account.

```
aws kms re-encrypt ^
    --ciphertext-blob fileb://ExampleEncryptedFile ^
    --source-key-id 1234abcd-12ab-34cd-56ef-1234567890ab ^
    --destination-key-id 0987dcba-09fe-87dc-65ba-ab0987654321 ^
    --query CiphertextBlob ^
    --output text > ExampleReEncryptedFile.base64
```

Utilizza quindi l'utilità `certutil`.

```
certutil -decode ExamplePlaintextFile.base64 ExamplePlaintextFile
```

Output:

```
Input Length = 18
Output Length = 12
CertUtil: -decode command completed successfully.
```

Per ulteriori informazioni, consulta la ReEncryptsezione AWS Key Management Service API Reference.

- Per i dettagli sull'API, consulta ReEncrypt AWS CLICommand Reference.

`retire-grant`

Il seguente esempio di codice mostra come utilizzare`retire-grant`.

AWS CLI

Come ritirare una concessione associata a una chiave principale del cliente

L'esempio `retire-grant` seguente elimina una concessione da una chiave KMS.

Il comando di esempio seguente specifica i parametri `grant-id` e `key-id`. Il valore del parametro `key-id` deve essere l'ARN della chiave KMS.

```
aws kms retire-grant \
    --grant-id 1234a2345b8a4e350500d432bccf8ecd6506710e1391880c4f7f7140160c9af3 \
    --key-id arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab
```

Questo comando non produce alcun output. Per confermare il ritiro della concessione, utilizza il comando `list-grants`.

Per ulteriori informazioni, consulta Ritirare e revocare le concessioni nella Guida per gli sviluppatori del Servizio AWS di gestione delle chiavi.

- Per i dettagli sull'API, consulta RetireGrant AWS CLICommand Reference.

`revoke-grant`

Il seguente esempio di codice mostra come utilizzare`revoke-grant`.

AWS CLI

Come revocare una concessione associata a una chiave principale del cliente

L'esempio `revoke-grant` seguente elimina una concessione da una chiave KMS. Il comando di esempio seguente specifica i parametri `grant-id` e `key-id`. Il valore del parametro `key-id` può essere l'ID o l'ARN della chiave KMS.

```
aws kms revoke-grant \
    --grant-id 1234a2345b8a4e350500d432bccf8ecd6506710e1391880c4f7f7140160c9af3 \
    --key-id 1234abcd-12ab-34cd-56ef-1234567890ab
```

Questo comando non produce alcun output. Per confermare la revoca della concessione, utilizza il comando `list-grants`.

Per ulteriori informazioni, consulta Ritirare e revocare le concessioni nella Guida per gli sviluppatori del Servizio AWS di gestione delle chiavi.

- Per i dettagli sull'API, consulta RevokeGrant AWS CLICommand Reference.

`rotate-key-on-demand`

Il seguente esempio di codice mostra come utilizzare`rotate-key-on-demand`.

AWS CLI

Come eseguire la rotazione on demand di una chiave KMS

L'esempio `rotate-key-on-demand` seguente avvia immediatamente la rotazione del materiale della chiave per la chiave KMS specificata.

```
aws kms rotate-key-on-demand \
    --key-id 1234abcd-12ab-34cd-56ef-1234567890ab
```

Output:

```
{
    "KeyId": "1234abcd-12ab-34cd-56ef-1234567890ab"
}
```

Per ulteriori informazioni, consulta Come eseguire la rotazione delle chiavi on-demand nella Guida per gli sviluppatori del Servizio AWS di gestione delle chiavi.

- Per i dettagli sull'API, consulta RotateKeyOnDemand AWS CLICommand Reference.

`schedule-key-deletion`

Il seguente esempio di codice mostra come utilizzare`schedule-key-deletion`.

AWS CLI

Come pianificare l'eliminazione di una chiave KMS gestita dal cliente.

L'esempio `schedule-key-deletion` seguente pianifica l'eliminazione della chiave KMS gestita dal cliente specificata tra 15 giorni.

Il parametro `--key-id` identifica la chiave KMS. Questo esempio utilizza il valore di un ARN di chiave, ma puoi utilizzare l'ID o l'ARN della chiave KMS. Il parametro `--pending-window-in-days` specifica la durata del periodo di attesa (da 7 a 30 giorni). Per impostazione predefinita, il periodo di attesa è pari a 30 giorni. Questo esempio specifica il valore 15, che indica di AWS eliminare definitivamente la chiave KMS 15 giorni dopo il completamento del comando.

```
aws kms schedule-key-deletion \
    --key-id arn:aws:kms:us-
west-2:123456789012:key/1234abcd-12ab-34cd-56ef-1234567890ab \
    --pending-window-in-days 15
```

La risposta include l'ARN della chiave, lo stato della chiave, il periodo di attesa (`PendingWindowInDays`) e la data di eliminazione in formato Unix. Per visualizzare la data di eliminazione nell'ora locale, usa la AWS console KMS. Le chiavi KMS con stato `PendingDeletion` non possono essere utilizzate nelle operazioni di crittografia.

```
{
    "KeyId": "arn:aws:kms:us-
west-2:123456789012:key/1234abcd-12ab-34cd-56ef-1234567890ab",
    "DeletionDate": "2022-06-18T23:43:51.272000+00:00",
    "KeyState": "PendingDeletion",
    "PendingWindowInDays": 15
}
```

Per ulteriori informazioni, consulta Eliminazione delle chiavi nella Guida per gli sviluppatori del Servizio AWS di gestione delle chiavi.

- Per i dettagli sull'API, consulta ScheduleKeyDeletion AWS CLICommand Reference.

`sign`

Il seguente esempio di codice mostra come utilizzare`sign`.

AWS CLI

Esempio 1: come generare una firma digitale per un messaggio

L'esempio `sign` seguente genera una firma crittografica per un messaggio breve. L'output del comando include un campo `Signature` crittografato in formato Base64 che è possibile verificare utilizzando il comando `verify`.

È necessario specificare un messaggio da firmare e un algoritmo di firma supportato dalla chiave KMS asimmetrica. Per ottenere gli algoritmi di firma per la chiave KMS, utilizza il comando `describe-key`.

In AWS CLI v2, il valore del `message` parametro deve essere codificato in Base64. In alternativa, puoi salvare il messaggio in un file e utilizzare il `fileb://` prefisso, che indica alla AWS CLI di leggere i dati binari dal file.

Prima di eseguire questo comando, sostituisci l'ID della chiave di esempio con un ID chiave valido del tuo AWS account. L'ID chiave deve rappresentare una chiave KMS asimmetrica il cui utilizzo è definito da SIGN_VERIFY.

```
msg=(echo 'Hello World' | base64)

aws kms sign \
    --key-id 1234abcd-12ab-34cd-56ef-1234567890ab \
    --message fileb://UnsignedMessage \
    --message-type RAW \
    --signing-algorithm RSASSA_PKCS1_V1_5_SHA_256
```

Output:

```
{
    "KeyId": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
    "Signature": "ABCDEFhpyVYyTxbafE74ccSvEJLJr3zuoV1Hfymz4qv+/
fxmxNLA7SE1SiF8lHw80fKZZ3bJ...",
    "SigningAlgorithm": "RSASSA_PKCS1_V1_5_SHA_256"
}
```

Per ulteriori informazioni sull'utilizzo delle chiavi KMS asimmetriche in KMS, consulta Chiavi asimmetriche in AWS KMS nella AWS Key Management Service Developer Guide.AWS

Esempio 2: come salvare una firma digitale in un file (Linux e macOS)

L'esempio `sign` seguente genera una firma crittografica per un messaggio breve archiviato in un file locale. Il comando ottiene anche la `Signature` proprietà dalla risposta, Base64-la decodifica e la salva nel file. ExampleSignature Puoi utilizzare il file di firma in un comando `verify` che verifica la firma.

Il comando `sign` richiede un messaggio crittografato in formato Base64 e un algoritmo di firma supportato dalla chiave KMS asimmetrica. Per ottenere gli algoritmi di firma supportati dalla chiave KMS, utilizza il comando `describe-key`.

Prima di eseguire questo comando, sostituite l'ID della chiave di esempio con un ID chiave valido del vostro account. AWS L'ID chiave deve rappresentare una chiave KMS asimmetrica il cui utilizzo è definito da SIGN_VERIFY.

```
echo 'hello world' | base64 > EncodedMessage

aws kms sign \
    --key-id 1234abcd-12ab-34cd-56ef-1234567890ab \
    --message fileb://EncodedMessage \
    --message-type RAW \
    --signing-algorithm RSASSA_PKCS1_V1_5_SHA_256 \
    --output text \
    --query Signature | base64 --decode > ExampleSignature
```

Questo comando non produce alcun output. Questo esempio estrae la proprietà `Signature` dell'output e la salva in un file.

Per ulteriori informazioni sull'utilizzo delle chiavi KMS asimmetriche in KMS, consulta Chiavi asimmetriche in AWS KMS nella AWS Key Management Service Developer Guide.AWS

- Per informazioni dettagliate sull'API, consulta Sign nella documentazione di riferimento dei comandi della AWS CLI .

`tag-resource`

Il seguente esempio di codice mostra come utilizzare. `tag-resource`

AWS CLI

Come aggiungere un tag a una chiave KMS

L'esempio `tag-resource` seguente aggiunge i tag `"Purpose":"Test"` e `"Dept":"IT"` a una chiave KMS gestita dal cliente. Puoi utilizzare tag simili a quelli riportati di seguito per etichettare le chiavi KMS e creare categorie di chiavi KMS per le autorizzazioni e il controllo.

Per specificare la chiave KMS, utilizza il parametro `key-id`. Questo esempio utilizza un valore di ID chiave, ma in questo comando puoi utilizzare un ID o un ARN di chiave.

```
aws kms tag-resource \
    --key-id 1234abcd-12ab-34cd-56ef-1234567890ab \
    --tags TagKey='Purpose',TagValue='Test' TagKey='Dept',TagValue='IT'
```

Questo comando non produce alcun output. Per visualizzare i tag su una chiave AWS KMS KMS, usa il `list-resource-tags` comando.

Per ulteriori informazioni sull'utilizzo dei tag in AWS KMS, consulta Tagging keys nella AWS Key Management Service Developer Guide.

- Per i dettagli sull'API, consulta AWS CLI Command TagResourceReference.

`untag-resource`

Il seguente esempio di codice mostra come utilizzare`untag-resource`.

AWS CLI

Come eliminare un tag da una chiave KMS

L'esempio `untag-resource` seguente elimina il tag con la chiave `"Purpose"` da una chiave KMS gestita dal cliente.

Per specificare la chiave KMS, utilizza il parametro `key-id`. Questo esempio utilizza un valore di ID chiave, ma in questo comando puoi utilizzare un ID o un ARN di chiave. Prima di eseguire questo comando, sostituisci l'ID della chiave di esempio con un ID chiave valido del tuo AWS account.

```
aws kms untag-resource \
    --key-id 1234abcd-12ab-34cd-56ef-1234567890ab \
    --tag-key 'Purpose'
```

Questo comando non produce alcun output. Per visualizzare i tag su una chiave AWS KMS KMS, usa il `list-resource-tags` comando.

Per ulteriori informazioni sull'utilizzo dei tag in AWS KMS, consulta Tagging keys nella AWS Key Management Service Developer Guide.

- Per i dettagli sull'API, consulta AWS CLI Command UntagResourceReference.

`update-alias`

Il seguente esempio di codice mostra come utilizzare`update-alias`.

AWS CLI

Come associare un alias a una chiave KMS diversa

L'esempio `update-alias` seguente associa l'alias `alias/test-key` a una chiave KMS diversa.

Il parametro `--alias-name` specifica l'alias. Il valore del nome dell'alias deve iniziare con `alias/`. Il parametro `--target-key-id` specifica la chiave KMS da associare all'alias. Non è necessario specificare la chiave KMS corrente per l'alias.

```
aws kms update-alias \
    --alias-name alias/test-key \
    --target-key-id 1234abcd-12ab-34cd-56ef-1234567890ab
```

Questo comando non produce alcun output. Per cercare l'alias, utilizza il comando `list-aliases`.

Per ulteriori informazioni, consulta Aggiornamento degli alias nella Guida per gli sviluppatori del Servizio AWS di gestione delle chiavi.

- Per i dettagli sull'API, consulta UpdateAlias AWS CLICommand Reference.

`update-custom-key-store`

Il seguente esempio di codice mostra come utilizzare`update-custom-key-store`.

AWS CLI

Esempio 1: come modificare il nome descrittivo di un archivio chiavi personalizzato

L'esempio `update-custom-key-store` seguente modifica il nome dell'archivio chiavi personalizzato. Questo esempio funziona per un key store AWS CloudHSM o un key store esterno.

Utilizza `custom-key-store-id` per identificare l'archivio chiavi. Utilizza il parametro `new-custom-key-store-name` per specificare il nuovo nome descrittivo.

Per aggiornare il nome descrittivo di un AWS key store CloudHSM, devi prima disconnetterlo, ad esempio usando il comando. `disconnect-custom-key-store` Puoi aggiornare il nome descrittivo di un archivio chiavi esterno mentre è connesso o disconnesso. Per individuare lo stato di connessione dell'archivio chiavi personalizzato, utilizza il comando `describe-custom-key-store`.

```
aws kms update-custom-key-store \
    --custom-key-store-id cks-1234567890abcdef0 \
    --new-custom-key-store-name ExampleKeyStore
```

Questo comando non restituisce dati. Per verificare che il comando funzioni, utilizza un comando `describe-custom-key-stores`.

Per ulteriori informazioni sull'aggiornamento di un key store AWS CloudHSM, consulta Modifica delle impostazioni del key store di AWS CloudHSM nella Key Management Service Developer Guide.AWS

Per informazioni sull'aggiornamento di un archivio chiavi esterno, consulta Modifica delle proprietà dell'archivio chiavi esterno nella Guida per gli sviluppatori del Servizio AWS di gestione delle chiavi.

Esempio 2: modificare la password kmsuser di un key store AWS CloudHSM

L'esempio `update-custom-key-store` seguente aggiorna il valore della password `kmsuser` con la password corrente per `kmsuser` nel cluster CloudHSM associato all'archivio chiavi specificato. Questo comando non modifica la password `kmsuser` del cluster. Indica semplicemente a AWS KMS la password corrente. Se KMS non dispone della `kmsuser` password corrente, non può connettersi al key store di AWS CloudHSM.

NOTA: prima di aggiornare un key store AWS CloudHSM, è necessario disconnetterlo. Utilizza il comando `disconnect-custom-key-store`. Una volta completato il comando, puoi ricollegare l'archivio di chiavi CloudHSM AWS . Utilizza il comando `connect-custom-key-store`.

```
aws kms update-custom-key-store \
    --custom-key-store-id cks-1234567890abcdef0 \
    --key-store-password ExamplePassword
```

Questo comando non restituisce alcun output. Per verificare che il comando sia stato eseguito correttamente, utilizza un comando `describe-custom-key-stores`.

Per ulteriori informazioni sull'aggiornamento di un key store AWS CloudHSM, consulta Modifica delle impostazioni del key store di AWS CloudHSM nella Key Management Service Developer Guide.AWS

Esempio 3: modificare il cluster AWS CloudHSM di un AWS key store CloudHSM

L'esempio seguente modifica il cluster AWS CloudHSM associato a un key store AWS CloudHSM in un cluster correlato, ad esempio un backup diverso dello stesso cluster.

NOTA: prima di aggiornare un key store AWS CloudHSM, è necessario disconnetterlo. Utilizza il comando `disconnect-custom-key-store`. Una volta completato il comando, puoi ricollegare l'archivio di chiavi CloudHSM AWS . Utilizza il comando `connect-custom-key-store`.

```
aws kms update-custom-key-store \
    --custom-key-store-id cks-1234567890abcdef0 \
    --cloud-hsm-cluster-id cluster-1a23b4cdefg
```

Questo comando non restituisce alcun output. Per verificare che il comando sia stato eseguito correttamente, utilizza un comando `describe-custom-key-stores`.

Per ulteriori informazioni sull'aggiornamento di un key store AWS CloudHSM, consulta Modifica delle impostazioni del key store di AWS CloudHSM nella Key Management Service Developer Guide.AWS

Esempio 4: come modificare le credenziali di autenticazione proxy di un archivio chiavi esterno

L'esempio seguente aggiorna le credenziali di autenticazione proxy per l'archivio chiavi esterno. Devi specificare `raw-secret-access-key` e `access-key-id`, anche se modifichi solo uno dei valori. È possibile utilizzare questa funzionalità per correggere credenziali non valide o per modificarle quando il proxy dell'archivio chiavi esterno le ruota.

Stabilisci le credenziali di autenticazione proxy per AWS KMS sul tuo archivio di chiavi esterno. Quindi usa questo comando per fornire la credenziale a KMS. AWS AWS KMS utilizza questa credenziale per firmare le sue richieste al proxy di archiviazione delle chiavi esterno.

Puoi aggiornare le credenziali di autenticazione proxy mentre l'archivio chiavi esterno è connesso o disconnesso. Per individuare lo stato di connessione dell'archivio chiavi personalizzato, utilizza il comando `describe-custom-key-store`.

```
aws kms update-custom-key-store \
    --custom-key-store-id cks-1234567890abcdef0 \
    --xks-proxy-authentication-credential "AccessKeyId=ABCDE12345670EXAMPLE,
 RawSecretAccessKey=DXjSUawnel2fr6SKC7G25CNxTyWKE5PF9XX6H/u9pSo="
```

Questo comando non restituisce alcun output. Per verificare che il comando sia stato eseguito correttamente, utilizza un comando `describe-custom-key-stores`.

Per informazioni sull'aggiornamento di un archivio chiavi esterno, consulta Modifica delle proprietà dell'archivio chiavi esterno nella Guida per gli sviluppatori del Servizio AWS di gestione delle chiavi.

Esempio 5: come modificare la connettività proxy di un archivio chiavi esterno

L'esempio seguente modifica l'opzione di connettività proxy dell'archivio chiavi esterno dalla connettività dell'endpoint pubblico alla connettività di servizio dell'endpoint VPC. Oltre a modificare il valore `xks-proxy-connectivity`, è necessario modificare il valore `xks-proxy-uri-endpoint` in modo che rifletta il nome DNS privato associato al servizio dell'endpoint VPC. È inoltre necessario aggiungere un valore `xks-proxy-vpc-endpoint-service-name`.

NOTA: prima di aggiornare la connettività proxy di un archivio esterno, è necessario disconnetterlo. Utilizza il comando `disconnect-custom-key-store`. Al completamento del comando, è possibile riconnettere l'archivio chiavi esterno utilizzando il comando `connect-custom-key-store`.

```
aws kms update-custom-key-store \
    --custom-key-store-id cks-1234567890abcdef0 \
    --xks-proxy-connectivity VPC_ENDPOINT_SERVICE \
    --xks-proxy-uri-endpoint "https://myproxy-private.xks.example.com" \
    --xks-proxy-vpc-endpoint-service-name "com.amazonaws.vpce.us-east-1.vpce-svc-
example"
```

Questo comando non restituisce alcun output. Per verificare che il comando sia stato eseguito correttamente, utilizza un comando `describe-custom-key-stores`.

Per informazioni sull'aggiornamento di un archivio chiavi esterno, consulta Modifica delle proprietà dell'archivio chiavi esterno nella Guida per gli sviluppatori del Servizio AWS di gestione delle chiavi.

- Per i dettagli sull'API, consulta AWS CLI Command UpdateCustomKeyStoreReference.

`update-key-description`

Il seguente esempio di codice mostra come utilizzareupdate-key-description.

AWS CLI

Esempio 1: come aggiungere o modificare una descrizione in una chiave KMS gestita dal cliente

L'esempio update-key-description seguente aggiunge una descrizione a una chiave KMS gestita dal cliente. Puoi utilizzare lo stesso comando per modificare una descrizione esistente.

Il parametro --key-id identifica la chiave KMS nel comando. Questo esempio utilizza un valore ARN di chiave, ma è possibile utilizzare l'ID chiave o l'ARN della chiave KMS. Il parametro --description specifica la nuova descrizione. Il valore di questo parametro sostituisce l'eventuale descrizione corrente della chiave KMS.

```
aws kms update-key-description \
    --key-id arn:aws:kms:us-
west-2:123456789012:key/1234abcd-12ab-34cd-56ef-1234567890ab \
    --description "IT Department test key"
```

Questo comando non produce alcun output. Per visualizzare la descrizione di una chiave KMS, utilizza il comando describe-key.

Per ulteriori informazioni, vedere UpdateKeyDescriptionnel AWS Key Management Service API Reference.

Esempio 2: come eliminare la descrizione di una chiave KMS gestita dal cliente

L'esempio update-key-description seguente elimina la descrizione da una chiave KMS gestita dal cliente.

Il parametro --key-id identifica la chiave KMS nel comando. Questo esempio utilizza un valore ID chiave, ma è possibile utilizzare l'ID chiave o l'ARN della chiave KMS. Il parametro --description con un valore di stringa vuoto (“) elimina la descrizione esistente.

```
aws kms update-key-description \
    --key-id 0987dcba-09fe-87dc-65ba-ab0987654321 \
    --description ''
```

Questo comando non produce alcun output. Per visualizzare la descrizione di una chiave KMS, utilizza il comando describe-key.

Per ulteriori informazioni, consulta la UpdateKeyDescriptionsezione AWS Key Management Service API Reference.

- Per i dettagli sull'API, consulta UpdateKeyDescription AWS CLICommand Reference.

`verify-mac`

Il seguente esempio di codice mostra come utilizzare`verify-mac`.

AWS CLI

Esempio 1: come verificare un codice HMAC

Il comando `verify-mac` seguente verifica un codice HMAC per un particolare messaggio, chiavi KMS HMAC e un algoritmo MAC. Un valore 'true' nel MacValid valore della risposta indica che l'HMAC è valido.

In AWS CLI v2, il valore del `message` parametro deve essere codificato in Base64. In alternativa, puoi salvare il messaggio in un file e utilizzare il `fileb://` prefisso, che indica alla AWS CLI di leggere i dati binari dal file.

Il codice MAC specificato non può essere crittografato in formato Base64. Per informazioni sulla decrittografia del codice MAC restituito dal comando `generate-mac`, consulta gli esempi del comando `generate-mac`.

Prima di eseguire questo comando, sostituisci l'ID della chiave di esempio con un ID chiave valido del tuo AWS account. L'ID chiave deve rappresentare una chiave KMS HMAC con un utilizzo chiave di `GENERATE_VERIFY_MAC`.

```
msg=(echo 'Hello World' | base64)

aws kms verify-mac \
    --key-id 1234abcd-12ab-34cd-56ef-1234567890ab \
```

```
    --message fileb://Message \
    --mac-algorithm HMAC_SHA_384 \
    --mac fileb://ExampleMac
```

Output:

```
{
    "KeyId": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
    "MacValid": true,
    "MacAlgorithm": "HMAC_SHA_384"
}
```

Per ulteriori informazioni sull'utilizzo delle chiavi HMAC KMS in AWS KMS, consulta le chiavi HMAC in KMS nella AWSAWS Key Management Service Developer Guide.

- Per i dettagli sull'API, consulta Command Reference. VerifyMacAWS CLI

`verify`

Il seguente esempio di codice mostra come utilizzare`verify`.

AWS CLI

Come verificare una firma digitale

Il comando `verify` seguente verifica una firma crittografica per un breve messaggio crittografato in formato Base64. L'ID chiave, il messaggio, il tipo di messaggio e l'algoritmo di firma devono essere gli stessi utilizzati per firmare il messaggio.

In AWS CLI v2, il valore del `message` parametro deve essere codificato in Base64. In alternativa, puoi salvare il messaggio in un file e utilizzare il `fileb://` prefisso, che indica alla AWS CLI di leggere i dati binari dal file.

La firma specificata non può essere crittografata in formato Base64. Per informazioni sulla decrittografia della firma restituita dal comando `sign`, consulta gli esempi del comando `sign`.

L'output del comando include un campo booleano `SignatureValid` che indica che la firma è stata verificata. Se la convalida della firma ha esito negativo, anche il comando `verify` ha esito negativo.

Prima di eseguire questo comando, sostituisci l'ID della chiave di esempio con un ID chiave valido del tuo AWS account.

```
aws kms verify \
    --key-id 1234abcd-12ab-34cd-56ef-1234567890ab \
    --message fileb://EncodedMessage \
    --message-type RAW \
    --signing-algorithm RSASSA_PKCS1_V1_5_SHA_256 \
    --signature fileb://ExampleSignature
```

Output:

```
{
    "KeyId": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
    "SignatureValid": true,
    "SigningAlgorithm": "RSASSA_PKCS1_V1_5_SHA_256"
}
```

Per ulteriori informazioni sull'utilizzo delle chiavi KMS asimmetriche in AWS KMS, consulta Using asimmetric keys nella Key Management Service Developer Guide.AWS

- Per informazioni dettagliate sull'API, consulta Verify in AWS CLI Command Reference.

Esempi di Lake Formation utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with Lake Formation.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`add-lf-tags-to-resource`

Il seguente esempio di codice mostra come utilizzare`add-lf-tags-to-resource`.

AWS CLI

Come aggiungere uno o più tag LF a una risorsa esistente

L'esempio `add-lf-tags-to-resource` seguente aggiunge un determinato tag LF alla risorsa della tabella.

```
aws lakeformation add-lf-tags-to-resource \
    --cli-input-json file://input.json
```

Contenuto di `input.json`:

```
{
    "CatalogId": "123456789111",
    "Resource": {
        "Table": {
            "CatalogId": "123456789111",
            "DatabaseName": "tpc",
            "Name": "dl_tpc_promotion"
        }
    },
    "LFTags": [{
        "CatalogId": "123456789111",
        "TagKey": "usergroup",
        "TagValues": [
            "analyst"
        ]
    }]
}
```

Output:

```
{
    "Failures": []
}
```

Per ulteriori informazioni, consulta Assegnazione di tag LF alle risorse del Catalogo dati nella Guida per gli sviluppatori di AWS Lake Formation.

- Per i dettagli sull'API, consulta AddLfTagsToResource AWS CLICommand Reference.

`batch-grant-permissions`

Il seguente esempio di codice mostra come utilizzare`batch-grant-permissions`.

AWS CLI

Come concedere in blocco le autorizzazioni sulle risorse ai principali

L'esempio `batch-grant-permissions` seguente concede in blocco ai principali l'accesso a risorse specifiche.

```
aws lakeformation batch-grant-permissions \
    --cli-input-json file://input.json
```

Contenuto di `input.json`:

```
{
    "CatalogId": "123456789111",
    "Entries": [{
            "Id": "1",
            "Principal": {
                "DataLakePrincipalIdentifier": "arn:aws:iam::123456789111:user/lf-
developer"
            },
            "Resource": {
                "Table": {
                    "CatalogId": "123456789111",
                    "DatabaseName": "tpc",
                    "Name": "dl_tpc_promotion"
                }
            },
            "Permissions": [
                "ALL"
            ],
            "PermissionsWithGrantOption": [
                "ALL"
            ]
        },
```

```
        {
            "Id": "2",
            "Principal": {
                "DataLakePrincipalIdentifier": "arn:aws:iam::123456789111:user/lf-
developer"
            },
            "Resource": {
                "Table": {
                    "CatalogId": "123456789111",
                    "DatabaseName": "tpc",
                    "Name": "dl_tpc_customer"
                }
            },
            "Permissions": [
                "ALL"
            ],
            "PermissionsWithGrantOption": [
                "ALL"
            ]
        },
        {
            "Id": "3",
            "Principal": {
                "DataLakePrincipalIdentifier": "arn:aws:iam::123456789111:user/lf-
business-analyst"
            },
            "Resource": {
                "Table": {
                    "CatalogId": "123456789111",
                    "DatabaseName": "tpc",
                    "Name": "dl_tpc_promotion"
                }
            },
            "Permissions": [
                "ALL"
            ],
            "PermissionsWithGrantOption": [
                "ALL"
            ]
        },
        {
            "Id": "4",
            "Principal": {
```

```
                "DataLakePrincipalIdentifier": "arn:aws:iam::123456789111:user/lf-
developer"
            },
            "Resource": {
                "DataCellsFilter": {
                    "TableCatalogId": "123456789111",
                    "DatabaseName": "tpc",
                    "TableName": "dl_tpc_item",
                    "Name": "developer_item"
                }
            },
            "Permissions": [
                "SELECT"
            ],
            "PermissionsWithGrantOption": []
        }
    ]
}
```

Output:

```
{
    "Failures": []
}
```

Per ulteriori informazioni sulla concessione e la revoca dell'accesso a Lake Formation, consulta la pagina Concessione e revoca delle autorizzazioni delle risorse del catalogo dati nella Guida per gli sviluppatori di AWS .

- Per i dettagli sull'API, consulta BatchGrantPermissions AWS CLICommand Reference.

`batch-revoke-permissions`

Il seguente esempio di codice mostra come utilizzare`batch-revoke-permissions`.

AWS CLI

Come revocare in blocco le autorizzazioni sulle risorse ai principali

L'esempio `batch-revoke-permissions` seguente revoca in blocco ai principali l'accesso a risorse specifiche.

```
aws lakeformation batch-revoke-permissions \
```

```
    --cli-input-json file://input.json
```

Contenuto di `input.json`:

```
{
    "CatalogId": "123456789111",
    "Entries": [{
            "Id": "1",
            "Principal": {
                "DataLakePrincipalIdentifier": "arn:aws:iam::123456789111:user/lf-
developer"
            },
            "Resource": {
                "Table": {
                    "CatalogId": "123456789111",
                    "DatabaseName": "tpc",
                    "Name": "dl_tpc_promotion"
                }
            },
            "Permissions": [
                "ALL"
            ],
            "PermissionsWithGrantOption": [
                "ALL"
            ]
        },
        {
            "Id": "2",
            "Principal": {
                "DataLakePrincipalIdentifier": "arn:aws:iam::123456789111:user/lf-
business-analyst"
            },
            "Resource": {
                "Table": {
                    "CatalogId": "123456789111",
                    "DatabaseName": "tpc",
                    "Name": "dl_tpc_promotion"
                }
            },
            "Permissions": [
                "ALL"
            ],
            "PermissionsWithGrantOption": [
```

```
                "ALL"
            ]
        }
    ]
}
```

Output:

```
{
    "Failures": []
}
```

Per ulteriori informazioni sulla concessione e la revoca dell'accesso a Lake Formation, consulta la pagina Concessione e revoca delle autorizzazioni delle risorse del catalogo dati nella Guida per gli sviluppatori di AWS .

- Per i dettagli sull'API, consulta BatchRevokePermissions AWS CLICommand Reference.

`cancel-transaction`

Il seguente esempio di codice mostra come utilizzare`cancel-transaction`.

AWS CLI

Come annullare una transazione

L'esempio `cancel-transaction` seguente annulla la transazione.

```
aws lakeformation cancel-transaction \
    --transaction-id='b014d972ca8347b89825e33c5774aec4'
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Lettura e scrittura di dati nel data lake all'interno delle transazioni nella Guida per gli sviluppatori di AWS Lake Formation.

- Per i dettagli sull'API, consulta CancelTransaction AWS CLICommand Reference.

`commit-transaction`

Il seguente esempio di codice mostra come utilizzare`commit-transaction`.

AWS CLI

Come confermare una transazione

L'esempio `commit-transaction` seguente conferma la transazione.

```
aws lakeformation commit-transaction \
    --transaction-id='b014d972ca8347b89825e33c5774aec4'
```

Output:

```
{
    "TransactionStatus": "committed"
}
```

Per ulteriori informazioni, consulta Lettura e scrittura di dati nel data lake all'interno delle transazioni nella Guida per gli sviluppatori di AWS Lake Formation.

- Per i dettagli sull'API, consulta CommitTransaction AWS CLICommand Reference.

`create-data-cells-filter`

Il seguente esempio di codice mostra come utilizzare`create-data-cells-filter`.

AWS CLI

Esempio 1: come creare un filtro per le celle di dati

L'esempio `create-data-cells-filter` seguente crea un filtro sulle celle di dati per consentire l'accesso a determinate colonne in base alle condizioni della riga.

```
aws lakeformation create-data-cells-filter \
    --cli-input-json file://input.json
```

Contenuto di `input.json`:

```
{
    "TableData": {
        "ColumnNames": ["p_channel_details", "p_start_date_sk", "p_promo_name"],
        "DatabaseName": "tpc",
```

```
        "Name": "developer_promotion",
        "RowFilter": {
            "FilterExpression": "p_promo_name='ese'"
        },
        "TableCatalogId": "123456789111",
        "TableName": "dl_tpc_promotion"
    }
}
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Filtro dei dati e sicurezza a livello di cella in Lake Formation nella Guida per gli sviluppatori di AWS Lake Formation.

Esempio 2: come creare un filtro per le colonne

L'esempio `create-data-cells-filter` seguente crea un filtro dati per consentire l'accesso a determinate colonne.

```
aws lakeformation create-data-cells-filter \
    --cli-input-json file://input.json
```

Contenuto di `input.json`:

```
{
    "TableData": {
        "ColumnNames": ["p_channel_details", "p_start_date_sk", "p_promo_name"],
        "DatabaseName": "tpc",
        "Name": "developer_promotion_allrows",
        "RowFilter": {
            "AllRowsWildcard": {}
        },
        "TableCatalogId": "123456789111",
        "TableName": "dl_tpc_promotion"
    }
}
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Filtro dei dati e sicurezza a livello di cella in Lake Formation nella Guida per gli sviluppatori di AWS Lake Formation.

Esempio 3: come creare un filtro dati con colonne di esclusione

L'esempio `create-data-cells-filter` seguente crea un filtro dati per consentire l'accesso a tutte le colonne tranne quelle menzionate.

```
aws lakeformation create-data-cells-filter \
    --cli-input-json file://input.json
```

Contenuto di `input.json`:

```
{
    "TableData": {
        "ColumnWildcard": {
            "ExcludedColumnNames": ["p_channel_details", "p_start_date_sk"]
        },
        "DatabaseName": "tpc",
        "Name": "developer_promotion_excludecolumn",
        "RowFilter": {
            "AllRowsWildcard": {}
        },
        "TableCatalogId": "123456789111",
        "TableName": "dl_tpc_promotion"
    }
}
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Filtro dei dati e sicurezza a livello di cella in Lake Formation nella Guida per gli sviluppatori di AWS Lake Formation.

- Per i dettagli sull'API, consulta CreateDataCellsFilter AWS CLICommand Reference.

`create-lf-tag`

Il seguente esempio di codice mostra come utilizzare`create-lf-tag`.

AWS CLI

Come creare un tag LF

L'esempio `create-lf-tag` seguente crea un tag LF con il nome e i valori specificati.

```
aws lakeformation create-lf-tag \
    --catalog-id '123456789111' \
    --tag-key 'usergroup' \
    --tag-values '["developer","analyst","campaign"]'
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Gestione dei tag LF per il controllo dell'accesso ai metadati nella Guida per gli sviluppatori di AWS Lake Formation.

- Per i dettagli sull'API, consulta CreateLfTag AWS CLICommand Reference.

`delete-data-cells-filter`

Il seguente esempio di codice mostra come utilizzare`delete-data-cells-filter`.

AWS CLI

Come eliminare il filtro per le celle di dati

L'esempio `delete-data-cells-filter` seguente elimina un determinato filtro per le celle di dati.

```
aws lakeformation delete-data-cells-filter \
    --cli-input-json file://input.json
```

Contenuto di `input.json`:

```
{
    "TableCatalogId": "123456789111",
    "DatabaseName": "tpc",
    "TableName": "dl_tpc_promotion",
    "Name": "developer_promotion"
}
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Filtro dei dati e sicurezza a livello di cella in Lake Formation nella Guida per gli sviluppatori di AWS Lake Formation.

- Per i dettagli sull'API, consulta DeleteDataCellsFilter AWS CLICommand Reference.

`delete-lf-tag`

Il seguente esempio di codice mostra come utilizzare`delete-lf-tag`.

AWS CLI

Come eliminare la definizione del tag LF

L'esempio `delete-lf-tag` seguente elimina la definizione del tag LF.

```
aws lakeformation delete-lf-tag \
    --catalog-id '123456789111' \
    --tag-key 'usergroup'
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Gestione dei tag LF per il controllo dell'accesso ai metadati nella Guida per gli sviluppatori di AWS Lake Formation.

- Per i dettagli sull'API, consulta DeleteLfTag AWS CLICommand Reference.

`delete-objects-on-cancel`

Il seguente esempio di codice mostra come utilizzare`delete-objects-on-cancel`.

AWS CLI

Come eliminare l'oggetto quando la transazione viene annullata

L'esempio `delete-objects-on-cancel` seguente elimina l'oggetto s3 elencato quando la transazione viene annullata.

```
aws lakeformation delete-objects-on-cancel \
    --cli-input-json file://input.json
```

Contenuto di `input.json`:

```
{
    "CatalogId": "012345678901",
    "DatabaseName": "tpc",
    "TableName": "dl_tpc_household_demographics_gov",
```

```
    "TransactionId": "1234d972ca8347b89825e33c5774aec4",
    "Objects": [{
        "Uri": "s3://lf-data-lake-012345678901/target/
dl_tpc_household_demographics_gov/run-unnamed-1-part-block-0-r-00000-snappy-
ff26b17504414fe88b302cd795eabd00.parquet",
        "ETag": "1234ab1fc50a316b149b4e1f21a73800"
    }]
}
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Lettura e scrittura di dati nel data lake all'interno delle transazioni nella Guida per gli sviluppatori di AWS Lake Formation.

- Per i dettagli sull'API, consulta DeleteObjectsOnCancel AWS CLICommand Reference.

`deregister-resource`

Il seguente esempio di codice mostra come utilizzare`deregister-resource`.

AWS CLI

Come annullare la registrazione dell'archiviazione nel data lake

L'esempio `deregister-resource` seguente annulla la registrazione della risorsa come gestita da Lake Formation.

```
aws lakeformation deregister-resource \
    --cli-input-json file://input.json
```

Contenuto di `input.json`:

```
{
    "ResourceArn": "arn:aws:s3:::lf-emr-athena-result-123"
}
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Aggiunta di una posizione Amazon S3 al data lake nella Guida per gli sviluppatori di AWS Lake Formation.

- Per i dettagli sull'API, consulta DeregisterResource AWS CLICommand Reference.

`describe-transaction`

Il seguente esempio di codice mostra come utilizzare`describe-transaction`.

AWS CLI

Come recuperare i dettagli di una transazione

L'esempio `describe-transaction` seguente restituisce i dettagli di una singola transazione.

```
aws lakeformation describe-transaction \
    --transaction-id='8cb4b1a7cc8d486fbaca9a64e7d9f5ce'
```

Output:

```
{
    "TransactionDescription": {
        "TransactionId": "12345972ca8347b89825e33c5774aec4",
        "TransactionStatus": "committed",
        "TransactionStartTime": "2022-08-10T14:29:04.046000+00:00",
        "TransactionEndTime": "2022-08-10T14:29:09.681000+00:00"
    }
}
```

Per ulteriori informazioni, consulta Lettura e scrittura di dati nel data lake all'interno delle transazioni nella Guida per gli sviluppatori di AWS Lake Formation.

- Per i dettagli sull'API, consulta DescribeTransaction AWS CLICommand Reference.

`extend-transaction`

Il seguente esempio di codice mostra come utilizzare`extend-transaction`.

AWS CLI

Come estendere una transazione

L'esempio `extend-transaction` seguente estende la transazione.

```
aws lakeformation extend-transaction \
    --transaction-id='8cb4b1a7cc8d486fbaca9a64e7d9f5ce'
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Lettura e scrittura di dati nel data lake all'interno delle transazioni nella Guida per gli sviluppatori di AWS Lake Formation.

- Per i dettagli sull'API, consulta ExtendTransaction AWS CLICommand Reference.

`get-data-lake-settings`

Il seguente esempio di codice mostra come utilizzare`get-data-lake-settings`.

AWS CLI

Per recuperare le impostazioni AWS del data lake gestite da Lake Formation

L'esempio `get-data-lake-settings` seguente recupera l'elenco degli amministratori del data lake e altre impostazioni del data lake.

```
aws lakeformation get-data-lake-settings \
    --cli-input-json file://input.json
```

Contenuto di `input.json`:

```
{
    "CatalogId": "123456789111"
}
```

Output:

```
{
    "DataLakeSettings": {
        "DataLakeAdmins": [{
            "DataLakePrincipalIdentifier": "arn:aws:iam::123456789111:user/lf-admin"
        }],
        "CreateDatabaseDefaultPermissions": [],
        "CreateTableDefaultPermissions": [
                {
                        "Principal": {
                            "DataLakePrincipalIdentifier": "IAM_ALLOWED_PRINCIPALS"
                        },
                        "Permissions": [
```

```
                            "ALL"
                        ]
                }
        ],
        "TrustedResourceOwners": [],
        "AllowExternalDataFiltering": true,
        "ExternalDataFilteringAllowList": [{
            "DataLakePrincipalIdentifier": "123456789111"
        }],
        "AuthorizedSessionTagValueList": [
            "Amazon EMR"
        ]
    }
}
```

Per ulteriori informazioni, consulta Modifica delle impostazioni di sicurezza predefinite per il data lake nella Guida per gli sviluppatori di AWS Lake Formation.

- Per i dettagli sull'API, consulta GetDataLakeSettingsCommand Reference.AWS CLI

`get-effective-permissions-for-path`

Il seguente esempio di codice mostra come utilizzare`get-effective-permissions-for-path`.

AWS CLI

Come recuperare le autorizzazioni sulle risorse che si trovano in un percorso specifico

L'esempio `get-effective-permissions-for-path` seguente restituisce le autorizzazioni di Lake Formation per una risorsa della tabella o del database specificata situata in un percorso in Amazon S3.

```
aws lakeformation get-effective-permissions-for-path \
    --cli-input-json file://input.json
```

Contenuto di `input.json`:

```
{
    "CatalogId": "123456789111",
    "ResourceArn": "arn:aws:s3:::lf-data-lake-123456789111"
}
```

Output:

```
{
    "Permissions": [{
            "Principal": {
                "DataLakePrincipalIdentifier": "arn:aws:iam::123456789111:user/lf-
campaign-manager"
            },
            "Resource": {
                "Database": {
                    "Name": "tpc"
                }
            },
            "Permissions": [
                "DESCRIBE"
            ],
            "PermissionsWithGrantOption": []
        },
        {
            "Principal": {
                "DataLakePrincipalIdentifier": "arn:aws:iam::123456789111:role/EMR-
RuntimeRole"
            },
            "Resource": {
                "Database": {
                    "Name": "tpc"
                }
            },
            "Permissions": [
                "ALL"
            ],
            "PermissionsWithGrantOption": []
        },
        {
            "Principal": {
                "DataLakePrincipalIdentifier": "arn:aws:iam::123456789111:saml-
provider/oktaSAMLProvider:user/emr-developer"
            },
            "Resource": {
                "Database": {
                    "Name": "tpc"
                }
            },
            "Permissions": [
```

```
                "ALL",
                "DESCRIBE"
            ],
            "PermissionsWithGrantOption": []
        },
        {
            "Principal": {
                "DataLakePrincipalIdentifier": "arn:aws:iam::123456789111:user/lf-
admin"
            },
            "Resource": {
                "Database": {
                    "Name": "tpc"
                }
            },
            "Permissions": [
                "ALL",
                "ALTER",
                "CREATE_TABLE",
                "DESCRIBE",
                "DROP"
            ],
            "PermissionsWithGrantOption": [
                "ALL",
                "ALTER",
                "CREATE_TABLE",
                "DESCRIBE",
                "DROP"
            ]
        },
        {
            "Principal": {
                "DataLakePrincipalIdentifier": "arn:aws:iam::123456789111:role/LF-
GlueServiceRole"
            },
            "Resource": {
                "Database": {
                    "Name": "tpc"
                }
            },
            "Permissions": [
                "CREATE_TABLE"
            ],
            "PermissionsWithGrantOption": []
```

```
        }
    ],
    "NextToken":
 "E5SlJDSTZleUp6SWpvaU9UQTNORE0zTXpFeE5Ua3pJbjE5TENKbGVIQnBjbUYwYVc5dUlqcDdJbk5sWTI5dVpITWlP
}
```

Per ulteriori informazioni, consulta Gestione delle autorizzazioni di Lake Formation nella Guida per gli sviluppatori di AWS Lake Formation.

- Per i dettagli sull'API, consulta GetEffectivePermissionsForPath AWS CLICommand Reference.

`get-lf-tag`

Il seguente esempio di codice mostra come utilizzare`get-lf-tag`.

AWS CLI

Come recuperare la definizione del tag LF

L'esempio `get-lf-tag` seguente recupera la definizione del tag LF.

```
aws lakeformation get-lf-tag \
    --catalog-id '123456789111' \
    --tag-key 'usergroup'
```

Output:

```
{
    "CatalogId": "123456789111",
    "TagKey": "usergroup",
    "TagValues": [
        "analyst",
        "campaign",
        "developer"
    ]
}
```

Per ulteriori informazioni, consulta Gestione dei tag LF per il controllo dell'accesso ai metadati nella Guida per gli sviluppatori di AWS Lake Formation.

- Per i dettagli sull'API, consulta GetLfTag AWS CLICommand Reference.

`get-query-state`

Il seguente esempio di codice mostra come utilizzare`get-query-state`.

AWS CLI

Come recuperare lo stato di una query inviata

L'esempio `get-query-state` seguente restituisce lo stato di una query inviata in precedenza.

```
aws lakeformation get-query-state \
    --query-id='1234273f-4a62-4cda-8d98-69615ee8be9b'
```

Output:

```
{
    "State": "FINISHED"
}
```

Per ulteriori informazioni, consulta Operazioni sui dati transazionali nella Guida per gli sviluppatori di AWS Lake Formation.

- Per i dettagli sull'API, consulta GetQueryState AWS CLICommand Reference.

`get-query-statistics`

Il seguente esempio di codice mostra come utilizzare`get-query-statistics`.

AWS CLI

Come recuperare le statistiche delle query

L'esempio `get-query-statistics` seguente recupera le statistiche sulla pianificazione e l'esecuzione di una query.

```
aws lakeformation get-query-statistics \
    --query-id='1234273f-4a62-4cda-8d98-69615ee8be9b'
```

Output:

```
{
```

```
    "ExecutionStatistics": {
        "AverageExecutionTimeMillis": 0,
        "DataScannedBytes": 0,
        "WorkUnitsExecutedCount": 0
    },
    "PlanningStatistics": {
        "EstimatedDataToScanBytes": 43235,
        "PlanningTimeMillis": 2377,
        "QueueTimeMillis": 440,
        "WorkUnitsGeneratedCount": 1
    },
    "QuerySubmissionTime": "2022-08-11T02:14:38.641870+00:00"
}
```

Per ulteriori informazioni, consulta Operazioni sui dati transazionali nella Guida per gli sviluppatori di AWS Lake Formation.

- Per i dettagli sull'API, consulta GetQueryStatistics AWS CLICommand Reference.

`get-resource-lf-tags`

Il seguente esempio di codice mostra come utilizzare`get-resource-lf-tags`.

AWS CLI

Come elencare i tag LF

L'esempio `list-lf-tags` seguente restituisce un elenco di tag LF che il richiedente è autorizzato a visualizzare.

```
aws lakeformation list-lf-tags \
    --cli-input-json file://input.json
```

Contenuto di `input.json`:

```
{
    "CatalogId": "123456789111",
    "ResourceShareType": "ALL",
    "MaxResults": 2
}
```

Output:

```
{
"LFTags": [{
        "CatalogId": "123456789111",
        "TagKey": "category",
        "TagValues": [
            "private",
            "public"
        ]
    },
    {
        "CatalogId": "123456789111",
        "TagKey": "group",
        "TagValues": [
            "analyst",
            "campaign",
            "developer"
        ]
    }],
    "NextToken": "kIiwiZXhwaXJhdGlvbiI6eyJzZWNvbmRzIjoxNjYwMDY4dCI6ZmFsc2V9"
}
```

Per ulteriori informazioni, consulta Gestione dei tag LF per il controllo dell'accesso ai metadati nella Guida per gli sviluppatori di AWS Lake Formation.

- Per i dettagli sull'API, consulta GetResourceLfTags AWS CLICommand Reference.

`get-table-objects`

Il seguente esempio di codice mostra come utilizzare`get-table-objects`.

AWS CLI

Come elencare gli oggetti di una tabella governata

L'esempio `get-table-objects` seguente restituisce l'insieme di oggetti Amazon S3 che costituiscono la tabella governata specificata.

```
aws lakeformation get-table-objects \
    --cli-input-json file://input.json
```

Contenuto di `input.json`:

```
{
    "CatalogId": "012345678901",
    "DatabaseName": "tpc",
    "TableName": "dl_tpc_household_demographics_gov",
    "QueryAsOfTime": "2022-08-10T15:00:00"
}
```

Output:

```
{
    "Objects": [{
        "PartitionValues": [],
        "Objects": [{
            "Uri": "s3://lf-data-lake-012345678901/target/
dl_tpc_household_demographics_gov/run-unnamed-1-part-block-0-r-00000-snappy-
ff26b17504414fe88b302cd795eabd00.parquet",
            "ETag": "12345b1fc50a316b149b4e1f21a73800",
            "Size": 43235
        }]
    }]
}
```

Per ulteriori informazioni, consulta Lettura e scrittura di dati nel data lake all'interno delle transazioni nella Guida per gli sviluppatori di AWS Lake Formation.

- Per i dettagli sull'API, consulta GetTableObjects AWS CLICommand Reference.

`get-work-unit-results`

Il seguente esempio di codice mostra come utilizzare`get-work-unit-results`.

AWS CLI

Come recuperare le unità di lavoro di una determinata query

L'esempio `get-work-unit-results` seguente restituisce le unità di lavoro risultanti dalla query.

```
aws lakeformation get-work-units \
    --query-id='1234273f-4a62-4cda-8d98-69615ee8be9b' \
    --work-unit-id '0' \
```

```
    --work-unit-token 'B2fMSdmQXe9umX8Ux8XCo4=' outfile
```

Output:

```
outfile with Blob content.
```

Per ulteriori informazioni, consulta Operazioni sui dati transazionali nella Guida per gli sviluppatori di AWS Lake Formation.

- Per i dettagli sull'API, consulta GetWorkUnitResults AWS CLICommand Reference.

`get-work-units`

Il seguente esempio di codice mostra come utilizzare`get-work-units`.

AWS CLI

Come recuperare le unità di lavoro

L'`get-work-units`esempio seguente recupera le unità di lavoro generate dall' StartQueryPlanning operazione.

```
aws lakeformation get-work-units \
    --query-id='1234273f-4a62-4cda-8d98-69615ee8be9b'
```

Output:

```
{
    "WorkUnitRanges": [{
        "WorkUnitIdMax": 0,
        "WorkUnitIdMin": 0,
        "WorkUnitToken":
 "1234eMAk4kLO4umqEL4Z5WuxL04AXwABABVhd3MtY3J5cHRvLXB1YmxpYy1rZXkAREEwYm9QbkhINmFYTWphbmMxZW
+f88jzGrYq22gE6jkQlpOB
+Oet2eqNUmFudAAAAfjB8BgkqhkiG9w0BBwagbzBtAgEAMGgGCSqGSIb3DQEHATAeBglghkgBZQMEAS4wEQQMCOEWRda
wAAAAEAAAAAAAAAAAAAAAEAAACX3/w5h75QAPomfKH+cyEKYU1yccUmBl
+VSojiGOtdsUk7vcjYXUUboYm3dvqRqX2s4gROMOn
+Ij8R0/8jYmnHkpvyAFNVRPyETyIKg7k5Z9+5I1c2d3446Jw/moWGGxjH8AEG9h27ytmOhozxDOEi/
F2ZoXz6wlGDfGUo/2WxCkYOhTyNaw6TM
+7drTM7yrW4iNVLUM0LX0xnFjIAhLhooWJek6vjQZUAZzBlAjBH8okRtYP8R7AY2Wls/
hqFBhG0V4l42AC0LxsuZbMQrE2SzWZUZ0E9Uew7/n0cyX4CMQDR79INyv4ysMByW9kKGGKyba+cCNklExMR
+btBQBmMuB2fMSdmQXe9umX8Ux8XCo4="
```

```
    }],
    "QueryId": "1234273f-4a62-4cda-8d98-69615ee8be9b"
}
```

Per ulteriori informazioni, consulta Operazioni sui dati transazionali nella Guida per gli sviluppatori di AWS Lake Formation.

- Per i dettagli sull'API, vedere GetWorkUnitsin AWS CLI Command Reference.

`grant-permissions`

Il seguente esempio di codice mostra come utilizzare`grant-permissions`.

AWS CLI

Esempio 1: come concedere al responsabile le autorizzazioni sulle risorse utilizzando i tag LF

L'esempio `grant-permissions` seguente concede al responsabile TUTTE le autorizzazioni sulla risorsa di database che corrisponde alla policy sui tag LF.

```
aws lakeformation grant-permissions \
    --cli-input-json file://input.json
```

Contenuto di `input.json`:

```
{
    "CatalogId": "123456789111",
    "Principal": {
        "DataLakePrincipalIdentifier": "arn:aws:iam::123456789111:user/lf-admin"
    },
    "Resource": {
        "LFTagPolicy": {
            "CatalogId": "123456789111",
            "ResourceType": "DATABASE",
            "Expression": [{
                "TagKey": "usergroup",
                "TagValues": [
                    "analyst",
                    "developer"
                ]
            }]
        }
    },
```

```
    "Permissions": [
        "ALL"
    ],
    "PermissionsWithGrantOption": [
        "ALL"
    ]
}
```

Questo comando non produce alcun output.

Per ulteriori informazioni sulla concessione e la revoca dell'accesso a Lake Formation, consulta la pagina Concessione e revoca delle autorizzazioni delle risorse del catalogo dati nella Guida per gli sviluppatori di AWS .

Esempio 2: come concedere al responsabile le autorizzazioni a livello di colonna

L'esempio `grant-permissions` seguente concede al responsabile l'autorizzazione a selezionare una colonna specifica.

```
aws lakeformation grant-permissions \
    --cli-input-json file://input.json
```

Contenuto di `input.json`:

```
{
    "CatalogId": "123456789111",
    "Principal": {
        "DataLakePrincipalIdentifier": "arn:aws:iam::123456789111:user/lf-developer"
    },
    "Resource": {
        "TableWithColumns": {
            "CatalogId": "123456789111",
            "ColumnNames": ["p_end_date_sk"],
            "DatabaseName": "tpc",
            "Name": "dl_tpc_promotion"
        }
    },
    "Permissions": [
        "SELECT"
    ],
    "PermissionsWithGrantOption": []
}
```

Questo comando non produce alcun output.

Per ulteriori informazioni sulla concessione e la revoca dell'accesso a Lake Formation, consulta la pagina Concessione e revoca delle autorizzazioni delle risorse del catalogo dati nella Guida per gli sviluppatori di AWS .

Esempio 3: come concedere al responsabile le autorizzazioni per le tabelle

L'esempio `grant-permissions` seguente concede al responsabile l'autorizzazione per la selezione su tutte le tabelle di un determinato database.

```
aws lakeformation grant-permissions \
    --cli-input-json file://input.json
```

Contenuto di `input.json`:

```
{
    "CatalogId": "123456789111",
    "Principal": {
        "DataLakePrincipalIdentifier": "arn:aws:iam::123456789111:user/lf-developer"
    },
    "Resource": {
        "Table": {
            "CatalogId": "123456789111",
            "DatabaseName": "tpc",
            "TableWildcard": {}
        }
    },
    "Permissions": [
        "SELECT"
    ],
    "PermissionsWithGrantOption": []
}
```

Questo comando non produce alcun output.

Per ulteriori informazioni sulla concessione e la revoca dell'accesso a Lake Formation, consulta la pagina Concessione e revoca delle autorizzazioni delle risorse del catalogo dati nella Guida per gli sviluppatori di AWS .

Esempio 4: come concedere al responsabile le autorizzazioni sui tag LF

L'esempio `grant-permissions` seguente concede al responsabile l'autorizzazione per l'associazione sui tag LF.

```
aws lakeformation grant-permissions \
    --cli-input-json file://input.json
```

Contenuto di `input.json`:

```
{
    "CatalogId": "123456789111",
    "Principal": {
        "DataLakePrincipalIdentifier": "arn:aws:iam::123456789111:user/lf-developer"
    },
    "Resource": {
        "LFTag": {
            "CatalogId": "123456789111",
            "TagKey": "category",
            "TagValues": [
                "private", "public"
            ]
        }

    },
    "Permissions": [
        "ASSOCIATE"
    ],
    "PermissionsWithGrantOption": []
}
```

Questo comando non produce alcun output.

Per ulteriori informazioni sulla concessione e la revoca dell'accesso a Lake Formation, consulta la pagina Concessione e revoca delle autorizzazioni delle risorse del catalogo dati nella Guida per gli sviluppatori di AWS .

Esempio 5: come concedere al responsabile le autorizzazioni sulle posizioni dei dati

L'esempio `grant-permissions` seguente concede al responsabile l'autorizzazione sulla posizione dei dati.

```
aws lakeformation grant-permissions \
```

```
    --cli-input-json file://input.json
```

Contenuto di `input.json`:

```
{
    "CatalogId": "123456789111",
    "Principal": {
        "DataLakePrincipalIdentifier": "arn:aws:iam::123456789111:user/lf-developer"
    },
    "Resource": {
        "DataLocation": {
            "CatalogId": "123456789111",
            "ResourceArn": "arn:aws:s3:::lf-data-lake-123456789111"
        }
    },
    "Permissions": [
        "DATA_LOCATION_ACCESS"
    ],
    "PermissionsWithGrantOption": []
}
```

Questo comando non produce alcun output.

Per ulteriori informazioni sulla concessione e la revoca dell'accesso a Lake Formation, consulta la pagina Concessione e revoca delle autorizzazioni delle risorse del catalogo dati nella Guida per gli sviluppatori di AWS .

- Per i dettagli sull'API, consulta GrantPermissions AWS CLICommand Reference.

`list-data-cells-filter`

Il seguente esempio di codice mostra come utilizzare`list-data-cells-filter`.

AWS CLI

Come elencare i filtri delle celle di dati

L'esempio `list-data-cells-filter` seguente elenca il filtro delle celle di dati per una determinata tabella.

```
aws lakeformation list-data-cells-filter \
    --cli-input-json file://input.json
```

Contenuto di `input.json`:

```
{
    "MaxResults": 2,
    "Table": {
        "CatalogId": "123456789111",
        "DatabaseName": "tpc",
        "Name": "dl_tpc_promotion"
    }
}
```

Output:

```
{
    "DataCellsFilters": [{
            "TableCatalogId": "123456789111",
            "DatabaseName": "tpc",
            "TableName": "dl_tpc_promotion",
            "Name": "developer_promotion",
            "RowFilter": {
                "FilterExpression": "p_promo_name='ese'"
            },
            "ColumnNames": [
                "p_channel_details",
                "p_start_date_sk",
                "p_purpose",
                "p_promo_id",
                "p_promo_name",
                "p_end_date_sk",
                "p_discount_active"
            ]
        },
        {
            "TableCatalogId": "123456789111",
            "DatabaseName": "tpc",
            "TableName": "dl_tpc_promotion",
            "Name": "developer_promotion_allrows",
            "RowFilter": {
                "FilterExpression": "TRUE",
                "AllRowsWildcard": {}
            },
            "ColumnNames": [
                "p_channel_details",
```

```
                "p_start_date_sk",
                "p_promo_name"
            ]
        }
    ],
    "NextToken": "2MDA2MTgwNiwibmFub3MiOjE0MDAwMDAwMH19"
}
```

Per ulteriori informazioni, consulta Filtro dei dati e sicurezza a livello di cella in Lake Formation nella Guida per gli sviluppatori di AWS Lake Formation.

- Per i dettagli sull'API, consulta ListDataCellsFilter AWS CLICommand Reference.

`list-permissions`

Il seguente esempio di codice mostra come utilizzare`list-permissions`.

AWS CLI

Esempio 1: come recuperare l'elenco delle autorizzazioni del responsabile sulla risorsa

L'esempio `list-permissions` seguente restituisce un elenco delle autorizzazioni del responsabile sulle risorse del database.

```
aws lakeformation list-permissions \
    --cli-input-json file://input.json
```

Contenuto di `input.json`:

```
{
    "CatalogId": "123456789111",
    "ResourceType": "DATABASE",
    "MaxResults": 2
}
```

Output:

```
{
    "PrincipalResourcePermissions": [{
        "Principal": {
            "DataLakePrincipalIdentifier": "arn:aws:iam::123456789111:user/lf-
campaign-manager"
```

```
        },
        "Resource": {
            "Database": {
                "CatalogId": "123456789111",
                "Name": "tpc"
            }
        },
        "Permissions": [
            "DESCRIBE"
        ],
        "PermissionsWithGrantOption": []
    }],
    "NextToken":
 "E5SlJDSTZleUp6SWpvaU9UQTNORE0zTXpFeE5Ua3pJbjE5TENKbGVIQnBjbUYwYVc5dUlqcDdJbk5sWTI5dVpITWlP
}
```

Per ulteriori informazioni, consulta Gestione delle autorizzazioni di Lake Formation nella Guida per gli sviluppatori di AWS Lake Formation.

Esempio 2: come recuperare l'elenco delle autorizzazioni del responsabile sulla tabella con filtri di dati

L'esempio `list-permissions` seguente elenca le autorizzazioni sulla tabella con i relativi filtri di dati concesse al responsabile.

```
aws lakeformation list-permissions \
    --cli-input-json file://input.json
```

Contenuto di `input.json`:

```
{
    "CatalogId": "123456789111",
    "Resource": {
        "Table": {
            "CatalogId": "123456789111",
            "DatabaseName": "tpc",
            "Name": "dl_tpc_customer"
        }
    },
    "IncludeRelated": "TRUE",
    "MaxResults": 10
}
```

Output:

```
{
    "PrincipalResourcePermissions": [{
            "Principal": {
                "DataLakePrincipalIdentifier": "arn:aws:iam::123456789111:role/
Admin"
            },
            "Resource": {
                "Table": {
                    "CatalogId": "123456789111",
                    "DatabaseName": "customer",
                    "Name": "customer_invoice"
                }
            },
            "Permissions": [
                "ALL",
                "ALTER",
                "DELETE",
                "DESCRIBE",
                "DROP",
                "INSERT"
            ],
            "PermissionsWithGrantOption": [
                "ALL",
                "ALTER",
                "DELETE",
                "DESCRIBE",
                "DROP",
                "INSERT"
            ]
        },
        {
            "Principal": {
                "DataLakePrincipalIdentifier": "arn:aws:iam::123456789111:role/
Admin"
            },
            "Resource": {
                "TableWithColumns": {
                    "CatalogId": "123456789111",
                    "DatabaseName": "customer",
                    "Name": "customer_invoice",
                    "ColumnWildcard": {}
                }
```

```
            },
            "Permissions": [
                "SELECT"
            ],
            "PermissionsWithGrantOption": [
                "SELECT"
            ]
        },
        {
            "Principal": {
                "DataLakePrincipalIdentifier": "arn:aws:iam::123456789111:role/
Admin"
            },
            "Resource": {
                "DataCellsFilter": {
                    "TableCatalogId": "123456789111",
                    "DatabaseName": "customer",
                    "TableName": "customer_invoice",
                    "Name": "dl_us_customer"
                }
            },
            "Permissions": [
                "DESCRIBE",
                "SELECT",
                "DROP"
            ],
            "PermissionsWithGrantOption": []
        }
    ],
    "NextToken": "VyeUFjY291bnRQZXJtaXNzaW9ucyI6ZmFsc2V9"
}
```

Per ulteriori informazioni, consulta Gestione delle autorizzazioni di Lake Formation nella Guida per gli sviluppatori di AWS Lake Formation.

Esempio 3: come recuperare l'elenco delle autorizzazioni del responsabile sui tag LF

L'esempio `list-permissions` seguente elenca le autorizzazioni sui tag LF concesse al responsabile.

```
aws lakeformation list-permissions \
    --cli-input-json file://input.json
```

Contenuto di `input.json`:

```
{
    "CatalogId": "123456789111",
    "Resource": {
        "LFTag": {
            "CatalogId": "123456789111",
            "TagKey": "category",
            "TagValues": [
                "private"
            ]
        }
    },
    "MaxResults": 10
}
```

Output:

```
{
    "PrincipalResourcePermissions": [{
            "Principal": {
                "DataLakePrincipalIdentifier": "arn:aws:iam::123456789111:user/lf-
admin"
            },
            "Resource": {
                "LFTag": {
                    "CatalogId": "123456789111",
                    "TagKey": "category",
                    "TagValues": [
                        "*"
                    ]
                }
            },
            "Permissions": [
                "DESCRIBE"
            ],
            "PermissionsWithGrantOption": [
                "DESCRIBE"
            ]
        },
        {
            "Principal": {
```

```
                "DataLakePrincipalIdentifier": "arn:aws:iam::123456789111:user/lf-
admin"
            },
            "Resource": {
                "LFTag": {
                    "CatalogId": "123456789111",
                    "TagKey": "category",
                    "TagValues": [
                        "*"
                    ]
                }
            },
            "Permissions": [
                "ASSOCIATE"
            ],
            "PermissionsWithGrantOption": [
                "ASSOCIATE"
            ]
        }
    ],
    "NextToken": "EJwY21GMGFXOXVJanA3SW5Ocm1pc3Npb25zIjpmYWxzZX0="
}
```

Per ulteriori informazioni, consulta Gestione delle autorizzazioni di Lake Formation nella Guida per gli sviluppatori di AWS Lake Formation.

- Per i dettagli sull'API, consulta ListPermissions AWS CLICommand Reference.

`list-resources`

Il seguente esempio di codice mostra come utilizzare`list-resources`.

AWS CLI

Come elencare le risorse gestite da Lake Formation

L'esempio `list-resources` seguente elenca le risorse che corrispondono alla condizione gestita da Lake Formation.

```
aws lakeformation list-resources \
    --cli-input-json file://input.json
```

Contenuto di `input.json`:

```
{
    "FilterConditionList": [{
        "Field": "ROLE_ARN",
        "ComparisonOperator": "CONTAINS",
        "StringValueList": [
            "123456789111"
        ]
    }],
    "MaxResults": 10
}
```

Output:

```
{
    "ResourceInfoList": [{
            "ResourceArn": "arn:aws:s3:::lf-data-lake-123456789111",
            "RoleArn": "arn:aws:iam::123456789111:role/LF-GlueServiceRole",
            "LastModified": "2022-07-21T02:12:46.669000+00:00"
        },
        {
            "ResourceArn": "arn:aws:s3:::lf-emr-test-123456789111",
            "RoleArn": "arn:aws:iam::123456789111:role/EMRLFS3Role",
            "LastModified": "2022-07-29T16:22:03.211000+00:00"
        }
    ]
}
```

Per ulteriori informazioni, consulta Gestione delle autorizzazioni di Lake Formation nella Guida per gli sviluppatori di AWS Lake Formation.

- Per i dettagli sull'API, consulta ListResources AWS CLICommand Reference.

`list-transactions`

Il seguente esempio di codice mostra come utilizzare`list-transactions`.

AWS CLI

Come elencare tutti i dettagli delle transazioni

L'esempio `list-transactions` seguente restituisce i metadati relativi alle transazioni e al loro stato.

```
aws lakeformation list-transactions \
    --cli-input-json file://input.json
```

Contenuto di `input.json`:

```
{
    "CatalogId": "123456789111",
    "StatusFilter": "ALL",
    "MaxResults": 3
}
```

Output:

```
{
    "Transactions": [{
            "TransactionId": "1234569f08804cb790d950d4d0fe485e",
            "TransactionStatus": "committed",
            "TransactionStartTime": "2022-08-10T14:32:29.220000+00:00",
            "TransactionEndTime": "2022-08-10T14:32:33.751000+00:00"
        },
        {
            "TransactionId": "12345972ca8347b89825e33c5774aec4",
            "TransactionStatus": "committed",
            "TransactionStartTime": "2022-08-10T14:29:04.046000+00:00",
            "TransactionEndTime": "2022-08-10T14:29:09.681000+00:00"
        },
        {
            "TransactionId": "12345daf6cb047dbba8ad9b0414613b2",
            "TransactionStatus": "committed",
            "TransactionStartTime": "2022-08-10T13:56:51.261000+00:00",
            "TransactionEndTime": "2022-08-10T13:56:51.547000+00:00"
        }
    ],
    "NextToken": "77X1ebypsI7os+X2lhHsZLGNCDK3nNGpwRdFpicSOHgcX1/
QMoniUAKcpR3kj3ts3PVdMA=="
}
```

Per ulteriori informazioni, consulta Lettura e scrittura di dati nel data lake all'interno delle transazioni nella Guida per gli sviluppatori di AWS Lake Formation.

- Per i dettagli sull'API, consulta ListTransactions AWS CLICommand Reference.

`put-data-lake-settings`

Il seguente esempio di codice mostra come utilizzare`put-data-lake-settings`.

AWS CLI

Per configurare le impostazioni AWS del data lake gestito da Lake Formation

L'esempio `put-data-lake-settings` seguente imposta l'elenco degli amministratori del data lake e altre impostazioni del data lake.

```
aws lakeformation put-data-lake-settings \
    --cli-input-json file://input.json
```

Contenuto di `input.json`:

```
{
    "DataLakeSettings": {
        "DataLakeAdmins": [{
                "DataLakePrincipalIdentifier": "arn:aws:iam::123456789111:user/lf-
admin"
            }
        ],
        "CreateDatabaseDefaultPermissions": [],
        "CreateTableDefaultPermissions": [],
        "TrustedResourceOwners": [],
        "AllowExternalDataFiltering": true,
        "ExternalDataFilteringAllowList": [{
            "DataLakePrincipalIdentifier ": "123456789111"
        }],
        "AuthorizedSessionTagValueList": ["Amazon EMR"]
    }
}
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Modifica delle impostazioni di sicurezza predefinite per il data lake nella Guida per gli sviluppatori di AWS Lake Formation.

- Per i dettagli sull'API, consulta Command PutDataLakeSettingsReference AWS CLI .

`register-resource`

Il seguente esempio di codice mostra come utilizzare`register-resource`.

AWS CLI

Esempio 1: come registrare l'archiviazione nel data lake utilizzando un ruolo collegato ai servizi

L'esempio `register-resource` seguente registra la risorsa come gestita da Lake Formation utilizzando un ruolo collegato ai servizi.

```
aws lakeformation register-resource \
    --cli-input-json file://input.json
```

Contenuto di `input.json`:

```
{
    "ResourceArn": "arn:aws:s3:::lf-emr-athena-result-123",
    "UseServiceLinkedRole": true
}
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Aggiunta di una posizione Amazon S3 al data lake nella Guida per gli sviluppatori di AWS Lake Formation.

Esempio 2: come registrare l'archiviazione nel data lake utilizzando una regola personalizzata

L'esempio `register-resource` seguente registra la risorsa come gestita da Lake Formation utilizzando un ruolo personalizzato.

```
aws lakeformation register-resource \
    --cli-input-json file://input.json
```

Contenuto di `input.json`:

```
{
    "ResourceArn": "arn:aws:s3:::lf-emr-athena-result-123",
    "UseServiceLinkedRole": false,
    "RoleArn": "arn:aws:iam::123456789111:role/LF-GlueServiceRole"
}
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Aggiunta di una posizione Amazon S3 al data lake nella Guida per gli sviluppatori di AWS Lake Formation.

- Per i dettagli sull'API, consulta RegisterResource AWS CLICommand Reference.

`remove-lf-tags-from-resource`

Il seguente esempio di codice mostra come utilizzare`remove-lf-tags-from-resource`.

AWS CLI

Come rimuovere il tag LF da una risorsa

L'esempio `remove-lf-tags-from-resource` seguente rimuove l'associazione tra il tag LF e la risorsa della tabella.

```
aws lakeformation remove-lf-tags-from-resource \
    --cli-input-json file://input.json
```

Contenuto di `input.json`:

```
{
    "CatalogId": "123456789111",
    "Resource": {
        "Table": {
            "CatalogId": "123456789111",
            "DatabaseName": "tpc",
            "Name": "dl_tpc_promotion"
        }
    },
    "LFTags": [{
        "CatalogId": "123456789111",
        "TagKey": "usergroup",
        "TagValues": [
            "developer"
        ]
    }]
}
```

Output:

```
{
    "Failures": []
}
```

Per ulteriori informazioni, consulta Assegnazione di tag LF alle risorse del Catalogo dati nella Guida per gli sviluppatori di AWS Lake Formation.

- Per i dettagli sull'API, consulta RemoveLfTagsFromResource AWS CLICommand Reference.

`revoke-permissions`

Il seguente esempio di codice mostra come utilizzare`revoke-permissions`.

AWS CLI

Come revocare al responsabile le autorizzazioni sulle risorse

L'esempio `revoke-permissions` seguente revoca al responsabile l'accesso a una tabella specifica di un determinato database.

```
aws lakeformation revoke-permissions \
    --cli-input-json file://input.json
```

Contenuto di `input.json`:

```
{
    "CatalogId": "123456789111",
    "Principal": {
        "DataLakePrincipalIdentifier": "arn:aws:iam::123456789111:user/lf-developer"
    },
    "Resource": {
        "Table": {
            "CatalogId": "123456789111",
            "DatabaseName": "tpc",
            "Name": "dl_tpc_promotion"
        }
    },
    "Permissions": [
        "ALL"
    ],
    "PermissionsWithGrantOption": []
}
```

Questo comando non produce alcun output.

Per ulteriori informazioni sulla concessione e la revoca dell'accesso a Lake Formation, consulta la pagina Concessione e revoca delle autorizzazioni delle risorse del catalogo dati nella Guida per gli sviluppatori di AWS .

- Per i dettagli sull'API, consulta RevokePermissions AWS CLICommand Reference.

`search-databases-by-lf-tags`

Il seguente esempio di codice mostra come utilizzare`search-databases-by-lf-tags`.

AWS CLI

Per cercare nelle risorse del database per LFTags

L'`search-databases-by-lf-tags`esempio seguente cerca le risorse del database che corrispondono all' LFTag espressione.

```
aws lakeformation search-databases-by-lf-tags \
    --cli-input-json file://input.json
```

Contenuto di `input.json`:

```
{
    "MaxResults": 1,
    "CatalogId": "123456789111",
    "Expression": [{
        "TagKey": "usergroup",
        "TagValues": [
            "developer"
        ]
    }]
}
```

Output:

```
{
    "DatabaseList": [{
        "Database": {
            "CatalogId": "123456789111",
            "Name": "tpc"
```

```
        },
        "LFTags": [{
            "CatalogId": "123456789111",
            "TagKey": "usergroup",
            "TagValues": [
                "developer"
            ]
        }]
    }]
}
```

Per ulteriori informazioni, consulta Visualizzazione delle risorse a cui è assegnato un tag LF nella Guida per gli sviluppatori di AWS Lake Formation.

- Per i dettagli sull'API, vedere SearchDatabasesByLfTagsin AWS CLI Command Reference.

`search-tables-by-lf-tags`

Il seguente esempio di codice mostra come utilizzare`search-tables-by-lf-tags`.

AWS CLI

Per cercare nelle risorse della tabella per LFTags

L'`search-tables-by-lf-tags`esempio seguente esegue una ricerca sulle risorse della tabella che corrispondono all' LFTag espressione.

```
aws lakeformation search-tables-by-lf-tags \
    --cli-input-json file://input.json
```

Contenuto di `input.json`:

```
{
    "MaxResults": 2,
    "CatalogId": "123456789111",
    "Expression": [{
        "TagKey": "usergroup",
        "TagValues": [
            "developer"
        ]
    }]
}
```

Output:


```
{
    "NextToken": "c2VhcmNoQWxsVGFnc0luVGFibGVzIjpmYWxzZX0=",
    "TableList": [{
        "Table": {
            "CatalogId": "123456789111",
            "DatabaseName": "tpc",
            "Name": "dl_tpc_item"
        },
        "LFTagOnDatabase": [{
            "CatalogId": "123456789111",
            "TagKey": "usergroup",
            "TagValues": [
                "developer"
            ]
        }],
        "LFTagsOnTable": [{
            "CatalogId": "123456789111",
            "TagKey": "usergroup",
            "TagValues": [
                "developer"
            ]
        }],
        "LFTagsOnColumns": [{
                "Name": "i_item_desc",
                "LFTags": [{
                    "CatalogId": "123456789111",
                    "TagKey": "usergroup",
                    "TagValues": [
                        "developer"
                    ]
                }]
            },
            {
                "Name": "i_container",
                "LFTags": [{
                    "CatalogId": "123456789111",
                    "TagKey": "usergroup",
                    "TagValues": [
                        "developer"
                    ]
                }]
            },
```

```
            {
                "Name": "i_wholesale_cost",
                "LFTags": [{
                    "CatalogId": "123456789111",
                    "TagKey": "usergroup",
                    "TagValues": [
                        "developer"
                    ]
                }]
            },
            {
                "Name": "i_manufact_id",
                "LFTags": [{
                    "CatalogId": "123456789111",
                    "TagKey": "usergroup",
                    "TagValues": [
                        "developer"
                    ]
                }]
            },
            {
                "Name": "i_brand_id",
                "LFTags": [{
                    "CatalogId": "123456789111",
                    "TagKey": "usergroup",
                    "TagValues": [
                        "developer"
                    ]
                }]
            },
            {
                "Name": "i_formulation",
                "LFTags": [{
                    "CatalogId": "123456789111",
                    "TagKey": "usergroup",
                    "TagValues": [
                        "developer"
                    ]
                }]
            },
            {
                "Name": "i_current_price",
                "LFTags": [{
                    "CatalogId": "123456789111",
```

```
                    "TagKey": "usergroup",
                    "TagValues": [
                        "developer"
                    ]
                }]
            },
            {
                "Name": "i_size",
                "LFTags": [{
                    "CatalogId": "123456789111",
                    "TagKey": "usergroup",
                    "TagValues": [
                        "developer"
                    ]
                }]
            },
            {
                "Name": "i_rec_start_date",
                "LFTags": [{
                    "CatalogId": "123456789111",
                    "TagKey": "usergroup",
                    "TagValues": [
                        "developer"
                    ]
                }]
            },
            {
                "Name": "i_manufact",
                "LFTags": [{
                    "CatalogId": "123456789111",
                    "TagKey": "usergroup",
                    "TagValues": [
                        "developer"
                    ]
                }]
            },
            {
                "Name": "i_item_sk",
                "LFTags": [{
                    "CatalogId": "123456789111",
                    "TagKey": "usergroup",
                    "TagValues": [
                        "developer"
                    ]
```

```
            }]
        },
        {
            "Name": "i_manager_id",
            "LFTags": [{
                "CatalogId": "123456789111",
                "TagKey": "usergroup",
                "TagValues": [
                    "developer"
                ]
            }]
        },
        {
            "Name": "i_item_id",
            "LFTags": [{
                "CatalogId": "123456789111",
                "TagKey": "usergroup",
                "TagValues": [
                    "developer"
                ]
            }]
        },
        {
            "Name": "i_class_id",
            "LFTags": [{
                "CatalogId": "123456789111",
                "TagKey": "usergroup",
                "TagValues": [
                    "developer"
                ]
            }]
        },
        {
            "Name": "i_class",
            "LFTags": [{
                "CatalogId": "123456789111",
                "TagKey": "usergroup",
                "TagValues": [
                    "developer"
                ]
            }]
        },
        {
            "Name": "i_category",
```

```
                "LFTags": [{
                    "CatalogId": "123456789111",
                    "TagKey": "usergroup",
                    "TagValues": [
                        "developer"
                    ]
                }]
            },
            {
                "Name": "i_category_id",
                "LFTags": [{
                    "CatalogId": "123456789111",
                    "TagKey": "usergroup",
                    "TagValues": [
                        "developer"
                    ]
                }]
            },
            {
                "Name": "i_brand",
                "LFTags": [{
                    "CatalogId": "123456789111",
                    "TagKey": "usergroup",
                    "TagValues": [
                        "developer"
                    ]
                }]
            },
            {
                "Name": "i_units",
                "LFTags": [{
                    "CatalogId": "123456789111",
                    "TagKey": "usergroup",
                    "TagValues": [
                        "developer"
                    ]
                }]
            },
            {
                "Name": "i_rec_end_date",
                "LFTags": [{
                    "CatalogId": "123456789111",
                    "TagKey": "usergroup",
                    "TagValues": [
```

```
                        "developer"
                    ]
                }]
            },
            {
                "Name": "i_color",
                "LFTags": [{
                    "CatalogId": "123456789111",
                    "TagKey": "usergroup",
                    "TagValues": [
                        "developer"
                    ]
                }]
            },
            {
                "Name": "i_product_name",
                "LFTags": [{
                    "CatalogId": "123456789111",
                    "TagKey": "usergroup",
                    "TagValues": [
                        "developer"
                    ]
                }]
            }
        ]
    }]
}
```

Per ulteriori informazioni, consulta Visualizzazione delle risorse a cui è assegnato un tag LF nella Guida per gli sviluppatori di AWS Lake Formation.

- Per i dettagli sull'API, vedere SearchTablesByLfTagsin AWS CLI Command Reference.

`start-query-planning`

Il seguente esempio di codice mostra come utilizzare`start-query-planning`.

AWS CLI

Come elaborare l'istruzione di una query

L'esempio `start-query-planning` seguente invia una richiesta per elaborare l'istruzione di una query.

```
aws lakeformation start-query-planning \
    --cli-input-json file://input.json
```

Contenuto di `input.json`:

```
{
    "QueryPlanningContext": {
        "CatalogId": "012345678901",
        "DatabaseName": "tpc"
    },
    "QueryString": "select * from dl_tpc_household_demographics_gov where
 hd_income_band_sk=9"
}
```

Output:

```
{
    "QueryId": "772a273f-4a62-4cda-8d98-69615ee8be9b"
}
```

Per ulteriori informazioni, consulta Lettura e scrittura di dati nel data lake all'interno delle transazioni nella Guida per gli sviluppatori di AWS Lake Formation.

- Per i dettagli sull'API, consulta StartQueryPlanning AWS CLICommand Reference.

`start-transaction`

Il seguente esempio di codice mostra come utilizzare`start-transaction`.

AWS CLI

Come avviare una nuova transazione

L'esempio `start-transaction` seguente avvia una nuova transazione e restituisce il relativo ID.

```
aws lakeformation start-transaction \
    --transaction-type = 'READ_AND_WRITE'
```

Output:

```
{
    "TransactionId": "b014d972ca8347b89825e33c5774aec4"
}
```

Per ulteriori informazioni, consulta Lettura e scrittura di dati nel data lake all'interno delle transazioni nella Guida per gli sviluppatori di AWS Lake Formation.

- Per i dettagli sull'API, consulta StartTransaction AWS CLICommand Reference.

`update-lf-tag`

Il seguente esempio di codice mostra come utilizzare`update-lf-tag`.

AWS CLI

Come aggiornare la definizione del tag LF

L'esempio `update-lf-tag` seguente aggiorna la definizione del tag LF.

```
aws lakeformation update-lf-tag \
    --catalog-id '123456789111' \
    --tag-key 'usergroup' \
    --tag-values-to-add '["admin"]'
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Gestione dei tag LF per il controllo dell'accesso ai metadati nella Guida per gli sviluppatori di AWS Lake Formation.

- Per i dettagli sull'API, consulta UpdateLfTag AWS CLICommand Reference.

`update-table-objects`

Il seguente esempio di codice mostra come utilizzare`update-table-objects`.

AWS CLI

Come modificare gli oggetti della tabella governata

L'esempio `update-table-objects` seguente aggiunge gli oggetti S3 forniti alla tabella governata specificata.

```
aws lakeformation update-table-objects \
    --cli-input-json file://input.json
```

Contenuto di `input.json`:

```
{
    "CatalogId": "012345678901",
    "DatabaseName": "tpc",
    "TableName": "dl_tpc_household_demographics_gov",
    "TransactionId": "12347a9f75424b9b915f6ff201d2a190",
    "WriteOperations": [{
        "AddObject": {
            "Uri": "s3://lf-data-lake-012345678901/target/
dl_tpc_household_demographics_gov/run-unnamed-1-part-block-0-r-00000-snappy-
ff26b17504414fe88b302cd795eabd00.parquet",
            "ETag": "1234ab1fc50a316b149b4e1f21a73800",
            "Size": 42200
        }
    }]
}
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Lettura e scrittura di dati nel data lake all'interno delle transazioni nella Guida per gli sviluppatori di AWS Lake Formation.

- Per i dettagli sull'API, consulta UpdateTableObjects AWS CLICommand Reference.

Esempi di utilizzo di Lambda AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with Lambda.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`add-layer-version-permission`

Il seguente esempio di codice mostra come utilizzare`add-layer-version-permission`.

AWS CLI

Come aggiungere autorizzazioni a una versione del livello

L'esempio `add-layer-version-permission` seguente concede all'account specificato l'autorizzazione a utilizzare la versione 1 del livello `my-layer`.

```
aws lambda add-layer-version-permission \
    --layer-name my-layer \
    --statement-id xaccount \
    --action lambda:GetLayerVersion  \
    --principal 123456789012 \
    --version-number 1
```

Output:

```
{
    "RevisionId": "35d87451-f796-4a3f-a618-95a3671b0a0c",
    "Statement":
    {
        "Sid":"xaccount",
        "Effect":"Allow",
        "Principal":{
            "AWS":"arn:aws:iam::210987654321:root"
        },
        "Action":"lambda:GetLayerVersion",
        "Resource":"arn:aws:lambda:us-east-2:123456789012:layer:my-layer:1"
    }
}
```

Per ulteriori informazioni, consulta Livelli di AWS Lambda nella Guida per gli sviluppatori di AWS Lambda.

- Per i dettagli sull'API, consulta AddLayerVersionPermission AWS CLICommand Reference.

`add-permission`

Il seguente esempio di codice mostra come utilizzare`add-permission`.

AWS CLI

Come aggiungere le autorizzazioni a una funzione Lambda esistente

Il comando `add-permission` seguente concede al servizio Amazon SNS l'autorizzazione a invocare la funzione denominata `my-function`.

```
aws lambda add-permission \
    --function-name my-function \
    --action lambda:InvokeFunction \
    --statement-id sns \
    --principal sns.amazonaws.com
```

Output:

```
{
    "Statement":
    {
        "Sid":"sns",
        "Effect":"Allow",
        "Principal":{
            "Service":"sns.amazonaws.com"
        },
        "Action":"lambda:InvokeFunction",
        "Resource":"arn:aws:lambda:us-east-2:123456789012:function:my-function"
    }
}
```

Per ulteriori informazioni, consulta Using Resource-based Policies for Lambda nella AWS Lambda Developer Guide.AWS

- Per i dettagli sull'API, consulta Command Reference. AddPermissionAWS CLI

`create-alias`

Il seguente esempio di codice mostra come utilizzare`create-alias`.

AWS CLI

Come creare un alias per una funzione Lambda

L'esempio `create-alias` seguente crea un alias denominato `LIVE` che punta alla versione 1 della funzione Lambda `my-function`.

```
aws lambda create-alias \
    --function-name my-function \
    --description "alias for live version of function" \
    --function-version 1 \
    --name LIVE
```

Output:

```
{
    "FunctionVersion": "1",
    "Name": "LIVE",
    "AliasArn": "arn:aws:lambda:us-west-2:123456789012:function:my-function:LIVE",
    "RevisionId": "873282ed-4cd3-4dc8-a069-d0c647e470c6",
    "Description": "alias for live version of function"
}
```

Per ulteriori informazioni, consulta Configurazione degli alias delle funzioni AWS Lambda nella AWS Lambda Developer Guide.

- Per i dettagli sull'API, consulta CreateAliasCommand Reference.AWS CLI

`create-event-source-mapping`

Il seguente esempio di codice mostra come utilizzare`create-event-source-mapping`.

AWS CLI

Per creare una mappatura tra un'origine di eventi e una funzione AWS Lambda

L'esempio `create-event-source-mapping` seguente crea una mappatura tra una coda SQS e la funzione Lambda `my-function`.

```
aws lambda create-event-source-mapping \
    --function-name my-function \
    --batch-size 5 \
```

```
    --event-source-arn arn:aws:sqs:us-west-2:123456789012:mySQSqueue
```

Output:

```
{
    "UUID": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
    "StateTransitionReason": "USER_INITIATED",
    "LastModified": 1569284520.333,
    "BatchSize": 5,
    "State": "Creating",
    "FunctionArn": "arn:aws:lambda:us-west-2:123456789012:function:my-function",
    "EventSourceArn": "arn:aws:sqs:us-west-2:123456789012:mySQSqueue"
}
```

Per ulteriori informazioni, consulta Strumento di mappatura dell'origine degli eventi AWS Lambda nella Guida per gli sviluppatori di AWS Lambda.

- Per i dettagli sull'API, consulta AWS CLI Command CreateEventSourceMappingReference.

`create-function`

Il seguente esempio di codice mostra come utilizzare`create-function`.

AWS CLI

Come creare una funzione Lambda

L'esempio di `create-function` seguente crea una funzione Lambda denominata `my-function`.

```
aws lambda create-function \
    --function-name my-function \
    --runtime nodejs22.x \
    --zip-file fileb://my-function.zip \
    --handler my-function.handler \
    --role arn:aws:iam::123456789012:role/service-role/MyTestFunction-role-tges6bf4
```

Contenuto di `my-function.zip`:

```
This file is a deployment package that contains your function code and any
 dependencies.
```

Output:

```
{
    "TracingConfig": {
        "Mode": "PassThrough"
    },
    "CodeSha256": "PFn4S+er27qk+UuZSTKEQfNKG/XNn7QJs90mJgq6oH8=",
    "FunctionName": "my-function",
    "CodeSize": 308,
    "RevisionId": "873282ed-4cd3-4dc8-a069-d0c647e470c6",
    "MemorySize": 128,
    "FunctionArn": "arn:aws:lambda:us-west-2:123456789012:function:my-function",
    "Version": "$LATEST",
    "Role": "arn:aws:iam::123456789012:role/service-role/MyTestFunction-role-
zgur6bf4",
    "Timeout": 3,
    "LastModified": "2025-10-14T22:26:11.234+0000",
    "Handler": "my-function.handler",
    "Runtime": "nodejs22.x",
    "Description": ""
}
```

Per ulteriori informazioni, consulta Configurazione della memoria della funzione Lambda nella Guida per gli sviluppatori Lambda AWS .

- Per i dettagli sull'API, consulta CreateFunction AWS CLICommand Reference.

`delete-alias`

Il seguente esempio di codice mostra come utilizzare`delete-alias`.

AWS CLI

Come eliminare un alias di una funzione Lambda

L'esempio `delete-alias` seguente elimina l'alias denominato `LIVE` dalla funzione Lambda `my-function`.

```
aws lambda delete-alias \
    --function-name my-function \
    --name LIVE
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Configurazione degli alias delle funzioni AWS Lambda nella AWS Lambda Developer Guide.

- Per i dettagli sull'API, consulta DeleteAliasCommand Reference.AWS CLI

`delete-event-source-mapping`

Il seguente esempio di codice mostra come utilizzare`delete-event-source-mapping`.

AWS CLI

Per eliminare la mappatura tra un'origine di eventi e una funzione AWS Lambda

L'esempio `delete-event-source-mapping` seguente elimina la mappatura tra una coda SQS e la funzione Lambda `my-function`.

```
aws lambda delete-event-source-mapping \
    --uuid  a1b2c3d4-5678-90ab-cdef-11111EXAMPLE
```

Output:

```
{
    "UUID": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
    "StateTransitionReason": "USER_INITIATED",
    "LastModified": 1569285870.271,
    "BatchSize": 5,
    "State": "Deleting",
    "FunctionArn": "arn:aws:lambda:us-west-2:123456789012:function:my-function",
    "EventSourceArn": "arn:aws:sqs:us-west-2:123456789012:mySQSqueue"
}
```

Per ulteriori informazioni, consulta Strumento di mappatura dell'origine degli eventi AWS Lambda nella Guida per gli sviluppatori di AWS Lambda.

- Per i dettagli sull'API, consulta AWS CLI Command DeleteEventSourceMappingReference.

`delete-function-concurrency`

Il seguente esempio di codice mostra come utilizzare`delete-function-concurrency`.

AWS CLI

Come rimuovere il limite di esecuzione simultanea riservato da una funzione

L'esempio `delete-function-concurrency` seguente elimina il limite di esecuzione simultanea riservato dalla funzione `my-function`.

```
aws lambda delete-function-concurrency \
    --function-name  my-function
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Prenotazione della simultaneità per una funzione Lambda nella Guida per gli sviluppatori di AWS Lambda.

- Per i dettagli sull'API, consulta DeleteFunctionConcurrency AWS CLICommand Reference.

`delete-function-event-invoke-config`

Il seguente esempio di codice mostra come utilizzare`delete-function-event-invoke-config`.

AWS CLI

Come eliminare una configurazione di invocazione asincrona

L'esempio `delete-function-event-invoke-config` seguente elimina la configurazione di invocazione asincrona per l'alias `GREEN` della funzione specificata.

```
aws lambda delete-function-event-invoke-config --function-name my-function:GREEN
```

- Per i dettagli sull'API, consulta DeleteFunctionEventInvokeConfig AWS CLICommand Reference.

`delete-function`

Il seguente esempio di codice mostra come utilizzare`delete-function`.

AWS CLI

Esempio 1: come eliminare una funzione Lambda in base al nome della funzione

L'esempio di `delete-function` seguente elimina la funzione Lambda denominata `my-function` specificandone il nome.

```
aws lambda delete-function \
    --function-name my-function
```

Questo comando non produce alcun output.

Esempio 2: eliminazione di una funzione Lambda in base all'ARN della funzione

L'esempio di `delete-function` seguente elimina la funzione Lambda denominata `my-function` specificando l'ARN della funzione.

```
aws lambda delete-function \
    --function-name arn:aws:lambda:us-west-2:123456789012:function:my-function
```

Questo comando non produce alcun output.

Esempio 3: eliminazione di una funzione Lambda in base all'ARN parziale della funzione

L'esempio di `delete-function` seguente elimina la funzione Lambda denominata `my-function` specificando l'ARN parziale della funzione.

```
aws lambda delete-function \
    --function-name 123456789012:function:my-function
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Configurazione delle funzioni Lambda AWS nella Guida per gli sviluppatori di AWS .

- Per i dettagli sull'API, consulta DeleteFunction AWS CLICommand Reference.

`delete-layer-version`

Il seguente esempio di codice mostra come utilizzare`delete-layer-version`.

AWS CLI

Come eliminare la versione di un livello Lambda

L'esempio `delete-layer-version` seguente elimina la versione 2 del livello denominato `my-layer`.

```
aws lambda delete-layer-version \
    --layer-name my-layer \
    --version-number 2
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Livelli di AWS Lambda nella Guida per gli sviluppatori di AWS Lambda.

- Per i dettagli sull'API, consulta DeleteLayerVersion AWS CLICommand Reference.

`delete-provisioned-concurrency-config`

Il seguente esempio di codice mostra come utilizzare`delete-provisioned-concurrency-config`.

AWS CLI

Come eliminare una configurazione della simultaneità fornita

L'esempio `delete-provisioned-concurrency-config` seguente elimina la configurazione di simultaneità fornita per l'alias `GREEN` della funzione specificata.

```
aws lambda delete-provisioned-concurrency-config \
    --function-name my-function \
    --qualifier GREEN
```

- Per i dettagli sull'API, consulta DeleteProvisionedConcurrencyConfig AWS CLICommand Reference.

`get-account-settings`

Il seguente esempio di codice mostra come utilizzare`get-account-settings`.

AWS CLI

Per recuperare i dettagli sul tuo account in una regione AWS

L'esempio `get-account-settings` seguente mostra i limiti Lambda e le informazioni sull'utilizzo per il tuo account.

```
aws lambda get-account-settings
```

Output:

```
{
    "AccountLimit": {
       "CodeSizeUnzipped": 262144000,
       "UnreservedConcurrentExecutions": 1000,
       "ConcurrentExecutions": 1000,
       "CodeSizeZipped": 52428800,
       "TotalCodeSize": 80530636800
    },
    "AccountUsage": {
       "FunctionCount": 4,
       "TotalCodeSize": 9426
    }
}
```

Per ulteriori informazioni, consulta Limiti di AWS Lambda nella Guida per gli sviluppatori di AWS .

- Per i dettagli sull'API, consulta GetAccountSettings AWS CLICommand Reference.

`get-alias`

Il seguente esempio di codice mostra come utilizzare`get-alias`.

AWS CLI

Come recuperare i dettagli dell'alias di una funzione

L'esempio `get-alias` seguente visualizza i dettagli per l'alias denominato `LIVE` dalla funzione Lambda `my-function`.

```
aws lambda get-alias \
    --function-name my-function \
    --name LIVE
```

Output:

```
{
    "FunctionVersion": "3",
    "Name": "LIVE",
    "AliasArn": "arn:aws:lambda:us-west-2:123456789012:function:my-function:LIVE",
    "RevisionId": "594f41fb-b85f-4c20-95c7-6ca5f2a92c93",
    "Description": "alias for live version of function"
}
```

Per ulteriori informazioni, consulta Configurazione degli alias delle funzioni AWS Lambda nella AWS Lambda Developer Guide.

- Per i dettagli sull'API, consulta GetAliasCommand Reference.AWS CLI

`get-event-source-mapping`

Il seguente esempio di codice mostra come utilizzare`get-event-source-mapping`.

AWS CLI

Come recuperare i dettagli relativi a uno strumento di mappatura dell'origine degli eventi

L'esempio `get-event-source-mapping` seguente mostra i dettagli per la mappatura tra una coda SQS e la funzione Lambda `my-function`.

```
aws lambda get-event-source-mapping \
    --uuid "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE"
```

Output:

```
{
    "UUID": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
    "StateTransitionReason": "USER_INITIATED",
    "LastModified": 1569284520.333,
    "BatchSize": 5,
    "State": "Enabled",
    "FunctionArn": "arn:aws:lambda:us-west-2:123456789012:function:my-function",
    "EventSourceArn": "arn:aws:sqs:us-west-2:123456789012:mySQSqueue"
}
```

Per ulteriori informazioni, consulta Strumento di mappatura dell'origine degli eventi AWS Lambda nella Guida per gli sviluppatori di AWS Lambda.

- Per i dettagli sull'API, consulta GetEventSourceMapping AWS CLICommand Reference.

`get-function-concurrency`

Il seguente esempio di codice mostra come utilizzare`get-function-concurrency`.

AWS CLI

Come visualizzare l'impostazione di simultaneità riservata per una funzione

L'esempio `get-function-concurrency` seguente recupera l'impostazione della simultaneità riservata per la funzione specificata.

```
aws lambda get-function-concurrency \
    --function-name my-function
```

Output:

```
{
    "ReservedConcurrentExecutions": 250
}
```

- Per i dettagli sull'API, consulta GetFunctionConcurrency AWS CLICommand Reference.

`get-function-configuration`

Il seguente esempio di codice mostra come utilizzare`get-function-configuration`.

AWS CLI

Come recuperare le impostazioni specifiche della versione di una funzione Lambda

L'esempio `get-function-configuration` seguente visualizza le impostazioni per la versione 2 della funzione `my-function`.

```
aws lambda get-function-configuration \
    --function-name  my-function:2
```

Output:

```
{
```

```
    "FunctionName": "my-function",
    "LastModified": "2019-09-26T20:28:40.438+0000",
    "RevisionId": "e52502d4-9320-4688-9cd6-152a6ab7490d",
    "MemorySize": 256,
    "Version": "2",
    "Role": "arn:aws:iam::123456789012:role/service-role/my-function-role-uy3l9qyq",
    "Timeout": 3,
    "Runtime": "nodejs10.x",
    "TracingConfig": {
        "Mode": "PassThrough"
    },
    "CodeSha256": "5tT2qgzYUHaqwR716pZ2dpkn/0J1FrzJmlKidWoaCgk=",
    "Description": "",
    "VpcConfig": {
        "SubnetIds": [],
        "VpcId": "",
        "SecurityGroupIds": []
    },
    "CodeSize": 304,
    "FunctionArn": "arn:aws:lambda:us-west-2:123456789012:function:my-function:2",
    "Handler": "index.handler"
}
```


Per ulteriori informazioni, consulta Configurazione delle funzioni Lambda AWS nella Guida per gli sviluppatori di AWS .

- Per i dettagli sull'API, consulta GetFunctionConfiguration AWS CLICommand Reference.

`get-function-event-invoke-config`

Il seguente esempio di codice mostra come utilizzare`get-function-event-invoke-config`.

AWS CLI

Come visualizzare una configurazione di invocazione asincrona

L'esempio `get-function-event-invoke-config` seguente recupera la configurazione di invocazione asincrona per l'alias BLUE della funzione specificata.

```
aws lambda get-function-event-invoke-config \
    --function-name my-function:BLUE
```

Output:

```
{
    "LastModified": 1577824396.653,
    "FunctionArn": "arn:aws:lambda:us-east-2:123456789012:function:my-
function:BLUE",
    "MaximumRetryAttempts": 0,
    "MaximumEventAgeInSeconds": 3600,
    "DestinationConfig": {
        "OnSuccess": {},
        "OnFailure": {
            "Destination": "arn:aws:sqs:us-east-2:123456789012:failed-invocations"
        }
    }
}
```

- Per i dettagli sull'API, consulta GetFunctionEventInvokeConfig AWS CLICommand Reference.

`get-function`

Il seguente esempio di codice mostra come utilizzare`get-function`.

AWS CLI

Come recuperare le informazioni relative a una funzione

Nell'esempio di `get-function` seguente vengono visualizzate informazioni sulla funzione `my-function`.

```
aws lambda get-function \
    --function-name  my-function
```

Output:

```
{
    "Concurrency": {
        "ReservedConcurrentExecutions": 100
    },
    "Code": {
        "RepositoryType": "S3",
        "Location": "https://awslambda-us-west-2-tasks.s3.us-west-2.amazonaws.com/
snapshots/123456789012/my-function..."
    },
```

```
    "Configuration": {
        "TracingConfig": {
            "Mode": "PassThrough"
        },
        "Version": "$LATEST",
        "CodeSha256": "5tT2qgzYUHoqwR616pZ2dpkn/0J1FrzJmlKidWaaCgk=",
        "FunctionName": "my-function",
        "VpcConfig": {
            "SubnetIds": [],
            "VpcId": "",
            "SecurityGroupIds": []
        },
        "MemorySize": 128,
        "RevisionId": "28f0fb31-5c5c-43d3-8955-03e76c5c1075",
        "CodeSize": 304,
        "FunctionArn": "arn:aws:lambda:us-west-2:123456789012:function:my-function",
        "Handler": "index.handler",
        "Role": "arn:aws:iam::123456789012:role/service-role/helloWorldPython-role-
uy3l9qyq",
        "Timeout": 3,
        "LastModified": "2025-09-24T18:20:35.054+0000",
        "Runtime": "nodejs22.x",
        "Description": ""
    }
}
```

Per ulteriori informazioni, consulta Configurazione della memoria della funzione Lambda nella Guida per gli sviluppatori Lambda AWS .

- Per i dettagli sull'API, consulta GetFunction AWS CLICommand Reference.

`get-layer-version-by-arn`

Il seguente esempio di codice mostra come utilizzare`get-layer-version-by-arn`.

AWS CLI

Come recuperare informazioni su una versione del livello Lambda

L'esempio `get-layer-version-by-arn` seguente visualizza informazioni sulla versione del livello con il nome della risorsa Amazon (ARN) specificato.

```
aws lambda get-layer-version-by-arn \
```

```
    --arn "arn:aws:lambda:us-west-2:123456789012:layer:AWSLambda-Python311-
SciPy1x:2"
```

Output:

```
{
    "LayerVersionArn": "arn:aws:lambda:us-west-2:123456789012:layer:AWSLambda-
Python311-SciPy1x:2",
    "Description": "AWS Lambda SciPy layer for Python 3.11 (scipy-1.1.0,
 numpy-1.15.4) https://github.com/scipy/scipy/releases/tag/v1.1.0 https://
github.com/numpy/numpy/releases/tag/v1.15.4",
    "CreatedDate": "2023-10-12T10:09:38.398+0000",
    "LayerArn": "arn:aws:lambda:us-west-2:123456789012:layer:AWSLambda-Python311-
SciPy1x",
    "Content": {
        "CodeSize": 41784542,
        "CodeSha256": "GGmv8ocUw4cly0T8HL0Vx/f5V4RmSCGNjDIslY4VskM=",
        "Location": "https://awslambda-us-west-2-layers.s3.us-west-2.amazonaws.com/
snapshots/123456789012/..."
    },
    "Version": 2,
    "CompatibleRuntimes": [
        "python3.11"
    ],
    "LicenseInfo": "SciPy: https://github.com/scipy/scipy/blob/main/LICENSE.txt,
 NumPy: https://github.com/numpy/numpy/blob/main/LICENSE.txt"
}
```

Per ulteriori informazioni, consulta Livelli di AWS Lambda nella Guida per gli sviluppatori di AWS Lambda.

- Per i dettagli sull'API, consulta GetLayerVersionByArn AWS CLICommand Reference.

`get-layer-version-policy`

Il seguente esempio di codice mostra come utilizzare`get-layer-version-policy`.

AWS CLI

Come recuperare la policy delle autorizzazioni per una versione del livello Lambda

L'esempio `get-layer-version-policy` seguente visualizza le informazioni sulla policy relative alla versione 1 per il livello denominato `my-layer`.

```
aws lambda get-layer-version-policy \
    --layer-name my-layer \
    --version-number 1
```

Output:

```
{
    "Policy": {
        "Version":"2012-10-17",
        "Id":"default",
        "Statement":
        [
            {
                "Sid":"xaccount",
                "Effect":"Allow",
                "Principal": {"AWS":"arn:aws:iam::123456789012:root"},
                "Action":"lambda:GetLayerVersion",
                "Resource":"arn:aws:lambda:us-west-2:123456789012:layer:my-layer:1"
            }
        ]
    },
    "RevisionId": "c68f21d2-cbf0-4026-90f6-1375ee465cd0"
}
```

Per ulteriori informazioni, consulta Livelli di AWS Lambda nella Guida per gli sviluppatori di AWS Lambda.

- Per i dettagli sull'API, consulta GetLayerVersionPolicy AWS CLICommand Reference.

`get-layer-version`

Il seguente esempio di codice mostra come utilizzare`get-layer-version`.

AWS CLI

Come recuperare informazioni su una versione del livello Lambda

L'esempio `get-layer-version` seguente visualizza informazioni sulla versione 1 del livello denominato `my-layer`.

```
aws lambda get-layer-version \
    --layer-name my-layer \
```

```
    --version-number 1
```

Output:

```
{
    "Content": {
        "Location": "https://awslambda-us-east-2-layers.s3.us-east-2.amazonaws.com/
snapshots/123456789012/my-layer-4aaa2fbb-ff77-4b0a-ad92-5b78a716a96a?
versionId=27iWyA73cCAYqyH...",
        "CodeSha256": "tv9jJO+rPbXUUXuRKi7CwHzKtLDkDRJLB3cC3Z/ouXo=",
        "CodeSize": 169
    },
    "LayerArn": "arn:aws:lambda:us-east-2:123456789012:layer:my-layer",
    "LayerVersionArn": "arn:aws:lambda:us-east-2:123456789012:layer:my-layer:1",
    "Description": "My Python layer",
    "CreatedDate": "2018-11-14T23:03:52.894+0000",
    "Version": 1,
    "LicenseInfo": "MIT",
    "CompatibleRuntimes": [
        "python3.10",
        "python3.11"
    ]
}
```

Per ulteriori informazioni, consulta Livelli di AWS Lambda nella Guida per gli sviluppatori di AWS Lambda.

- Per i dettagli sull'API, consulta GetLayerVersion AWS CLICommand Reference.

`get-policy`

Il seguente esempio di codice mostra come utilizzare`get-policy`.

AWS CLI

Come recuperare la policy IAM basata sulle risorse per una funzione, una versione o un alias

L'esempio `get-policy` seguente mostra le informazioni sulla policy sulla funzione Lambda `my-function`.

```
aws lambda get-policy \
    --function-name my-function
```

Output:

```
{
    "Policy": {
        "Version":"2012-10-17",
        "Id":"default",
        "Statement":
        [
            {
                "Sid":"iot-events",
                "Effect":"Allow",
                "Principal": {"Service":"iotevents.amazonaws.com"},
                "Action":"lambda:InvokeFunction",
                "Resource":"arn:aws:lambda:us-west-2:123456789012:function:my-
function"
            }
        ]
    },
    "RevisionId": "93017fc9-59cb-41dc-901b-4845ce4bf668"
}
```

Per ulteriori informazioni, consulta Using Resource-based Policies for Lambda nella AWS Lambda Developer Guide.AWS

- Per i dettagli sull'API, consulta Command Reference. GetPolicyAWS CLI

`get-provisioned-concurrency-config`

Il seguente esempio di codice mostra come utilizzare`get-provisioned-concurrency-config`.

AWS CLI

Come visualizzare una configurazione della simultaneità fornita

L'esempio `get-provisioned-concurrency-config` seguente visualizza i dettagli della configurazione di simultaneità fornita per l'alias `BLUE` della funzione specificata.

```
aws lambda get-provisioned-concurrency-config \
    --function-name my-function \
    --qualifier BLUE
```

Output:

```
{
    "RequestedProvisionedConcurrentExecutions": 100,
    "AvailableProvisionedConcurrentExecutions": 100,
    "AllocatedProvisionedConcurrentExecutions": 100,
    "Status": "READY",
    "LastModified": "2019-12-31T20:28:49+0000"
}
```

- Per i dettagli sull'API, consulta GetProvisionedConcurrencyConfig AWS CLICommand Reference.

`invoke`

Il seguente esempio di codice mostra come utilizzare`invoke`.

AWS CLI

Esempio 1: come invocare una funzione Lambda in modo sincrono

L'esempio di `invoke` seguente invoca la funzione `my-function` in modo sincrono. L'`cli-binary-format`opzione è obbligatoria se utilizzi la versione 2 della AWS CLI. Per ulteriori informazioni, consulta Opzioni della riga di comando globali supportate da AWS CLI nella Guida per l'utente dell'Interfaccia della linea di comando AWS (AWS CLI).

```
aws lambda invoke \
    --function-name my-function \
    --cli-binary-format raw-in-base64-out \
    --payload '{ "name": "Bob" }' \
    response.json
```

Output:

```
{
    "ExecutedVersion": "$LATEST",
    "StatusCode": 200
}
```

Per ulteriori informazioni, consulta Invocare una funzione Lambda in modo sincrono nella Guida per gli sviluppatori di AWS Lambda.

Esempio 2: come invocare una funzione Lambda in modo asincrono

L'esempio di `invoke` seguente invoca la funzione `my-function` in modo asincrono. L'`cli-binary-format`opzione è obbligatoria se utilizzi la versione 2 della AWS CLI. Per ulteriori informazioni, consulta Opzioni della riga di comando globali supportate da AWS CLI nella Guida per l'utente dell'Interfaccia della linea di comando AWS (AWS CLI).

```
aws lambda invoke \
    --function-name my-function \
    --invocation-type Event \
    --cli-binary-format raw-in-base64-out \
    --payload '{ "name": "Bob" }' \
    response.json
```

Output:

```
{
    "StatusCode": 202
}
```

Per ulteriori informazioni, consulta Invocare una funzione Lambda in modo asincrono nella Guida per gli sviluppatori di AWS Lambda.

- Per informazioni dettagliate sull'API, consulta Invoke nella documentazione di riferimento dei comandi della AWS CLI .

`list-aliases`

Il seguente esempio di codice mostra come usare`list-aliases`.

AWS CLI

Come recuperare l'elenco di alias per una funzione Lambda

L'esempio `list-aliases` seguente visualizza un elenco degli alias per la funzione Lambda `my-function`.

```
aws lambda list-aliases \
    --function-name my-function
```

Output:

```
{
```

```
    "Aliases": [
        {
            "AliasArn": "arn:aws:lambda:us-west-2:123456789012:function:my-
function:BETA",
            "RevisionId": "a410117f-ab16-494e-8035-7e204bb7933b",
            "FunctionVersion": "2",
            "Name": "BETA",
            "Description": "alias for beta version of function"
        },
        {
            "AliasArn": "arn:aws:lambda:us-west-2:123456789012:function:my-
function:LIVE",
            "RevisionId": "21d40116-f8b1-40ba-9360-3ea284da1bb5",
            "FunctionVersion": "1",
            "Name": "LIVE",
            "Description": "alias for live version of function"
        }
    ]
}
```

Per ulteriori informazioni, consulta Configurazione degli alias delle funzioni AWS Lambda nella AWS Lambda Developer Guide.

- Per i dettagli sull'API, consulta ListAliasesCommand Reference.AWS CLI

`list-event-source-mappings`

Il seguente esempio di codice mostra come utilizzare`list-event-source-mappings`.

AWS CLI

Come elencare gli strumenti di mappatura dell'origine degli eventi per una funzione

L'esempio `list-event-source-mappings` seguente visualizza un elenco degli strumenti di mappatura dell'origine degli eventi per la funzione Lambda `my-function`.

```
aws lambda list-event-source-mappings \
    --function-name my-function
```

Output:

```
{
    "EventSourceMappings": [
```

```
        {
            "UUID": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
            "StateTransitionReason": "USER_INITIATED",
            "LastModified": 1569284520.333,
            "BatchSize": 5,
            "State": "Enabled",
            "FunctionArn": "arn:aws:lambda:us-west-2:123456789012:function:my-
function",
            "EventSourceArn": "arn:aws:sqs:us-west-2:123456789012:mySQSqueue"
        }
    ]
}
```

Per ulteriori informazioni, consulta Strumento di mappatura dell'origine degli eventi AWS Lambda nella Guida per gli sviluppatori di AWS Lambda.

- Per i dettagli sull'API, consulta ListEventSourceMappings AWS CLICommand Reference.

`list-function-event-invoke-configs`

Il seguente esempio di codice mostra come utilizzare`list-function-event-invoke-configs`.

AWS CLI

Come visualizzare un elenco di configurazioni di invocazione asincrona

L'esempio `list-function-event-invoke-configs` seguente elenca le configurazioni di invocazione asincrona per la funzione specificata.

```
aws lambda list-function-event-invoke-configs \
    --function-name my-function
```

Output:

```
{
    "FunctionEventInvokeConfigs": [
        {
            "LastModified": 1577824406.719,
            "FunctionArn": "arn:aws:lambda:us-east-2:123456789012:function:my-
function:GREEN",
            "MaximumRetryAttempts": 2,
            "MaximumEventAgeInSeconds": 1800
        },
```

```
        {
            "LastModified": 1577824396.653,
            "FunctionArn": "arn:aws:lambda:us-east-2:123456789012:function:my-
function:BLUE",
            "MaximumRetryAttempts": 0,
            "MaximumEventAgeInSeconds": 3600
        }
    ]
}
```

- Per i dettagli sull'API, consulta ListFunctionEventInvokeConfigs AWS CLICommand Reference.

`list-functions`

Il seguente esempio di codice mostra come utilizzare`list-functions`.

AWS CLI

Come recuperare un elenco di funzioni Lambda

L'esempio di `list-functions` seguente visualizza un elenco di tutte le funzioni per l'utente attuale.

```
aws lambda list-functions
```

Output:

```
{
    "Functions": [
        {
            "TracingConfig": {
                "Mode": "PassThrough"
            },
            "Version": "$LATEST",
            "CodeSha256": "dBG9m8SGdmlEjw/JYXlhhvCrAv5TxvXsbL/RMr0fT/I=",
            "FunctionName": "helloworld",
            "MemorySize": 128,
            "RevisionId": "1718e831-badf-4253-9518-d0644210af7b",
            "CodeSize": 294,
            "FunctionArn": "arn:aws:lambda:us-
west-2:123456789012:function:helloworld",
            "Handler": "helloworld.handler",
```

```
            "Role": "arn:aws:iam::123456789012:role/service-role/MyTestFunction-
role-zgur6bf4",
            "Timeout": 3,
            "LastModified": "2025-09-23T18:32:33.857+0000",
            "Runtime": "nodejs22.x",
            "Description": ""
        },
        {
            "TracingConfig": {
                "Mode": "PassThrough"
            },
            "Version": "$LATEST",
            "CodeSha256": "sU0cJ2/hOZevwV/lTxCuQqK3gDZP3i8gUoqUUVRmY6E=",
            "FunctionName": "my-function",
            "VpcConfig": {
                "SubnetIds": [],
                "VpcId": "",
                "SecurityGroupIds": []
            },
            "MemorySize": 256,
            "RevisionId": "93017fc9-59cb-41dc-901b-4845ce4bf668",
            "CodeSize": 266,
            "FunctionArn": "arn:aws:lambda:us-west-2:123456789012:function:my-
function",
            "Handler": "index.handler",
            "Role": "arn:aws:iam::123456789012:role/service-role/helloWorldPython-
role-uy3l9qyq",
            "Timeout": 3,
            "LastModified": "2025-10-01T16:47:28.490+0000",
            "Runtime": "nodejs22.x",
            "Description": ""
        },
        {
            "Layers": [
                {
                    "CodeSize": 41784542,
                    "Arn": "arn:aws:lambda:us-west-2:420165488524:layer:AWSLambda-
Python37-SciPy1x:2"
                },
                {
                    "CodeSize": 4121,
                    "Arn": "arn:aws:lambda:us-
west-2:123456789012:layer:pythonLayer:1"
                }
```

```
            ],
            "TracingConfig": {
                "Mode": "PassThrough"
            },
            "Version": "$LATEST",
            "CodeSha256": "ZQukCqxtkqFgyF2cU41Avj99TKQ/hNihPtDtRcc08mI=",
            "FunctionName": "my-python-function",
            "VpcConfig": {
                "SubnetIds": [],
                "VpcId": "",
                "SecurityGroupIds": []
            },
            "MemorySize": 128,
            "RevisionId": "80b4eabc-acf7-4ea8-919a-e874c213707d",
            "CodeSize": 299,
            "FunctionArn": "arn:aws:lambda:us-west-2:123456789012:function:my-
python-function",
            "Handler": "lambda_function.lambda_handler",
            "Role": "arn:aws:iam::123456789012:role/service-role/my-python-function-
role-z5g7dr6n",
            "Timeout": 3,
            "LastModified": "2025-10-01T19:40:41.643+0000",
            "Runtime": "python3.11",
            "Description": ""
        }
    ]
}
```

Per ulteriori informazioni, consulta Configurazione della memoria della funzione Lambda nella Guida per gli sviluppatori Lambda AWS .

- Per i dettagli sull'API, consulta ListFunctions AWS CLICommand Reference.

`list-layer-versions`

Il seguente esempio di codice mostra come utilizzare`list-layer-versions`.

AWS CLI

Per elencare le versioni di un layer AWS Lambda

L'esempio `list-layers-versions` seguente visualizza informazioni sulle versioni del livello denominato `my-layer`.

```
aws lambda list-layer-versions \
    --layer-name my-layer
```

Output:

```
{
    "Layers": [
        {
            "LayerVersionArn": "arn:aws:lambda:us-east-2:123456789012:layer:my-
layer:2",
            "Version": 2,
            "Description": "My layer",
            "CreatedDate": "2023-11-15T00:37:46.592+0000",
            "CompatibleRuntimes": [
                "python3.10",
                "python3.11"
            ]
        }
    ]
}
```

Per ulteriori informazioni, consulta Livelli di AWS Lambda nella Guida per gli sviluppatori di AWS Lambda.

- Per i dettagli sull'API, consulta ListLayerVersions AWS CLICommand Reference.

`list-layers`

Il seguente esempio di codice mostra come utilizzare`list-layers`.

AWS CLI

Come elencare i livelli compatibili con il runtime della funzione

L'esempio `list-layers` seguente visualizza informazioni sui livelli compatibili con il runtime di Python 3.11.

```
aws lambda list-layers \
    --compatible-runtime python3.11
```

Output:

```
{
    "Layers": [
        {
            "LayerName": "my-layer",
            "LayerArn": "arn:aws:lambda:us-east-2:123456789012:layer:my-layer",
            "LatestMatchingVersion": {
                "LayerVersionArn": "arn:aws:lambda:us-east-2:123456789012:layer:my-
layer:2",
                "Version": 2,
                "Description": "My layer",
                "CreatedDate": "2023-11-15T00:37:46.592+0000",
                "CompatibleRuntimes": [
                    "python3.10",
                    "python3.11"
                ]
            }
        }
    ]
}
```

Per ulteriori informazioni, consulta Livelli di AWS Lambda nella Guida per gli sviluppatori di AWS Lambda.

- Per i dettagli sull'API, consulta ListLayers AWS CLICommand Reference.

`list-provisioned-concurrency-configs`

Il seguente esempio di codice mostra come utilizzare`list-provisioned-concurrency-configs`.

AWS CLI

Come ottenere un elenco di configurazioni di simultaneità fornita

L'esempio `list-provisioned-concurrency-configs` seguente elenca le configurazioni di simultaneità fornita per la funzione specificata.

```
aws lambda list-provisioned-concurrency-configs \
    --function-name my-function
```

Output:

```
{
    "ProvisionedConcurrencyConfigs": [
        {
            "FunctionArn": "arn:aws:lambda:us-east-2:123456789012:function:my-
function:GREEN",
            "RequestedProvisionedConcurrentExecutions": 100,
            "AvailableProvisionedConcurrentExecutions": 100,
            "AllocatedProvisionedConcurrentExecutions": 100,
            "Status": "READY",
            "LastModified": "2019-12-31T20:29:00+0000"
        },
        {
            "FunctionArn": "arn:aws:lambda:us-east-2:123456789012:function:my-
function:BLUE",
            "RequestedProvisionedConcurrentExecutions": 100,
            "AvailableProvisionedConcurrentExecutions": 100,
            "AllocatedProvisionedConcurrentExecutions": 100,
            "Status": "READY",
            "LastModified": "2019-12-31T20:28:49+0000"
        }
    ]
}
```

- Per i dettagli sull'API, consulta ListProvisionedConcurrencyConfigs AWS CLICommand Reference.

`list-tags`

Il seguente esempio di codice mostra come utilizzare`list-tags`.

AWS CLI

Come recuperare l'elenco di tag per una funzione Lambda

L'esempio `list-tags` seguente visualizza i tag collegati alla funzione Lambda `my-function`.

```
aws lambda list-tags \
    --resource arn:aws:lambda:us-west-2:123456789012:function:my-function
```

Output:

```
{
```

```
    "Tags": {
        "Category": "Web Tools",
        "Department": "Sales"
    }
}
```

Per ulteriori informazioni, consulta Tagging delle funzioni Lambda nella Guida per gli sviluppatori di AWS Lambda.

- Per i dettagli sull'API, consulta ListTags AWS CLICommand Reference.

`list-versions-by-function`

Il seguente esempio di codice mostra come utilizzare`list-versions-by-function`.

AWS CLI

Come recuperare un elenco delle versioni di una funzione

L'esempio `list-versions-by-function` seguente visualizza l'elenco di versioni per la funzione Lambda `my-function`.

```
aws lambda list-versions-by-function \
    --function-name my-function
```

Output:

```
{
    "Versions": [
        {
            "TracingConfig": {
                "Mode": "PassThrough"
            },
            "Version": "$LATEST",
            "CodeSha256": "sU0cJ2/hOZevwV/lTxCuQqK3gDZP3i8gUoqUUVRmY6E=",
            "FunctionName": "my-function",
            "VpcConfig": {
                "SubnetIds": [],
                "VpcId": "",
                "SecurityGroupIds": []
            },
            "MemorySize": 256,
            "RevisionId": "93017fc9-59cb-41dc-901b-4845ce4bf668",
```

```
            "CodeSize": 266,
            "FunctionArn": "arn:aws:lambda:us-west-2:123456789012:function:my-
function:$LATEST",
            "Handler": "index.handler",
            "Role": "arn:aws:iam::123456789012:role/service-role/helloWorldPython-
role-uy3l9qyq",
            "Timeout": 3,
            "LastModified": "2019-10-01T16:47:28.490+0000",
            "Runtime": "nodejs10.x",
            "Description": ""
        },
        {
            "TracingConfig": {
                "Mode": "PassThrough"
            },
            "Version": "1",
            "CodeSha256": "5tT2qgzYUHoqwR616pZ2dpkn/0J1FrzJmlKidWaaCgk=",
            "FunctionName": "my-function",
            "VpcConfig": {
                "SubnetIds": [],
                "VpcId": "",
                "SecurityGroupIds": []
            },
            "MemorySize": 256,
            "RevisionId": "949c8914-012e-4795-998c-e467121951b1",
            "CodeSize": 304,
            "FunctionArn": "arn:aws:lambda:us-west-2:123456789012:function:my-
function:1",
            "Handler": "index.handler",
            "Role": "arn:aws:iam::123456789012:role/service-role/helloWorldPython-
role-uy3l9qyq",
            "Timeout": 3,
            "LastModified": "2019-09-26T20:28:40.438+0000",
            "Runtime": "nodejs10.x",
            "Description": "new version"
        },
        {
            "TracingConfig": {
                "Mode": "PassThrough"
            },
            "Version": "2",
            "CodeSha256": "sU0cJ2/hOZevwV/lTxCuQqK3gDZP3i8gUoqUUVRmY6E=",
            "FunctionName": "my-function",
            "VpcConfig": {
```

```
                "SubnetIds": [],
                "VpcId": "",
                "SecurityGroupIds": []
            },
            "MemorySize": 256,
            "RevisionId": "cd669f21-0f3d-4e1c-9566-948837f2e2ea",
            "CodeSize": 266,
            "FunctionArn": "arn:aws:lambda:us-west-2:123456789012:function:my-
function:2",
            "Handler": "index.handler",
            "Role": "arn:aws:iam::123456789012:role/service-role/helloWorldPython-
role-uy3l9qyq",
            "Timeout": 3,
            "LastModified": "2019-10-01T16:47:28.490+0000",
            "Runtime": "nodejs10.x",
            "Description": "newer version"
        }
    ]
}
```

Per ulteriori informazioni, consulta Configurazione degli alias delle funzioni AWS Lambda nella AWS Lambda Developer Guide.

- Per i dettagli sull'API, consulta ListVersionsByFunctionCommand Reference.AWS CLI

`publish-layer-version`

Il seguente esempio di codice mostra come utilizzare`publish-layer-version`.

AWS CLI

Come creare una versione del livello Lambda

L'esempio `publish-layer-version` seguente crea una nuova versione del livello di libreria Python. Il comando recupera il contenuto del livello in un file denominato `layer.zip` nel bucket S3 specificato.

```
aws lambda publish-layer-version \
    --layer-name my-layer \
    --description "My Python layer" \
    --license-info "MIT" \
    --content S3Bucket=lambda-layers-us-west-2-123456789012,S3Key=layer.zip \
```

```
    --compatible-runtimes python3.10 python3.11
```

Output:

```
{
    "Content": {
        "Location": "https://awslambda-us-west-2-layers.s3.us-west-2.amazonaws.com/
snapshots/123456789012/my-layer-4aaa2fbb-ff77-4b0a-ad92-5b78a716a96a?
versionId=27iWyA73cCAYqyH...",
        "CodeSha256": "tv9jJO+rPbXUUXuRKi7CwHzKtLDkDRJLB3cC3Z/ouXo=",
        "CodeSize": 169
    },
    "LayerArn": "arn:aws:lambda:us-west-2:123456789012:layer:my-layer",
    "LayerVersionArn": "arn:aws:lambda:us-west-2:123456789012:layer:my-layer:1",
    "Description": "My Python layer",
    "CreatedDate": "2023-11-14T23:03:52.894+0000",
    "Version": 1,
    "LicenseInfo": "MIT",
    "CompatibleRuntimes": [
        "python3.10",
        "python3.11"
    ]
}
```

Per ulteriori informazioni, consulta Livelli di AWS Lambda nella Guida per gli sviluppatori di AWS Lambda.

- Per i dettagli sull'API, consulta PublishLayerVersion AWS CLICommand Reference.

`publish-version`

Il seguente esempio di codice mostra come utilizzare`publish-version`.

AWS CLI

Come pubblicare una nuova versione di una funzione

L'esempio `publish-version` seguente illustra la pubblicazione di una nuova versione della funzione Lambda `my-function`.

```
aws lambda publish-version \
    --function-name my-function
```

Output:

```
{
    "TracingConfig": {
        "Mode": "PassThrough"
    },
    "CodeSha256": "dBG9m8SGdmlEjw/JYXlhhvCrAv5TxvXsbL/RMr0fT/I=",
    "FunctionName": "my-function",
    "CodeSize": 294,
    "RevisionId": "f31d3d39-cc63-4520-97d4-43cd44c94c20",
    "MemorySize": 128,
    "FunctionArn": "arn:aws:lambda:us-west-2:123456789012:function:my-function:3",
    "Version": "2",
    "Role": "arn:aws:iam::123456789012:role/service-role/MyTestFunction-role-
zgur6bf4",
    "Timeout": 3,
    "LastModified": "2019-09-23T18:32:33.857+0000",
    "Handler": "my-function.handler",
    "Runtime": "nodejs10.x",
    "Description": ""
}
```

Per ulteriori informazioni, consulta Configurazione degli alias delle funzioni AWS Lambda nella AWS Lambda Developer Guide.

- Per i dettagli sull'API, consulta PublishVersionCommand Reference.AWS CLI

`put-function-concurrency`

Il seguente esempio di codice mostra come utilizzare`put-function-concurrency`.

AWS CLI

Come configurare un limite di simultaneità riservata per una funzione

L'esempio `put-function-concurrency` seguente configura 100 esecuzioni simultanee riservate per la funzione `my-function`.

```
aws lambda put-function-concurrency \
    --function-name  my-function  \
    --reserved-concurrent-executions 100
```

Output:

```
{
    "ReservedConcurrentExecutions": 100
}
```

Per ulteriori informazioni, consulta Reserving Concurrency for a Lambda Function nella Guida per gli sviluppatori di AWS Lambda.

- Per i dettagli sull'API, consulta PutFunctionConcurrency AWS CLICommand Reference.

`put-function-event-invoke-config`

Il seguente esempio di codice mostra come utilizzare`put-function-event-invoke-config`.

AWS CLI

Come configurare la gestione degli errori per l'invocazione asincrona

L'esempio `put-function-event-invoke-config` seguente imposta una durata massima degli eventi pari a un'ora e disattiva i nuovi tentativi per la funzione specificata.

```
aws lambda put-function-event-invoke-config \
    --function-name my-function \
    --maximum-event-age-in-seconds 3600 \
    --maximum-retry-attempts 0
```

Output:

```
{
    "LastModified": 1573686021.479,
    "FunctionArn": "arn:aws:lambda:us-east-2:123456789012:function:my-function:
$LATEST",
    "MaximumRetryAttempts": 0,
    "MaximumEventAgeInSeconds": 3600,
    "DestinationConfig": {
        "OnSuccess": {},
        "OnFailure": {}
    }
}
```

- Per i dettagli sull'API, consulta PutFunctionEventInvokeConfig AWS CLICommand Reference.

`put-provisioned-concurrency-config`

Il seguente esempio di codice mostra come utilizzare`put-provisioned-concurrency-config`.

AWS CLI

Come allocare la simultaneità fornita

L'esempio `put-provisioned-concurrency-config` seguente alloca 100 simultaneità fornita per l'alias BLUE della funzione specificata.

```
aws lambda put-provisioned-concurrency-config \
    --function-name my-function \
    --qualifier BLUE \
    --provisioned-concurrent-executions 100
```

Output:

```
{
    "Requested ProvisionedConcurrentExecutions": 100,
    "Allocated ProvisionedConcurrentExecutions": 0,
    "Status": "IN_PROGRESS",
    "LastModified": "2019-11-21T19:32:12+0000"
}
```

- Per i dettagli sull'API, consulta PutProvisionedConcurrencyConfig AWS CLICommand Reference.

`remove-layer-version-permission`

Il seguente esempio di codice mostra come utilizzare`remove-layer-version-permission`.

AWS CLI

Come eliminare le autorizzazioni relative alla versione del livello

L'esempio `remove-layer-version-permission` seguente elimina l'autorizzazione concessa a un account per configurare una versione del livello.

```
aws lambda remove-layer-version-permission \
    --layer-name my-layer \
    --statement-id xaccount \
```

```
    --version-number 1
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Livelli di AWS Lambda nella Guida per gli sviluppatori di AWS Lambda.

- Per i dettagli sull'API, consulta RemoveLayerVersionPermission AWS CLICommand Reference.

`remove-permission`

Il seguente esempio di codice mostra come utilizzare`remove-permission`.

AWS CLI

Come rimuovere le autorizzazioni da una funzione Lambda esistente

L'esempio `remove-permission` seguente concede rimuove l'autorizzazione per invocare una funzione denominata `my-function`.

```
aws lambda remove-permission \
    --function-name my-function \
    --statement-id sns
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Using Resource-based Policies for Lambda nella AWS Lambda Developer Guide.AWS

- Per i dettagli sull'API, consulta Command Reference. RemovePermissionAWS CLI

`tag-resource`

Il seguente esempio di codice mostra come utilizzare`tag-resource`.

AWS CLI

Come aggiungere tag a una funzione Lambda esistente

L'esempio `tag-resource` seguente aggiunge un tag con il nome della chiave `DEPARTMENT` e un valore di `Department A` alla funzione Lambda specificata.

```
aws lambda tag-resource \
```

```
    --resource arn:aws:lambda:us-west-2:123456789012:function:my-function \
    --tags "DEPARTMENT=Department A"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Tagging Lambda Functions nella Guida per gli sviluppatori di AWS Lambda.

- Per i dettagli sull'API, consulta TagResource AWS CLICommand Reference.

`untag-resource`

Il seguente esempio di codice mostra come utilizzare`untag-resource`.

AWS CLI

Come rimuovere i tag da una funzione Lambda esistente

L'esempio `untag-resource` seguente rimuove il tag con il tag `DEPARTMENT` per il nome della chiave dalla funzione Lambda `my-function`.

```
aws lambda untag-resource \
    --resource arn:aws:lambda:us-west-2:123456789012:function:my-function \
    --tag-keys DEPARTMENT
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Tagging Lambda Functions nella Guida per gli sviluppatori di AWS Lambda.

- Per i dettagli sull'API, consulta UntagResource AWS CLICommand Reference.

`update-alias`

Il seguente esempio di codice mostra come utilizzare`update-alias`.

AWS CLI

Come aggiornare l'alias di una funzione

L'esempio `update-alias` seguente aggiorna l'alias denominato `LIVE` che punti alla versione 3 della funzione Lambda `my-function`.

```
aws lambda update-alias \
    --function-name my-function \
    --function-version 3 \
    --name LIVE
```

Output:

```
{
    "FunctionVersion": "3",
    "Name": "LIVE",
    "AliasArn": "arn:aws:lambda:us-west-2:123456789012:function:my-function:LIVE",
    "RevisionId": "594f41fb-b85f-4c20-95c7-6ca5f2a92c93",
    "Description": "alias for live version of function"
}
```

Per ulteriori informazioni, consulta Configurazione degli alias delle funzioni AWS Lambda nella AWS Lambda Developer Guide.

- Per i dettagli sull'API, consulta UpdateAliasCommand Reference.AWS CLI

`update-event-source-mapping`

Il seguente esempio di codice mostra come utilizzare`update-event-source-mapping`.

AWS CLI

Per aggiornare la mappatura tra un'origine di eventi e una funzione AWS Lambda

L'esempio `update-event-source-mapping` seguente aggiorna la dimensione del batch a 8 nella mappatura specificata.

```
aws lambda update-event-source-mapping \
    --uuid  "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE" \
    --batch-size 8
```

Output:

```
{
    "UUID": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE",
    "StateTransitionReason": "USER_INITIATED",
    "LastModified": 1569284520.333,
    "BatchSize": 8,
```

```
    "State": "Updating",
    "FunctionArn": "arn:aws:lambda:us-west-2:123456789012:function:my-function",
    "EventSourceArn": "arn:aws:sqs:us-west-2:123456789012:mySQSqueue"
}
```

Per ulteriori informazioni, consulta Strumento di mappatura dell'origine degli eventi AWS Lambda nella Guida per gli sviluppatori di AWS Lambda.

- Per i dettagli sull'API, consulta AWS CLI Command UpdateEventSourceMappingReference.

`update-function-code`

Il seguente esempio di codice mostra come utilizzare`update-function-code`.

AWS CLI

Come aggiornare il codice di una funzione Lambda

L'esempio di `update-function-code` seguente sostituisce il codice della versione ($LATEST) non pubblicata della funzione `my-function` con i contenuti del file zip specificato.

```
aws lambda update-function-code \
    --function-name  my-function \
    --zip-file fileb://my-function.zip
```

Output:

```
{
    "FunctionName": "my-function",
    "LastModified": "2019-09-26T20:28:40.438+0000",
    "RevisionId": "e52502d4-9320-4688-9cd6-152a6ab7490d",
    "MemorySize": 256,
    "Version": "$LATEST",
    "Role": "arn:aws:iam::123456789012:role/service-role/my-function-role-uy3l9qyq",
    "Timeout": 3,
    "Runtime": "nodejs10.x",
    "TracingConfig": {
        "Mode": "PassThrough"
    },
    "CodeSha256": "5tT2qgzYUHaqwR716pZ2dpkn/0J1FrzJmlKidWoaCgk=",
    "Description": "",
    "VpcConfig": {
        "SubnetIds": [],
```

```
        "VpcId": "",
        "SecurityGroupIds": []
    },
    "CodeSize": 304,
    "FunctionArn": "arn:aws:lambda:us-west-2:123456789012:function:my-function",
    "Handler": "index.handler"
}
```

Per ulteriori informazioni, consulta Configurazione delle funzioni Lambda AWS nella Guida per gli sviluppatori di AWS .

- Per i dettagli sull'API, consulta UpdateFunctionCode AWS CLICommand Reference.

`update-function-configuration`

Il seguente esempio di codice mostra come utilizzare`update-function-configuration`.

AWS CLI

Come modificare la configurazione di una funzione

L'esempio di `update-function-configuration` seguente modifica la dimensione della memoria in 256 MB per la versione non pubblicata ($LATEST) della funzione `my-function`.

```
aws lambda update-function-configuration \
    --function-name  my-function \
    --memory-size 256
```

Output:

```
{
    "FunctionName": "my-function",
    "LastModified": "2019-09-26T20:28:40.438+0000",
    "RevisionId": "e52502d4-9320-4688-9cd6-152a6ab7490d",
    "MemorySize": 256,
    "Version": "$LATEST",
    "Role": "arn:aws:iam::123456789012:role/service-role/my-function-role-uy3l9qyq",
    "Timeout": 3,
    "Runtime": "nodejs10.x",
    "TracingConfig": {
        "Mode": "PassThrough"
    },
    "CodeSha256": "5tT2qgzYUHaqwR716pZ2dpkn/0J1FrzJmlKidWoaCgk=",
```

```
    "Description": "",
    "VpcConfig": {
        "SubnetIds": [],
        "VpcId": "",
        "SecurityGroupIds": []
    },
    "CodeSize": 304,
    "FunctionArn": "arn:aws:lambda:us-west-2:123456789012:function:my-function",
    "Handler": "index.handler"
}
```

Per ulteriori informazioni, consulta Configurazione delle funzioni Lambda AWS nella Guida per gli sviluppatori di AWS .

- Per i dettagli sull'API, consulta UpdateFunctionConfiguration AWS CLICommand Reference.

`update-function-event-invoke-config`

Il seguente esempio di codice mostra come utilizzare`update-function-event-invoke-config`.

AWS CLI

Come aggiornare una configurazione di invocazione asincrona

L'esempio `update-function-event-invoke-config` seguente aggiunge una destinazione on-failure alla configurazione di invocazione asincrona esistente per la funzione specificata.

```
aws lambda update-function-event-invoke-config \
    --function-name my-function \
    --destination-config '{"OnFailure":{"Destination": "arn:aws:sqs:us-
east-2:123456789012:destination"}}'
```

Output:

```
{
    "LastModified": 1573687896.493,
    "FunctionArn": "arn:aws:lambda:us-east-2:123456789012:function:my-function:
$LATEST",
    "MaximumRetryAttempts": 0,
    "MaximumEventAgeInSeconds": 3600,
    "DestinationConfig": {
        "OnSuccess": {},
        "OnFailure": {
```

```
            "Destination": "arn:aws:sqs:us-east-2:123456789012:destination"
        }
    }
}
```

- Per i dettagli sull'API, consulta UpdateFunctionEventInvokeConfig AWS CLICommand Reference.

Esempi di License Manager che utilizzano AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with License Manager.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`create-license-configuration`

Il seguente esempio di codice mostra come utilizzare`create-license-configuration`.

AWS CLI

Esempio 1: come creare una configurazione di licenza

L'esempio `create-license-configuration` seguente crea una configurazione di licenza con un limite rigido di 10 core.

```
aws license-manager create-license-configuration --name my-license-configuration \
    --license-counting-type Core \
    --license-count 10 \
    --license-count-hard-limit
```

Output:

```
{
  "LicenseConfigurationArn": "arn:aws:license-manager:us-
west-2:123456789012:license-configuration:lic-6eb6586f508a786a2ba41EXAMPLE1111"
}
```

Esempio 2: come creare una configurazione di licenza

L'`create-license-configuration`esempio seguente crea una configurazione di licenza con un limite flessibile di 100 vCPUs. Utilizza una regola per abilitare l'ottimizzazione della vCPU.

```
aws license-manager create-license-configuration --name my-license-configuration
    --license-counting-type vCPU \
    --license-count 100 \
    --license-rules "#honorVcpuOptimization=true"
```

Output:

```
{
  "LicenseConfigurationArn": "arn:aws:license-manager:us-
west-2:123456789012:license-configuration:lic-6eb6586f508a786a2ba41EXAMPLE2222"
}
```

- Per i dettagli sull'API, consulta CreateLicenseConfiguration AWS CLICommand Reference.

`delete-license-configuration`

Il seguente esempio di codice mostra come utilizzare`delete-license-configuration`.

AWS CLI

Come eliminare una configurazione della licenza

L'esempio `delete-license-configuration` seguente elimina la configurazione di licenza specificata.

```
aws license-manager delete-license-configuration \
    --license-configuration-arn arn:aws:license-manager:us-
west-2:123456789012:license-configuration:lic-6eb6586f508a786a2ba4f56c1EXAMPLE
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta DeleteLicenseConfiguration AWS CLICommand Reference.

`get-license-configuration`

Il seguente esempio di codice mostra come utilizzare`get-license-configuration`.

AWS CLI

Come ottenere informazioni sulla configurazione della licenza

L'esempio `get-license-configuration` seguente mostra i dettagli della configurazione di licenza specificata.

```
aws license-manager get-license-configuration \
    --license-configuration-arn arn:aws:license-manager:us-
west-2:123456789012:license-configuration:lic-38b658717b87478aaa7c00883EXAMPLE
```

Output:

```
{
    "LicenseConfigurationId": "lic-38b658717b87478aaa7c00883EXAMPLE",
    "LicenseConfigurationArn": "arn:aws:license-manager:us-
west-2:123456789012:license-configuration:lic-38b658717b87478aaa7c00883EXAMPLE",
    "Name": "my-license-configuration",
    "LicenseCountingType": "vCPU",
    "LicenseRules": [],
    "LicenseCountHardLimit": false,
    "ConsumedLicenses": 0,
    "Status": "AVAILABLE",
    "OwnerAccountId": "123456789012",
    "ConsumedLicenseSummaryList": [
        {
            "ResourceType": "EC2_INSTANCE",
            "ConsumedLicenses": 0
        },
        {
            "ResourceType": "EC2_HOST",
            "ConsumedLicenses": 0
        },
        {
            "ResourceType": "SYSTEMS_MANAGER_MANAGED_INSTANCE",
```

```
            "ConsumedLicenses": 0
        }
    ],
    "ManagedResourceSummaryList": [
        {
            "ResourceType": "EC2_INSTANCE",
            "AssociationCount": 0
        },
        {
            "ResourceType": "EC2_HOST",
            "AssociationCount": 0
        },
        {
            "ResourceType": "EC2_AMI",
            "AssociationCount": 2
        },
        {
            "ResourceType": "SYSTEMS_MANAGER_MANAGED_INSTANCE",
            "AssociationCount": 0
        }
    ]
}
```

- Per i dettagli sull'API, consulta GetLicenseConfiguration AWS CLICommand Reference.

`get-service-settings`

Il seguente esempio di codice mostra come utilizzare`get-service-settings`.

AWS CLI

Come ottenere le impostazioni dello Strumento di gestione delle licenze

L'esempio `get-service-settings` seguente visualizza le impostazioni del servizio per lo Strumento di gestione delle licenze nella Regione corrente.

```
aws license-manager get-service-settings
```

Di seguito viene illustrato un esempio di output se il rilevamento di risorse tra account è disabilitato.

```
{
```

```
    "OrganizationConfiguration": {
        "EnableIntegration": false
    },
    "EnableCrossAccountsDiscovery": false
}
```

Di seguito viene mostrato un esempio di output se il rilevamento di risorse tra account è abilitato.

```
{
    "S3BucketArn": "arn:aws:s3:::aws-license-manager-service-c22d6279-35c4-47c4-bb",
    "OrganizationConfiguration": {
        "EnableIntegration": true
    },
    "EnableCrossAccountsDiscovery": true
}
```

- Per i dettagli sull'API, consulta GetServiceSettings AWS CLICommand Reference.

`list-associations-for-license-configuration`

Il seguente esempio di codice mostra come utilizzare`list-associations-for-license-configuration`.

AWS CLI

Come ottenere le associazioni per la configurazione di una licenza

L'esempio `list-associations-for-license-configuration` seguente visualizza informazioni dettagliate per le associazioni della configurazione di licenza specificata.

```
aws license-manager list-associations-for-license-configuration \
    --license-configuration-arn arn:aws:license-manager:us-
west-2:123456789012:license-configuration:lic-38b658717b87478aaa7c00883EXAMPLE
```

Output:

```
{
    "LicenseConfigurationAssociations": [
        {
            "ResourceArn": "arn:aws:ec2:us-west-2::image/ami-1234567890abcdef0",
            "ResourceType": "EC2_AMI",
            "ResourceOwnerId": "123456789012",
```

```
            "AssociationTime": 1568825118.617
        },
        {
            "ResourceArn": "arn:aws:ec2:us-west-2::image/ami-0abcdef1234567890",
            "ResourceType": "EC2_AMI",
            "ResourceOwnerId": "123456789012",
            "AssociationTime": 1568825118.946
        }
    ]
}
```

- Per i dettagli sull'API, consulta ListAssociationsForLicenseConfiguration AWS CLICommand Reference.

`list-license-configurations`

Il seguente esempio di codice mostra come utilizzare`list-license-configurations`.

AWS CLI

Esempio 1: come elencare tutte le configurazioni delle licenze

L'esempio `list-license-configurations` seguente elenca tutte le configurazioni di licenza.

```
aws license-manager list-license-configurations
```

Output:

```
{
    "LicenseConfigurations": [
        {
            "LicenseConfigurationId": "lic-6eb6586f508a786a2ba4f56c1EXAMPLE",
            "LicenseConfigurationArn": "arn:aws:license-manager:us-
west-2:123456789012:license-configuration:lic-6eb6586f508a786a2ba4f56c1EXAMPLE",
            "Name": "my-license-configuration",
            "LicenseCountingType": "Core",
            "LicenseRules": [],
            "LicenseCount": 10,
            "LicenseCountHardLimit": true,
            "ConsumedLicenses": 0,
            "Status": "AVAILABLE",
            "OwnerAccountId": "123456789012",
```

```
            "ConsumedLicenseSummaryList": [
                {
                    "ResourceType": "EC2_INSTANCE",
                    "ConsumedLicenses": 0
                },
                {
                    "ResourceType": "EC2_HOST",
                    "ConsumedLicenses": 0
                },
                {
                    "ResourceType": "SYSTEMS_MANAGER_MANAGED_INSTANCE",
                    "ConsumedLicenses": 0
                }
            ],
            "ManagedResourceSummaryList": [
                {
                    "ResourceType": "EC2_INSTANCE",
                    "AssociationCount": 0
                },
                {
                    "ResourceType": "EC2_HOST",
                    "AssociationCount": 0
                },
                {
                    "ResourceType": "EC2_AMI",
                    "AssociationCount": 0
                },
                {
                    "ResourceType": "SYSTEMS_MANAGER_MANAGED_INSTANCE",
                    "AssociationCount": 0
                }
            ]
        },
        {
            ...
        }
    ]
}
```

Esempio 2: come elencare una configurazione della licenza specifica

L'esempio `list-license-configurations` seguente elenca solo la configurazione di licenza specificata.

```
aws license-manager list-license-configurations \
    --license-configuration-arns arn:aws:license-manager:us-
west-2:123456789012:license-configuration:lic-38b658717b87478aaa7c00883EXAMPLE
```

- Per i dettagli sull'API, consulta ListLicenseConfigurations AWS CLICommand Reference.

`list-license-specifications-for-resource`

Il seguente esempio di codice mostra come utilizzare`list-license-specifications-for-resource`.

AWS CLI

Come elencare le configurazioni delle licenze per una risorsa

L'esempio `list-license-specifications-for-resource` seguente elenca le configurazioni di licenza associate all'Amazon Machine Image (AMI) specificata.

```
aws license-manager list-license-specifications-for-resource \
    --resource-arn arn:aws:ec2:us-west-2::image/ami-1234567890abcdef0
```

Output:

```
{
    "LicenseConfigurationArn": "arn:aws:license-manager:us-
west-2:123456789012:license-configuration:lic-38b658717b87478aaa7c00883EXAMPLE"
}
```

- Per i dettagli sull'API, consulta ListLicenseSpecificationsForResource AWS CLICommand Reference.

`list-resource-inventory`

Il seguente esempio di codice mostra come utilizzare`list-resource-inventory`.

AWS CLI

Come elencare le risorse nell'inventario delle risorse

L'esempio `list-resource-inventory` seguente elenca le risorse gestite utilizzando l'inventario di Systems Manager.

```
aws license-manager list-resource-inventory
```

Output:

```
{
    "ResourceInventoryList": [
        {
            "Platform": "Red Hat Enterprise Linux Server",
            "ResourceType": "EC2Instance",
            "PlatformVersion": "7.4",
            "ResourceArn": "arn:aws:ec2:us-west-2:1234567890129:instance/
i-05d3cdfb05bd36376",
            "ResourceId": "i-05d3cdfb05bd36376",
            "ResourceOwningAccountId": "1234567890129"
        },
        {
            "Platform": "Amazon Linux",
            "ResourceType": "EC2Instance",
            "PlatformVersion": "2",
            "ResourceArn": "arn:aws:ec2:us-west-2:1234567890129:instance/
i-0b1d036cfd4594808",
            "ResourceId": "i-0b1d036cfd4594808",
            "ResourceOwningAccountId": "1234567890129"
        },
        {
            "Platform": "Microsoft Windows Server 2019 Datacenter",
            "ResourceType": "EC2Instance",
            "PlatformVersion": "10.0.17763",
            "ResourceArn": "arn:aws:ec2:us-west-2:1234567890129:instance/
i-0cdb3b54a2a8246ad",
            "ResourceId": "i-0cdb3b54a2a8246ad",
            "ResourceOwningAccountId": "1234567890129"
        }
    ]
}
```

- Per i dettagli sull'API, consulta ListResourceInventory AWS CLICommand Reference.

`list-tags-for-resource`

Il seguente esempio di codice mostra come utilizzare`list-tags-for-resource`.

AWS CLI

Come elencare i tag per una configurazione della licenza

L'esempio `list-tags-for-resource` seguente elenca i tag per la configurazione di licenza specificata.

```
aws license-manager list-tags-for-resource \
    --resource-arn arn:aws:license-manager:us-west-2:123456789012:license-configuration:lic-6eb6586f508a786a2ba4f56c1EXAMPLE
```

Output:

```
{
    "Tags": [
        {
            "Key": "project",
            "Value": "lima"
        }
    ]
}
```

- Per i dettagli sull'API, consulta ListTagsForResource AWS CLICommand Reference.

`list-usage-for-license-configuration`

Il seguente esempio di codice mostra come utilizzare`list-usage-for-license-configuration`.

AWS CLI

Come elencare le licenze in uso per una configurazione della licenza

L'esempio `list-usage-for-license-configuration` seguente elenca le informazioni sulle risorse che utilizzano le licenze per la configurazione di licenza specificata. Ad esempio, se il tipo di licenza è vCPU, tutte le istanze impiegano una licenza per vCPU.

```
aws license-manager list-usage-for-license-configuration \
    --license-configuration-arn arn:aws:license-manager:us-west-2:123456789012:license-configuration:lic-38b658717b87478aaa7c00883EXAMPLE
```

Output:

```
{
    "LicenseConfigurationUsageList": [
        {
            "ResourceArn": "arn:aws:ec2:us-west-2:123456789012:instance/
i-04a636d18e83cfacb",
            "ResourceType": "EC2_INSTANCE",
            "ResourceStatus": "running",
            "ResourceOwnerId": "123456789012",
            "AssociationTime": 1570892850.519,
            "ConsumedLicenses": 2
        }
    ]
}
```

- Per i dettagli sull'API, consulta ListUsageForLicenseConfiguration AWS CLICommand Reference.

`tag-resource`

Il seguente esempio di codice mostra come utilizzare`tag-resource`.

AWS CLI

Come aggiungere un tag a una configurazione della licenza

L'esempio `tag-resource` seguente aggiunge il tag specificato (nome e valore di chiave) alla configurazione di licenza specificata.

```
aws license-manager tag-resource \
    --tags Key=project,Value=lima \
    --resource-arn arn:aws:license-manager:us-west-2:123456789012:license-
configuration:lic-6eb6586f508a786a2ba4f56c1EXAMPLE
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta TagResource AWS CLICommand Reference.

`untag-resource`

Il seguente esempio di codice mostra come utilizzare`untag-resource`.

AWS CLI

Come rimuovere i tag da una configurazione della licenza

L'esempio `untag-resource` seguente rimuove il tag specificato (nome di chiave e risorsa) dalla configurazione di licenza specificata.

```
aws license-manager untag-resource \
    --tag-keys project \
    --resource-arn arn:aws:license-manager:us-west-2:123456789012:license-
configuration:lic-6eb6586f508a786a2ba4f56c1EXAMPLE
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta UntagResource AWS CLICommand Reference.

`update-license-configuration`

Il seguente esempio di codice mostra come utilizzare`update-license-configuration`.

AWS CLI

Come aggiornare una configurazione della licenza

L'esempio `update-license-configuration` seguente aggiorna la configurazione di licenza specificata per rimuovere il limite rigido.

```
aws license-manager update-license-configuration \
    --no-license-count-hard-limit \
    --license-configuration-arn arn:aws:license-manager:us-
west-2:880185128111:license-configuration:lic-6eb6586f508a786a2ba4f56c1EXAMPLE
```

Questo comando non produce alcun output.

L'esempio `update-license-configuration` seguente aggiorna la configurazione di licenza specificata per modificarne lo stato in `DISABLED`.

```
aws license-manager update-license-configuration \
    --license-configuration-status DISABLED
    --license-configuration-arn arn:aws:license-manager:us-
west-2:880185128111:license-configuration:lic-6eb6586f508a786a2ba4f56c1EXAMPLE
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta UpdateLicenseConfiguration AWS CLICommand Reference.

`update-license-specifications-for-resource`

Il seguente esempio di codice mostra come utilizzare`update-license-specifications-for-resource`.

AWS CLI

Come aggiornare le configurazioni delle licenze per una risorsa

L'esempio `update-license-specifications-for-resource` seguente sostituisce la configurazione di licenza associata all'Amazon Machine Image (AMI) specificata rimuovendo una configurazione di licenza e aggiungendone un'altra.

```
aws license-manager update-license-specifications-for-resource \
    --resource-arn arn:aws:ec2:us-west-2::image/ami-1234567890abcdef0 \
    --remove-license-specifications LicenseConfigurationArn=arn:aws:license-
manager:us-west-2:123456789012:license-
configuration:lic-38b658717b87478aaa7c00883EXAMPLE \
    --add-license-specifications LicenseConfigurationArn=arn:aws:license-manager:us-
west-2:123456789012:license-configuration:lic-42b6deb06e5399a980d555927EXAMPLE
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta UpdateLicenseSpecificationsForResource AWS CLICommand Reference.

`update-service-settings`

Il seguente esempio di codice mostra come utilizzare`update-service-settings`.

AWS CLI

Come aggiornare le impostazioni dello Strumento di gestione delle licenze

L'`update-service-settings`esempio seguente abilita l'individuazione delle risorse tra account per License Manager nella AWS regione corrente. Il bucket Amazon S3 è lo strumento di sincronizzazione dei dati della risorsa necessario per l'inventario di Systems Manager.

```
aws license-manager update-service-settings \
    --organization-configuration EnableIntegration=true \
    --enable-cross-accounts-discovery \
    --s3-bucket-arn arn:aws:s3:::aws-license-manager-service-abcd1234EXAMPLE
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, vedere UpdateServiceSettingsin AWS CLI Command Reference.

Esempi di utilizzo di Lightsail AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface con Lightsail.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`allocate-static-ip`

Il seguente esempio di codice mostra come usare. `allocate-static-ip`

AWS CLI

Come creare un IP statico

L'esempio `allocate-static-ip` seguente crea l'IP statico specificato, che è possibile collegare a un'istanza.

```
aws lightsail allocate-static-ip \
    --static-ip-name StaticIp-1
```

Output:

```
{
    "operations": [
        {
            "id": "b5d06d13-2f19-4683-889f-dEXAMPLEed79",
            "resourceName": "StaticIp-1",
            "resourceType": "StaticIp",
            "createdAt": 1571071325.076,
            "location": {
                "availabilityZone": "all",
                "regionName": "us-west-2"
            },
            "isTerminal": true,
            "operationType": "AllocateStaticIp",
            "status": "Succeeded",
            "statusChangedAt": 1571071325.274
        }
    ]
}
```

- Per i dettagli sull'API, consulta AllocateStaticIp AWS CLICommand Reference.

`attach-disk`

Il seguente esempio di codice mostra come utilizzare`attach-disk`.

AWS CLI

Come collegare un disco di archiviazione a blocchi a un'istanza

L'esempio `attach-disk` seguente collega il disco `Disk-1` all'istanza `WordPress_Multisite-1` con il percorso del disco `/dev/xvdf`.

```
aws lightsail attach-disk \
    --disk-name Disk-1 \
    --disk-path /dev/xvdf \
    --instance-name WordPress_Multisite-1
```

Output:

```
{
```

```
    "operations": [
        {
            "id": "10a08267-19ce-43be-b913-6EXAMPLE7e80",
            "resourceName": "Disk-1",
            "resourceType": "Disk",
            "createdAt": 1571071465.472,
            "location": {
                "availabilityZone": "us-west-2a",
                "regionName": "us-west-2"
            },
            "isTerminal": false,
            "operationDetails": "WordPress_Multisite-1",
            "operationType": "AttachDisk",
            "status": "Started",
            "statusChangedAt": 1571071465.472
        },
        {
            "id": "2912c477-5295-4539-88c9-bEXAMPLEd1f0",
            "resourceName": "WordPress_Multisite-1",
            "resourceType": "Instance",
            "createdAt": 1571071465.474,
            "location": {
                "availabilityZone": "us-west-2a",
                "regionName": "us-west-2"
            },
            "isTerminal": false,
            "operationDetails": "Disk-1",
            "operationType": "AttachDisk",
            "status": "Started",
            "statusChangedAt": 1571071465.474
        }
    ]
}
```


- Per i dettagli sull'API, consulta AttachDisk AWS CLICommand Reference.

`attach-instances-to-load-balancer`

Il seguente esempio di codice mostra come utilizzare`attach-instances-to-load-balancer`.

AWS CLI

Come collegare le istanze a un bilanciatore del carico

L'esempio `attach-instances-to-load-balancer` seguente collega le istanze `MEAN-1`, `MEAN-2` e `MEAN-3` al bilanciatore del carico `LoadBalancer-1`.

```
aws lightsail attach-instances-to-load-balancer \
    --instance-names {"MEAN-1","MEAN-2","MEAN-3"} \
    --load-balancer-name LoadBalancer-1
```

Output:

```
{
    "operations": [
        {
            "id": "8055d19d-abb2-40b9-b527-1EXAMPLE3c7b",
            "resourceName": "LoadBalancer-1",
            "resourceType": "LoadBalancer",
            "createdAt": 1571071699.892,
            "location": {
                "availabilityZone": "all",
                "regionName": "us-west-2"
            },
            "isTerminal": false,
            "operationDetails": "MEAN-2",
            "operationType": "AttachInstancesToLoadBalancer",
            "status": "Started",
            "statusChangedAt": 1571071699.892
        },
        {
            "id": "c35048eb-8538-456a-a118-0EXAMPLEfb73",
            "resourceName": "MEAN-2",
            "resourceType": "Instance",
            "createdAt": 1571071699.887,
            "location": {
                "availabilityZone": "all",
                "regionName": "us-west-2"
            },
            "isTerminal": false,
            "operationDetails": "LoadBalancer-1",
            "operationType": "AttachInstancesToLoadBalancer",
            "status": "Started",
            "statusChangedAt": 1571071699.887
        },
        {
            "id": "910d09e0-adc5-4372-bc2e-0EXAMPLEd891",
```

```
            "resourceName": "LoadBalancer-1",
            "resourceType": "LoadBalancer",
            "createdAt": 1571071699.882,
            "location": {
                "availabilityZone": "all",
                "regionName": "us-west-2"
            },
            "isTerminal": false,
            "operationDetails": "MEAN-3",
            "operationType": "AttachInstancesToLoadBalancer",
            "status": "Started",
            "statusChangedAt": 1571071699.882
        },
        {
            "id": "178b18ac-43e8-478c-9bed-1EXAMPLE4755",
            "resourceName": "MEAN-3",
            "resourceType": "Instance",
            "createdAt": 1571071699.901,
            "location": {
                "availabilityZone": "all",
                "regionName": "us-west-2"
            },
            "isTerminal": false,
            "operationDetails": "LoadBalancer-1",
            "operationType": "AttachInstancesToLoadBalancer",
            "status": "Started",
            "statusChangedAt": 1571071699.901
        },
        {
            "id": "fb62536d-2a98-4190-a6fc-4EXAMPLE7470",
            "resourceName": "LoadBalancer-1",
            "resourceType": "LoadBalancer",
            "createdAt": 1571071699.885,
            "location": {
                "availabilityZone": "all",
                "regionName": "us-west-2"
            },
            "isTerminal": false,
            "operationDetails": "MEAN-1",
            "operationType": "AttachInstancesToLoadBalancer",
            "status": "Started",
            "statusChangedAt": 1571071699.885
        },
        {
```

```
            "id": "787dac0d-f98d-46c3-8571-3EXAMPLE5a85",
            "resourceName": "MEAN-1",
            "resourceType": "Instance",
            "createdAt": 1571071699.901,
            "location": {
                "availabilityZone": "all",
                "regionName": "us-west-2"
            },
            "isTerminal": false,
            "operationDetails": "LoadBalancer-1",
            "operationType": "AttachInstancesToLoadBalancer",
            "status": "Started",
            "statusChangedAt": 1571071699.901
        }
    ]
}
```

- Per i dettagli sull'API, consulta AttachInstancesToLoadBalancer AWS CLICommand Reference.

`attach-load-balancer-tls-certificate`

Il seguente esempio di codice mostra come utilizzare`attach-load-balancer-tls-certificate`.

AWS CLI

Come collegare un certificato TLS a un bilanciatore del carico

L'esempio `attach-load-balancer-tls-certificate` seguente collega il certificato TLS `Certificate2` al bilanciatore del carico `LoadBalancer-1`.

```
aws lightsail attach-load-balancer-tls-certificate \
    --certificate-name Certificate2 \
    --load-balancer-name LoadBalancer-1
```

Output:

```
{
    "operations": [
        {
            "id": "cf1ad6e3-3cbb-4b8a-a7f2-3EXAMPLEa118",
```

```
            "resourceName": "LoadBalancer-1",
            "resourceType": "LoadBalancer",
            "createdAt": 1571072255.416,
            "location": {
                "availabilityZone": "all",
                "regionName": "us-west-2"
            },
            "isTerminal": true,
            "operationDetails": "Certificate2",
            "operationType": "AttachLoadBalancerTlsCertificate",
            "status": "Succeeded",
            "statusChangedAt": 1571072255.416
        },
        {
            "id": "dae1bcfb-d531-4c06-b4ea-bEXAMPLEc04e",
            "resourceName": "Certificate2",
            "resourceType": "LoadBalancerTlsCertificate",
            "createdAt": 1571072255.416,
            "location": {
                "availabilityZone": "all",
                "regionName": "us-west-2"
            },
            "isTerminal": true,
            "operationDetails": "LoadBalancer-1",
            "operationType": "AttachLoadBalancerTlsCertificate",
            "status": "Succeeded",
            "statusChangedAt": 1571072255.416
        }
    ]
}
```

- Per i dettagli sull'API, consulta AttachLoadBalancerTlsCertificate AWS CLICommand Reference.

`attach-static-ip`

Il seguente esempio di codice mostra come utilizzare`attach-static-ip`.

AWS CLI

Come collegare un IP statico a un'istanza

L'esempio `attach-static-ip` seguente collega l'IP statico `StaticIp-1` all'istanza `MEAN-1`.

```
aws lightsail attach-static-ip \
    --static-ip-name StaticIp-1 \
    --instance-name MEAN-1
```

Output:

```
{
    "operations": [
        {
            "id": "45e6fa13-4808-4b8d-9292-bEXAMPLE20b2",
            "resourceName": "StaticIp-1",
            "resourceType": "StaticIp",
            "createdAt": 1571072569.375,
            "location": {
                "availabilityZone": "all",
                "regionName": "us-west-2"
            },
            "isTerminal": true,
            "operationDetails": "MEAN-1",
            "operationType": "AttachStaticIp",
            "status": "Succeeded",
            "statusChangedAt": 1571072569.375
        },
        {
            "id": "9ee09a17-863c-4e51-8a6d-3EXAMPLE5475",
            "resourceName": "MEAN-1",
            "resourceType": "Instance",
            "createdAt": 1571072569.376,
            "location": {
                "availabilityZone": "us-west-2a",
                "regionName": "us-west-2"
            },
            "isTerminal": true,
            "operationDetails": "StaticIp-1",
            "operationType": "AttachStaticIp",
            "status": "Succeeded",
            "statusChangedAt": 1571072569.376
        }
    ]
}
```

- Per i dettagli sull'API, consulta AttachStaticIp AWS CLICommand Reference.

close-instance-public-ports

Il seguente esempio di codice mostra come utilizzare`close-instance-public-ports`.

AWS CLI

Come chiudere le porte del firewall per un'istanza

L'esempio `close-instance-public-ports` seguente chiude la porta TCP 22 sull'istanza `MEAN-2`.

```
aws lightsail close-instance-public-ports \
    --instance-name MEAN-2 \
    --port-info fromPort=22,protocol=TCP,toPort=22
```

Output:

```
{
    "operation": {
        "id": "4f328636-1c96-4649-ae6d-1EXAMPLEf446",
        "resourceName": "MEAN-2",
        "resourceType": "Instance",
        "createdAt": 1571072845.737,
        "location": {
            "availabilityZone": "us-west-2a",
            "regionName": "us-west-2"
        },
        "isTerminal": true,
        "operationDetails": "22/tcp",
        "operationType": "CloseInstancePublicPorts",
        "status": "Succeeded",
        "statusChangedAt": 1571072845.737
    }
}
```

- Per i dettagli sull'API, consulta CloseInstancePublicPorts AWS CLICommand Reference.

copy-snapshot

Il seguente esempio di codice mostra come utilizzare`copy-snapshot`.

AWS CLI

Esempio 1: copiare un'istantanea all'interno della stessa regione AWS

L'`copy-snapshot`esempio seguente copia un'istantanea dell'istanza `MEAN-1-1571075291` come istantanea dell'istanza `MEAN-1-Copy` all'interno della stessa regione. AWS `us-west-2`

```
aws lightsail copy-snapshot \
    --source-snapshot-name MEAN-1-1571075291 \
    --target-snapshot-name MEAN-1-Copy \
    --source-region us-west-2
```

Output:

```
{
    "operations": [
        {
            "id": "ced16fc1-f401-4556-8d82-1EXAMPLEb982",
            "resourceName": "MEAN-1-Copy",
            "resourceType": "InstanceSnapshot",
            "createdAt": 1571075581.498,
            "location": {
                "availabilityZone": "all",
                "regionName": "us-west-2"
            },
            "isTerminal": false,
            "operationDetails": "us-west-2:MEAN-1-1571075291",
            "operationType": "CopySnapshot",
            "status": "Started",
            "statusChangedAt": 1571075581.498
        }
    ]
}
```

Per ulteriori informazioni, consulta Copiare gli snapshot da una AWS regione all'altra in Amazon Lightsail nella Lightsail Dev Guide.

Esempio 2: copiare uno snapshot da una regione all'altra AWS

L'`copy-snapshot`esempio seguente copia lo snapshot dell'istanza `MEAN-1-1571075291` come istantanea dell'istanza `MEAN-1-1571075291-Copy` da AWS Region a. `us-west-2` `us-east-1`

```
aws lightsail copy-snapshot \
```

```
    --source-snapshot-name MEAN-1-1571075291 \
    --target-snapshot-name MEAN-1-1571075291-Copy \
    --source-region us-west-2 \
    --region us-east-1
```

Output:

```
{
    "operations": [
        {
            "id": "91116b79-119c-4451-b44a-dEXAMPLEd97b",
            "resourceName": "MEAN-1-1571075291-Copy",
            "resourceType": "InstanceSnapshot",
            "createdAt": 1571075695.069,
            "location": {
                "availabilityZone": "all",
                "regionName": "us-east-1"
            },
            "isTerminal": false,
            "operationDetails": "us-west-2:MEAN-1-1571075291",
            "operationType": "CopySnapshot",
            "status": "Started",
            "statusChangedAt": 1571075695.069
        }
    ]
}
```

Per ulteriori informazioni, consulta Copiare gli snapshot da una AWS regione all'altra in Amazon Lightsail nella Lightsail Dev Guide.

Esempio 3: copiare uno snapshot automatico all'interno della stessa regione AWS

L'`copy-snapshot`esempio seguente copia un'istantanea automatica `2019-10-14` dell'istanza `WordPress-1` come istantanea manuale `WordPress-1-10142019` nella regione. AWS `us-west-2`

```
aws lightsail copy-snapshot \
    --source-resource-name WordPress-1 \
    --restore-date 2019-10-14 \
    --target-snapshot-name WordPress-1-10142019 \
    --source-region us-west-2
```

Output:

```
{
    "operations": [
        {
            "id": "be3e6754-cd1d-48e6-ad9f-2EXAMPLE1805",
            "resourceName": "WordPress-1-10142019",
            "resourceType": "InstanceSnapshot",
            "createdAt": 1571082412.311,
            "location": {
                "availabilityZone": "all",
                "regionName": "us-west-2"
            },
            "isTerminal": false,
            "operationDetails": "us-west-2:WordPress-1",
            "operationType": "CopySnapshot",
            "status": "Started",
            "statusChangedAt": 1571082412.311
        }
    ]
}
```

Per ulteriori informazioni, consulta Conservazione di istantanee automatiche di istanze o dischi in Amazon Lightsail nella Lightsail Dev Guide.

Esempio 4: copiare uno snapshot automatico da una regione all'altra AWS

L'`copy-snapshot`esempio seguente copia un'istantanea automatica `2019-10-14` dell'istanza `WordPress-1` come istantanea manuale `WordPress-1-10142019` dalla AWS regione a. `us-west-2` `us-east-1`

```
aws lightsail copy-snapshot \
    --source-resource-name WordPress-1 \
    --restore-date 2019-10-14 \
    --target-snapshot-name WordPress-1-10142019 \
    --source-region us-west-2 \
    --region us-east-1
```

Output:

```
{
```

```
    "operations": [
        {
            "id": "dffa128b-0b07-476e-b390-bEXAMPLE3775",
            "resourceName": "WordPress-1-10142019",
            "resourceType": "InstanceSnapshot",
            "createdAt": 1571082493.422,
            "location": {
                "availabilityZone": "all",
                "regionName": "us-east-1"
            },
            "isTerminal": false,
            "operationDetails": "us-west-2:WordPress-1",
            "operationType": "CopySnapshot",
            "status": "Started",
            "statusChangedAt": 1571082493.422
        }
    ]
}
```

Per ulteriori informazioni, consulta Conservazione di istantanee automatiche di istanze o dischi in Amazon Lightsail nella Lightsail Dev Guide.

- Per i dettagli sull'API, consulta Command Reference. CopySnapshotAWS CLI

`create-disk-from-snapshot`

Il seguente esempio di codice mostra come utilizzare`create-disk-from-snapshot`.

AWS CLI

Come creare un disco in base a uno snapshot di disco

L'esempio `create-disk-from-snapshot` seguente crea un disco di archiviazione a blocchi denominato `Disk-2` in base allo snapshot del disco di archiviazione a blocchi specificato. Il disco viene creato nella AWS regione e nella zona di disponibilità specificate, con 32 GB di spazio di archiviazione.

```
aws lightsail create-disk-from-snapshot \
    --disk-name Disk-2 \
    --disk-snapshot-name Disk-1-1566839161 \
    --availability-zone us-west-2a \
    --size-in-gb 32
```

Output:

```
{
    "operations": [
        {
            "id": "d42b605d-5ef1-4b4a-8791-7a3e8b66b5e7",
            "resourceName": "Disk-2",
            "resourceType": "Disk",
            "createdAt": 1569624941.471,
            "location": {
                "availabilityZone": "us-west-2a",
                "regionName": "us-west-2"
            },
            "isTerminal": false,
            "operationType": "CreateDiskFromSnapshot",
            "status": "Started",
            "statusChangedAt": 1569624941.791
        }
    ]
}
```

Per ulteriori informazioni, consulta Creazione di un disco di storage a blocchi da uno snapshot in Amazon Lightsail nella Lightsail Developer Guide.

- Per i dettagli sull'API, consulta Command Reference. CreateDiskFromSnapshotAWS CLI

`create-disk-snapshot`

Il seguente esempio di codice mostra come utilizzare`create-disk-snapshot`.

AWS CLI

Esempio 1: come creare uno snapshot di un disco

L'esempio `create-disk-snapshot` seguente crea uno snapshot denominato `DiskSnapshot-1` del disco di archiviazione a blocchi specificato.

```
aws lightsail create-disk-snapshot \
    --disk-name Disk-1 \
    --disk-snapshot-name DiskSnapshot-1
```

Output:

```
{
    "operations": [
        {
            "id": "fa74c6d2-03a3-4f42-a7c7-792f124d534b",
            "resourceName": "DiskSnapshot-1",
            "resourceType": "DiskSnapshot",
            "createdAt": 1569625129.739,
            "location": {
                "availabilityZone": "all",
                "regionName": "us-west-2"
            },
            "isTerminal": false,
            "operationDetails": "Disk-1",
            "operationType": "CreateDiskSnapshot",
            "status": "Started",
            "statusChangedAt": 1569625129.739
        },
        {
            "id": "920a25df-185c-4528-87cd-7b85f5488c06",
            "resourceName": "Disk-1",
            "resourceType": "Disk",
            "createdAt": 1569625129.739,
            "location": {
                "availabilityZone": "us-west-2a",
                "regionName": "us-west-2"
            },
            "isTerminal": false,
            "operationDetails": "DiskSnapshot-1",
            "operationType": "CreateDiskSnapshot",
            "status": "Started",
            "statusChangedAt": 1569625129.739
        }
    ]
}
```

Esempio 2: come creare uno snapshot del disco di sistema di un'istanza

L'esempio `create-disk-snapshot` seguente crea uno snapshot del disco di sistema dell'istanza specificata.

```
aws lightsail create-disk-snapshot \
    --instance-name WordPress-1 \
```

```
    --disk-snapshot-name SystemDiskSnapshot-1
```

Output:

```
{
    "operations": [
        {
            "id": "f508cf1c-6597-42a6-a4c3-4aebd75af0d9",
            "resourceName": "SystemDiskSnapshot-1",
            "resourceType": "DiskSnapshot",
            "createdAt": 1569625294.685,
            "location": {
                "availabilityZone": "all",
                "regionName": "us-west-2"
            },
            "isTerminal": false,
            "operationDetails": "WordPress-1",
            "operationType": "CreateDiskSnapshot",
            "status": "Started",
            "statusChangedAt": 1569625294.685
        },
        {
            "id": "0bb9f712-da3b-4d99-b508-3bf871d989e5",
            "resourceName": "WordPress-1",
            "resourceType": "Instance",
            "createdAt": 1569625294.685,
            "location": {
                "availabilityZone": "us-west-2a",
                "regionName": "us-west-2"
            },
            "isTerminal": false,
            "operationDetails": "SystemDiskSnapshot-1",
            "operationType": "CreateDiskSnapshot",
            "status": "Started",
            "statusChangedAt": 1569625294.685
        }
    ]
}
```

Per ulteriori informazioni, consulta Snapshot in Amazon Lightsail e Creazione di un'istantanea del volume root di un'istanza in Amazon Lightsail nella Lightsail Developer Guide.

- Per i dettagli sull'API, consulta Command Reference. CreateDiskSnapshotAWS CLI

`create-disk`

Il seguente esempio di codice mostra come utilizzare`create-disk`.

AWS CLI

Come creare un disco di archiviazione a blocchi

L'`create-disk`esempio seguente crea un disco di archiviazione a blocchi `Disk-1` nella AWS regione e nella zona di disponibilità specificate, con 32 GB di spazio di archiviazione.

```
aws lightsail create-disk \
    --disk-name Disk-1 \
    --availability-zone us-west-2a \
    --size-in-gb 32
```

Output:

```
{
    "operations": [
        {
            "id": "1c85e2ec-86ba-4697-b936-77f4d3dc013a",
            "resourceName": "Disk-1",
            "resourceType": "Disk",
            "createdAt": 1569449220.36,
            "location": {
                "availabilityZone": "us-west-2a",
                "regionName": "us-west-2"
            },
            "isTerminal": false,
            "operationType": "CreateDisk",
            "status": "Started",
            "statusChangedAt": 1569449220.588
        }
    ]
}
```

- Per i dettagli sull'API, vedere CreateDiskin AWS CLI Command Reference.

`create-domain-entry`

Il seguente esempio di codice mostra come utilizzare`create-domain-entry`.

AWS CLI

Come creare una voce di dominio (record DNS)

L'esempio `create-domain-entry` seguente crea un record DNS (A) per l'apex del dominio specificato che punta all'indirizzo IP di un'istanza.

Nota: le operazioni API di Lightsail relative al dominio sono disponibili solo nella regione. `us-east-1` Se il profilo della CLI è configurato per l'utilizzo di una Regione diversa, è necessario includere il parametro `--region us-east-1`, altrimenti il comando ha esito negativo.

```
aws lightsail create-domain-entry \
    --region us-east-1 \
    --domain-name example.com \
    --domain-entry name=example.com,type=A,target=192.0.2.0
```

Output:

```
{
    "operation": {
        "id": "5be4494d-56f4-41fc-8730-693dcd0ef9e2",
        "resourceName": "example.com",
        "resourceType": "Domain",
        "createdAt": 1569865296.519,
        "location": {
            "availabilityZone": "all",
            "regionName": "global"
        },
        "isTerminal": true,
        "operationType": "CreateDomainEntry",
        "status": "Succeeded",
        "statusChangedAt": 1569865296.519
    }
}
```

Per ulteriori informazioni, consulta DNS in Amazon Lightsail e Creazione di una zona DNS per gestire i record DNS del dominio in Amazon Lightsail nella Lightsail Developer Guide.

- Per i dettagli sull'API, consulta Command Reference. CreateDomainEntryAWS CLI

`create-domain`

Il seguente esempio di codice mostra come utilizzare`create-domain`.

AWS CLI

Come creare un dominio (zona DNS)

L'esempio `create-domain` seguente crea una zona DNS per il dominio specificato.

Nota: le operazioni API di Lightsail relative al dominio sono disponibili solo nella regione. `us-east-1` Se il profilo della CLI è configurato per l'utilizzo di una Regione diversa, è necessario includere il parametro `--region us-east-1`, altrimenti il comando ha esito negativo.

```
aws lightsail create-domain \
    --region us-east-1 \
    --domain-name example.com
```

Output:

```
{
    "operation": {
        "id": "64e522c8-9ae1-4c05-9b65-3f237324dc34",
        "resourceName": "example.com",
        "resourceType": "Domain",
        "createdAt": 1569864291.92,
        "location": {
            "availabilityZone": "all",
            "regionName": "global"
        },
        "isTerminal": true,
        "operationType": "CreateDomain",
        "status": "Succeeded",
        "statusChangedAt": 1569864292.109
    }
}
```

Per ulteriori informazioni, consulta DNS in Amazon Lightsail e Creazione di una zona DNS per gestire i record DNS del dominio in Amazon Lightsail nella Lightsail Developer Guide.

- Per i dettagli sull'API, consulta Command Reference. CreateDomainAWS CLI

`create-instance-snapshot`

Il seguente esempio di codice mostra come utilizzare`create-instance-snapshot`.

AWS CLI

Come creare uno snapshot di un'istanza

L'esempio `create-instance-snapshot` seguente crea uno snapshot dall'istanza specificata.

```
aws lightsail create-instance-snapshot \
    --instance-name WordPress-1 \
    --instance-snapshot-name WordPress-Snapshot-1
```

Output:

```
{
    "operations": [
        {
            "id": "4c3db559-9dd0-41e7-89c0-2cb88c19786f",
            "resourceName": "WordPress-Snapshot-1",
            "resourceType": "InstanceSnapshot",
            "createdAt": 1569866438.48,
            "location": {
                "availabilityZone": "all",
                "regionName": "us-west-2"
            },
            "isTerminal": false,
            "operationDetails": "WordPress-1",
            "operationType": "CreateInstanceSnapshot",
            "status": "Started",
            "statusChangedAt": 1569866438.48
        },
        {
            "id": "c04fdc45-2981-488c-88b5-d6d2fd759a6a",
            "resourceName": "WordPress-1",
            "resourceType": "Instance",
            "createdAt": 1569866438.48,
            "location": {
                "availabilityZone": "us-west-2a",
                "regionName": "us-west-2"
            },
            "isTerminal": false,
```

```
            "operationDetails": "WordPress-Snapshot-1",
            "operationType": "CreateInstanceSnapshot",
            "status": "Started",
            "statusChangedAt": 1569866438.48
        }
    ]
}
```

- Per i dettagli sull'API, consulta CreateInstanceSnapshot AWS CLICommand Reference.

`create-instances-from-snapshot`

Il seguente esempio di codice mostra come utilizzare`create-instances-from-snapshot`.

AWS CLI

Come creare un'istanza da uno snapshot

L'`create-instances-from-snapshot`esempio seguente crea un'istanza dallo snapshot dell'istanza specificata, nella AWS regione e nella zona di disponibilità specificate, utilizzando il pacchetto da $12 USD.

Nota: il bundle specificato deve avere specifiche uguali o superiori a quelle del bundle dell'istanza di origine utilizzata per creare lo snapshot.

```
aws lightsail create-instances-from-snapshot \
    --instance-snapshot-name WordPress-1-1569866208 \
    --instance-names WordPress-2 \
    --availability-zone us-west-2a \
    --bundle-id small_3_0
```

Output:

```
{
    "operations": [
        {
            "id": "003f8271-b711-464d-b9b8-7f3806cb496e",
            "resourceName": "WordPress-2",
            "resourceType": "Instance",
            "createdAt": 1569865914.908,
            "location": {
```

```
                "availabilityZone": "us-west-2a",
                "regionName": "us-west-2"
            },
            "isTerminal": false,
            "operationType": "CreateInstancesFromSnapshot",
            "status": "Started",
            "statusChangedAt": 1569865914.908
        }
    ]
}
```

- Per i dettagli sull'API, consulta Command CreateInstancesFromSnapshotReference AWS CLI .

`create-instances`

Il seguente esempio di codice mostra come utilizzare`create-instances`.

AWS CLI

Esempio 1: come creare una singola istanza

L'`create-instances`esempio seguente crea un'istanza nella AWS regione e nella zona di disponibilità specificate, utilizzando il WordPress blueprint e il pacchetto da $5,00 USD.

```
aws lightsail create-instances \
    --instance-names Instance-1 \
    --availability-zone us-west-2a \
    --blueprint-id wordpress \
    --bundle-id nano_3_0
```

Output:

```
{
    "operations": [
        {
            "id": "9a77158f-7be3-4d6d-8054-cf5ae2b720cc",
            "resourceName": "Instance-1",
            "resourceType": "Instance",
            "createdAt": 1569447986.061,
            "location": {
                "availabilityZone": "us-west-2a",
```

```
                "regionName": "us-west-2"
            },
            "isTerminal": false,
            "operationType": "CreateInstance",
            "status": "Started",
            "statusChangedAt": 1569447986.061
        }
    ]
}
```

Esempio 2: come creare più istanze contemporaneamente

L'`create-instances`esempio seguente crea tre istanze nella AWS regione e nella zona di disponibilità specificate, utilizzando il WordPress blueprint e il pacchetto da $5,00 USD.

```
aws lightsail create-instances \
    --instance-names {"Instance1","Instance2","Instance3"} \
    --availability-zone us-west-2a \
    --blueprint-id wordpress \
    --bundle-id nano_3_0
```

Output:

```
{
    "operations": [
        {
            "id": "5492f015-9d2e-48c6-8eea-b516840e6903",
            "resourceName": "Instance1",
            "resourceType": "Instance",
            "createdAt": 1569448780.054,
            "location": {
                "availabilityZone": "us-west-2a",
                "regionName": "us-west-2"
            },
            "isTerminal": false,
            "operationType": "CreateInstance",
            "status": "Started",
            "statusChangedAt": 1569448780.054
        },
        {
            "id": "c58b5f46-2676-44c8-b95c-3ad375898515",
            "resourceName": "Instance2",
```

```
            "resourceType": "Instance",
            "createdAt": 1569448780.054,
            "location": {
                "availabilityZone": "us-west-2a",
                "regionName": "us-west-2"
            },
            "isTerminal": false,
            "operationType": "CreateInstance",
            "status": "Started",
            "statusChangedAt": 1569448780.054
        },
        {
            "id": "a5ad8006-9bee-4499-9eb7-75e42e6f5882",
            "resourceName": "Instance3",
            "resourceType": "Instance",
            "createdAt": 1569448780.054,
            "location": {
                "availabilityZone": "us-west-2a",
                "regionName": "us-west-2"
            },
            "isTerminal": false,
            "operationType": "CreateInstance",
            "status": "Started",
            "statusChangedAt": 1569448780.054
        }
    ]
}
```

- Per i dettagli sull'API, vedere in Command Reference. CreateInstancesAWS CLI

`create-key-pair`

Il seguente esempio di codice mostra come utilizzare`create-key-pair`.

AWS CLI

Come creare una coppia di chiavi

L'esempio `create-key-pair` seguente crea una coppia di chiavi che è possibile utilizzare per autenticarsi e connettersi a un'istanza.

```
aws lightsail create-key-pair \
```

```
    --key-pair-name MyPersonalKeyPair
```

L'output fornisce il valore Base64 della chiave privata che è possibile utilizzare per autenticarsi alle istanze che utilizzano la coppia di chiavi creata. Nota: copia il valore Base64 della chiave privata e incollalo in una posizione sicura perché non potrai recuperarlo in un secondo momento.


```
{
    "keyPair": {
        "name": "MyPersonalKeyPair",
        "arn": "arn:aws:lightsail:us-west-2:111122223333:KeyPair/55025c71-198f-403b-
b42f-a69433e724fb",
        "supportCode": "621291663362/MyPersonalKeyPair",
        "createdAt": 1569866556.567,
        "location": {
            "availabilityZone": "all",
            "regionName": "us-west-2"
        },
        "resourceType": "KeyPair"
    },
    "publicKeyBase64": "ssh-rsa ssh-rsa
 AAAAB3NzaC1yc2EAAAADAQABAAABAQCV0xUEwx96amPERH7K1bVT1tTFl9OmNk6o7m5YVHk9xlOdMbDRbFvhtXvw4jz
+BHUgedGUXno6uF7agqxZNO1kPLJBIVTW26SSYBJ0tE
+y804UyVsjrbUqCaMXDhmfXpWulMPwuXhwcKh7e8hwoTfkiX0E6Ql
+KqF/MiA3w6DCjEqvvdIO7SiEZJFsuGNfYDDN3w60Rel5MUhmn3OJdn4y/
A7NWb3IxL4pPfVE4rgFRKU8n1jp9kwRnlVMVBOWuGXk6n+H6M2f1 ",
    "privateKeyBase64": "-----BEGIN RSA PRIVATE KEY-----
EXAMPLETCCAfICCQD6m7oRw0uXOjANBgkqhkiG9w0BAQUFADCBiDELMAkGA1UEBhMC
\nVVMxCzAJBgNVBAgTAldBMRAwDgYDVQQHEwdTZWF0dGxlMQ8wDQYDVQQKEwZBbWF6\nb24xFDASBgNVBAsTC0lBTSBD
\nBgkqhkiG9w0BCQEWEG5vb25lQGFtYXpvbi5jb20wHhcNMTEwNDI1MjA0NTIxWhcN
\nMTIwNDI0MjA0NTIxWjCBiDELMAkGA1UEBhMCVVMxCzAJBgNVBAgTAldBMRAwDgYD
\nVQQHEwdTZWF0dGxlMQ8wDQEXAMPLEwZBbWF6b24xFDASBgNVBAsTC0lBTSBDb25z
\nb2xlMRIwEAYDVQQDEwlUZXN0Q2lsYWMxHzAdBgkqhkiG9w0BCQEWEG5vb25lQGFt
\nYXpvbi5jb20wgZ8wDQYJKoZIhvcNAQEBBQADgY0AMIGJAoGBAMEXAMPLE4GmWIWJ
\n21uUSfwfEvySWtC2XADZ4nB+BLYgVIk60CpiwsZ3G93vUEIO3IyNoH/f0wYK8m9T
\nrDHudUZg3qX4waLG5M43q7Wgc/MbQITxOUSQv7c7ugFFDzQGBzZswY6786m86gpE
\nIbb3OhjZnzcvQAaREXAMPLEMm2nrAgMBAAEwDQYJKoZIhvcNAQEFBQADgYEAtCu4\nnnUhVVxYUntneD9+h8Mg9q6q
+auNKyExzyLwaxlAoo7TJHidbtS4J5iNmZgXL0Fkb
\nFFBjvSfpJIlJ00zbhNYS5f6GuoEDmFJl0ZxBHjJnyp378OEXAMPLELvjx79LjSTb
\nNYiytVbZPQUQ5Yaxu2jXnimvw3rrszlaEXAMPLE=\n-----END RSA PRIVATE KEY-----",
    "operation": {
        "id": "67f984db-9994-45fe-ad38-59bafcaf82ef",
        "resourceName": "MyPersonalKeyPair",
        "resourceType": "KeyPair",
```

```
        "createdAt": 1569866556.567,
        "location": {
            "availabilityZone": "all",
            "regionName": "us-west-2"
        },
        "isTerminal": true,
        "operationType": "CreateKeyPair",
        "status": "Succeeded",
        "statusChangedAt": 1569866556.704
    }
}
```

- Per i dettagli sull'API, consulta CreateKeyPair AWS CLICommand Reference.

`create-load-balancer-tls-certificate`

Il seguente esempio di codice mostra come utilizzare`create-load-balancer-tls-certificate`.

AWS CLI

Come creare un certificato TLS per un bilanciatore del carico

L'esempio `create-load-balancer-tls-certificate` seguente crea un certificato TLS collegato al bilanciatore del carico specificato. Il certificato creato si applica ai domini specificati. Nota: per un bilanciatore del carico è possibile creare solo due certificati.

```
aws lightsail create-load-balancer-tls-certificate \
    --certificate-alternative-names abc.example.com \
    --certificate-domain-name example.com \
    --certificate-name MySecondCertificate \
    --load-balancer-name MyFirstLoadBalancer
```

Output:

```
{
    "operations": [
        {
            "id": "be663aed-cb46-41e2-9b23-e2f747245bd4",
            "resourceName": "MySecondCertificate",
            "resourceType": "LoadBalancerTlsCertificate",
```

```
            "createdAt": 1569867364.971,
            "location": {
                "availabilityZone": "all",
                "regionName": "us-west-2"
            },
            "isTerminal": true,
            "operationDetails": "MyFirstLoadBalancer",
            "operationType": "CreateLoadBalancerTlsCertificate",
            "status": "Succeeded",
            "statusChangedAt": 1569867365.219
        },
        {
            "id": "f3dfa930-969e-41cc-ac7d-337178716f6d",
            "resourceName": "MyFirstLoadBalancer",
            "resourceType": "LoadBalancer",
            "createdAt": 1569867364.971,
            "location": {
                "availabilityZone": "all",
                "regionName": "us-west-2"
            },
            "isTerminal": true,
            "operationDetails": "MySecondCertificate",
            "operationType": "CreateLoadBalancerTlsCertificate",
            "status": "Succeeded",
            "statusChangedAt": 1569867365.219
        }
    ]
}
```

- Per i dettagli sull'API, consulta CreateLoadBalancerTlsCertificate AWS CLICommand Reference.

`create-load-balancer`

Il seguente esempio di codice mostra come utilizzare`create-load-balancer`.

AWS CLI

Come creare un bilanciatore del carico

L'esempio `create-load-balancer` seguente crea un bilanciatore del carico con un certificato TLS. Il certificato TLS si applica ai domini specificati e instrada il traffico verso le istanze sulla porta 80.

```
aws lightsail create-load-balancer \
    --certificate-alternative-names www.example.com test.example.com \
    --certificate-domain-name example.com \
    --certificate-name Certificate-1 \
    --instance-port 80 \
    --load-balancer-name LoadBalancer-1
```

Output:


```
{
    "operations": [
        {
            "id": "cc7b920a-83d8-4762-a74e-9174fe1540be",
            "resourceName": "LoadBalancer-1",
            "resourceType": "LoadBalancer",
            "createdAt": 1569867169.406,
            "location": {
                "availabilityZone": "all",
                "regionName": "us-west-2"
            },
            "isTerminal": false,
            "operationType": "CreateLoadBalancer",
            "status": "Started",
            "statusChangedAt": 1569867169.406
        },
        {
            "id": "658ed43b-f729-42f3-a8e4-3f8024d3c98d",
            "resourceName": "LoadBalancer-1",
            "resourceType": "LoadBalancerTlsCertificate",
            "createdAt": 1569867170.193,
            "location": {
                "availabilityZone": "all",
                "regionName": "us-west-2"
            },
            "isTerminal": true,
            "operationDetails": "LoadBalancer-1",
            "operationType": "CreateLoadBalancerTlsCertificate",
            "status": "Succeeded",
            "statusChangedAt": 1569867170.54
        },
        {
            "id": "4757a342-5181-4870-b1e0-227eebc35ab5",
            "resourceName": "LoadBalancer-1",
```

```
            "resourceType": "LoadBalancer",
            "createdAt": 1569867170.193,
            "location": {
                "availabilityZone": "all",
                "regionName": "us-west-2"
            },
            "isTerminal": true,
            "operationDetails": "Certificate-1",
            "operationType": "CreateLoadBalancerTlsCertificate",
            "status": "Succeeded",
            "statusChangedAt": 1569867170.54
        }
    ]
}
```

Per ulteriori informazioni, consulta i sistemi di bilanciamento del carico di Lightsail nella Lightsail Developer Guide.

- Per i dettagli sull'API, consulta Command Reference. CreateLoadBalancerAWS CLI

`create-relational-database-from-snapshot`

Il seguente esempio di codice mostra come utilizzare`create-relational-database-from-snapshot`.

AWS CLI

Come creare un database gestito a partire da uno snapshot

L'`create-relational-database-from-snapshot`esempio seguente crea un database gestito dall'istantanea specificata nella AWS regione e nella zona di disponibilità specificate, utilizzando il pacchetto di database standard da $15 USD. Nota: il bundle specificato deve avere specifiche uguali o superiori a quelle del bundle del database di origine utilizzato per creare lo snapshot.

```
aws lightsail create-relational-database-from-snapshot \
    --relational-database-snapshot-name Database-Oregon-1-1566839359 \
    --relational-database-name Database-1 \
    --availability-zone us-west-2a \
    --relational-database-bundle-id micro_1_0 \
    --no-publicly-accessible
```

Output:

```
{
    "operations": [
        {
            "id": "ad6d9193-9d5c-4ea1-97ae-8fe6de600b4c",
            "resourceName": "Database-1",
            "resourceType": "RelationalDatabase",
            "createdAt": 1569867916.938,
            "location": {
                "availabilityZone": "us-west-2a",
                "regionName": "us-west-2"
            },
            "isTerminal": false,
            "operationType": "CreateRelationalDatabaseFromSnapshot",
            "status": "Started",
            "statusChangedAt": 1569867918.643
        }
    ]
}
```

- Per i dettagli sull'API, vedere CreateRelationalDatabaseFromSnapshotin AWS CLI Command Reference.

`create-relational-database-snapshot`

Il seguente esempio di codice mostra come utilizzare`create-relational-database-snapshot`.

AWS CLI

Come creare uno snapshot di un database gestito

L'esempio `create-relational-database-snapshot` seguente crea uno snapshot del database gestito specificato.

```
aws lightsail create-relational-database-snapshot \
    --relational-database-name Database1 \
    --relational-database-snapshot-name RelationalDatabaseSnapshot1
```

Output:

```
{
```

```
    "operations": [
        {
            "id": "853667fb-ea91-4c02-8d20-8fc5fd43b9eb",
            "resourceName": "RelationalDatabaseSnapshot1",
            "resourceType": "RelationalDatabaseSnapshot",
            "createdAt": 1569868074.645,
            "location": {
                "availabilityZone": "all",
                "regionName": "us-west-2"
            },
            "isTerminal": false,
            "operationDetails": "Database1",
            "operationType": "CreateRelationalDatabaseSnapshot",
            "status": "Started",
            "statusChangedAt": 1569868074.645
        },
        {
            "id": "fbafa521-3cac-4be8-9773-1c143780b239",
            "resourceName": "Database1",
            "resourceType": "RelationalDatabase",
            "createdAt": 1569868074.645,
            "location": {
                "availabilityZone": "us-west-2a",
                "regionName": "us-west-2"
            },
            "isTerminal": false,
            "operationDetails": "RelationalDatabaseSnapshot1",
            "operationType": "CreateRelationalDatabaseSnapshot",
            "status": "Started",
            "statusChangedAt": 1569868074.645
        }
    ]
}
```


- Per i dettagli sull'API, consulta CreateRelationalDatabaseSnapshot AWS CLICommand Reference.

`create-relational-database`

Il seguente esempio di codice mostra come utilizzare`create-relational-database`.

AWS CLI

Come creare un database gestito

L'`create-relational-database`esempio seguente crea un database gestito nella AWS regione e nella zona di disponibilità specificate, utilizzando il motore di database MySQL 5.6 (mysql_5_6) e il pacchetto di database standard da $15 USD (micro_1_0). Per il database gestito è già specificato un nome utente master e non è accessibile pubblicamente.

```
aws lightsail create-relational-database \
    --relational-database-name Database-1 \
    --availability-zone us-west-2a \
    --relational-database-blueprint-id mysql_5_6 \
    --relational-database-bundle-id micro_1_0 \
    --master-database-name dbmaster \
    --master-username user \
    --no-publicly-accessible
```

Output:

```
{
    "operations": [
        {
            "id": "b52bedee-73ed-4798-8d2a-9c12df89adcd",
            "resourceName": "Database-1",
            "resourceType": "RelationalDatabase",
            "createdAt": 1569450017.244,
            "location": {
                "availabilityZone": "us-west-2a",
                "regionName": "us-west-2"
            },
            "isTerminal": false,
            "operationType": "CreateRelationalDatabase",
            "status": "Started",
            "statusChangedAt": 1569450018.637
        }
    ]
}
```

- Per i dettagli sull'API, vedere in Command Reference. CreateRelationalDatabaseAWS CLI

`delete-auto-snapshot`

Il seguente esempio di codice mostra come utilizzare`delete-auto-snapshot`.

AWS CLI

Come eliminare uno snapshot automatico

L'esempio `delete-auto-snapshot` seguente elimina lo snapshot automatico `2019-10-10` dell'istanza `WordPress-1`.

```
aws lightsail delete-auto-snapshot \
    --resource-name WordPress-1 \
    --date 2019-10-10
```

Output:

```
{
    "operations": [
        {
            "id": "31c36e09-3d52-46d5-b6d8-7EXAMPLE534a",
            "resourceName": "WordPress-1",
            "resourceType": "Instance",
            "createdAt": 1571088141.501,
            "location": {
                "availabilityZone": "us-west-2",
                "regionName": "us-west-2"
            },
            "isTerminal": true,
            "operationDetails": "DeleteAutoSnapshot-2019-10-10",
            "operationType": "DeleteAutoSnapshot",
            "status": "Succeeded"
        }
    ]
}
```

Per ulteriori informazioni, consulta Eliminazione di istantanee automatiche di istanze o dischi in Amazon Lightsail nella Lightsail Dev Guide.

- Per i dettagli sull'API, consulta Command Reference. DeleteAutoSnapshotAWS CLI

`delete-disk-snapshot`

Il seguente esempio di codice mostra come utilizzare`delete-disk-snapshot`.

AWS CLI

Come eliminare uno snapshot di un disco di archiviazione a blocchi

L'esempio `delete-disk-snapshot` seguente elimina lo snapshot specificato di un disco di archiviazione a blocchi.

```
aws lightsail delete-disk-snapshot \
    --disk-snapshot-name DiskSnapshot-1
```

Output:

```
{
    "operations": [
        {
            "id": "d1e5766d-b81e-4595-ad5d-02afbccfcd5d",
            "resourceName": "DiskSnapshot-1",
            "resourceType": "DiskSnapshot",
            "createdAt": 1569873552.79,
            "location": {
                "availabilityZone": "all",
                "regionName": "us-west-2"
            },
            "isTerminal": true,
            "operationType": "DeleteDiskSnapshot",
            "status": "Succeeded",
            "statusChangedAt": 1569873552.79
        }
    ]
}
```

- Per i dettagli sull'API, consulta DeleteDiskSnapshot AWS CLICommand Reference.

`delete-disk`

Il seguente esempio di codice mostra come utilizzare`delete-disk`.

AWS CLI

Come eliminare un disco di archiviazione a blocchi

L'esempio `delete-disk` seguente elimina il disco di archiviazione a blocchi specificato.

```
aws lightsail delete-disk \
    --disk-name Disk-1
```

Output:

```
{
    "operations": [
        {
            "id": "6378c70f-4d75-4f7a-ab66-730fca0bb2fc",
            "resourceName": "Disk-1",
            "resourceType": "Disk",
            "createdAt": 1569872887.864,
            "location": {
                "availabilityZone": "us-west-2a",
                "regionName": "us-west-2"
            },
            "isTerminal": true,
            "operationType": "DeleteDisk",
            "status": "Succeeded",
            "statusChangedAt": 1569872887.864
        }
    ]
}
```

- Per i dettagli sull'API, consulta DeleteDisk AWS CLICommand Reference.

`delete-domain-entry`

Il seguente esempio di codice mostra come utilizzare`delete-domain-entry`.

AWS CLI

Come eliminare una voce di dominio (record DNS)

L'esempio `delete-domain-entry` seguente elimina la voce di dominio specificata da un dominio esistente.

Nota: le operazioni API di Lightsail relative al dominio sono disponibili solo nella regione. `us-east-1` Se il profilo della CLI è configurato per l'utilizzo di una Regione diversa, è necessario includere il parametro `--region us-east-1`, altrimenti il comando ha esito negativo.

```
aws lightsail delete-domain-entry \
    --region us-east-1 \
    --domain-name example.com \
    --domain-entry name=123.example.com,target=192.0.2.0,type=A
```

Output:

```
{
    "operation": {
        "id": "06eacd01-d785-420e-8daa-823150c7dca1",
        "resourceName": "example.com ",
        "resourceType": "Domain",
        "createdAt": 1569874157.005,
        "location": {
            "availabilityZone": "all",
            "regionName": "global"
        },
        "isTerminal": true,
        "operationType": "DeleteDomainEntry",
        "status": "Succeeded",
        "statusChangedAt": 1569874157.005
    }
}
```

- Per i dettagli sull'API, consulta Command Reference. DeleteDomainEntryAWS CLI

`delete-domain`

Il seguente esempio di codice mostra come utilizzare`delete-domain`.

AWS CLI

Come eliminare un dominio (zona DNS)

L'esempio `delete-domain` seguente elimina il dominio specificato e tutte le voci in esso contenute (record DNS).

Nota: le operazioni API di Lightsail relative al dominio sono disponibili solo nella regione. `us-east-1` Se il profilo della CLI è configurato per l'utilizzo di una Regione diversa, è necessario includere il parametro `--region us-east-1`, altrimenti il comando ha esito negativo.

```
aws lightsail delete-domain \
    --region us-east-1 \
    --domain-name example.com
```

Output:

```
{
    "operation": {
        "id": "fcef5265-5af1-4a46-a3d7-90b5e18b9b32",
        "resourceName": "example.com",
        "resourceType": "Domain",
        "createdAt": 1569873788.13,
        "location": {
            "availabilityZone": "all",
            "regionName": "global"
        },
        "isTerminal": true,
        "operationType": "DeleteDomain",
        "status": "Succeeded",
        "statusChangedAt": 1569873788.13
    }
}
```

- Per i dettagli sull'API, consulta Command Reference. DeleteDomainAWS CLI

`delete-instance-snapshot`

Il seguente esempio di codice mostra come utilizzare`delete-instance-snapshot`.

AWS CLI

titolo

L'esempio `delete-instance-snapshot` seguente elimina lo snapshot specificato di un'istanza.

```
aws lightsail delete-instance-snapshot \
```

```
    --instance-snapshot-name WordPress-1-Snapshot-1
```

Output:

```
{
    "operations": [
        {
            "id": "14dad182-976a-46c6-bfd4-9480482bf0ea",
            "resourceName": "WordPress-1-Snapshot-1",
            "resourceType": "InstanceSnapshot",
            "createdAt": 1569874524.562,
            "location": {
                "availabilityZone": "all",
                "regionName": "us-west-2"
            },
            "isTerminal": true,
            "operationType": "DeleteInstanceSnapshot",
            "status": "Succeeded",
            "statusChangedAt": 1569874524.562
        }
    ]
}
```

- Per i dettagli sull'API, consulta DeleteInstanceSnapshot AWS CLICommand Reference.

`delete-instance`

Il seguente esempio di codice mostra come utilizzare`delete-instance`.

AWS CLI

Come eliminare un'istanza

L'esempio `delete-instance` seguente elimina l'istanza specificata.

```
aws lightsail delete-instance \
    --instance-name WordPress-1
```

Output:

```
{
    "operations": [
```

```
        {
            "id": "d77345a3-8f80-4d2e-b47d-aaa622718df2",
            "resourceName": "Disk-1",
            "resourceType": "Disk",
            "createdAt": 1569874357.469,
            "location": {
                "availabilityZone": "us-west-2a",
                "regionName": "us-west-2"
            },
            "isTerminal": false,
            "operationDetails": "WordPress-1",
            "operationType": "DetachDisk",
            "status": "Started",
            "statusChangedAt": 1569874357.469
        },
        {
            "id": "708fa606-2bfd-4e48-a2c1-0b856585b5b1",
            "resourceName": "WordPress-1",
            "resourceType": "Instance",
            "createdAt": 1569874357.465,
            "location": {
                "availabilityZone": "us-west-2a",
                "regionName": "us-west-2"
            },
            "isTerminal": false,
            "operationDetails": "Disk-1",
            "operationType": "DetachDisk",
            "status": "Started",
            "statusChangedAt": 1569874357.465
        },
        {
            "id": "3187e823-8acb-405d-b098-fad5ceb17bec",
            "resourceName": "WordPress-1",
            "resourceType": "Instance",
            "createdAt": 1569874357.829,
            "location": {
                "availabilityZone": "us-west-2a",
                "regionName": "us-west-2"
            },
            "isTerminal": true,
            "operationType": "DeleteInstance",
            "status": "Succeeded",
            "statusChangedAt": 1569874357.829
        }
```

```
    ]
}
```

- Per i dettagli sull'API, consulta DeleteInstance AWS CLICommand Reference.

`delete-key-pair`

Il seguente esempio di codice mostra come utilizzare`delete-key-pair`.

AWS CLI

Come eliminare una coppia di chiavi

L'esempio `delete-key-pair` seguente elimina la coppia di chiavi specificata.

```
aws lightsail delete-key-pair \
    --key-pair-name MyPersonalKeyPair
```

Output:

```
{
    "operation": {
        "id": "81621463-df38-4810-b866-6e801a15abbf",
        "resourceName": "MyPersonalKeyPair",
        "resourceType": "KeyPair",
        "createdAt": 1569874626.466,
        "location": {
            "availabilityZone": "all",
            "regionName": "us-west-2"
        },
        "isTerminal": true,
        "operationType": "DeleteKeyPair",
        "status": "Succeeded",
        "statusChangedAt": 1569874626.685
    }
}
```

- Per i dettagli sull'API, consulta DeleteKeyPair AWS CLICommand Reference.

`delete-known-host-keys`

Il seguente esempio di codice mostra come utilizzare`delete-known-host-keys`.

AWS CLI

Come eliminare le chiavi host note da un'istanza

L'esempio `delete-known-host-keys` seguente elimina la chiave host nota dall'istanza specificata.

```
aws lightsail delete-known-host-keys \
    --instance-name Instance-1
```

Output:

```
{
    "operations": [
        {
            "id": "c61afe9c-45a4-41e6-a97e-d212364da3f5",
            "resourceName": "Instance-1",
            "resourceType": "Instance",
            "createdAt": 1569874760.201,
            "location": {
                "availabilityZone": "us-west-2a",
                "regionName": "us-west-2"
            },
            "isTerminal": true,
            "operationType": "DeleteKnownHostKeys",
            "status": "Succeeded",
            "statusChangedAt": 1569874760.201
        }
    ]
}
```

Per ulteriori informazioni, consulta Risoluzione dei problemi di connessione con il client SSH o RDP basato su browser Amazon Lightsail nella Lightsail Dev Guide.

- Per i dettagli sull'API, consulta Command Reference. DeleteKnownHostKeysAWS CLI

`delete-load-balancer-tls-certificate`

Il seguente esempio di codice mostra come utilizzare`delete-load-balancer-tls-certificate`.

AWS CLI

Come eliminare un certificato TLS per un bilanciatore del carico

L'esempio `delete-load-balancer-tls-certificate` seguente elimina il certificato TLS specificato dal bilanciatore del carico specificato.

```
aws lightsail delete-load-balancer-tls-certificate \
    --load-balancer-name MyFirstLoadBalancer \
    --certificate-name MyFirstCertificate
```

Output:

```
{
    "operations": [
        {
            "id": "50bec274-e45e-4caa-8a69-b763ef636583",
            "resourceName": "MyFirstCertificate",
            "resourceType": "LoadBalancerTlsCertificate",
            "createdAt": 1569874989.48,
            "location": {
                "availabilityZone": "all",
                "regionName": "us-west-2"
            },
            "isTerminal": false,
            "operationType": "DeleteLoadBalancerTlsCertificate",
            "status": "Started",
            "statusChangedAt": 1569874989.48
        },
        {
            "id": "78c58cdc-a59a-4b27-8213-500638634a8f",
            "resourceName": "MyFirstLoadBalancer",
            "resourceType": "LoadBalancer",
            "createdAt": 1569874989.48,
            "location": {
                "availabilityZone": "all",
                "regionName": "us-west-2"
            },
            "isTerminal": false,
            "operationType": "DeleteLoadBalancerTlsCertificate",
            "status": "Started",
            "statusChangedAt": 1569874989.48
        }
```

```
    ]
}
```

- Per i dettagli sull'API, consulta DeleteLoadBalancerTlsCertificate AWS CLICommand Reference.

`delete-load-balancer`

Il seguente esempio di codice mostra come utilizzare`delete-load-balancer`.

AWS CLI

Come eliminare un bilanciatore del carico

L'esempio `delete-load-balancer` seguente elimina il bilanciatore del carico specificato e i certificati TLS associati.

```
aws lightsail delete-load-balancer \
    --load-balancer-name MyFirstLoadBalancer
```

Output:

```
{
    "operations": [
        {
            "id": "a8c968c7-72a3-4680-a714-af8f03eea535",
            "resourceName": "MyFirstLoadBalancer",
            "resourceType": "LoadBalancer",
            "createdAt": 1569875092.125,
            "location": {
                "availabilityZone": "all",
                "regionName": "us-west-2"
            },
            "isTerminal": true,
            "operationType": "DeleteLoadBalancer",
            "status": "Succeeded",
            "statusChangedAt": 1569875092.125
        },
        {
            "id": "f91a29fc-8ce3-4e69-a227-ea70ca890bf5",
            "resourceName": "MySecondCertificate",
            "resourceType": "LoadBalancerTlsCertificate",
```

```
            "createdAt": 1569875091.938,
            "location": {
                "availabilityZone": "all",
                "regionName": "us-west-2"
            },
            "isTerminal": false,
            "operationType": "DeleteLoadBalancerTlsCertificate",
            "status": "Started",
            "statusChangedAt": 1569875091.938
        },
        {
            "id": "cf64c060-154b-4eb4-ba57-84e2e41563d6",
            "resourceName": "MyFirstLoadBalancer",
            "resourceType": "LoadBalancer",
            "createdAt": 1569875091.94,
            "location": {
                "availabilityZone": "all",
                "regionName": "us-west-2"
            },
            "isTerminal": false,
            "operationType": "DeleteLoadBalancerTlsCertificate",
            "status": "Started",
            "statusChangedAt": 1569875091.94
        }
    ]
}
```

Per ulteriori informazioni, consulta il titolo nella guida.

- Per i dettagli sull'API, consulta DeleteLoadBalancer AWS CLICommand Reference.

`delete-relational-database-snapshot`

Il seguente esempio di codice mostra come utilizzare`delete-relational-database-snapshot`.

AWS CLI

Come eliminare uno snapshot di un database gestito

L'esempio `delete-relational-database-snapshot` seguente elimina lo snapshot specificato di un database gestito.

```
aws lightsail delete-relational-database-snapshot \
```

```
    --relational-database-snapshot-name Database-Oregon-1-1566839359
```

Output:

```
{
    "operations": [
        {
            "id": "b99acae8-735b-4823-922f-30af580e3729",
            "resourceName": "Database-Oregon-1-1566839359",
            "resourceType": "RelationalDatabaseSnapshot",
            "createdAt": 1569875293.58,
            "location": {
                "availabilityZone": "all",
                "regionName": "us-west-2"
            },
            "isTerminal": true,
            "operationType": "DeleteRelationalDatabaseSnapshot",
            "status": "Succeeded",
            "statusChangedAt": 1569875293.58
        }
    ]
}
```

- Per i dettagli sull'API, consulta DeleteRelationalDatabaseSnapshot AWS CLICommand Reference.

`delete-relational-database`

Il seguente esempio di codice mostra come utilizzare`delete-relational-database`.

AWS CLI

Come eliminare un database gestito

L'esempio `delete-relational-database` seguente elimina il database gestito specificato.

```
aws lightsail delete-relational-database \
    --relational-database-name Database-1
```

Output:

```
{
```

```
    "operations": [
        {
            "id": "3b0c41c1-053d-46f0-92a3-14f76141dc86",
            "resourceName": "Database-1",
            "resourceType": "RelationalDatabase",
            "createdAt": 1569875210.999,
            "location": {
                "availabilityZone": "us-west-2a",
                "regionName": "us-west-2"
            },
            "isTerminal": false,
            "operationType": "DeleteRelationalDatabase",
            "status": "Started",
            "statusChangedAt": 1569875210.999
        },
        {
            "id": "01ddeae8-a87a-4a4b-a1f3-092c71bf9180",
            "resourceName": "Database-1",
            "resourceType": "RelationalDatabase",
            "createdAt": 1569875211.029,
            "location": {
                "availabilityZone": "us-west-2a",
                "regionName": "us-west-2"
            },
            "isTerminal": false,
            "operationDetails": "Database-1-FinalSnapshot-1569875210793",
            "operationType": "CreateRelationalDatabaseSnapshot",
            "status": "Started",
            "statusChangedAt": 1569875211.029
        },
        {
            "id": "74d73681-30e8-4532-974e-1f23cd3f9f73",
            "resourceName": "Database-1-FinalSnapshot-1569875210793",
            "resourceType": "RelationalDatabaseSnapshot",
            "createdAt": 1569875211.029,
            "location": {
                "availabilityZone": "all",
                "regionName": "us-west-2"
            },
            "isTerminal": false,
            "operationDetails": "Database-1",
            "operationType": "CreateRelationalDatabaseSnapshot",
            "status": "Started",
            "statusChangedAt": 1569875211.029
```

```
        }
    ]
}
```

- Per i dettagli sull'API, consulta DeleteRelationalDatabase AWS CLICommand Reference.

`detach-static-ip`

Il seguente esempio di codice mostra come utilizzare`detach-static-ip`.

AWS CLI

Come scollegare un IP statico da un'istanza

L'esempio `detach-static-ip` seguente scollega l'IP statico `StaticIp-1` da qualsiasi istanza collegata.

```
aws lightsail detach-static-ip \
    --static-ip-name StaticIp-1
```

Output:

```
{
    "operations": [
        {
            "id": "2a43d8a3-9f2d-4fe7-bdd0-eEXAMPLE3cf3",
            "resourceName": "StaticIp-1",
            "resourceType": "StaticIp",
            "createdAt": 1571088261.999,
            "location": {
                "availabilityZone": "all",
                "regionName": "us-west-2"
            },
            "isTerminal": true,
            "operationDetails": "MEAN-1",
            "operationType": "DetachStaticIp",
            "status": "Succeeded",
            "statusChangedAt": 1571088261.999
        },
        {
            "id": "41a7d40c-74e8-4d2e-a837-cEXAMPLEf747",
            "resourceName": "MEAN-1",
```

```
            "resourceType": "Instance",
            "createdAt": 1571088262.022,
            "location": {
                "availabilityZone": "us-west-2a",
                "regionName": "us-west-2"
            },
            "isTerminal": true,
            "operationDetails": "StaticIp-1",
            "operationType": "DetachStaticIp",
            "status": "Succeeded",
            "statusChangedAt": 1571088262.022
        }
    ]
}
```

- Per i dettagli sull'API, consulta DetachStaticIp AWS CLICommand Reference.

`get-active-names`

Il seguente esempio di codice mostra come utilizzare`get-active-names`.

AWS CLI

Come ottenere i nomi delle risorse attive

L'`get-active-names`esempio seguente restituisce i nomi delle risorse attive nella AWS regione configurata.

```
aws lightsail get-active-names
```

Output:

```
{
    "activeNames": [
        "WordPress-1",
        "StaticIp-1",
        "MEAN-1",
        "Plesk_Hosting_Stack_on_Ubuntu-1"
    ]
}
```

- Per i dettagli sull'API, vedere GetActiveNamesin AWS CLI Command Reference.

`get-auto-snapshots`

Il seguente esempio di codice mostra come utilizzare`get-auto-snapshots`.

AWS CLI

Come ottenere gli snapshot automatici disponibili per un'istanza

L'esempio `get-auto-snapshots` seguente restituisce gli snapshot automatici disponibili per l'istanza `WordPress-1`.

```
aws lightsail get-auto-snapshots \
    --resource-name WordPress-1
```

Output:

```
{
    "resourceName": "WordPress-1",
    "resourceType": "Instance",
    "autoSnapshots": [
        {
            "date": "2019-10-14",
            "createdAt": 1571033872.0,
            "status": "Success",
            "fromAttachedDisks": []
        },
        {
            "date": "2019-10-13",
            "createdAt": 1570947473.0,
            "status": "Success",
            "fromAttachedDisks": []
        },
        {
            "date": "2019-10-12",
            "createdAt": 1570861072.0,
            "status": "Success",
            "fromAttachedDisks": []
        },
        {
            "date": "2019-10-11",
            "createdAt": 1570774672.0,
            "status": "Success",
            "fromAttachedDisks": []
```

```
        }
    ]
}
```

Per ulteriori informazioni, consulta Conservazione di istantanee automatiche di istanze o dischi in Amazon Lightsail nella Lightsail Dev Guide.

- Per i dettagli sull'API, consulta Command Reference. GetAutoSnapshotsAWS CLI

`get-blueprints`

Il seguente esempio di codice mostra come utilizzare`get-blueprints`.

AWS CLI

Come ottenere i blueprint per le nuove istanze

L'`get-blueprints`esempio seguente mostra i dettagli su tutti i blueprint disponibili che possono essere utilizzati per creare nuove istanze in Amazon Lightsail.

```
aws lightsail get-blueprints
```

Output:

```
{
    "blueprints": [
        {
            "blueprintId": "wordpress",
            "name": "WordPress",
            "group": "wordpress",
            "type": "app",
            "description": "Bitnami, the leaders in application packaging, and
 Automattic, the experts behind WordPress, have teamed up to offer this official
 WordPress image. This image is a pre-configured, ready-to-run image for running
 WordPress on Amazon Lightsail. WordPress is the world's most popular content
 management platform. Whether it's for an enterprise or small business website, or
 a personal or corporate blog, content authors can easily create content using its
 new Gutenberg editor, and developers can extend the base platform with additional
 features. Popular plugins like Jetpack, Akismet, All in One SEO Pack, WP Mail,
 Google Analytics for WordPress, and Amazon Polly are all pre-installed in this
 image. Let's Encrypt SSL certificates are supported through an auto-configuration
 script.",
            "isActive": true,
```

```
            "minPower": 0,
            "version": "6.5.3-0",
            "versionCode": "1",
            "productUrl": "https://aws.amazon.com/marketplace/pp/B00NN8Y43U",
            "licenseUrl": "https://aws.amazon.com/marketplace/pp/B00NN8Y43U#pdp-
usage",
            "platform": "LINUX_UNIX"
        },
        {
            "blueprintId": "lamp_8_bitnami",
            "name": "LAMP (PHP 8)",
            "group": "lamp_8",
            "type": "app",
            "description": "LAMP with PHP 8.X packaged by Bitnami enables you
 to quickly start building your websites and applications by providing a coding
 framework. As a developer, it provides standalone project directories to store your
 applications. This blueprint is configured for production environments. It includes
 SSL auto-configuration with Let's Encrypt certificates, and the latest releases of
 PHP, Apache, and MariaDB on Linux. This application also includes phpMyAdmin, PHP
 main modules and Composer.",
            "isActive": true,
            "minPower": 0,
            "version": "8.2.18-4",
            "versionCode": "1",
            "productUrl": "https://aws.amazon.com/marketplace/pp/
prodview-6g3gzfcih6dvu",
            "licenseUrl": "https://aws.amazon.com/marketplace/pp/
prodview-6g3gzfcih6dvu#pdp-usage",
            "platform": "LINUX_UNIX"
        },
        {
            "blueprintId": "nodejs",
            "name": "Node.js",
            "group": "node",
            "type": "app",
            "description": "Node.js packaged by Bitnami is a pre-configured, ready
 to run image for Node.js on Amazon EC2. It includes the latest version of Node.js,
 Apache, Python and Redis. The image supports multiple Node.js applications, each
 with its own virtual host and project directory. It is configured for production
 use and is secure by default, as all ports except HTTP, HTTPS and SSH ports are
 closed. Let's Encrypt SSL certificates are supported through an auto-configuration
 script. Developers benefit from instant access to a secure, update and consistent
 Node.js environment without having to manually install and configure multiple
 components and libraries.",
```

```
            "isActive": true,
            "minPower": 0,
            "version": "18.20.2-0",
            "versionCode": "1",
            "productUrl": "https://aws.amazon.com/marketplace/pp/B00NNZUAKO",
            "licenseUrl": "https://aws.amazon.com/marketplace/pp/B00NNZUAKO#pdp-
usage",
            "platform": "LINUX_UNIX"
        },
        ...
        }
    ]
}
```

- Per i dettagli sull'API, consulta GetBlueprintsCommand Reference.AWS CLI

`get-bundles`

Il seguente esempio di codice mostra come utilizzare`get-bundles`.

AWS CLI

Come ottenere i bundle per le nuove istanze

L'`get-bundles`esempio seguente mostra i dettagli su tutti i pacchetti disponibili che possono essere utilizzati per creare nuove istanze in Amazon Lightsail.

```
aws lightsail get-bundles
```

Output:

```
{
    "bundles": [
        {
            "price": 5.0,
            "cpuCount": 2,
            "diskSizeInGb": 20,
            "bundleId": "nano_3_0",
            "instanceType": "nano",
            "isActive": true,
            "name": "Nano",
```

```
            "power": 298,
            "ramSizeInGb": 0.5,
            "transferPerMonthInGb": 1024,
            "supportedPlatforms": [
                "LINUX_UNIX"
            ]
        },
        {
            "price": 7.0,
            "cpuCount": 2,
            "diskSizeInGb": 40,
            "bundleId": "micro_3_0",
            "instanceType": "micro",
            "isActive": true,
            "name": "Micro",
            "power": 500,
            "ramSizeInGb": 1.0,
            "transferPerMonthInGb": 2048,
            "supportedPlatforms": [
                "LINUX_UNIX"
            ]
        },
        {
            "price": 12.0,
            "cpuCount": 2,
            "diskSizeInGb": 60,
            "bundleId": "small_3_0",
            "instanceType": "small",
            "isActive": true,
            "name": "Small",
            "power": 1000,
            "ramSizeInGb": 2.0,
            "transferPerMonthInGb": 3072,
            "supportedPlatforms": [
                "LINUX_UNIX"
            ]
        },
        ...
        }
    ]
}
```

- Per i dettagli sull'API, consulta GetBundlesCommand Reference.AWS CLI

`get-cloud-formation-stack-records`

Il seguente esempio di codice mostra come utilizzare`get-cloud-formation-stack-records`.

AWS CLI

Per ottenere i record dello CloudFormation stack e gli stack associati

L'`get-cloud-formation-stack-records`esempio seguente mostra i dettagli sui record dello CloudFormation stack e sugli stack associati utilizzati per creare EC2 risorse Amazon da istantanee Amazon Lightsail esportate.

```
aws lightsail get-cloud-formation-stack-records
```

Output:

```
{
    "cloudFormationStackRecords": [
        {
            "name": "CloudFormationStackRecord-588a4243-
e2d1-490d-8200-3a7513ecebdf",
            "arn": "arn:aws:lightsail:us-
west-2:111122223333:CloudFormationStackRecord/28d646ab-27bc-48d9-a422-1EXAMPLE6d37",
            "createdAt": 1565301666.586,
            "location": {
                "availabilityZone": "all",
                "regionName": "us-west-2"
            },
            "resourceType": "CloudFormationStackRecord",
            "state": "Succeeded",
            "sourceInfo": [
                {
                    "resourceType": "ExportSnapshotRecord",
                    "name": "ExportSnapshotRecord-
e02f23d7-0453-4aa9-9c95-91aa01a141dd",
                    "arn": "arn:aws:lightsail:us-
west-2:111122223333:ExportSnapshotRecord/f12b8792-f3ea-4d6f-b547-2EXAMPLE8796"
                }
            ],
            "destinationInfo": {
                "id": "arn:aws:cloudformation:us-west-2:111122223333:stack/
Lightsail-Stack-588a4243-e2d1-490d-8200-3EXAMPLEebdf/063203b0-
ba28-11e9-838b-0EXAMPLE8b00",
```

```
                "service": "Aws::CloudFormation::Stack"
            }
        }
    ]
}
```

- Per i dettagli sull'API, consulta Command Reference. GetCloudFormationStackRecordsAWS CLI

`get-disk-snapshot`

Il seguente esempio di codice mostra come utilizzare`get-disk-snapshot`.

AWS CLI

Come ottenere le informazioni su uno snapshot del disco

L'esempio `get-disk-snapshot` seguente mostra i dettagli relativi allo snapshot del disco `Disk-1-1566839161`.

```
aws lightsail get-disk-snapshot \
    --disk-snapshot-name Disk-1-1566839161
```

Output:

```
{
    "diskSnapshot": {
        "name": "Disk-1-1566839161",
        "arn": "arn:aws:lightsail:us-west-2:111122223333:DiskSnapshot/
e2d0fa53-8ee0-41a0-8e56-0EXAMPLE1051",
        "supportCode": "6EXAMPLE3362/snap-0EXAMPLE06100d09",
        "createdAt": 1566839163.749,
        "location": {
            "availabilityZone": "all",
            "regionName": "us-west-2"
        },
        "resourceType": "DiskSnapshot",
        "tags": [],
        "sizeInGb": 8,
        "state": "completed",
        "progress": "100%",
        "fromDiskName": "Disk-1",
```

```
        "fromDiskArn": "arn:aws:lightsail:us-west-2:111122223333:Disk/
c21cfb0a-07f2-44ae-9a23-bEXAMPLE8096",
        "isFromAutoSnapshot": false
    }
}
```

Per ulteriori informazioni, consulta il titolo nella guida.

- Per i dettagli sull'API, consulta GetDiskSnapshot AWS CLICommand Reference.

`get-disk-snapshots`

Il seguente esempio di codice mostra come utilizzare`get-disk-snapshots`.

AWS CLI

Come ottenere le informazioni su tutti gli snapshot del disco

L'`get-disk-snapshots`esempio seguente visualizza i dettagli su tutte le istantanee del disco nella AWS regione configurata.

```
aws lightsail get-disk-snapshots
```

Output:

```
{
    "diskSnapshots": [
        {
            "name": "Disk-2-1571090588",
            "arn": "arn:aws:lightsail:us-
west-2:111122223333:DiskSnapshot/32e889a9-38d4-4687-9f21-eEXAMPLE7839",
            "supportCode": "6EXAMPLE3362/snap-0EXAMPLE1ca192a4",
            "createdAt": 1571090591.226,
            "location": {
                "availabilityZone": "all",
                "regionName": "us-west-2"
            },
            "resourceType": "DiskSnapshot",
            "tags": [],
            "sizeInGb": 8,
            "state": "completed",
            "progress": "100%",
```

```
            "fromDiskName": "Disk-2",
            "fromDiskArn": "arn:aws:lightsail:us-
west-2:111122223333:Disk/6a343ff8-6341-422d-86e2-bEXAMPLE16c2",
            "isFromAutoSnapshot": false
        },
        {
            "name": "Disk-1-1566839161",
            "arn": "arn:aws:lightsail:us-west-2:111122223333:DiskSnapshot/
e2d0fa53-8ee0-41a0-8e56-0EXAMPLE1051",
            "supportCode": "6EXAMPLE3362/snap-0EXAMPLEe06100d09",
            "createdAt": 1566839163.749,
            "location": {
                "availabilityZone": "all",
                "regionName": "us-west-2"
            },
            "resourceType": "DiskSnapshot",
            "tags": [],
            "sizeInGb": 8,
            "state": "completed",
            "progress": "100%",
            "fromDiskName": "Disk-1",
            "fromDiskArn": "arn:aws:lightsail:us-west-2:111122223333:Disk/
c21cfb0a-07f2-44ae-9a23-bEXAMPLE8096",
            "isFromAutoSnapshot": false
        }
    ]
}
```

- Per i dettagli sull'API, vedere GetDiskSnapshotsin AWS CLI Command Reference.

`get-disk`

Il seguente esempio di codice mostra come utilizzare`get-disk`.

AWS CLI

Come ottenere le informazioni su un disco di archiviazione a blocchi

L'esempio `get-disk` seguente mostra i dettagli relativi al disco `Disk-1`.

```
aws lightsail get-disk \
    --disk-name Disk-1
```

Output:

```
{
    "disk": {
        "name": "Disk-1",
        "arn": "arn:aws:lightsail:us-west-2:111122223333:Disk/
c21cfb0a-07f2-44ae-9a23-bEXAMPLE8096",
        "supportCode": "6EXAMPLE3362/vol-0EXAMPLEf2f88b32f",
        "createdAt": 1566585439.587,
        "location": {
            "availabilityZone": "us-west-2a",
            "regionName": "us-west-2"
        },
        "resourceType": "Disk",
        "tags": [],
        "sizeInGb": 8,
        "isSystemDisk": false,
        "iops": 100,
        "path": "/dev/xvdf",
        "state": "in-use",
        "attachedTo": "WordPress_Multisite-1",
        "isAttached": true,
        "attachmentState": "attached"
    }
}
```

Per ulteriori informazioni, consulta il titolo nella guida.

- Per i dettagli sull'API, consulta GetDisk AWS CLICommand Reference.

`get-disks`

Il seguente esempio di codice mostra come utilizzare`get-disks`.

AWS CLI

Come ottenere le informazioni su tutti i dischi di archiviazione a blocchi

L'`get-disks`esempio seguente visualizza i dettagli su tutti i dischi nella AWS regione configurata.

```
aws lightsail get-disks
```

Output:


```
{
    "disks": [
        {
            "name": "Disk-2",
            "arn": "arn:aws:lightsail:us-
west-2:111122223333:Disk/6a343ff8-6341-422d-86e2-bEXAMPLE16c2",
            "supportCode": "6EXAMPLE3362/vol-0EXAMPLE929602087",
            "createdAt": 1571090461.634,
            "location": {
                "availabilityZone": "us-west-2a",
                "regionName": "us-west-2"
            },
            "resourceType": "Disk",
            "tags": [],
            "sizeInGb": 8,
            "isSystemDisk": false,
            "iops": 100,
            "state": "available",
            "isAttached": false,
            "attachmentState": "detached"
        },
        {
            "name": "Disk-1",
            "arn": "arn:aws:lightsail:us-west-2:111122223333:Disk/
c21cfb0a-07f2-44ae-9a23-bEXAMPLE8096",
            "supportCode": "6EXAMPLE3362/vol-0EXAMPLEf2f88b32f",
            "createdAt": 1566585439.587,
            "location": {
                "availabilityZone": "us-west-2a",
                "regionName": "us-west-2"
            },
            "resourceType": "Disk",
            "tags": [],
            "sizeInGb": 8,
            "isSystemDisk": false,
            "iops": 100,
            "path": "/dev/xvdf",
            "state": "in-use",
            "attachedTo": "WordPress_Multisite-1",
            "isAttached": true,
            "attachmentState": "attached"
        }
```

```
    ]
}
```

- Per i dettagli sull'API, vedere GetDisksin AWS CLI Command Reference.

`get-domain`

Il seguente esempio di codice mostra come utilizzare`get-domain`.

AWS CLI

Come ottenere le informazioni su un dominio

L'esempio `get-domain` seguente mostra i dettagli relativi al dominio `example.com`.

Nota: le operazioni API di Lightsail relative al dominio sono disponibili solo nella regione. `us-east-1` AWS Se il profilo della CLI è configurato per l'utilizzo di una Regione diversa, è necessario includere il parametro ``--region us-east-1``, altrimenti il comando ha esito negativo.

```
aws lightsail get-domain \
    --domain-name example.com \
    --region us-east-1
```

Output:

```
{
    "domain": {
        "name": "example.com",
        "arn":
 "arn:aws:lightsail:global:111122223333:Domain/28cda903-3f15-44b2-9baf-3EXAMPLEb304",
        "supportCode": "6EXAMPLE3362//hostedzone/ZEXAMPLEONGSC1",
        "createdAt": 1570728588.6,
        "location": {
            "availabilityZone": "all",
            "regionName": "global"
        },
        "resourceType": "Domain",
        "tags": [],
        "domainEntries": [
            {
                "id": "-1682899164",
                "name": "example.com",
```

```
                "target": "192.0.2.0",
                "isAlias": false,
                "type": "A"
            },
            {
                "id": "1703104243",
                "name": "example.com",
                "target": "ns-137.awsdns-17.com",
                "isAlias": false,
                "type": "NS"
            },
            {
                "id": "-1038331153",
                "name": "example.com",
                "target": "ns-1710.awsdns-21.co.uk",
                "isAlias": false,
                "type": "NS"
            },
            {
                "id": "-2107289565",
                "name": "example.com",
                "target": "ns-692.awsdns-22.net",
                "isAlias": false,
                "type": "NS"
            },
            {
                "id": "1582095705",
                "name": "example.com",
                "target": "ns-1436.awsdns-51.org",
                "isAlias": false,
                "type": "NS"
            },
            {
                "id": "-1769796132",
                "name": "example.com",
                "target": "ns-1710.awsdns-21.co.uk. awsdns-hostmaster.amazon.com. 1
 7200 900 1209600 86400",
                "isAlias": false,
                "type": "SOA"
            }
        ]
    }
}
```

- Per i dettagli sull'API, consulta Command Reference. GetDomainAWS CLI

`get-domains`

Il seguente esempio di codice mostra come utilizzare`get-domains`.

AWS CLI

Come ottenere le informazioni su tutti i domini

L'`get-domains`esempio seguente mostra i dettagli su tutti i domini nella AWS regione configurata.

Nota: le operazioni API di Lightsail relative ai domini sono disponibili solo nella regione. `us-east-1` AWS Se il profilo della CLI è configurato per l'utilizzo di una Regione diversa, è necessario includere il parametro `--region us-east-1`, altrimenti il comando ha esito negativo.

```
aws lightsail get-domains \
    --region us-east-1
```

Output:

```
{
    "domains": [
        {
            "name": "example.com",
            "arn":
 "arn:aws:lightsail:global:111122223333:Domain/28cda903-3f15-44b2-9baf-3EXAMPLEb304",
            "supportCode": "6EXAMPLE3362//hostedzone/ZEXAMPLEONGSC1",
            "createdAt": 1570728588.6,
            "location": {
                "availabilityZone": "all",
                "regionName": "global"
            },
            "resourceType": "Domain",
            "tags": [],
            "domainEntries": [
                {
                    "id": "-1682899164",
                    "name": "example.com",
                    "target": "192.0.2.0",
```

```
                    "isAlias": false,
                    "type": "A"
                },
                {
                    "id": "1703104243",
                    "name": "example.com",
                    "target": "ns-137.awsdns-17.com",
                    "isAlias": false,
                    "type": "NS"
                },
                {
                    "id": "-1038331153",
                    "name": "example.com",
                    "target": "ns-4567.awsdns-21.co.uk",
                    "isAlias": false,
                    "type": "NS"
                },
                {
                    "id": "-2107289565",
                    "name": "example.com",
                    "target": "ns-333.awsdns-22.net",
                    "isAlias": false,
                    "type": "NS"
                },
                {
                    "id": "1582095705",
                    "name": "example.com",
                    "target": "ns-1111.awsdns-51.org",
                    "isAlias": false,
                    "type": "NS"
                },
                {
                    "id": "-1769796132",
                    "name": "example.com",
                    "target": "ns-1234.awsdns-21.co.uk. awsdns-
hostmaster.amazon.com. 1 7200 900 1209600 86400",
                    "isAlias": false,
                    "type": "SOA"
                },
                {
                    "id": "1029454894",
                    "name": "_dead6a124ede046a0319eb44a4eb3cbc.example.com",
                    "target": "_be133b0a0899fb7b6bf79d9741d1a383.hkvuiqjoua.acm-
validations.aws",
```

```
                    "isAlias": false,
                    "type": "CNAME"
                }
            ]
        },
        {
            "name": "example.net",
            "arn": "arn:aws:lightsail:global:111122223333:Domain/9c9f0d70-
c92e-4753-86c2-6EXAMPLE029d",
            "supportCode": "6EXAMPLE3362//hostedzone/ZEXAMPLE5TPKMV",
            "createdAt": 1556661071.384,
            "location": {
                "availabilityZone": "all",
                "regionName": "global"
            },
            "resourceType": "Domain",
            "tags": [],
            "domainEntries": [
                {
                    "id": "-766320943",
                    "name": "example.net",
                    "target": "192.0.2.2",
                    "isAlias": false,
                    "type": "A"
                },
                {
                    "id": "-453913825",
                    "name": "example.net",
                    "target": "ns-123.awsdns-10.net",
                    "isAlias": false,
                    "type": "NS"
                },
                {
                    "id": "1553601564",
                    "name": "example.net",
                    "target": "ns-4444.awsdns-47.co.uk",
                    "isAlias": false,
                    "type": "NS"
                },
                {
                    "id": "1653797661",
                    "name": "example.net",
                    "target": "ns-7890.awsdns-61.org",
                    "isAlias": false,
```

```
                    "type": "NS"
                },
                {
                    "id": "706414698",
                    "name": "example.net",
                    "target": "ns-123.awsdns-44.com",
                    "isAlias": false,
                    "type": "NS"
                },
                {
                    "id": "337271745",
                    "name": "example.net",
                    "target": "ns-4444.awsdns-47.co.uk. awsdns-
hostmaster.amazon.com. 1 7200 900 1209600 86400",
                    "isAlias": false,
                    "type": "SOA"
                },
                {
                    "id": "-1785431096",
                    "name": "www.example.net",
                    "target": "192.0.2.2",
                    "isAlias": false,
                    "type": "A"
                }
            ]
        },
        {
            "name": "example.org",
            "arn": "arn:aws:lightsail:global:111122223333:Domain/
f0f13ba3-3df0-4fdc-8ebb-1EXAMPLEf26e",
            "supportCode": "6EXAMPLE3362//hostedzone/ZEXAMPLEAFO38",
            "createdAt": 1556661199.106,
            "location": {
                "availabilityZone": "all",
                "regionName": "global"
            },
            "resourceType": "Domain",
            "tags": [],
            "domainEntries": [
                {
                    "id": "2065301345",
                    "name": "example.org",
                    "target": "192.0.2.4",
                    "isAlias": false,
```

```
                    "type": "A"
                },
                {
                    "id": "-447198516",
                    "name": "example.org",
                    "target": "ns-123.awsdns-45.com",
                    "isAlias": false,
                    "type": "NS"
                },
                {
                    "id": "136463022",
                    "name": "example.org",
                    "target": "ns-9999.awsdns-15.co.uk",
                    "isAlias": false,
                    "type": "NS"
                },
                {
                    "id": "1395941679",
                    "name": "example.org",
                    "target": "ns-555.awsdns-01.net",
                    "isAlias": false,
                    "type": "NS"
                },
                {
                    "id": "872052569",
                    "name": "example.org",
                    "target": "ns-6543.awsdns-38.org",
                    "isAlias": false,
                    "type": "NS"
                },
                {
                    "id": "1001949377",
                    "name": "example.org",
                    "target": "ns-1234.awsdns-15.co.uk. awsdns-
hostmaster.amazon.com. 1 7200 900 1209600 86400",
                    "isAlias": false,
                    "type": "SOA"
                },
                {
                    "id": "1046191192",
                    "name": "www.example.org",
                    "target": "192.0.2.4",
                    "isAlias": false,
                    "type": "A"
```

```
                }
            ]
        }
    ]
}
```

- Per i dettagli sull'API, consulta Command Reference. GetDomainsAWS CLI

`get-export-snapshot-record`

Il seguente esempio di codice mostra come utilizzare`get-export-snapshot-record`.

AWS CLI

Per ottenere i record delle istantanee esportate su Amazon EC2

L'`get-export-snapshot-record`esempio seguente mostra i dettagli sulle istanze di Amazon Lightsail o sugli snapshot del disco esportati in Amazon. EC2

```
aws lightsail get-export-snapshot-records
```

Output:

```
{
    "exportSnapshotRecords": [
        {
            "name": "ExportSnapshotRecord-d2da10ce-0b3c-4ae1-ab3a-2EXAMPLEa586",
            "arn": "arn:aws:lightsail:us-
west-2:111122223333:ExportSnapshotRecord/076c7060-b0cc-4162-98f0-2EXAMPLEe28e",
            "createdAt": 1543534665.678,
            "location": {
                "availabilityZone": "all",
                "regionName": "us-west-2"
            },
            "resourceType": "ExportSnapshotRecord",
            "state": "Succeeded",
            "sourceInfo": {
                "resourceType": "InstanceSnapshot",
                "createdAt": 1540339310.706,
                "name": "WordPress-512MB-Oregon-1-1540339219",
                "arn": "arn:aws:lightsail:us-
west-2:111122223333:InstanceSnapshot/5446f534-ed60-4c17-b4a5-bEXAMPLEf8b7",
```

```
                "fromResourceName": "WordPress-512MB-Oregon-1",
                "fromResourceArn": "arn:aws:lightsail:us-
west-2:111122223333:Instance/4b8f1f24-e4d1-4cf3-88ff-cEXAMPLEa397",
                "instanceSnapshotInfo": {
                    "fromBundleId": "nano_2_0",
                    "fromBlueprintId": "wordpress_4_9_8",
                    "fromDiskInfo": [
                        {
                            "path": "/dev/sda1",
                            "sizeInGb": 20,
                            "isSystemDisk": true
                        }
                    ]
                }
            },
            "destinationInfo": {
                "id": "ami-0EXAMPLEc0d65058e",
                "service": "Aws::EC2::Image"
            }
        },
        {
            "name": "ExportSnapshotRecord-1c94e884-40ff-4fe1-9302-0EXAMPLE14c2",
            "arn": "arn:aws:lightsail:us-west-2:111122223333:ExportSnapshotRecord/
fb392ce8-6567-4013-9bfd-3EXAMPLE5b4c",
            "createdAt": 1543432110.2,
            "location": {
                "availabilityZone": "all",
                "regionName": "us-west-2"
            },
            "resourceType": "ExportSnapshotRecord",
            "state": "Succeeded",
            "sourceInfo": {
                "resourceType": "InstanceSnapshot",
                "createdAt": 1540833603.545,
                "name": "LAMP_PHP_5-512MB-Oregon-1-1540833565",
                "arn": "arn:aws:lightsail:us-
west-2:111122223333:InstanceSnapshot/82334399-b5f2-49ec-8382-0EXAMPLEe45f",
                "fromResourceName": "LAMP_PHP_5-512MB-Oregon-1",
                "fromResourceArn": "arn:aws:lightsail:us-
west-2:111122223333:Instance/863b9f35-ab1e-4418-bdd2-1EXAMPLEbab2",
                "instanceSnapshotInfo": {
                    "fromBundleId": "nano_2_0",
                    "fromBlueprintId": "lamp_5_6_37_2",
                    "fromDiskInfo": [
```

```
                        {
                            "path": "/dev/sda1",
                            "sizeInGb": 20,
                            "isSystemDisk": true
                        }
                    ]
                }
            },
            "destinationInfo": {
                "id": "ami-0EXAMPLE7c5ec84e2",
                "service": "Aws::EC2::Image"
            }
        }
    ]
}
```

- Per i dettagli sull'API, consulta GetExportSnapshotRecordCommand Reference.AWS CLI

`get-instance-access-details`

Il seguente esempio di codice mostra come utilizzare`get-instance-access-details`.

AWS CLI

Come ottenere le informazioni sulla chiave dell'host per un'istanza

L'esempio `get-instance-access-details` seguente mostra le informazioni relative alla chiave dell'host per l'istanza `WordPress_Multisite-1`.

```
aws lightsail get-instance-access-details \
    --instance-name WordPress_Multisite-1
```

Output:

```
{
    "accessDetails": {
        "certKey": "ssh-rsa-cert-v01@openssh.com
 AEXAMPLEaC1yc2EtY2VydC12MDFAb3BlbnNzaC5jb20AAAAgNf076Dt3ppmPd0fPxZVMmS491aEAYYH9cHqAJ3fNML8
vEXAMPLE2eBWJyQvn7ol/
i0+s966h5sx8qUD79lPB7q5UESd5VZGFtytrykfQJnjiwqe7EV5agzvjblLj26Fb37EKda9HVfCOu8pWbvky7Tyn9w29
+xMfQM9xVz0rXZmqx8uJidJpRgLCMTviofwQJU/
K1EXAMPLEAAAAAAAABAAAALS00MzMzMDU4MzA4ODg1MTY2NjM4Onp6UWlndHk4UElRSG9STitOTG5QSEE9PQAAAAsAAA
```

```
+LiB+ozNbUA0cdNL9Y67x7qPv/R7XhTc21+2A+8+GuVpK/Kz9dqDMKNAEXAMPLE+YYN
+tiXm7Y8OgziK+7iDB7xUuQ4vghmn4+qgz9mKwYgWvVe2+0XLuV7cnWPB7iUlHQg
+E3LUKrV4ZFw9pj7X2dFdNKfMxwWgI1ISWKimEXAMPLEeHjrf1Rqc/
QH6TpWCvPfcx8uvwVqdwTfkE/SfA5BCzbGGI1UmIUadh8nHcb5FamQ1hK7kECy47K/x9FMn/
KwmM7pCwJbSLDMO7n9bnbvck6m8ZoB2N2YLMG5dW7BerEXAMPLEobqfdtyYJHHel1EyyEJs1fWNU3D5JIGlgzcPAV
+ZlbQyUCZXf0oslSa+HE85fO/
FRq9SVSBSHrmbeb0frlPhgMzgSmqLeyhlbr6wwWIDbREXAMPLEJZ49H7RdQxdKyYrZPWvRgcr0qI2EL0tAajnpQQ8UZq
Aqter0xN5PhFL0J49OWTacwCGRAjLhibAx7K1t/1ZXWo6c+ijq8clll327EXAMPLE/
e89GC89KcmKCxfGQniDAUgF8UqofIbq3ZOUgiAAYCVXclI4L68NhVXyoWuQXPBRQSEXAMPLEWm74tDL9tFN3c7tSe/
Oz0cTR+4sAAAIPAAAAB3NzaC1yc2EAAAIAQnG/
L0DqiSnLrWhEox4aHqMgd0m0oLLAYx6OQH9F0TM9EXAMPLE961rzSCMon7ZgsWNnL0OwZQgDG
+rtJ4N0B7HOVwns4ynUFbzNQ3qFGGeE3lKwX1L41vV1iSy7sDk8aI0LmrKJi1LE1Qc1l8uboRlwoXOYEXAMPLEaUCeX
+10+WEXAMPLEg6Y4U4ZvE2B3xyRdpvysb5TGFNtk5qPslacnVkoLOGsZZXMpLGJnG4OBpQLLtpj9sNMxAgZPCAUjhkqk
+nxO9O4NUZ2pTWbVSUaV1gm6pug9xbwNO1Im21t34JeLlKTqxcJ6zzS8W0c0KKpAm5c4hWkseMbyutS2jav/4hiS
+BhrYgptzfwe5qRXEXAMPLEHZQr3YfGzYoBJ/
lLK3NHhxOihhsfAYwMei0BFZT1F/7CT3IH4iitEkIgodi06/
Mw6UDqMPozyQCK1lEA6LFhYCOZG9drWcoRa74lM4kY9TP028Za8gDMh1WpkXLq9Gixon5OHP8aM/
sEXAMPLEr2+fnkw+1BtoO5L6+VKoPlXaGqZ/fBYEXAMPLEAMQHjnLM1JYNvtEEPhp+TNzXHzuixWf/
Ht04m0AVpXrzIDXaS1O2tXY=",
        "ipAddress": "192.0.2.0",
        "privateKey": "-----BEGIN RSA PRIVATE KEY-----
\nEXAMPLEBAAKCAQEA+AD3qeU2toBy5O5v7wnRLVo/tngVickL5+6Jf4tPrPeuoebM
\nfKlA+/ZTwe6uVBEneVWRhbcra8pH0CZ44sKnuxFeWoM7425S49uhW9+xCnWvR1Xw
\njrvKVm75Mu08p/cNvfWugrBuaPB65DspgxNnOfZWMVxpIpSqOSPWmSwQHV597d6C
\nrEXAMPLEo8hJmqz2KFQ09X7fB2lBruGgr9aXiNPmWmovYKqwFmrnFvR7odFmDecq
\n5EXAMPLE9dyU1ZsrWhGby77eYrVaFl0GNGQ8qy1HGUiscquZ9NDIL49n4mXbfsTH
\n0EXAMPLE12ZqsfLiYnSaUYCwjE74qH8ECVPytQIDAQABAoIBAHeZV9Z58JHAjifz
\nCEXAMPLEEqC3doOVDgXSlkKI92qNo4z2VcUEho878paCuVVXVHcCGgSnGeyIh2tN
\nMEXAMPLESohR427BhH3YLA+3Z5SIvnejbTgYPfLC37B8khTaYqkqMvdZiFVZK5qn
\nIEXAMPLEM93oF9eSZCjcLKB/jGHsfb0eCDMP8BshHE2beuqzVMoK1DxOnvoP3+Fp
\nAEXAMPLESq6pDpCo9YVUX8g1u3Ro9cPl2LXHDy+oVEY5KhbZQJ7VU1I72WOvppWW
\nOEXAMPLEkgYlq7p6qYtYcSgTEjz14gDiMfQ7SyHB3alkIoNONQ9ZPaWHyJvymeud
\noQTNuz0CgYEA/LFWNTEZrzdzdR1kJmyNRmAermU0B6utyNENChAlHGSHkB+1lVSh
\nbEXAMPLEQo9ooUeW5UxO3YwacZLoDT1mwxw1Ptc1+PNycZoLe1fE9UdARrdmGTob
\n8l7CPLSXp3xuR8VqSp2fnIc7hfiQs/NrPX9gm/EOrB0we0RKyDSzWScCgYEA+z/r
\niob+nJZq0YbnOSuP6oMULP4vnWniWj8MIhUJU53LwSAM8DeJdONKDdkuiOd52aAL
\nVgn7nLo88rVWKhJwVc4tu/rNgZLcR3bP4+kL6zand0KQnMLyOzNA2Ys26aa5udH1\nqWl0WTt9WEm/
h10ndC1knOMectrvsG17b38y5sMCgYEA54NiRGGz8oCPW6GN/FZA
\nKEXAMPLE5tw34GEH3Uxlc9n3CejDaQmczOATwX4nIwRZDEqWyYZcS0btg1jhGiBD\nYEXAMPLEkc8Z71L/
agZEAaVCEog9FqfSqwB
+XTfoKh8qur74X1yCu9p6gof1q6k9\neEXAMPLEchJcNNOg4ETIfMkCgYBdVORRhE4mqvWpOdzA7v66FdEz2YSkjAXKk
\naEXAMPLE8Z/8yBSmuBv1Qv03XA12my462uB92uzzGAuW
+1yBc2Kn1sXqYTy0y1z0\ngEXAMPLEBogjw4MqHKL1bPKMHyQU8/
q24PaYgzHPzy13wlH6pTYf1XqlHdE2D6Vv\nyEXAMPLEgQC3i/
```

```
kVVhky/2XRwRVlC7JO2Bg3QGTx38hpmDa5IuofKANjA+Wa3/zy\nbEXAMPLE6ytQgD9GN/YtBq+uhO
+2ZkvXPL+CWRi0ZRXpPwYDBBFU9Cw0AuWWGlL8\nwEXAMPLExMlcysRgcWB9RNgf3AuOpFd2i6XT/
riNsvvkpmJ+VooU8g==\n-----END RSA PRIVATE KEY-----\n",
        "protocol": "ssh",
        "instanceName": "WordPress_Multisite-1",
        "username": "bitnami",
        "hostKeys": [
            {
                "algorithm": "ssh-rsa",
                "publicKey":
 "AEXAMPLEaC1yc2EAAAADAQABAAABAQCoeR9ieZTjQ3pXCHczuAYZFjlF7t
+uBkXuqeGMRex78pCvmS+DiEXAMPLEuJ1Q8dcKhrQL4HpXbD9dosVCTaJnJwb4MQqsuSVFdHFzy3guP
+BKclWqtxJEXAMPLEsBGqZZlrIv6a9bTA0TCplZ8AD+hSRTaSXXqg6FT
+Qf16IktH0XlMs7xIEXAMPLEmNtjCpzZiGXDHzytoMvUgwa8uHPp44Og36EUu4VqQxoUHPJKoXvcQizyk3K8ym0hP0Tp
Ot6y9HwvykEXAMPLEAfbKjbR42+u6+OSlkr4d339q2U1sTDytJhhs8HUel1wTfGRfp",
                "witnessedAt": 1570744377.699,
                "fingerprintSHA1": "SHA1:GEXAMPLEMoYgUg0ucadqU9Bt3Lk",
                "fingerprintSHA256": "SHA256:IEXAMPLEcB5vgxnAUoJawbdZ
+MwELhIp6FUxuwq/LIU"
            },
            {
                "algorithm": "ssh-ed25519",
                "publicKey":
 "AEXAMPLEaC1lZDI1NTE5AAAAIC1gwGPDfGaONxEXAMPLEJX3UNap781QxHQmn8nzlrUv",
                "witnessedAt": 1570744377.697,
                "fingerprintSHA1": "SHA1:VEXAMPLE5ReqSmTgv03sSUw9toU",
                "fingerprintSHA256": "SHA256:0EXAMPLEdE6tI95k3TJpG
+qhJbAoknB0yz9nAEaDt3A"
            },
            {
                "algorithm": "ecdsa-sha2-nistp256",
                "publicKey":
 "AEXAMPLEZHNhLXNoYTItbmlzdHAyNTYAAAAIbmlzdHAyNTYAAABEXAMPLE9B4mZy8YSsZW7cixCDq5yHSAAxjJkDo5
+EnKlDCsYtUkxxEXAMPLE6VOWL2z63RTKa2AUPgd8irjxWI=",
                "witnessedAt": 1570744377.707,
                "fingerprintSHA1": "SHA1:UEXAMPLEOYCfXsCf2G6tDg+7YG0",
                "fingerprintSHA256": "SHA256:wEXAMPLEQ9a/
iEXAMPLEhRufm6U9vFU4cpkMPHnBsNA"
            }
        ]
    }
}
```

- Per i dettagli sull'API, consulta GetInstanceAccessDetails AWS CLICommand Reference.

`get-instance-metric-data`

Il seguente esempio di codice mostra come utilizzare`get-instance-metric-data`.

AWS CLI

Come ottenere dati delle metriche per un'istanza

L'esempio `get-instance-metric-data` seguente restituisce la percentuale media di `CPUUtilization` ogni `7200` secondi (2 ore) tra `1571342400` e `1571428800` per l'istanza `MEAN-1`.

Per identificare l'ora di inizio e quella di fine, è consigliabile utilizzare un convertitore di data e ora Unix.

```
aws lightsail get-instance-metric-data \
    --instance-name MEAN-1 \
    --metric-name CPUUtilization \
    --period 7200 \
    --start-time 1571342400 \
    --end-time 1571428800 \
    --unit Percent \
    --statistics Average
```

Output:

```
{
    "metricName": "CPUUtilization",
    "metricData": [
        {
            "average": 0.2611371877012072 5,
            "timestamp": 1571342400.0,
            "unit": "Percent"
        },
        {
            "average": 0.26861268928111953,
            "timestamp": 1571392800.0,
            "unit": "Percent"
        },
        {
            "average": 0.28187475104748777,
            "timestamp": 1571378400.0,
            "unit": "Percent"
```

```
        },
        {
            "average": 0.2651936960458352,
            "timestamp": 1571421600.0,
            "unit": "Percent"
        },
        {
            "average": 0.2561856213712188,
            "timestamp": 1571371200.0,
            "unit": "Percent"
        },
        {
            "average": 0.3021383254607764,
            "timestamp": 1571356800.0,
            "unit": "Percent"
        },
        {
            "average": 0.2618381649223539,
            "timestamp": 1571407200.0,
            "unit": "Percent"
        },
        {
            "average": 0.26331929394825787,
            "timestamp": 1571400000.0,
            "unit": "Percent"
        },
        {
            "average": 0.2576348407007818,
            "timestamp": 1571385600.0,
            "unit": "Percent"
        },
        {
            "average": 0.2513008454658378,
            "timestamp": 1571364000.0,
            "unit": "Percent"
        },
        {
            "average": 0.26329974562758346,
            "timestamp": 1571414400.0,
            "unit": "Percent"
        },
        {
            "average": 0.2667092536656445,
            "timestamp": 1571349600.0,
```

```
            "unit": "Percent"
        }
    ]
}
```

- Per i dettagli sull'API, consulta GetInstanceMetricData AWS CLICommand Reference.

`get-instance-port-states`

Il seguente esempio di codice mostra come utilizzare`get-instance-port-states`.

AWS CLI

Come ottenere le informazioni sul firewall per un'istanza

L'esempio `get-instance-port-states` seguente restituisce le porte del firewall configurate per l'istanza `MEAN-1`.

```
aws lightsail get-instance-port-states \
    --instance-name MEAN-1
```

Output:

```
{
    "portStates": [
        {
            "fromPort": 80,
            "toPort": 80,
            "protocol": "tcp",
            "state": "open"
        },
        {
            "fromPort": 22,
            "toPort": 22,
            "protocol": "tcp",
            "state": "open"
        },
        {
            "fromPort": 443,
            "toPort": 443,
            "protocol": "tcp",
```

```
            "state": "open"
        }
    ]
}
```

- Per i dettagli sull'API, consulta GetInstancePortStates AWS CLICommand Reference.

`get-instance-snapshot`

Il seguente esempio di codice mostra come utilizzare`get-instance-snapshot`.

AWS CLI

Come ottenere le informazioni su uno snapshot di istanza specificato

L'esempio `get-instance-snapshot` seguente mostra i dettagli relativi allo snapshot di istanza specificato.

```
aws lightsail get-instance-snapshot \
    --instance-snapshot-name MEAN-1-1571419854
```

Output:

```
{
    "instanceSnapshot": {
        "name": "MEAN-1-1571419854",
        "arn": "arn:aws:lightsail:us-west-2:111122223333:InstanceSnapshot/
ac54700c-48a8-40fd-b065-2EXAMPLEac8f",
        "supportCode": "6EXAMPLE3362/ami-0EXAMPLE67a73020d",
        "createdAt": 1571419891.927,
        "location": {
            "availabilityZone": "all",
            "regionName": "us-west-2"
        },
        "resourceType": "InstanceSnapshot",
        "tags": [],
        "state": "available",
        "fromAttachedDisks": [],
        "fromInstanceName": "MEAN-1",
        "fromInstanceArn": "arn:aws:lightsail:us-west-2:111122223333:Instance/
bd470fc5-a68b-44c5-8dbc-8EXAMPLEbada",
        "fromBlueprintId": "mean",
```

```
        "fromBundleId": "medium_3_0",
        "isFromAutoSnapshot": false,
        "sizeInGb": 80
    }
}
```

- Per i dettagli sull'API, consulta GetInstanceSnapshot AWS CLICommand Reference.

`get-instance-snapshots`

Il seguente esempio di codice mostra come utilizzare`get-instance-snapshots`.

AWS CLI

Come ottenere le informazioni su tutti gli snapshot delle istanze

L'`get-instance-snapshots`esempio seguente mostra i dettagli su tutte le istantanee delle istanze nella AWS regione configurata.

```
aws lightsail get-instance-snapshots
```

Output:

```
{
    "instanceSnapshots": [
        {
            "name": "MEAN-1-1571421498",
            "arn": "arn:aws:lightsail:us-west-2:111122223333:InstanceSnapshot/
a20e6ebe-b0ee-4ae4-a750-3EXAMPLEcb0c",
            "supportCode": "6EXAMPLE3362/ami-0EXAMPLEe33cabfa1",
            "createdAt": 1571421527.755,
            "location": {
                "availabilityZone": "all",
                "regionName": "us-west-2"
            },
            "resourceType": "InstanceSnapshot",
            "tags": [
                {
                    "key": "no_delete"
                }
            ],
            "state": "available",
```

```
            "fromAttachedDisks": [],
            "fromInstanceName": "MEAN-1",
            "fromInstanceArn": "arn:aws:lightsail:us-
west-2:111122223333:Instance/1761aa0a-6038-4f25-8b94-2EXAMPLE19fd",
            "fromBlueprintId": "wordpress",
            "fromBundleId": "micro_3_0",
            "isFromAutoSnapshot": false,
            "sizeInGb": 40
        },
        {
            "name": "MEAN-1-1571419854",
            "arn": "arn:aws:lightsail:us-west-2:111122223333:InstanceSnapshot/
ac54700c-48a8-40fd-b065-2EXAMPLEac8f",
            "supportCode": "6EXAMPLE3362/ami-0EXAMPLE67a73020d",
            "createdAt": 1571419891.927,
            "location": {
                "availabilityZone": "all",
                "regionName": "us-west-2"
            },
            "resourceType": "InstanceSnapshot",
            "tags": [],
            "state": "available",
            "fromAttachedDisks": [],
            "fromInstanceName": "MEAN-1",
            "fromInstanceArn": "arn:aws:lightsail:us-west-2:111122223333:Instance/
bd470fc5-a68b-44c5-8dbc-8EXAMPLEbada",
            "fromBlueprintId": "mean",
            "fromBundleId": "medium_3_0",
            "isFromAutoSnapshot": false,
            "sizeInGb": 80
        }
    ]
}
```

- Per i dettagli sull'API, vedere GetInstanceSnapshotsin AWS CLI Command Reference.

`get-instance-state`

Il seguente esempio di codice mostra come utilizzare`get-instance-state`.

AWS CLI

Come ottenere le informazioni sullo stato di un'istanza

L'esempio `get-instance-state` seguente restituisce lo stato dell'istanza specificata.

```
aws lightsail get-instance-state \
    --instance-name MEAN-1
```

Output:

```
{
    "state": {
        "code": 16,
        "name": "running"
    }
}
```

- Per i dettagli sull'API, consulta GetInstanceState AWS CLICommand Reference.

`get-instance`

Il seguente esempio di codice mostra come utilizzare`get-instance`.

AWS CLI

Come ottenere le informazioni su un'istanza

L'esempio `get-instance` seguente mostra i dettagli relativi all'istanza `MEAN-1`.

```
aws lightsail get-instance \
    --instance-name MEAN-1
```

Output:

```
{
    "instance": {
        "name": "MEAN-1",
        "arn": "arn:aws:lightsail:us-west-2:111122223333:Instance/bd470fc5-
a68b-44c5-8dbc-EXAMPLE4bada",
        "supportCode": "6EXAMPLE3362/i-05EXAMPLE407c97d3",
        "createdAt": 1570635023.124,
        "location": {
            "availabilityZone": "us-west-2a",
            "regionName": "us-west-2"
        },
```

```
        "resourceType": "Instance",
        "tags": [],
        "blueprintId": "mean",
        "blueprintName": "MEAN",
        "bundleId": "medium_3_0",
        "isStaticIp": false,
        "privateIpAddress": "192.0.2.0",
        "publicIpAddress": "192.0.2.0",
        "hardware": {
            "cpuCount": 2,
            "disks": [
                {
                    "createdAt": 1570635023.124,
                    "sizeInGb": 80,
                    "isSystemDisk": true,
                    "iops": 240,
                    "path": "/dev/xvda",
                    "attachedTo": "MEAN-1",
                    "attachmentState": "attached"
                }
            ],
            "ramSizeInGb": 4.0
        },
        "networking": {
            "monthlyTransfer": {
                "gbPerMonthAllocated": 4096
            },
            "ports": [
                {
                    "fromPort": 80,
                    "toPort": 80,
                    "protocol": "tcp",
                    "accessFrom": "Anywhere (0.0.0.0/0)",
                    "accessType": "public",
                    "commonName": "",
                    "accessDirection": "inbound"
                },
                {
                    "fromPort": 22,
                    "toPort": 22,
                    "protocol": "tcp",
                    "accessFrom": "Anywhere (0.0.0.0/0)",
                    "accessType": "public",
                    "commonName": "",
```

```
                    "accessDirection": "inbound"
                },
                {
                    "fromPort": 443,
                    "toPort": 443,
                    "protocol": "tcp",
                    "accessFrom": "Anywhere (0.0.0.0/0)",
                    "accessType": "public",
                    "commonName": "",
                    "accessDirection": "inbound"
                }
            ]
        },
        "state": {
            "code": 16,
            "name": "running"
        },
        "username": "bitnami",
        "sshKeyName": "MyKey"
    }
}
```

- Per i dettagli sull'API, consulta GetInstance AWS CLICommand Reference.

`get-instances`

Il seguente esempio di codice mostra come utilizzare`get-instances`.

AWS CLI

Come ottenere le informazioni su tutte le istanze

L'`get-instances`esempio seguente mostra i dettagli su tutte le istanze nella AWS regione configurata.

```
aws lightsail get-instances
```

Output:

```
{
    "instances": [
        {
```

```
            "name": "Windows_Server_2022-1",
            "arn": "arn:aws:lightsail:us-
west-2:111122223333:Instance/0f44fbb9-8f55-4e47-a25e-EXAMPLE04763",
            "supportCode": "62EXAMPLE362/i-0bEXAMPLE71a686b9",
            "createdAt": 1571332358.665,
            "location": {
                "availabilityZone": "us-west-2a",
                "regionName": "us-west-2"
            },
            "resourceType": "Instance",
            "tags": [],
            "blueprintId": "windows_server_2022",
            "blueprintName": "Windows Server 2022",
            "bundleId": "large_win_3_0",
            "isStaticIp": false,
            "privateIpAddress": "192.0.2.0",
            "publicIpAddress": "192.0.2.0",
            "hardware": {
                "cpuCount": 1,
                "disks": [
                    {
                        "createdAt": 1571332358.665,
                        "sizeInGb": 160,
                        "isSystemDisk": true,
                        "iops": 180,
                        "path": "/dev/sda1",
                        "attachedTo": "Windows_Server_2022-1",
                        "attachmentState": "attached"
                    },
                    {
                        "name": "my-disk-for-windows-server",
                        "arn": "arn:aws:lightsail:us-
west-2:111122223333:Disk/4123a81c-484c-49ea-afea-5EXAMPLEda87",
                        "supportCode": "6EXAMPLE3362/vol-0EXAMPLEb2b99ca3d",
                        "createdAt": 1571355063.494,
                        "location": {
                            "availabilityZone": "us-west-2a",
                            "regionName": "us-west-2"
                        },
                        "resourceType": "Disk",
                        "tags": [],
                        "sizeInGb": 128,
                        "isSystemDisk": false,
                        "iops": 384,
```

```
                    "path": "/dev/xvdf",
                    "state": "in-use",
                    "attachedTo": "Windows_Server_2022-1",
                    "isAttached": true,
                    "attachmentState": "attached"
                }
            ],
            "ramSizeInGb": 8.0
        },
        "networking": {
            "monthlyTransfer": {
                "gbPerMonthAllocated": 3072
            },
            "ports": [
                {
                    "fromPort": 80,
                    "toPort": 80,
                    "protocol": "tcp",
                    "accessFrom": "Anywhere (0.0.0.0/0)",
                    "accessType": "public",
                    "commonName": "",
                    "accessDirection": "inbound"
                },
                {
                    "fromPort": 22,
                    "toPort": 22,
                    "protocol": "tcp",
                    "accessFrom": "Anywhere (0.0.0.0/0)",
                    "accessType": "public",
                    "commonName": "",
                    "accessDirection": "inbound"
                },
                {
                    "fromPort": 3389,
                    "toPort": 3389,
                    "protocol": "tcp",
                    "accessFrom": "Anywhere (0.0.0.0/0)",
                    "accessType": "public",
                    "commonName": "",
                    "accessDirection": "inbound"
                }
            ]
        },
        "state": {
```

```
                "code": 16,
                "name": "running"
            },
            "username": "Administrator",
            "sshKeyName": "LightsailDefaultKeyPair"
        },
        {
            "name": "MEAN-1",
            "arn": "arn:aws:lightsail:us-west-2:111122223333:Instance/bd470fc5-
a68b-44c5-8dbc-8EXAMPLEbada",
            "supportCode": "6EXAMPLE3362/i-0EXAMPLEa407c97d3",
            "createdAt": 1570635023.124,
            "location": {
                "availabilityZone": "us-west-2a",
                "regionName": "us-west-2"
            },
            "resourceType": "Instance",
            "tags": [],
            "blueprintId": "mean",
            "blueprintName": "MEAN",
            "bundleId": "medium_3_0",
            "isStaticIp": false,
            "privateIpAddress": "192.0.2.0",
            "publicIpAddress": "192.0.2.0",
            "hardware": {
                "cpuCount": 2,
                "disks": [
                    {
                        "name": "Disk-1",
                        "arn": "arn:aws:lightsail:us-west-2:111122223333:Disk/
c21cfb0a-07f2-44ae-9a23-bEXAMPLE8096",
                        "supportCode": "6EXAMPLE3362/vol-0EXAMPLEf2f88b32f",
                        "createdAt": 1566585439.587,
                        "location": {
                            "availabilityZone": "us-west-2a",
                            "regionName": "us-west-2"
                        },
                        "resourceType": "Disk",
                        "tags": [
                            {
                                "key": "test"
                            }
                        ],
                        "sizeInGb": 8,
```

```
                    "isSystemDisk": false,
                    "iops": 240,
                    "path": "/dev/xvdf",
                    "state": "in-use",
                    "attachedTo": "MEAN-1",
                    "isAttached": true,
                    "attachmentState": "attached"
                },
                {
                    "createdAt": 1570635023.124,
                    "sizeInGb": 80,
                    "isSystemDisk": true,
                    "iops": 240,
                    "path": "/dev/sda1",
                    "attachedTo": "MEAN-1",
                    "attachmentState": "attached"
                }
            ],
            "ramSizeInGb": 4.0
        },
        "networking": {
            "monthlyTransfer": {
                "gbPerMonthAllocated": 4096
            },
            "ports": [
                {
                    "fromPort": 80,
                    "toPort": 80,
                    "protocol": "tcp",
                    "accessFrom": "Anywhere (0.0.0.0/0)",
                    "accessType": "public",
                    "commonName": "",
                    "accessDirection": "inbound"
                },
                {
                    "fromPort": 22,
                    "toPort": 22,
                    "protocol": "tcp",
                    "accessFrom": "Anywhere (0.0.0.0/0)",
                    "accessType": "public",
                    "commonName": "",
                    "accessDirection": "inbound"
                },
                {
```

```
                        "fromPort": 443,
                        "toPort": 443,
                        "protocol": "tcp",
                        "accessFrom": "Anywhere (0.0.0.0/0)",
                        "accessType": "public",
                        "commonName": "",
                        "accessDirection": "inbound"
                    }
                ]
            },
            "state": {
                "code": 16,
                "name": "running"
            },
            "username": "bitnami",
            "sshKeyName": "MyTestKey"
        }
    ]
}
```

- Per i dettagli sull'API, vedere GetInstancesin AWS CLI Command Reference.

`get-key-pair`

Il seguente esempio di codice mostra come utilizzare`get-key-pair`.

AWS CLI

Come ottenere le informazioni su una coppia di chiavi

L'esempio `get-key-pair` seguente mostra i dettagli relativi alla coppia di chiavi specificata.

```
aws lightsail get-key-pair \
    --key-pair-name MyKey1
```

Output:

```
{
    "keyPair": {
        "name": "MyKey1",
        "arn": "arn:aws:lightsail:us-
west-2:111122223333:KeyPair/19a4efdf-3054-43d6-91fd-eEXAMPLE21bf",
```

```
        "supportCode": "6EXAMPLE3362/MyKey1",
        "createdAt": 1571255026.975,
        "location": {
            "availabilityZone": "all",
            "regionName": "us-west-2"
        },
        "resourceType": "KeyPair",
        "tags": [],
        "fingerprint": "00:11:22:33:44:55:66:77:88:99:aa:bb:cc:dd:ee:ff:gg:hh:ii:jj"
    }
}
```

- Per i dettagli sull'API, consulta GetKeyPair AWS CLICommand Reference.

`get-key-pairs`

Il seguente esempio di codice mostra come utilizzare`get-key-pairs`.

AWS CLI

Come ottenere le informazioni su tutte le coppie di chiavi

L'`get-key-pairs`esempio seguente mostra i dettagli su tutte le coppie di chiavi nella AWS regione configurata.

```
aws lightsail get-key-pairs
```

Output:

```
{
    "keyPairs": [
        {
            "name": "MyKey1",
            "arn": "arn:aws:lightsail:us-
west-2:111122223333:KeyPair/19a4efdf-3054-43d6-91fd-eEXAMPLE21bf",
            "supportCode": "6EXAMPLE3362/MyKey1",
            "createdAt": 1571255026.975,
            "location": {
                "availabilityZone": "all",
                "regionName": "us-west-2"
            },
            "resourceType": "KeyPair",
```

```
            "tags": [],
            "fingerprint":
 "00:11:22:33:44:55:66:77:88:99:aa:bb:cc:dd:ee:ff:gg:hh:ii:jj"
        }
    ]
}
```

- Per i dettagli sull'API, vedere GetKeyPairsin AWS CLI Command Reference.

`get-load-balancer-tls-certificates`

Il seguente esempio di codice mostra come utilizzare`get-load-balancer-tls-certificates`.

AWS CLI

Come ottenere le informazioni sui certificati TLS per un bilanciatore del carico

L'esempio `get-load-balancer-tls-certificates` seguente mostra i dettagli relativi ai certificati TLS per il bilanciatore del carico specificato.

```
aws lightsail get-load-balancer-tls-certificates \
    --load-balancer-name LoadBalancer-1
```

Output:

```
{
    "tlsCertificates": [
        {
            "name": "example-com",
            "arn": "arn:aws:lightsail:us-
west-2:111122223333:LoadBalancerTlsCertificate/d7bf4643-6a02-4cd4-b3c4-
fEXAMPLE9b4d",
            "supportCode": "6EXAMPLE3362/arn:aws:acm:us-
west-2:333322221111:certificate/9af8e32c-a54e-4a67-8c63-cEXAMPLEb314",
            "createdAt": 1571678025.3,
            "location": {
                "availabilityZone": "all",
                "regionName": "us-west-2"
            },
            "resourceType": "LoadBalancerTlsCertificate",
            "loadBalancerName": "LoadBalancer-1",
            "isAttached": false,
```

```
            "status": "ISSUED",
            "domainName": "example.com",
            "domainValidationRecords": [
                {
                    "name": "_dEXAMPLE4ede046a0319eb44a4eb3cbc.example.com.",
                    "type": "CNAME",
                    "value": "_bEXAMPLE0899fb7b6bf79d9741d1a383.hkvuiqjoua.acm-
validations.aws.",
                    "validationStatus": "SUCCESS",
                    "domainName": "example.com"
                }
            ],
            "issuedAt": 1571678070.0,
            "issuer": "Amazon",
            "keyAlgorithm": "RSA-2048",
            "notAfter": 1605960000.0,
            "notBefore": 1571616000.0,
            "serial": "00:11:22:33:44:55:66:77:88:99:aa:bb:cc:dd:ee:ff",
            "signatureAlgorithm": "SHA256WITHRSA",
            "subject": "CN=example.com",
            "subjectAlternativeNames": [
                "example.com"
            ]
        }
    ]
}
```

- Per i dettagli sull'API, consulta GetLoadBalancerTlsCertificates AWS CLICommand Reference.

`get-load-balancer`

Il seguente esempio di codice mostra come utilizzare`get-load-balancer`.

AWS CLI

Come ottenere le informazioni su un bilanciatore del carico

L'esempio `get-load-balancer` seguente mostra i dettagli relativi al bilanciatore del carico specificato.

```
aws lightsail get-load-balancer \
    --load-balancer-name LoadBalancer-1
```

Output:


```
{
    "loadBalancer": {
        "name": "LoadBalancer-1",
        "arn": "arn:aws:lightsail:us-
west-2:111122223333:LoadBalancer/40486b2b-1ad0-4152-83e4-cEXAMPLE6f4b",
        "supportCode": "6EXAMPLE3362/arn:aws:elasticloadbalancing:us-
west-2:333322221111:loadbalancer/app/
bEXAMPLE128cb59d86f946a9395dd304/1EXAMPLE8dd9d77e",
        "createdAt": 1571677906.723,
        "location": {
            "availabilityZone": "all",
            "regionName": "us-west-2"
        },
        "resourceType": "LoadBalancer",
        "tags": [],
        "dnsName": "bEXAMPLE128cb59d86f946a9395dd304-1486911371.us-
west-2.elb.amazonaws.com",
        "state": "active",
        "protocol": "HTTP",
        "publicPorts": [
            80
        ],
        "healthCheckPath": "/",
        "instancePort": 80,
        "instanceHealthSummary": [
            {
                "instanceName": "MEAN-3",
                "instanceHealth": "healthy"
            },
            {
                "instanceName": "MEAN-1",
                "instanceHealth": "healthy"
            },
            {
                "instanceName": "MEAN-2",
                "instanceHealth": "healthy"
            }
        ],
        "tlsCertificateSummaries": [
            {
                "name": "example-com",
                "isAttached": false
```

```
            }
        ],
        "configurationOptions": {
            "SessionStickinessEnabled": "false",
            "SessionStickiness_LB_CookieDurationSeconds": "86400"
        }
    }
}
```

- Per i dettagli sull'API, consulta GetLoadBalancer AWS CLICommand Reference.

`get-load-balancers`

Il seguente esempio di codice mostra come utilizzare`get-load-balancers`.

AWS CLI

Come ottenere le informazioni su tutti i bilanciatori del carico

L'`get-load-balancers`esempio seguente mostra i dettagli su tutti i sistemi di bilanciamento del carico nella regione AWS configurata.

```
aws lightsail get-load-balancers
```

Output:

```
{
    "loadBalancers": [
        {
            "name": "LoadBalancer-1",
            "arn": "arn:aws:lightsail:us-
west-2:111122223333:LoadBalancer/40486b2b-1ad0-4152-83e4-cEXAMPLE6f4b",
            "supportCode": "6EXAMPLE3362/arn:aws:elasticloadbalancing:us-
west-2:333322221111:loadbalancer/app/
bEXAMPLE128cb59d86f946a9395dd304/1EXAMPLE8dd9d77e",
            "createdAt": 1571677906.723,
            "location": {
                "availabilityZone": "all",
                "regionName": "us-west-2"
            },
            "resourceType": "LoadBalancer",
```

```
            "tags": [],
            "dnsName": "bEXAMPLE128cb59d86f946a9395dd304-1486911371.us-
west-2.elb.amazonaws.com",
            "state": "active",
            "protocol": "HTTP",
            "publicPorts": [
                80
            ],
            "healthCheckPath": "/",
            "instancePort": 80,
            "instanceHealthSummary": [
                {
                    "instanceName": "MEAN-3",
                    "instanceHealth": "healthy"
                },
                {
                    "instanceName": "MEAN-1",
                    "instanceHealth": "healthy"
                },
                {
                    "instanceName": "MEAN-2",
                    "instanceHealth": "healthy"
                }
            ],
            "tlsCertificateSummaries": [
                {
                    "name": "example-com",
                    "isAttached": false
                }
            ],
            "configurationOptions": {
                "SessionStickinessEnabled": "false",
                "SessionStickiness_LB_CookieDurationSeconds": "86400"
            }
        }
    ]
}
```

- Per i dettagli sull'API, vedere GetLoadBalancersin AWS CLI Command Reference.

`get-operation`

Il seguente esempio di codice mostra come utilizzare`get-operation`.

AWS CLI

Come ottenere le informazioni su una singola operazione

L'esempio `get-operation` seguente mostra i dettagli relativi all'operazione specificata.

```
aws lightsail get-operation \
    --operation-id e5700e8a-daf2-4b49-bc01-3EXAMPLE910a
```

Output:

```
{
    "operation": {
        "id": "e5700e8a-daf2-4b49-bc01-3EXAMPLE910a",
        "resourceName": "Instance-1",
        "resourceType": "Instance",
        "createdAt": 1571679872.404,
        "location": {
            "availabilityZone": "us-west-2a",
            "regionName": "us-west-2"
        },
        "isTerminal": true,
        "operationType": "CreateInstance",
        "status": "Succeeded",
        "statusChangedAt": 1571679890.304
    }
}
```

- Per i dettagli sull'API, consulta GetOperation AWS CLICommand Reference.

`get-operations-for-resource`

Il seguente esempio di codice mostra come utilizzare`get-operations-for-resource`.

AWS CLI

Come ottenere tutte le operazioni per una risorsa

L'esempio `get-operations-for-resource` seguente mostra i dettagli relativi a tutte le operazioni per la risorsa specificata.

```
aws lightsail get-operations-for-resource \
```

```
    --resource-name LoadBalancer-1
```

Output:

```
{
    "operations": [
        {
            "id": "e2973046-43f8-4252-a4b4-9EXAMPLE69ce",
            "resourceName": "LoadBalancer-1",
            "resourceType": "LoadBalancer",
            "createdAt": 1571678786.071,
            "location": {
                "availabilityZone": "all",
                "regionName": "us-west-2"
            },
            "isTerminal": true,
            "operationDetails": "MEAN-1",
            "operationType": "DetachInstancesFromLoadBalancer",
            "status": "Succeeded",
            "statusChangedAt": 1571679087.57
        },
        {
            "id": "2d742a18-0e7f-48c8-9705-3EXAMPLEf98a",
            "resourceName": "LoadBalancer-1",
            "resourceType": "LoadBalancer",
            "createdAt": 1571678782.784,
            "location": {
                "availabilityZone": "all",
                "regionName": "us-west-2"
            },
            "isTerminal": true,
            "operationDetails": "MEAN-1",
            "operationType": "AttachInstancesToLoadBalancer",
            "status": "Succeeded",
            "statusChangedAt": 1571678798.465
        },
        {
            "id": "6c700fcc-4246-40ab-952b-1EXAMPLEdac2",
            "resourceName": "LoadBalancer-1",
            "resourceType": "LoadBalancer",
            "createdAt": 1571678775.297,
            "location": {
                "availabilityZone": "all",
```

```
                "regionName": "us-west-2"
            },
            "isTerminal": true,
            "operationDetails": "MEAN-3",
            "operationType": "AttachInstancesToLoadBalancer",
            "status": "Succeeded",
            "statusChangedAt": 1571678842.806
        },
        ...
        }
    ]
}
```

- Per i dettagli sull'API, consulta GetOperationsForResource AWS CLICommand Reference.

`get-operations`

Il seguente esempio di codice mostra come utilizzare`get-operations`.

AWS CLI

Come ottenere le informazioni su tutte le operazioni

L'`get-operations`esempio seguente mostra i dettagli su tutte le operazioni nella AWS regione configurata.

```
aws lightsail get-operations
```

Output:

```
{
    "operations": [
        {
            "id": "e5700e8a-daf2-4b49-bc01-3EXAMPLE910a",
            "resourceName": "Instance-1",
            "resourceType": "Instance",
            "createdAt": 1571679872.404,
            "location": {
                "availabilityZone": "us-west-2a",
                "regionName": "us-west-2"
            },
```

```
            "isTerminal": true,
            "operationType": "CreateInstance",
            "status": "Succeeded",
            "statusChangedAt": 1571679890.304
        },
        {
            "id": "701a3339-930e-4914-a9f9-7EXAMPLE68d7",
            "resourceName": "WordPress-1",
            "resourceType": "Instance",
            "createdAt": 1571678786.072,
            "location": {
                "availabilityZone": "all",
                "regionName": "us-west-2"
            },
            "isTerminal": true,
            "operationDetails": "LoadBalancer-1",
            "operationType": "DetachInstancesFromLoadBalancer",
            "status": "Succeeded",
            "statusChangedAt": 1571679086.399
        },
        {
            "id": "e2973046-43f8-4252-a4b4-9EXAMPLE69ce",
            "resourceName": "LoadBalancer-1",
            "resourceType": "LoadBalancer",
            "createdAt": 1571678786.071,
            "location": {
                "availabilityZone": "all",
                "regionName": "us-west-2"
            },
            "isTerminal": true,
            "operationDetails": "WordPress-1",
            "operationType": "DetachInstancesFromLoadBalancer",
            "status": "Succeeded",
            "statusChangedAt": 1571679087.57
        },
        ...
        }
    ]
}
```

- Per i dettagli sull'API, vedere GetOperationsin AWS CLI Command Reference.

`get-regions`

Il seguente esempio di codice mostra come utilizzare`get-regions`.

AWS CLI

Per ottenere tutte le AWS regioni per Amazon Lightsail

L'`get-regions`esempio seguente mostra i dettagli su tutte le AWS regioni per Amazon Lightsail.

```
aws lightsail get-regions
```

Output:

```
{
    "regions": [
        {
            "continentCode": "NA",
            "description": "This region is recommended to serve users in the eastern
 United States",
            "displayName": "Virginia",
            "name": "us-east-1",
            "availabilityZones": [],
            "relationalDatabaseAvailabilityZones": []
        },
        {
            "continentCode": "NA",
            "description": "This region is recommended to serve users in the eastern
 United States",
            "displayName": "Ohio",
            "name": "us-east-2",
            "availabilityZones": [],
            "relationalDatabaseAvailabilityZones": []
        },
        {
            "continentCode": "NA",
            "description": "This region is recommended to serve users in the
 northwestern United States, Alaska, and western Canada",
            "displayName": "Oregon",
            "name": "us-west-2",
            "availabilityZones": [],
            "relationalDatabaseAvailabilityZones": []
        },
        ...
```

```
        }
    ]
}
```

- Per i dettagli sull'API, consulta AWS CLI Command GetRegionsReference.

`get-relational-database-blueprints`

Il seguente esempio di codice mostra come utilizzare`get-relational-database-blueprints`.

AWS CLI

Come ottenere i blueprint per i nuovi database relazionali

L'`get-relational-database-blueprints`esempio seguente mostra i dettagli su tutti i blueprint di database relazionali disponibili che possono essere utilizzati per creare nuovi database relazionali in Amazon Lightsail.

```
aws lightsail get-relational-database-blueprints
```

Output:

```
{
    "blueprints": [
        {
            "blueprintId": "mysql_5_6",
            "engine": "mysql",
            "engineVersion": "5.6.44",
            "engineDescription": "MySQL Community Edition",
            "engineVersionDescription": "MySQL 5.6.44",
            "isEngineDefault": false
        },
        {
            "blueprintId": "mysql_5_7",
            "engine": "mysql",
            "engineVersion": "5.7.26",
            "engineDescription": "MySQL Community Edition",
            "engineVersionDescription": "MySQL 5.7.26",
            "isEngineDefault": true
        },
        {
            "blueprintId": "mysql_8_0",
```

```
            "engine": "mysql",
            "engineVersion": "8.0.16",
            "engineDescription": "MySQL Community Edition",
            "engineVersionDescription": "MySQL 8.0.16",
            "isEngineDefault": false
        },
        {
            "blueprintId": "postgres_9_6",
            "engine": "postgres",
            "engineVersion": "9.6.15",
            "engineDescription": "PostgreSQL",
            "engineVersionDescription": "PostgreSQL 9.6.15-R1",
            "isEngineDefault": false
        },
        {
            "blueprintId": "postgres_10",
            "engine": "postgres",
            "engineVersion": "10.10",
            "engineDescription": "PostgreSQL",
            "engineVersionDescription": "PostgreSQL 10.10-R1",
            "isEngineDefault": false
        },
        {
            "blueprintId": "postgres_11",
            "engine": "postgres",
            "engineVersion": "11.5",
            "engineDescription": "PostgreSQL",
            "engineVersionDescription": "PostgreSQL 11.5-R1",
            "isEngineDefault": true
        }
    ]
}
```

- Per i dettagli sulle API, consulta Command Reference. GetRelationalDatabaseBlueprintsAWS CLI

`get-relational-database-bundles`

Il seguente esempio di codice mostra come utilizzare`get-relational-database-bundles`.

AWS CLI

Come ottenere i bundle per i nuovi database relazionali

L'`get-relational-database-bundles`esempio seguente mostra i dettagli su tutti i pacchetti di database relazionali disponibili che possono essere utilizzati per creare nuovi database relazionali in Amazon Lightsail. È opportuno notare che la risposta non include i bundle inattivi perché il flag `--include-inactive` non è specificato nel comando. Non è possibile utilizzare bundle inattivi per creare nuovi database relazionali.

```
aws lightsail get-relational-database-bundles
```

Output:

```
{
    "bundles": [
        {
            "bundleId": "micro_2_0",
            "name": "Micro",
            "price": 15.0,
            "ramSizeInGb": 1.0,
            "diskSizeInGb": 40,
            "transferPerMonthInGb": 100,
            "cpuCount": 2,
            "isEncrypted": true,
            "isActive": true
        },
        {
            "bundleId": "micro_ha_2_0",
            "name": "Micro with High Availability",
            "price": 30.0,
            "ramSizeInGb": 1.0,
            "diskSizeInGb": 40,
            "transferPerMonthInGb": 100,
            "cpuCount": 2,
            "isEncrypted": true,
            "isActive": true
        },
        {
            "bundleId": "small_2_0",
            "name": "Small",
            "price": 30.0,
            "ramSizeInGb": 2.0,
            "diskSizeInGb": 80,
            "transferPerMonthInGb": 100,
            "cpuCount": 2,
```

```
            "isEncrypted": true,
            "isActive": true
        },
        {
            "bundleId": "small_ha_2_0",
            "name": "Small with High Availability",
            "price": 60.0,
            "ramSizeInGb": 2.0,
            "diskSizeInGb": 80,
            "transferPerMonthInGb": 100,
            "cpuCount": 2,
            "isEncrypted": true,
            "isActive": true
        },
        {
            "bundleId": "medium_2_0",
            "name": "Medium",
            "price": 60.0,
            "ramSizeInGb": 4.0,
            "diskSizeInGb": 120,
            "transferPerMonthInGb": 100,
            "cpuCount": 2,
            "isEncrypted": true,
            "isActive": true
        },
        {
            "bundleId": "medium_ha_2_0",
            "name": "Medium with High Availability",
            "price": 120.0,
            "ramSizeInGb": 4.0,
            "diskSizeInGb": 120,
            "transferPerMonthInGb": 100,
            "cpuCount": 2,
            "isEncrypted": true,
            "isActive": true
        },
        {
            "bundleId": "large_2_0",
            "name": "Large",
            "price": 115.0,
            "ramSizeInGb": 8.0,
            "diskSizeInGb": 240,
            "transferPerMonthInGb": 200,
            "cpuCount": 2,
```

```
            "isEncrypted": true,
            "isActive": true
        },
        {
            "bundleId": "large_ha_2_0",
            "name": "Large with High Availability",
            "price": 230.0,
            "ramSizeInGb": 8.0,
            "diskSizeInGb": 240,
            "transferPerMonthInGb": 200,
            "cpuCount": 2,
            "isEncrypted": true,
            "isActive": true
        }
    ]
}
```

Per ulteriori informazioni, consulta Creazione di un database in Amazon Lightsail nella Amazon Lightsail Developer Guide.

- Per i dettagli sull'API, consulta GetRelationalDatabaseBundlesCommand Reference.AWS CLI

`get-relational-database-events`

Il seguente esempio di codice mostra come utilizzare`get-relational-database-events`.

AWS CLI

Come ottenere gli eventi per un database relazionale

L'esempio `get-relational-database-events` seguente mostra i dettagli relativi agli eventi delle ultime 17 ore (1.020 minuti) per il database relazionale specificato.

```
aws lightsail get-relational-database-events \
    --relational-database-name Database-1 \
    --duration-in-minutes 1020
```

Output:

```
{
    "relationalDatabaseEvents": [
        {
```

```
            "resource": "Database-1",
            "createdAt": 1571654146.553,
            "message": "Backing up Relational Database",
            "eventCategories": [
                "backup"
            ]
        },
        {
            "resource": "Database-1",
            "createdAt": 1571654249.98,
            "message": "Finished Relational Database backup",
            "eventCategories": [
                "backup"
            ]
        }
    ]
}
```

- Per i dettagli sull'API, consulta GetRelationalDatabaseEvents AWS CLICommand Reference.

`get-relational-database-log-events`

Il seguente esempio di codice mostra come utilizzare`get-relational-database-log-events`.

AWS CLI

Come ottenere gli eventi di log per un database relazionale

L'esempio `get-relational-database-log-events` seguente mostra i dettagli relativi al log specificato tra `1570733176` e `1571597176` per il database relazionale `Database1`. Le informazioni restituite sono configurate in modo da iniziare da `head`.

Per identificare l'ora di inizio e quella di fine, è consigliabile utilizzare un convertitore di data e ora Unix.

```
aws lightsail get-relational-database-log-events \
    --relational-database-name Database1 \
    --log-stream-name error \
    --start-from-head \
    --start-time 1570733176 \
    --end-time 1571597176
```

Output:


```
{
    "resourceLogEvents": [
        {
            "createdAt": 1570820267.0,
            "message": "2019-10-11 18:57:47 20969 [Warning] IP address '192.0.2.0'
 could not be resolved: Name or service not known"
        },
        {
            "createdAt": 1570860974.0,
            "message": "2019-10-12 06:16:14 20969 [Warning] IP address '8192.0.2.0'
 could not be resolved: Temporary failure in name resolution"
        },
        {
            "createdAt": 1570860977.0,
            "message": "2019-10-12 06:16:17 20969 [Warning] IP address '192.0.2.0'
 could not be resolved: Temporary failure in name resolution"
        },
        {
            "createdAt": 1570860979.0,
            "message": "2019-10-12 06:16:19 20969 [Warning] IP address '192.0.2.0'
 could not be resolved: Temporary failure in name resolution"
        },
        {
            "createdAt": 1570860981.0,
            "message": "2019-10-12 06:16:21 20969 [Warning] IP address '192.0.2.0'
 could not be resolved: Temporary failure in name resolution"
        },
        {
            "createdAt": 1570860982.0,
            "message": "2019-10-12 06:16:22 20969 [Warning] IP address '192.0.2.0'
 could not be resolved: Temporary failure in name resolution"
        },
        {
            "createdAt": 1570860984.0,
            "message": "2019-10-12 06:16:24 20969 [Warning] IP address '192.0.2.0'
 could not be resolved: Temporary failure in name resolution"
        },
        {
            "createdAt": 1570860986.0,
            "message": "2019-10-12 06:16:26 20969 [Warning] IP address '192.0.2.0'
 could not be resolved: Temporary failure in name resolution"
        },
```

```
        ...
        }
    ],
    "nextBackwardToken":
 "eEXAMPLEZXJUZXh0IjoiZnRWb3F3cUpRSlQ5NndMYThxelRUZlFhR3J6c2dKWEEvM2kvajZMZzVVVWpqRDN0YjFXTj
    "nextForwardToken":
 "eEXAMPLEZXJUZXh0IjoiT09Lb0Z6ZFRJbHhaNEQ5N2tPbkkwRmwwNUxPZjFTbFFwUklQbzlSaWgvMWVXbEk4aG56VH
}
```

- Per i dettagli sull'API, consulta GetRelationalDatabaseLogEvents AWS CLICommand Reference.

`get-relational-database-log-streams`

Il seguente esempio di codice mostra come utilizzare`get-relational-database-log-streams`.

AWS CLI

Come ottenere i flussi di log per un database relazionale

L'esempio `get-relational-database-log-streams` seguente restituisce tutti i flussi di log disponibili per il database relazionale specificato.

```
aws lightsail get-relational-database-log-streams \
--relational-database-name Database1
```

Output:

```
{
    "logStreams": [
        "audit",
        "error",
        "general",
        "slowquery"
    ]
}
```

- Per i dettagli sull'API, consulta GetRelationalDatabaseLogStreams AWS CLICommand Reference.

`get-relational-database-master-user-password`

Il seguente esempio di codice mostra come utilizzare`get-relational-database-master-user-password`.

AWS CLI

Come ottenere la password dell'utente master per un database relazionale

L'esempio `get-relational-database-master-user-password` seguente restituisce le informazioni relative alla password dell'utente master per il database relazionale specificato.

```
aws lightsail get-relational-database-master-user-password \
    --relational-database-name Database-1
```

Output:

```
{
    "masterUserPassword": "VEXAMPLEec.9qvx,_t<)Wkf)kwboM,>2",
    "createdAt": 1571259453.959
}
```

- Per i dettagli sull'API, consulta GetRelationalDatabaseMasterUserPassword AWS CLICommand Reference.

`get-relational-database-metric-data`

Il seguente esempio di codice mostra come utilizzare`get-relational-database-metric-data`.

AWS CLI

Come ottenere i dati delle metriche per un database relazionale

L'esempio `get-relational-database-metric-data` seguente restituisce il risultato del conteggio della metrica `DatabaseConnections` nel periodo di 24 ore (`86400` secondi) tra `1570733176` e `1571597176` per il database relazionale `Database1`.

Per identificare l'ora di inizio e quella di fine, è consigliabile utilizzare un convertitore di data e ora Unix.

```
aws lightsail get-relational-database-metric-data \
    --relational-database-name Database1 \
    --metric-name DatabaseConnections \
    --period 86400 \
    --start-time 1570733176 \
    --end-time 1571597176 \
    --unit Count \
    --statistics Sum
```

Output:

```
{
    "metricName": "DatabaseConnections",
    "metricData": [
        {
            "sum": 1.0,
            "timestamp": 1571510760.0,
            "unit": "Count"
        },
        {
            "sum": 1.0,
            "timestamp": 1570733160.0,
            "unit": "Count"
        },
        {
            "sum": 1.0,
            "timestamp": 1570992360.0,
            "unit": "Count"
        },
        {
            "sum": 0.0,
            "timestamp": 1571251560.0,
            "unit": "Count"
        },
        {
            "sum": 721.0,
            "timestamp": 1570819560.0,
            "unit": "Count"
        },
        {
            "sum": 1.0,
            "timestamp": 1571078760.0,
            "unit": "Count"
```

```
        },
        {
            "sum": 2.0,
            "timestamp": 1571337960.0,
            "unit": "Count"
        },
        {
            "sum": 684.0,
            "timestamp": 1570905960.0,
            "unit": "Count"
        },
        {
            "sum": 0.0,
            "timestamp": 1571165160.0,
            "unit": "Count"
        },
        {
            "sum": 1.0,
            "timestamp": 1571424360.0,
            "unit": "Count"
        }
    ]
}
```

- Per i dettagli sull'API, consulta GetRelationalDatabaseMetricData AWS CLICommand Reference.

`get-relational-database-parameters`

Il seguente esempio di codice mostra come utilizzare`get-relational-database-parameters`.

AWS CLI

Come ottenere i parametri per un database relazionale

L'esempio `get-relational-database-parameters` seguente restituisce le informazioni relative a tutti i parametri disponibili per il database relazionale specificato.

```
aws lightsail get-relational-database-parameters \
    --relational-database-name Database-1
```

Output:

```
{
    "parameters": [
        {
            "allowedValues": "0,1",
            "applyMethod": "pending-reboot",
            "applyType": "dynamic",
            "dataType": "boolean",
            "description": "Automatically set all granted roles as active after the
 user has authenticated successfully.",
            "isModifiable": true,
            "parameterName": "activate_all_roles_on_login",
            "parameterValue": "0"
        },
        {
            "allowedValues": "0,1",
            "applyMethod": "pending-reboot",
            "applyType": "static",
            "dataType": "boolean",
            "description": "Controls whether user-defined functions that have only
 an xxx symbol for the main function can be loaded",
            "isModifiable": false,
            "parameterName": "allow-suspicious-udfs"
        },
        {
            "allowedValues": "0,1",
            "applyMethod": "pending-reboot",
            "applyType": "dynamic",
            "dataType": "boolean",
            "description": "Sets the autocommit mode",
            "isModifiable": true,
            "parameterName": "autocommit"
        },
        {
            "allowedValues": "0,1",
            "applyMethod": "pending-reboot",
            "applyType": "static",
            "dataType": "boolean",
            "description": "Controls whether the server autogenerates SSL key and
 certificate files in the data directory, if they do not already exist.",
            "isModifiable": false,
            "parameterName": "auto_generate_certs"
        },
        ...
```

```
        }
    ]
}
```

Per ulteriori informazioni, consulta Aggiornamento dei parametri del database in Amazon Lightsail nella Lightsail Dev Guide.

- Per i dettagli sull'API, consulta GetRelationalDatabaseParametersCommand Reference.AWS CLI

`get-relational-database-snapshot`

Il seguente esempio di codice mostra come utilizzare`get-relational-database-snapshot`.

AWS CLI

Come ottenere le informazioni su uno snapshot di database relazionale

L'esempio `get-relational-database-snapshot` seguente mostra i dettagli relativi allo snapshot di database relazionale specificato.

```
aws lightsail get-relational-database-snapshot \
    --relational-database-snapshot-name Database-1-1571350042
```

Output:

```
{
    "relationalDatabaseSnapshot": {
        "name": "Database-1-1571350042",
        "arn": "arn:aws:lightsail:us-
west-2:111122223333:RelationalDatabaseSnapshot/0389bbad-4b85-4c3d-9EXAMPLEaee3643d2",
        "supportCode": "6EXAMPLE3362/ls-8EXAMPLE2ba7ad041451946fafc2ad19cfbd9eb2",
        "createdAt": 1571350046.238,
        "location": {
            "availabilityZone": "all",
            "regionName": "us-west-2"
        },
        "resourceType": "RelationalDatabaseSnapshot",
        "tags": [],
        "engine": "mysql",
        "engineVersion": "8.0.16",
```

```
        "sizeInGb": 40,
        "state": "available",
        "fromRelationalDatabaseName": "Database-1",
        "fromRelationalDatabaseArn": "arn:aws:lightsail:us-
west-2:111122223333:RelationalDatabase/7ea932b1-b85a-4bd5-9b3e-bEXAMPLE8cc4",
        "fromRelationalDatabaseBundleId": "micro_1_0",
        "fromRelationalDatabaseBlueprintId": "mysql_8_0"
    }
}
```

- Per i dettagli sull'API, consulta GetRelationalDatabaseSnapshot AWS CLICommand Reference.

`get-relational-database-snapshots`

Il seguente esempio di codice mostra come utilizzare`get-relational-database-snapshots`.

AWS CLI

Come ottenere le informazioni su tutti gli snapshot di database relazionale

L'`get-relational-database-snapshots`esempio seguente visualizza i dettagli su tutte le istantanee del database relazionale nella regione AWS configurata.

```
aws lightsail get-relational-database-snapshots
```

Output:

```
{
    "relationalDatabaseSnapshots": [
        {
            "name": "Database-1-1571350042",
            "arn": "arn:aws:lightsail:us-
west-2:111122223333:RelationalDatabaseSnapshot/0389bbad-4b85-4c3d-9861-6EXAMPLE43d2",
            "supportCode": "6EXAMPLE3362/
ls-8EXAMPLE2ba7ad041451946fafc2ad19cfbd9eb2",
            "createdAt": 1571350046.238,
            "location": {
                "availabilityZone": "all",
                "regionName": "us-west-2"
            },
            "resourceType": "RelationalDatabaseSnapshot",
            "tags": [],
```

```
            "engine": "mysql",
            "engineVersion": "8.0.16",
            "sizeInGb": 40,
            "state": "available",
            "fromRelationalDatabaseName": "Database-1",
            "fromRelationalDatabaseArn": "arn:aws:lightsail:us-
west-2:111122223333:RelationalDatabase/7ea932b1-b85a-4bd5-9b3e-bEXAMPLE8cc4",
            "fromRelationalDatabaseBundleId": "micro_1_0",
            "fromRelationalDatabaseBlueprintId": "mysql_8_0"
        },
        {
            "name": "Database1-Console",
            "arn": "arn:aws:lightsail:us-
west-2:111122223333:RelationalDatabaseSnapshot/8b94136e-06ec-4b1a-
a3fb-5EXAMPLEe1e9",
            "supportCode": "6EXAMPLE3362/
ls-9EXAMPLE14b000d34c8d1c432734e137612d5b5c",
            "createdAt": 1571249981.025,
            "location": {
                "availabilityZone": "all",
                "regionName": "us-west-2"
            },
            "resourceType": "RelationalDatabaseSnapshot",
            "tags": [
                {
                    "key": "test"
                }
            ],
            "engine": "mysql",
            "engineVersion": "5.6.44",
            "sizeInGb": 40,
            "state": "available",
            "fromRelationalDatabaseName": "Database1",
            "fromRelationalDatabaseArn": "arn:aws:lightsail:us-
west-2:111122223333:RelationalDatabase/a6161cb7-4535-4f16-9dcf-8EXAMPLE3d4e",
            "fromRelationalDatabaseBundleId": "micro_1_0",
            "fromRelationalDatabaseBlueprintId": "mysql_5_6"
        }
    ]
}
```

- Per i dettagli sull'API, vedere GetRelationalDatabaseSnapshotsin AWS CLI Command Reference.

`get-relational-database`

Il seguente esempio di codice mostra come utilizzare`get-relational-database`.

AWS CLI

Come ottenere le informazioni su un database relazionale

L'esempio `get-relational-database` seguente mostra i dettagli relativi al database relazionale specificato.

```
aws lightsail get-relational-database \
    --relational-database-name Database-1
```

Output:

```
{
    "relationalDatabase": {
        "name": "Database-1",
        "arn": "arn:aws:lightsail:us-
west-2:111122223333:RelationalDatabase/7ea932b1-b85a-4bd5-9b3e-bEXAMPLE8cc4",
        "supportCode": "6EXAMPLE3362/ls-9EXAMPLE8ad863723b62cc8901a8aa6e794ae0d2",
        "createdAt": 1571259453.795,
        "location": {
            "availabilityZone": "us-west-2a",
            "regionName": "us-west-2"
        },
        "resourceType": "RelationalDatabase",
        "tags": [],
        "relationalDatabaseBlueprintId": "mysql_8_0",
        "relationalDatabaseBundleId": "micro_1_0",
        "masterDatabaseName": "dbmaster",
        "hardware": {
            "cpuCount": 1,
            "diskSizeInGb": 40,
            "ramSizeInGb": 1.0
        },
        "state": "available",
        "backupRetentionEnabled": false,
        "pendingModifiedValues": {},
        "engine": "mysql",
        "engineVersion": "8.0.16",
        "masterUsername": "dbmasteruser",
```

```
        "parameterApplyStatus": "in-sync",
        "preferredBackupWindow": "10:01-10:31",
        "preferredMaintenanceWindow": "sat:11:14-sat:11:44",
        "publiclyAccessible": true,
        "masterEndpoint": {
            "port": 3306,
            "address": "ls-9EXAMPLE8ad863723b62ccEXAMPLEa6e794ae0d2.czowadgeezqi.us-
west-2.rds.amazonaws.com"
        },
        "pendingMaintenanceActions": []
    }
}
```

- Per i dettagli sull'API, consulta GetRelationalDatabase AWS CLICommand Reference.

`get-relational-databases`

Il seguente esempio di codice mostra come utilizzare`get-relational-databases`.

AWS CLI

Come ottenere le informazioni su tutti i database relazionali

L'`get-relational-databases`esempio seguente visualizza i dettagli su tutti i database relazionali nella AWS regione configurata.

```
aws lightsail get-relational-databases
```

Output:

```
{
    "relationalDatabases": [
        {
            "name": "MySQL",
            "arn": "arn:aws:lightsail:us-
west-2:111122223333:RelationalDatabase/8529020c-3ab9-4d51-92af-5EXAMPLE8979",
            "supportCode": "6EXAMPLE3362/
ls-3EXAMPLEa995d8c3b06b4501356e5f2f28e1aeba",
            "createdAt": 1554306019.155,
            "location": {
                "availabilityZone": "us-west-2a",
                "regionName": "us-west-2"
```

```
            },
            "resourceType": "RelationalDatabase",
            "tags": [],
            "relationalDatabaseBlueprintId": "mysql_8_0",
            "relationalDatabaseBundleId": "micro_1_0",
            "masterDatabaseName": "dbmaster",
            "hardware": {
                "cpuCount": 1,
                "diskSizeInGb": 40,
                "ramSizeInGb": 1.0
            },
            "state": "available",
            "backupRetentionEnabled": true,
            "pendingModifiedValues": {},
            "engine": "mysql",
            "engineVersion": "8.0.15",
            "latestRestorableTime": 1571686200.0,
            "masterUsername": "dbmasteruser",
            "parameterApplyStatus": "in-sync",
            "preferredBackupWindow": "07:51-08:21",
            "preferredMaintenanceWindow": "tue:12:18-tue:12:48",
            "publiclyAccessible": true,
            "masterEndpoint": {
                "port": 3306,
                "address":
 "ls-3EXAMPLEa995d8c3b06b4501356e5f2fEXAMPLEa.czowadgeezqi.us-
west-2.rds.amazonaws.com"
            },
            "pendingMaintenanceActions": []
        },
        {
            "name": "Postgres",
            "arn": "arn:aws:lightsail:us-west-2:111122223333:RelationalDatabase/
e9780b6b-d0ab-4af2-85f1-1EXAMPLEac68",
            "supportCode": "6EXAMPLE3362/
ls-3EXAMPLEb4fffb5cec056220c734713e14bd5fcd",
            "createdAt": 1554306000.814,
            "location": {
                "availabilityZone": "us-west-2a",
                "regionName": "us-west-2"
            },
            "resourceType": "RelationalDatabase",
            "tags": [],
            "relationalDatabaseBlueprintId": "postgres_11",
```

```
            "relationalDatabaseBundleId": "micro_1_0",
            "masterDatabaseName": "dbmaster",
            "hardware": {
                "cpuCount": 1,
                "diskSizeInGb": 40,
                "ramSizeInGb": 1.0
            },
            "state": "available",
            "backupRetentionEnabled": true,
            "pendingModifiedValues": {},
            "engine": "postgres",
            "engineVersion": "11.1",
            "latestRestorableTime": 1571686339.0,
            "masterUsername": "dbmasteruser",
            "parameterApplyStatus": "in-sync",
            "preferredBackupWindow": "06:19-06:49",
            "preferredMaintenanceWindow": "sun:10:19-sun:10:49",
            "publiclyAccessible": false,
            "masterEndpoint": {
                "port": 5432,
                "address":
 "ls-3EXAMPLEb4fffb5cec056220c734713eEXAMPLEd.czowadgeezqi.us-
west-2.rds.amazonaws.com"
            },
            "pendingMaintenanceActions": []
        }
    ]
}
```

- Per i dettagli sull'API, vedere GetRelationalDatabasesin AWS CLI Command Reference.

`get-static-ip`

Il seguente esempio di codice mostra come utilizzare`get-static-ip`.

AWS CLI

Come ottenere le informazioni su un IP statico

L'esempio `get-static-ip` seguente mostra i dettagli relativi all'IP statico specificato.

```
aws lightsail get-static-ip \
    --static-ip-name StaticIp-1
```

Output:

```
{
    "staticIp": {
        "name": "StaticIp-1",
        "arn": "arn:aws:lightsail:us-
west-2:111122223333:StaticIp/2257cd76-1f0e-4ac0-82e2-2EXAMPLE23ad",
        "supportCode": "6EXAMPLE3362/192.0.2.0",
        "createdAt": 1571071325.076,
        "location": {
            "availabilityZone": "all",
            "regionName": "us-west-2"
        },
        "resourceType": "StaticIp",
        "ipAddress": "192.0.2.0",
        "isAttached": false
    }
}
```

- Per i dettagli sull'API, consulta GetStaticIp AWS CLICommand Reference.

`get-static-ips`

Il seguente esempio di codice mostra come utilizzare`get-static-ips`.

AWS CLI

Per ottenere informazioni su tutto ciò che è statico IPs

L'`get-static-ips`esempio seguente mostra i dettagli su tutte le informazioni statiche IPs nella AWS regione configurata.

```
aws lightsail get-static-ips
```

Output:

```
{
    "staticIps": [
        {
            "name": "StaticIp-1",
            "arn": "arn:aws:lightsail:us-
west-2:111122223333:StaticIp/2257cd76-1f0e-4ac0-8EXAMPLE16f9423ad",
```

```
            "supportCode": "6EXAMPLE3362/192.0.2.0",
            "createdAt": 1571071325.076,
            "location": {
                "availabilityZone": "all",
                "regionName": "us-west-2"
            },
            "resourceType": "StaticIp",
            "ipAddress": "192.0.2.0",
            "isAttached": false
        },
        {
            "name": "StaticIP-2",
            "arn": "arn:aws:lightsail:us-west-2:111122223333:StaticIp/c61edb40-
e5f0-4fd6-ae7c-8EXAMPLE19f8",
            "supportCode": "6EXAMPLE3362/192.0.2.2",
            "createdAt": 1568305385.681,
            "location": {
                "availabilityZone": "all",
                "regionName": "us-west-2"
            },
            "resourceType": "StaticIp",
            "ipAddress": "192.0.2.2",
            "attachedTo": "WordPress-1",
            "isAttached": true
        }
    ]
}
```

- Per i dettagli sull'API, vedere GetStaticIpsin AWS CLI Command Reference.

`is-vpc-peered`

Il seguente esempio di codice mostra come utilizzare`is-vpc-peered`.

AWS CLI

Per determinare se il tuo cloud privato virtuale Amazon Lightsail è peer-to-peer

L'`is-vpc-peered`esempio seguente restituisce lo stato di peering del cloud privato virtuale (VPC) Amazon Lightsail per la regione specificata. AWS

```
aws lightsail is-vpc-peered \
```

```
    --region us-west-2
```

Output:

```
{
    "isPeered": true
}
```

- Per i dettagli sull'API, consulta IsVpcPeeredCommand Reference.AWS CLI

`open-instance-public-ports`

Il seguente esempio di codice mostra come utilizzare`open-instance-public-ports`.

AWS CLI

Come aprire le porte del firewall per un'istanza

L'esempio `open-instance-public-ports` seguente apre la porta TCP 22 sull'istanza specificata.

```
aws lightsail open-instance-public-ports \
    --instance-name MEAN-2 \
    --port-info fromPort=22,protocol=TCP,toPort=22
```

Output:

```
{
    "operation": {
        "id": "719744f0-a022-46f2-9f11-6EXAMPLE4642",
        "resourceName": "MEAN-2",
        "resourceType": "Instance",
        "createdAt": 1571072906.849,
        "location": {
            "availabilityZone": "us-west-2a",
            "regionName": "us-west-2"
        },
        "isTerminal": true,
        "operationDetails": "22/tcp",
        "operationType": "OpenInstancePublicPorts",
        "status": "Succeeded",
```

```
        "statusChangedAt": 1571072906.849
    }
}
```

- Per i dettagli sull'API, consulta OpenInstancePublicPorts AWS CLICommand Reference.

`peer-vpc`

Il seguente esempio di codice mostra come utilizzare`peer-vpc`.

AWS CLI

Per eseguire il peering del cloud privato virtuale Amazon Lightsail

L'`peer-vpc`esempio seguente esegue il peering del cloud privato virtuale (VPC) Amazon Lightsail per la regione specificata. AWS

```
aws lightsail peer-vpc \
    --region us-west-2
```

Output:

```
{
    "operation": {
        "id": "787e846a-54ac-497f-bce2-9EXAMPLE5d91",
        "resourceName": "vpc-0EXAMPLEa5261efb3",
        "resourceType": "PeeredVpc",
        "createdAt": 1571694233.104,
        "location": {
            "availabilityZone": "all",
            "regionName": "us-west-2"
        },
        "isTerminal": true,
        "operationDetails": "vpc-e2b3eb9b",
        "operationType": "PeeredVpc",
        "status": "Succeeded",
        "statusChangedAt": 1571694233.104
    }
}
```

- Per i dettagli sull'API, consulta PeerVpcCommand Reference.AWS CLI

`reboot-instance`

Il seguente esempio di codice mostra come utilizzare`reboot-instance`.

AWS CLI

Come riavviare un'istanza

L'esempio `reboot-instance` seguente riavvia l'istanza specificata.

```
aws lightsail reboot-instance \
    --instance-name MEAN-1
```

Output:

```
{
    "operations": [
        {
            "id": "2b679f1c-8b71-4bb4-8e97-8EXAMPLEed93",
            "resourceName": "MEAN-1",
            "resourceType": "Instance",
            "createdAt": 1571694445.49,
            "location": {
                "availabilityZone": "us-west-2a",
                "regionName": "us-west-2"
            },
            "isTerminal": true,
            "operationDetails": "",
            "operationType": "RebootInstance",
            "status": "Succeeded",
            "statusChangedAt": 1571694445.49
        }
    ]
}
```

- Per i dettagli sull'API, consulta RebootInstance AWS CLICommand Reference.

`reboot-relational-database`

Il seguente esempio di codice mostra come utilizzare`reboot-relational-database`.

AWS CLI

Come riavviare un database relazionale

L'esempio `reboot-relational-database` seguente riavvia il database relazionale specificato.

```
aws lightsail reboot-relational-database \
    --relational-database-name Database-1
```

Output:

```
{
    "operations": [
        {
            "id": "e4c980c0-3137-496c-9c91-1EXAMPLEdec2",
            "resourceName": "Database-1",
            "resourceType": "RelationalDatabase",
            "createdAt": 1571694532.91,
            "location": {
                "availabilityZone": "us-west-2a",
                "regionName": "us-west-2"
            },
            "isTerminal": false,
            "operationDetails": "",
            "operationType": "RebootRelationalDatabase",
            "status": "Started",
            "statusChangedAt": 1571694532.91
        }
    ]
}
```

- Per i dettagli sull'API, consulta RebootRelationalDatabase AWS CLICommand Reference.

`release-static-ip`

Il seguente esempio di codice mostra come utilizzare`release-static-ip`.

AWS CLI

Come eliminare un IP statico

L'esempio `release-static-ip` seguente elimina l'IP statico specificato.

```
aws lightsail release-static-ip \
    --static-ip-name StaticIp-1
```

Output:

```
{
    "operations": [
        {
            "id": "e374c002-dc6d-4c7f-919f-2EXAMPLE13ce",
            "resourceName": "StaticIp-1",
            "resourceType": "StaticIp",
            "createdAt": 1571694962.003,
            "location": {
                "availabilityZone": "all",
                "regionName": "us-west-2"
            },
            "isTerminal": true,
            "operationType": "ReleaseStaticIp",
            "status": "Succeeded",
            "statusChangedAt": 1571694962.003
        }
    ]
}
```

- Per i dettagli sull'API, consulta ReleaseStaticIp AWS CLICommand Reference.

`start-instance`

Il seguente esempio di codice mostra come utilizzare`start-instance`.

AWS CLI

Come avviare un'istanza

L'esempio `start-instance` seguente avvia l'istanza specificata.

```
aws lightsail start-instance \
    --instance-name WordPress-1
```

Output:

```
{
```

```
    "operations": [
        {
            "id": "f88d2a93-7cea-4165-afce-2d688cb18f23",
            "resourceName": "WordPress-1",
            "resourceType": "Instance",
            "createdAt": 1571695583.463,
            "location": {
                "availabilityZone": "us-west-2a",
                "regionName": "us-west-2"
            },
            "isTerminal": false,
            "operationType": "StartInstance",
            "status": "Started",
            "statusChangedAt": 1571695583.463
        }
    ]
}
```

- Per i dettagli sull'API, consulta StartInstance AWS CLICommand Reference.

`start-relational-database`

Il seguente esempio di codice mostra come utilizzare`start-relational-database`.

AWS CLI

Come avviare un database relazionale

L'esempio `start-relational-database` seguente avvia il database relazionale specificato.

```
aws lightsail start-relational-database \
    --relational-database-name Database-1
```

Output:

```
{
    "operations": [
        {
            "id": "4d5294ec-a38a-4fda-9e37-aEXAMPLE0d24",
            "resourceName": "Database-1",
            "resourceType": "RelationalDatabase",
            "createdAt": 1571695998.822,
```

```
            "location": {
                "availabilityZone": "us-west-2a",
                "regionName": "us-west-2"
            },
            "isTerminal": false,
            "operationType": "StartRelationalDatabase",
            "status": "Started",
            "statusChangedAt": 1571695998.822
        }
    ]
}
```

- Per i dettagli sull'API, consulta StartRelationalDatabase AWS CLICommand Reference.

`stop-instance`

Il seguente esempio di codice mostra come utilizzare`stop-instance`.

AWS CLI

Come arrestare un'istanza

L'esempio `stop-instance` seguente arresta l'istanza database specificata.

```
aws lightsail stop-instance \
--instance-name WordPress-1
```

Output:

```
{
    "operations": [
        {
            "id": "265357e2-2943-4d51-888a-1EXAMPLE7585",
            "resourceName": "WordPress-1",
            "resourceType": "Instance",
            "createdAt": 1571695471.134,
            "location": {
                "availabilityZone": "us-west-2a",
                "regionName": "us-west-2"
            },
            "isTerminal": false,
            "operationType": "StopInstance",
            "status": "Started",
```

```
            "statusChangedAt": 1571695471.134
        }
    ]
}
```

- Per i dettagli sull'API, consulta StopInstance AWS CLICommand Reference.

`stop-relational-database`

Il seguente esempio di codice mostra come utilizzare`stop-relational-database`.

AWS CLI

Come arrestare un database relazionale

L'esempio `stop-relational-database` seguente arresta il database relazionale specificato.

```
aws lightsail stop-relational-database \
    --relational-database-name Database-1
```

Output:

```
{
    "operations": [
        {
            "id": "cc559c19-4adb-41e4-b75b-5EXAMPLE4e61",
            "resourceName": "Database-1",
            "resourceType": "RelationalDatabase",
            "createdAt": 1571695526.29,
            "location": {
                "availabilityZone": "us-west-2a",
                "regionName": "us-west-2"
            },
            "isTerminal": false,
            "operationType": "StopRelationalDatabase",
            "status": "Started",
            "statusChangedAt": 1571695526.29
        }
    ]
}
```

- Per i dettagli sull'API, consulta StopRelationalDatabase AWS CLICommand Reference.

`unpeer-vpc`

Il seguente esempio di codice mostra come utilizzareunpeer-vpc.

AWS CLI

Per rendere anonimo il cloud privato virtuale Amazon Lightsail

L'unpeer-vpcesempio seguente annulla il peering del cloud privato virtuale (VPC) Amazon Lightsail per la regione specificata. AWS

```
aws lightsail unpeer-vpc \
    --region us-west-2
```

Output:

```
{
    "operation": {
        "id": "531aca64-7157-47ab-84c6-eEXAMPLEd898",
        "resourceName": "vpc-0EXAMPLEa5261efb3",
        "resourceType": "PeeredVpc",
        "createdAt": 1571694109.945,
        "location": {
            "availabilityZone": "all",
            "regionName": "us-west-2"
        },
        "isTerminal": true,
        "operationDetails": "vpc-e2b3eb9b",
        "operationType": "UnpeeredVpc",
        "status": "Succeeded",
        "statusChangedAt": 1571694109.945
    }
}
```

- Per i dettagli sull'API, consulta Command Reference. UnpeerVpcAWS CLI

Esempi di Macie utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with Macie.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`describe-buckets`

Il seguente esempio di codice mostra come utilizzare`describe-buckets`.

AWS CLI

Come eseguire query sui dati in uno o più bucket S3 monitorati e analizzati da Amazon Macie per il tuo account

L'`describe-buckets`esempio seguente interroga i metadati per tutti i bucket S3 i cui nomi iniziano con amzn-s3-demo-bucket e si trovano nella regione corrente. AWS

```
aws macie2 describe-buckets \
    --criteria '{"bucketName":{"prefix":"amzn-s3-demo-bucket"}}'
```

Output:

```
{
    "buckets": [
        {
            "accountId": "123456789012",
            "allowsUnencryptedObjectUploads": "FALSE",
            "automatedDiscoveryMonitoringStatus": "MONITORED",
            "bucketArn": "arn:aws:s3:::amzn-s3-demo-bucket1",
            "bucketCreatedAt": "2020-05-18T19:54:00+00:00",
            "bucketName": "amzn-s3-demo-bucket1",
            "classifiableObjectCount": 13,
            "classifiableSizeInBytes": 1592088,
            "jobDetails": {
```

```
            "isDefinedInJob": "TRUE",
            "isMonitoredByJob": "TRUE",
            "lastJobId": "08c81dc4a2f3377fae45c9ddaEXAMPLE",
            "lastJobRunTime": "2024-08-19T14:55:30.270000+00:00"
        },
        "lastAutomatedDiscoveryTime": "2024-10-22T19:11:25.364000+00:00",
        "lastUpdated": "2024-10-25T07:33:06.337000+00:00",
        "objectCount": 13,
        "objectCountByEncryptionType": {
            "customerManaged": 0,
            "kmsManaged": 2,
            "s3Managed": 7,
            "unencrypted": 4,
            "unknown": 0
        },
        "publicAccess": {
            "effectivePermission": "NOT_PUBLIC",
            "permissionConfiguration": {
                "accountLevelPermissions": {
                    "blockPublicAccess": {
                        "blockPublicAcls": true,
                        "blockPublicPolicy": true,
                        "ignorePublicAcls": true,
                        "restrictPublicBuckets": true
                    }
                },
                "bucketLevelPermissions": {
                    "accessControlList": {
                        "allowsPublicReadAccess": false,
                        "allowsPublicWriteAccess": false
                    },
                    "blockPublicAccess": {
                        "blockPublicAcls": true,
                        "blockPublicPolicy": true,
                        "ignorePublicAcls": true,
                        "restrictPublicBuckets": true
                    },
                    "bucketPolicy": {
                        "allowsPublicReadAccess": false,
                        "allowsPublicWriteAccess": false
                    }
                }
            }
        },
```

```
            "region": "us-west-2",
            "replicationDetails": {
                "replicated": false,
                "replicatedExternally": false,
                "replicationAccounts": []
            },
            "sensitivityScore": 78,
            "serverSideEncryption": {
                "kmsMasterKeyId": null,
                "type": "NONE"
            },
            "sharedAccess": "NOT_SHARED",
            "sizeInBytes": 4549746,
            "sizeInBytesCompressed": 0,
            "tags": [
                {
                    "key": "Division",
                    "value": "HR"
                },
                {
                    "key": "Team",
                    "value": "Recruiting"
                }
            ],
            "unclassifiableObjectCount": {
                "fileType": 0,
                "storageClass": 0,
                "total": 0
            },
            "unclassifiableObjectSizeInBytes": {
                "fileType": 0,
                "storageClass": 0,
                "total": 0
            },
            "versioning": true
        },
        {
            "accountId": "123456789012",
            "allowsUnencryptedObjectUploads": "TRUE",
            "automatedDiscoveryMonitoringStatus": "MONITORED",
            "bucketArn": "arn:aws:s3:::amzn-s3-demo-bucket2",
            "bucketCreatedAt": "2020-11-25T18:24:38+00:00",
            "bucketName": "amzn-s3-demo-bucket2",
            "classifiableObjectCount": 8,
```

```
            "classifiableSizeInBytes": 133810,
            "jobDetails": {
                "isDefinedInJob": "TRUE",
                "isMonitoredByJob": "FALSE",
                "lastJobId": "188d4f6044d621771ef7d65f2EXAMPLE",
                "lastJobRunTime": "2024-07-09T19:37:11.511000+00:00"
            },
            "lastAutomatedDiscoveryTime": "2024-10-24T19:11:25.364000+00:00",
            "lastUpdated": "2024-10-25T07:33:06.337000+00:00",
            "objectCount": 8,
            "objectCountByEncryptionType": {
                "customerManaged": 0,
                "kmsManaged": 0,
                "s3Managed": 8,
                "unencrypted": 0,
                "unknown": 0
            },
            "publicAccess": {
                "effectivePermission": "NOT_PUBLIC",
                "permissionConfiguration": {
                    "accountLevelPermissions": {
                        "blockPublicAccess": {
                            "blockPublicAcls": true,
                            "blockPublicPolicy": true,
                            "ignorePublicAcls": true,
                            "restrictPublicBuckets": true
                        }
                    },
                    "bucketLevelPermissions": {
                        "accessControlList": {
                            "allowsPublicReadAccess": false,
                            "allowsPublicWriteAccess": false
                        },
                        "blockPublicAccess": {
                            "blockPublicAcls": true,
                            "blockPublicPolicy": true,
                            "ignorePublicAcls": true,
                            "restrictPublicBuckets": true
                        },
                        "bucketPolicy": {
                            "allowsPublicReadAccess": false,
                            "allowsPublicWriteAccess": false
                        }
                    }
```

```
                }
            },
            "region": "us-west-2",
            "replicationDetails": {
                "replicated": false,
                "replicatedExternally": false,
                "replicationAccounts": []
            },
            "sensitivityScore": 95,
            "serverSideEncryption": {
                "kmsMasterKeyId": null,
                "type": "AES256"
            },
            "sharedAccess": "EXTERNAL",
            "sizeInBytes": 175978,
            "sizeInBytesCompressed": 0,
            "tags": [
                {
                    "key": "Division",
                    "value": "HR"
                },
                {
                    "key": "Team",
                    "value": "Recruiting"
                }
            ],
            "unclassifiableObjectCount": {
                "fileType": 3,
                "storageClass": 0,
                "total": 3
            },
            "unclassifiableObjectSizeInBytes": {
                "fileType": 2999826,
                "storageClass": 0,
                "total": 2999826
            },
            "versioning": true
        }
    ]
}
```


Per ulteriori informazioni, consulta Filtraggio dell'inventario del bucket S3 nella Guida per l'utente di Amazon Macie.

- DescribeBucketsPer AWS CLI i dettagli sull'API, consulta Command Reference.

Esempi di Amazon Managed Grafana utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando Amazon Managed Grafana. AWS Command Line Interface

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`list-workspaces`

Il seguente esempio di codice mostra come usare`list-workspaces`.

AWS CLI

Come elencare gli spazi di lavoro per l'account nella Regione specificata dalle credenziali dell'utente

L'esempio `list-workspaces` seguente elenca le aree di lavoro Grafana per la Regione dell'account.

```
aws grafana list-workspaces
```

Output:

```
{
    "workspaces": [
        {
            "authentication": {
                "providers": [
```

```
                    "AWS_SSO"
                ]
            },
            "created": "2022-04-04T16:20:21.796000-07:00",
            "description": "to test tags",
            "endpoint": "g-949e7b44df.grafana-workspace.us-east-1.amazonaws.com",
            "grafanaVersion": "8.2",
            "id": "g-949e7b44df",
            "modified": "2022-04-04T16:20:21.796000-07:00",
            "name": "testtag2",
            "notificationDestinations": [
                "SNS"
            ],
            "status": "ACTIVE"
        },
        {
            "authentication": {
                "providers": [
                    "AWS_SSO"
                ]
            },
            "created": "2022-04-20T10:22:15.115000-07:00",
            "description": "ww",
            "endpoint": "g-bffa51ed1b.grafana-workspace.us-east-1.amazonaws.com",
            "grafanaVersion": "8.2",
            "id": "g-bffa51ed1b",
            "modified": "2022-04-20T10:22:15.115000-07:00",
            "name": "ww",
            "notificationDestinations": [
                "SNS"
            ],
            "status": "ACTIVE"
        }
    ]
}
```


- Per i dettagli sull'API, consulta ListWorkspaces AWS CLICommand Reference.

MediaConnect esempi utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with MediaConnect.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`add-flow-outputs`

Il seguente esempio di codice mostra come utilizzare`add-flow-outputs`.

AWS CLI

Come aggiungere output a un flusso

L'esempio `add-flow-outputs` seguente aggiunge gli output al flusso specificato.

```
aws mediaconnect add-flow-outputs \
--flow-arn arn:aws:mediaconnect:us-
east-1:111122223333:flow:1-23aBC45dEF67hiJ8-12AbC34DE5fG:BaseballGame \
--outputs Description='NYC
 stream',Destination=192.0.2.12,Name=NYC,Port=3333,Protocol=rtp-
fec,SmoothingLatency=100 Description='LA
 stream',Destination=203.0.113.9,Name=LA,Port=4444,Protocol=rtp-
fec,SmoothingLatency=100
```

Output:

```
{
    "Outputs": [
        {
            "Port": 3333,
            "OutputArn": "arn:aws:mediaconnect:us-
east-1:111122223333:output:2-3aBC45dEF67hiJ89-c34de5fG678h:NYC",
            "Name": "NYC",
            "Description": "NYC stream",
```

```
            "Destination": "192.0.2.12",
            "Transport": {
                "Protocol": "rtp-fec",
                "SmoothingLatency": 100
            }
        },
        {
            "Port": 4444,
            "OutputArn": "arn:aws:mediaconnect:us-
east-1:111122223333:output:2-987655dEF67hiJ89-c34de5fG678h:LA",
            "Name": "LA",
            "Description": "LA stream",
            "Destination": "203.0.113.9",
            "Transport": {
                "Protocol": "rtp-fec",
                "SmoothingLatency": 100
            }
        }
    ],
    "FlowArn": "arn:aws:mediaconnect:us-
east-1:111122223333:flow:1-23aBC45dEF67hiJ8-12AbC34DE5fG:BaseballGame"
}
```

Per ulteriori informazioni, consulta Aggiungere output a un flusso nella Guida per l' MediaConnect utente di AWS Elemental.

- Per i dettagli sull'API, consulta AWS CLI Command AddFlowOutputsReference.

`create-flow`

Il seguente esempio di codice mostra come utilizzare`create-flow`.

AWS CLI

Come creare un flusso

L'esempio `create-flow` seguente crea un flusso con la configurazione specificata.

```
aws mediaconnect create-flow \
    --availability-zone us-west-2c \
    --name ExampleFlow \
    --source Description='Example source,
 backup',IngestPort=1055,Name=BackupSource,Protocol=rtp,WhitelistCidr=10.24.34.0/23
```

Output:

```
{
    "Flow": {
        "FlowArn": "arn:aws:mediaconnect:us-
east-1:123456789012:flow:1-23aBC45dEF67hiJ8-12AbC34DE5fG:ExampleFlow",
        "AvailabilityZone": "us-west-2c",
        "EgressIp": "54.245.71.21",
        "Source": {
            "IngestPort": 1055,
            "SourceArn": "arn:aws:mediaconnect:us-
east-1:123456789012:source:2-3aBC45dEF67hiJ89-c34de5fG678h:BackupSource",
            "Transport": {
                "Protocol": "rtp",
                "MaxBitrate": 80000000
            },
            "Description": "Example source, backup",
            "IngestIp": "54.245.71.21",
            "WhitelistCidr": "10.24.34.0/23",
            "Name": "mySource"
        },
        "Entitlements": [],
        "Name": "ExampleFlow",
        "Outputs": [],
        "Status": "STANDBY",
        "Description": "Example source, backup"
    }
}
```

Per ulteriori informazioni, consulta Creazione di un flusso nella Guida MediaConnect utente di AWS Elemental.

- Per i dettagli sull'API, consulta CreateFlow AWS CLICommand Reference.

`delete-flow`

Il seguente esempio di codice mostra come utilizzare`delete-flow`.

AWS CLI

Come eliminare un flusso

L'esempio `delete-flow` seguente elimina il flusso specificato.

```
aws mediaconnect delete-flow \
    --flow-arn arn:aws:mediaconnect:us-
east-1:123456789012:flow:1-23aBC45dEF67hiJ8-12AbC34DE5fG:AwardsShow
```

Output:

```
{
    "FlowArn": "arn:aws:mediaconnect:us-
east-1:123456789012:flow:1-23aBC45dEF67hiJ8-12AbC34DE5fG:AwardsShow",
    "Status": "DELETING"
}
```

Per ulteriori informazioni, consulta Eliminazione di un flusso nella Guida per l' MediaConnect utente di AWS Elemental.

- Per i dettagli sull'API, consulta AWS CLI Command DeleteFlowReference.

`describe-flow`

Il seguente esempio di codice mostra come utilizzare`describe-flow`.

AWS CLI

Come visualizzare i dettagli di un flusso

L'esempio `describe-flow` seguente visualizza i dettagli del flusso specificato, come ARN, zona di disponibilità, stato, origine, diritti e output.

```
aws mediaconnect describe-flow \
    --flow-arn arn:aws:mediaconnect:us-
east-1:123456789012:flow:1-23aBC45dEF67hiJ8-12AbC34DE5fG:AwardsShow
```

Output:

```
{
    "Flow": {
        "EgressIp": "54.201.4.39",
        "AvailabilityZone": "us-west-2c",
        "Status": "ACTIVE",
        "FlowArn": "arn:aws:mediaconnect:us-
east-1:123456789012:flow:1-23aBC45dEF67hiJ8-12AbC34DE5fG:AwardsShow",
        "Entitlements": [
```

```
            {
                "EntitlementArn": "arn:aws:mediaconnect:us-
west-2:123456789012:entitlement:1-AaBb11CcDd22EeFf-34DE5fG12AbC:MyEntitlement",
                "Description": "Assign to this account",
                "Name": "MyEntitlement",
                "Subscribers": [
                    "444455556666"
                ]
            }
        ],
        "Description": "NYC awards show",
        "Name": "AwardsShow",
        "Outputs": [
            {
                "Port": 2355,
                "Name": "NYC",
                "Transport": {
                    "SmoothingLatency": 0,
                    "Protocol": "rtp-fec"
                },
                "OutputArn": "arn:aws:mediaconnect:us-
east-1:123456789012:output:2-3aBC45dEF67hiJ89-c34de5fG678h:NYC",
                "Destination": "192.0.2.0"
            },
            {
                "Port": 3025,
                "Name": "LA",
                "Transport": {
                    "SmoothingLatency": 0,
                    "Protocol": "rtp-fec"
                },
                "OutputArn": "arn:aws:mediaconnect:us-
east-1:123456789012:output:2-987655dEF67hiJ89-c34de5fG678h:LA",
                "Destination": "192.0.2.0"
            }
        ],
        "Source": {
            "IngestIp": "54.201.4.39",
            "SourceArn": "arn:aws:mediaconnect:us-
east-1:123456789012:source:3-4aBC56dEF78hiJ90-4de5fG6Hi78Jk:ShowSource",
            "Transport": {
                "MaxBitrate": 80000000,
                "Protocol": "rtp"
            },
```

```
            "IngestPort": 1069,
            "Description": "Saturday night show",
            "Name": "ShowSource",
            "WhitelistCidr": "10.24.34.0/23"
        }
    }
}
```

Per ulteriori informazioni, consulta Visualizzazione dei dettagli di un flusso nella Guida per l' MediaConnect utente di AWS Elemental.

- Per i dettagli sull'API, consulta DescribeFlow AWS CLICommand Reference.

`grant-flow-entitlements`

Il seguente esempio di codice mostra come utilizzare`grant-flow-entitlements`.

AWS CLI

Come concedere un diritto su un flusso

L'`grant-flow-entitlements`esempio seguente concede il diritto al flusso esistente specificato di condividere i tuoi contenuti con un altro account. AWS

```
aws mediaconnect grant-flow-entitlements \
    --flow-arn arn:aws:mediaconnect:us-
east-1:111122223333:flow:1-23aBC45dEF67hiJ8-12AbC34DE5fG:BaseballGame \
    --entitlements Description='For
 AnyCompany',Encryption={"Algorithm=aes128,KeyType=static-
key,RoleArn=arn:aws:iam::111122223333:role/MediaConnect-
ASM,SecretArn=arn:aws:secretsmanager:us-
west-2:111122223333:secret:mySecret1"},Name=AnyCompany_Entitlement,Subscribers=444455556666
 Description='For Example Corp',Name=ExampleCorp,Subscribers=777788889999
```

Output:

```
{
    "Entitlements": [
        {
            "Name": "AnyCompany_Entitlement",
            "EntitlementArn": "arn:aws:mediaconnect:us-
west-2:111122223333:entitlement:1-11aa22bb11aa22bb-3333cccc4444:AnyCompany_Entitlement",
            "Subscribers": [
```

```
                "444455556666"
            ],
            "Description": "For AnyCompany",
            "Encryption": {
                "SecretArn": "arn:aws:secretsmanager:us-
west-2:111122223333:secret:mySecret1",
                "Algorithm": "aes128",
                "RoleArn": "arn:aws:iam::111122223333:role/MediaConnect-ASM",
                "KeyType": "static-key"
            }
        },
        {
            "Name": "ExampleCorp",
            "EntitlementArn": "arn:aws:mediaconnect:us-
west-2:111122223333:entitlement:1-3333cccc4444dddd-1111aaaa2222:ExampleCorp",
            "Subscribers": [
                "777788889999"
            ],
            "Description": "For Example Corp"
        }
    ],
    "FlowArn": "arn:aws:mediaconnect:us-
east-1:111122223333:flow:1-23aBC45dEF67hiJ8-12AbC34DE5fG:BaseballGame"
}
```

Per ulteriori informazioni, consulta Concessione di un diritto su un flusso nella Guida per l'utente di Elemental.AWS MediaConnect

- Per i dettagli sull'API, consulta GrantFlowEntitlementsCommand Reference.AWS CLI

`list-entitlements`

Il seguente esempio di codice mostra come utilizzare`list-entitlements`.

AWS CLI

Come visualizzare un elenco di diritti

L'esempio `list-entitlements` seguente visualizza un elenco di tutti i diritti concessi all'account.

```
aws mediaconnect list-entitlements
```

Output:

```
{
    "Entitlements": [
        {
            "EntitlementArn": "arn:aws:mediaconnect:us-
west-2:111122223333:entitlement:1-11aa22bb11aa22bb-3333cccc4444:MyEntitlement",
            "EntitlementName": "MyEntitlement"
        }
    ]
}
```

Per ulteriori informazioni, consulta ListEntitlements AWSElemental MediaConnect API Reference.

- Per i dettagli sull'API, consulta ListEntitlements AWS CLICommand Reference.

`list-flows`

Il seguente esempio di codice mostra come utilizzare`list-flows`.

AWS CLI

Come visualizzare un elenco di flussi

L'esempio `list-flows` seguente visualizza un elenco di flussi.

```
aws mediaconnect list-flows
```

Output:

```
{
    "Flows": [
        {
            "Status": "STANDBY",
            "SourceType": "OWNED",
            "AvailabilityZone": "us-west-2a",
            "Description": "NYC awards show",
            "Name": "AwardsShow",
            "FlowArn": "arn:aws:mediaconnect:us-
east-1:111122223333:flow:1-23aBC45dEF67hiJ8-12AbC34DE5fG:AwardsShow"
        },
        {
```

```
            "Status": "STANDBY",
            "SourceType": "OWNED",
            "AvailabilityZone": "us-west-2c",
            "Description": "LA basketball game",
            "Name": "BasketballGame",
            "FlowArn": "arn:aws:mediaconnect:us-
east-1:111122223333:flow:1-23aBC45dEF67hiJ8-12AbC34DE5fG:BasketballGame"
        }
    ]
}
```

Per ulteriori informazioni, consulta Visualizzazione di un elenco di flussi nella Guida per l' MediaConnect utente di AWS Elemental.

- Per i dettagli sull'API, consulta ListFlows AWS CLICommand Reference.

`list-tags-for-resource`

Il seguente esempio di codice mostra come utilizzare`list-tags-for-resource`.

AWS CLI

Per elencare i tag di una MediaConnect risorsa

L'`list-tags-for-resource`esempio seguente visualizza le chiavi e i valori dei tag associati alla MediaConnect risorsa specificata.

```
aws mediaconnect list-tags-for-resource \
    --resource-arn arn:aws:mediaconnect:us-
east-1:123456789012:flow:1-23aBC45dEF67hiJ8-12AbC34DE5fG:BasketballGame
```

Output:

```
{
    "Tags": {
        "region": "west",
        "stage": "prod"
    }
}
```

Per ulteriori informazioni, consulta ListTagsForResource, TagResource, UntagResource nell'AWS Elemental MediaConnect API Reference.

- Per i dettagli sull'API, consulta ListTagsForResource AWS CLICommand Reference.

`remove-flow-output`

Il seguente esempio di codice mostra come utilizzare`remove-flow-output`.

AWS CLI

Come rimuovere un output da un flusso

L'esempio `remove-flow-output` seguente rimuove un output dal flusso specificato.

```
aws mediaconnect remove-flow-output \
    --flow-arn arn:aws:mediaconnect:us-
east-1:111122223333:flow:1-23aBC45dEF67hiJ8-12AbC34DE5fG:BaseballGame \
    --output-arn arn:aws:mediaconnect:us-
east-1:111122223333:output:2-3aBC45dEF67hiJ89-c34de5fG678h:NYC
```

Output:

```
{
    "FlowArn": "arn:aws:mediaconnect:us-
east-1:111122223333:flow:1-23aBC45dEF67hiJ8-12AbC34DE5fG:BaseballGame",
    "OutputArn": "arn:aws:mediaconnect:us-
east-1:111122223333:output:2-3aBC45dEF67hiJ89-c34de5fG678h:NYC"
}
```

Per ulteriori informazioni, consulta Rimuovere gli output da un flusso nella Guida per l' MediaConnect utente di AWS Elemental.

- Per i dettagli sull'API, consulta AWS CLI Command RemoveFlowOutputReference.

`revoke-flow-entitlement`

Il seguente esempio di codice mostra come utilizzare`revoke-flow-entitlement`.

AWS CLI

Come revocare un diritto

L'esempio seguente `revoke-flow-entitlement` revoca un diritto sul flusso specificato.

```
aws mediaconnect revoke-flow-entitlement \
    --flow-arn arn:aws:mediaconnect:us-
east-1:111122223333:flow:1-23aBC45dEF67hiJ8-12AbC34DE5fG:BaseballGame \
    --entitlement-arn arn:aws:mediaconnect:us-
west-2:111122223333:entitlement:1-11aa22bb11aa22bb-3333cccc4444:AnyCompany_Entitlement
```

Output:

```
{
    "FlowArn": "arn:aws:mediaconnect:us-
east-1:111122223333:flow:1-23aBC45dEF67hiJ8-12AbC34DE5fG:BaseballGame",
    "EntitlementArn": "arn:aws:mediaconnect:us-
west-2:111122223333:entitlement:1-11aa22bb11aa22bb-3333cccc4444:AnyCompany_Entitlement"
}
```

Per ulteriori informazioni, consulta Revoca di un diritto nella Guida per l'utente di ElementalAWS . MediaConnect

- Per i dettagli sull'API, consulta Command Reference. RevokeFlowEntitlementAWS CLI

`start-flow`

Il seguente esempio di codice mostra come utilizzare`start-flow`.

AWS CLI

Come avviare un flusso

L'esempio `start-flow` seguente avvia il flusso specificato.

```
aws mediaconnect start-flow \
    --flow-arn arn:aws:mediaconnect:us-
east-1:123456789012:flow:1-23aBC45dEF67hiJ8-12AbC34DE5fG:AwardsShow
```

Questo comando non produce alcun output. Output:

```
{
    "FlowArn": "arn:aws:mediaconnect:us-
east-1:123456789012:flow:1-23aBC45dEF67hiJ8-12AbC34DE5fG:AwardsShow",
    "Status": "STARTING"
```

```
}
```

Per ulteriori informazioni, consulta Avvio di un flusso nella Guida per l' MediaConnect utente di AWS Elemental.

- Per i dettagli sull'API, consulta StartFlow AWS CLICommand Reference.

`stop-flow`

Il seguente esempio di codice mostra come utilizzare`stop-flow`.

AWS CLI

Come arrestare un flusso

L'esempio `stop-flow` seguente arresta il flusso specificato.

```
aws mediaconnect stop-flow \
    --flow-arn arn:aws:mediaconnect:us-
east-1:123456789012:flow:1-23aBC45dEF67hiJ8-12AbC34DE5fG:AwardsShow
```

Output:

```
{
    "Status": "STOPPING",
    "FlowArn": "arn:aws:mediaconnect:us-
east-1:123456789012:flow:1-23aBC45dEF67hiJ8-12AbC34DE5fG:AwardsShow"
}
```

Per ulteriori informazioni, consulta Stopping a Flow nella AWS Elemental MediaConnect User Guide.

- Per i dettagli sull'API, consulta AWS CLI Command StopFlowReference.

`tag-resource`

Il seguente esempio di codice mostra come utilizzare`tag-resource`.

AWS CLI

Per aggiungere tag a una MediaConnect risorsa

L'`tag-resource`esempio seguente aggiunge un tag con un nome e un valore di chiave alla MediaConnect risorsa specificata.

```
aws mediaconnect tag-resource \
    --resource-arn arn:aws:mediaconnect:us-
east-1:123456789012:flow:1-23aBC45dEF67hiJ8-12AbC34DE5fG:BasketballGame
    --tags region=west
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta ListTagsForResource, TagResource, UntagResource nell'AWS Elemental MediaConnect API Reference.

- Per i dettagli sull'API, consulta TagResource AWS CLICommand Reference.

`untag-resource`

Il seguente esempio di codice mostra come utilizzare`untag-resource`.

AWS CLI

Per rimuovere tag da una MediaConnect risorsa

L'`untag-resource`esempio seguente rimuove il tag con il nome di chiave specificato e il relativo valore associato da una MediaConnect risorsa.

```
aws mediaconnect untag-resource \
    --resource-arn arn:aws:mediaconnect:us-
east-1:123456789012:flow:1-23aBC45dEF67hiJ8-12AbC34DE5fG:BasketballGame \
    --tag-keys region
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta ListTagsForResource, TagResource, UntagResource AWS Elemental MediaConnect API Reference.

- Per i dettagli sull'API, consulta UntagResource AWS CLICommand Reference.

`update-flow-entitlement`

Il seguente esempio di codice mostra come utilizzare`update-flow-entitlement`.

AWS CLI

Come aggiornare un diritto

L'esempio `update-flow-entitlement` seguente aggiorna l'autorizzazione specificata con una nuova descrizione e un nuovo abbonato.

```
aws mediaconnect update-flow-entitlement \
    --flow-arn arn:aws:mediaconnect:us-
east-1:111122223333:flow:1-23aBC45dEF67hiJ8-12AbC34DE5fG:BaseballGame \
    --entitlement-arn arn:aws:mediaconnect:us-
west-2:111122223333:entitlement:1-11aa22bb11aa22bb-3333cccc4444:AnyCompany_Entitlement
 \
    --description 'For AnyCompany Affiliate' \
    --subscribers 777788889999
```

Output:

```
{
    "FlowArn": "arn:aws:mediaconnect:us-
east-1:111122223333:flow:1-23aBC45dEF67hiJ8-12AbC34DE5fG:BaseballGame",
    "Entitlement": {
        "Name": "AnyCompany_Entitlement",
        "Description": "For AnyCompany Affiliate",
        "EntitlementArn": "arn:aws:mediaconnect:us-
west-2:111122223333:entitlement:1-11aa22bb11aa22bb-3333cccc4444:AnyCompany_Entitlement",
        "Encryption": {
            "KeyType": "static-key",
            "Algorithm": "aes128",
            "RoleArn": "arn:aws:iam::111122223333:role/MediaConnect-ASM",
            "SecretArn": "arn:aws:secretsmanager:us-
west-2:111122223333:secret:mySecret1"
        },
        "Subscribers": [
            "777788889999"
        ]
    }
}
```

Per ulteriori informazioni, consulta Updating an Entitlement nella AWS MediaConnect Elemental User Guide.

- Per i dettagli sull'API, consulta Command UpdateFlowEntitlementReference AWS CLI .

`update-flow-output`

Il seguente esempio di codice mostra come utilizzare`update-flow-output`.

AWS CLI

Come aggiornare un output su un flusso

L'esempio `update-flow-output` seguente aggiorna un output sul flusso specificato.

```
aws mediaconnect update-flow-output \
    --flow-arn arn:aws:mediaconnect:us-
east-1:111122223333:flow:1-23aBC45dEF67hiJ8-12AbC34DE5fG:BaseballGame \
    --output-arn arn:aws:mediaconnect:us-
east-1:111122223333:output:2-3aBC45dEF67hiJ89-c34de5fG678h:NYC \
    --port 3331
```

Output:

```
{
    "FlowArn": "arn:aws:mediaconnect:us-
east-1:111122223333:flow:1-23aBC45dEF67hiJ8-12AbC34DE5fG:BaseballGame",
    "Output": {
        "Name": "NYC",
        "Port": 3331,
        "Description": "NYC stream",
        "Transport": {
            "Protocol": "rtp-fec",
            "SmoothingLatency": 100
        },
        "OutputArn": "arn:aws:mediaconnect:us-
east-1:111122223333:output:2-3aBC45dEF67hiJ89-c34de5fG678h:NYC",
        "Destination": "192.0.2.12"
    }
}
```

Per ulteriori informazioni, consulta Aggiornamento degli output su un flusso nella Guida per l' MediaConnect utente di AWS Elemental.

- Per i dettagli sull'API, consulta AWS CLI Command UpdateFlowOutputReference.

`update-flow-source`

Il seguente esempio di codice mostra come utilizzare`update-flow-source`.

AWS CLI

Come aggiornare l'origine di un flusso esistente

L'esempio `update-flow-source` seguente aggiorna l'origine di un flusso esistente.

```
aws mediaconnect update-flow-source \
    --flow-arn arn:aws:mediaconnect:us-
east-1:111122223333:flow:1-23aBC45dEF67hiJ8-12AbC34DE5fG:AwardsShow \
    --source-arn arn:aws:mediaconnect:us-
east-1:111122223333:source:3-4aBC56dEF78hiJ90-4de5fG6Hi78Jk:ShowSource \
    --description 'Friday night show' \
    --ingest-port 3344 \
    --protocol rtp-fec \
    --whitelist-cidr 10.24.34.0/23
```

Output:

```
{
    "FlowArn": "arn:aws:mediaconnect:us-
east-1:111122223333:flow:1-23aBC45dEF67hiJ8-12AbC34DE5fG:AwardsShow",
    "Source": {
        "IngestIp": "34.210.136.56",
        "WhitelistCidr": "10.24.34.0/23",
        "Transport": {
            "Protocol": "rtp-fec"
        },
        "IngestPort": 3344,
        "Name": "ShowSource",
        "Description": "Friday night show",
        "SourceArn": "arn:aws:mediaconnect:us-
east-1:111122223333:source:3-4aBC56dEF78hiJ90-4de5fG6Hi78Jk:ShowSource"
    }
}
```

Per ulteriori informazioni, consulta Aggiornamento della fonte di un flusso nella Guida per l' MediaConnect utente di AWS Elemental.

- Per i dettagli sull'API, consulta UpdateFlowSource AWS CLICommand Reference.

MediaConvert esempi utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with MediaConvert.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`cancel-job`

Il seguente esempio di codice mostra come utilizzare`cancel-job`.

AWS CLI

Come annullare un processo in coda

L'esempio `cancel-job` seguente annulla il processo con l'ID `1234567891234-abc123`. Non è possibile annullare un processo dopo che il servizio ne ha avviato l'elaborazione.

```
aws mediaconvert cancel-job \
    --endpoint-url https://abcd1234.mediaconvert.region-name-1.amazonaws.com \
    --region region-name-1 \
    --id 1234567891234-abc123
```

Per ottenere un endpoint specifico per l'account, utilizza `describe-endpoints` o invia il comando senza l'endpoint. Il servizio restituisce un errore e il tuo endpoint.

Per ulteriori informazioni, consulta Lavorare con AWS Elemental MediaConvert Jobs nella Guida per l' MediaConvert utente di AWS Elemental.

- Per i dettagli sull'API, consulta AWS CLI Command CancelJobReference.

`create-job-template`

Il seguente esempio di codice mostra come utilizzare`create-job-template`.

AWS CLI

Come creare un modello di processo

L'esempio `create-job-template` seguente crea un modello di processo con le impostazioni di transcodifica specificate nel file `job-template.json` che si trova nel sistema.

```
aws mediaconvert create-job-template \
    --endpoint-url https://abcd1234.mediaconvert.region-name-1.amazonaws.com \
    --region region-name-1 \
    --name JobTemplate1 \
    --cli-input-json file://~/job-template.json
```

Se crei il file JSON del modello di processo utilizzando `get-job-template` e quindi modificando il file, rimuovi l'oggetto `JobTemplate`, ma conserva al suo interno l'oggetto secondario Settings. Inoltre, assicurati di rimuovere le seguenti coppie chiave-valore: `LastUpdated`, `Arn`, `Type` e `CreatedAt`. Puoi specificare la categoria, la descrizione, il nome e la coda nel file JSON o nella riga di comando.

Per ottenere un endpoint specifico per l'account, utilizza `describe-endpoints` o invia il comando senza l'endpoint. Il servizio restituisce un errore e il tuo endpoint.

Se la richiesta ha esito positivo, il servizio restituisce la specifica JSON per il modello di processo che hai creato.

Per ulteriori informazioni, consulta Working with AWS Elemental MediaConvert Job Templates nella Guida per l' MediaConvert utente di AWS Elemental.

- Per i dettagli sulle API, consultate AWS CLI Command CreateJobTemplateReference.

`create-job`

Il seguente esempio di codice mostra come utilizzare`create-job`.

AWS CLI

Come creare un processo

L'esempio `create-job` seguente crea un processo di transcodifica con le impostazioni specificate in un file `job.json` che si trova nel sistema da cui viene inviato il comando. Questa specifica del processo JSON può definire le singole impostazioni, fare riferimento a un modello di processo o a preimpostazioni di output.

```
aws mediaconvert create-job \
    --endpoint-url https://abcd1234.mediaconvert.region-name-1.amazonaws.com \
    --region region-name-1 \
    --cli-input-json file://~/job.json
```

Puoi utilizzare la MediaConvert console AWS Elemental per generare le specifiche del lavoro JSON scegliendo le impostazioni del lavoro e quindi scegliendo Mostra lavoro JSON nella parte inferiore della sezione Job.

Per ottenere un endpoint specifico per l'account, utilizza `describe-endpoints` o invia il comando senza l'endpoint. Il servizio restituisce un errore e il tuo endpoint.

Se la richiesta ha esito positivo, il servizio restituisce la specifica del processo JSON inviata con la richiesta.

Per ulteriori informazioni, consulta Lavorare con AWS Elemental MediaConvert Jobs nella Guida per l'utente di ElementalAWS . MediaConvert

- Per i dettagli sull'API, consulta AWS CLI Command CreateJobReference.

`create-preset`

Il seguente esempio di codice mostra come utilizzare`create-preset`.

AWS CLI

Come creare una preimpostazione di output personalizzata

L'esempio `create-preset` seguente crea una preimpostazione di output personalizzata basata sulle impostazioni di output specificate nel file `preset.json`. Puoi specificare la categoria, la descrizione e il nome nel file JSON o nella riga di comando.

```
aws mediaconvert create-preset \
    --endpoint-url https://abcd1234.mediaconvert.region-name-1.amazonaws.com
    --region region-name-1 \
    --cli-input-json file://~/preset.json
```

Se crei il file JSON con una preimpostazione utilizzando `get-preset` e quindi modifichi il file di output, assicurati di rimuovere le seguenti coppie chiave-valore: `LastUpdated`, `Arn`, `Type` e `CreatedAt`.

Per ottenere un endpoint specifico per l'account, utilizza `describe-endpoints` o invia il comando senza l'endpoint. Il servizio restituisce un errore e il tuo endpoint.

Per ulteriori informazioni, consulta Lavorare con i preset di MediaConvert output AWS elementali nella Guida per l'utente di AWS MediaConvert Elemental.

- Per i dettagli sulle API, consultate Command CreatePresetReference AWS CLI .

`create-queue`

Il seguente esempio di codice mostra come utilizzare`create-queue`.

AWS CLI

Come creare una coda personalizzata

L'esempio `create-queue` seguente crea una coda di transcodifica personalizzata.

```
aws mediaconvert create-queue \
    --endpoint-url https://abcd1234.mediaconvert.region-name-1.amazonaws.com \
    --region region-name-1 \
    --name Queue1 \
    --description "Keep this queue empty unless job is urgent."
```

Per ottenere un endpoint specifico per l'account, utilizza `describe-endpoints` o invia il comando senza l'endpoint. Il servizio restituisce un errore e il tuo endpoint.

Output:

```
{
    "Queue": {
        "Status": "ACTIVE",
        "Name": "Queue1",
        "LastUpdated": 1518034928,
        "Arn": "arn:aws:mediaconvert:region-name-1:012345678998:queues/Queue1",
        "Type": "CUSTOM",
        "CreatedAt": 1518034928,
        "Description": "Keep this queue empty unless job is urgent."
```

```
    }
}
```

Per ulteriori informazioni, consulta Working with AWS Elemental MediaConvert Queues nella Guida per l'utente di AWS MediaConvert Elemental.

- Per i dettagli sull'API, consulta Command CreateQueueReference AWS CLI .

`delete-job-template`

Il seguente esempio di codice mostra come utilizzare`delete-job-template`.

AWS CLI

Come eliminare un modello di processo

L'esempio `delete-job-template` seguente elimina il modello di processo specificato.

```
aws mediaconvert delete-job-template \
    --name "DASH Streaming" \
    --endpoint-url https://abcd1234.mediaconvert.us-west-2.amazonaws.com
```

Questo comando non produce alcun output. Esegui `aws mediaconvert list-job-templates` per verificare che il modello sia stato eliminato.

Per ulteriori informazioni, consulta Working with AWS Elemental MediaConvert Job Templates nella Guida per l' MediaConvert utente di AWS Elemental.

- Per i dettagli sulle API, consultate AWS CLI Command DeleteJobTemplateReference.

`delete-preset`

Il seguente esempio di codice mostra come utilizzare`delete-preset`.

AWS CLI

Come eliminare una coda on demand

L'esempio `delete-preset` seguente elimina la preimpostazione personalizzata specificata.

```
aws mediaconvert delete-preset \
    --name SimpleMP4 \
```

```
    --endpoint-url  https://abcd1234.mediaconvert.us-west-2.amazonaws.com
```

Questo comando non produce alcun output. Esegui `aws mediaconvert list-presets` per verificare che la preimpostazione sia stata eliminata.

Per ulteriori informazioni, consulta Lavorare con i preset di MediaConvert output AWS elementali nella Guida per l'utente di AWS MediaConvert Elemental.

- Per i dettagli sulle API, consultate Command DeletePresetReference AWS CLI .

`delete-queue`

Il seguente esempio di codice mostra come utilizzare`delete-queue`.

AWS CLI

Come eliminare una coda on demand

L'esempio `delete-queue` seguente elimina la coda on demand personalizzata specificata.

Non è possibile eliminare la coda predefinita. Non è possibile eliminare una coda riservata che ha un piano tariffario attivo o che contiene processi non elaborati.

```
aws mediaconvert delete-queue \
    --name Customer1 \
    --endpoint-url https://abcd1234.mediaconvert.us-west-2.amazonaws.com
```

Questo comando non produce alcun output. Esegui `aws mediaconvert list-queues` per verificare che la coda sia stata eliminata.

Per ulteriori informazioni, consulta Working with AWS Elemental MediaConvert Queues nella Guida per l'utente di AWS MediaConvert Elemental.

- Per i dettagli sull'API, consulta Command DeleteQueueReference AWS CLI .

`describe-endpoints`

Il seguente esempio di codice mostra come utilizzare`describe-endpoints`.

AWS CLI

Come recuperare l'endpoint specifico dell'account

L'esempio `describe-endpoints` seguente recupera l'endpoint necessario per inviare qualsiasi altra richiesta al servizio.

```
aws mediaconvert describe-endpoints
```

Output:

```
{
    "Endpoints": [
        {
            "Url": "https://abcd1234.mediaconvert.region-name-1.amazonaws.com"
        }
    ]
}
```

Per ulteriori informazioni, consulta Guida introduttiva all' MediaConvert utilizzo dell'API nell'AWS Elemental MediaConvert API Reference.

- Per i dettagli sull'API, consulta DescribeEndpoints AWS CLICommand Reference.

`get-job-template`

Il seguente esempio di codice mostra come utilizzare`get-job-template`.

AWS CLI

Come ottenere dettagli su un modello di processo

L'esempio `get-job-template` seguente visualizza la definizione JSON del modello di processo personalizzato specificato.

```
aws mediaconvert get-job-template \
    --name "DASH Streaming" \
    --endpoint-url https://abcd1234.mediaconvert.us-east-1.amazonaws.com
```

Output:

```
{
    "JobTemplate": {
        "StatusUpdateInterval": "SECONDS_60",
        "LastUpdated": 1568652998,
```

```
        "Description": "Create a DASH streaming ABR stack",
        "CreatedAt": 1568652998,
        "Priority": 0,
        "Name": "DASH Streaming",
        "Settings": {
            ...<truncatedforbrevity>...
        },
        "Arn": "arn:aws:mediaconvert:us-west-2:123456789012:jobTemplates/DASH
 Streaming",
        "Type": "CUSTOM"
    }
}
```

Per ulteriori informazioni, consulta Working with AWS Elemental MediaConvert Job Templates nella Guida per l' MediaConvert utente di AWS Elemental.

- Per i dettagli sulle API, consultate AWS CLI Command GetJobTemplateReference.

`get-job`

Il seguente esempio di codice mostra come utilizzare`get-job`.

AWS CLI

Come ottenere dettagli su un determinato processo

L'esempio seguente richiede le informazioni per il processo con ID `1234567890987-1ab2c3`, che in questo esempio è terminato con un errore.

```
aws mediaconvert get-job \
    --endpoint-url https://abcd1234.mediaconvert.region-name-1.amazonaws.com \
    --region region-name-1 \
    --id 1234567890987-1ab2c3
```

Per ottenere un endpoint specifico per l'account, utilizza `describe-endpoints` o invia il comando senza l'endpoint. Il servizio restituisce un errore e il tuo endpoint.

Se la richiesta ha esito positivo, il servizio restituisce un file JSON contenente le informazioni sul processo, incluse le relative impostazioni, eventuali errori restituiti e altri dati sul processo, come indicato di seguito:

```
{
```

```
    "Job": {
        "Status": "ERROR",
        "Queue": "arn:aws:mediaconvert:region-name-1:012345678998:queues/Queue1",
        "Settings": {
            ...<truncated for brevity>...
        },
        "ErrorMessage": "Unable to open input file [s3://my-input-bucket/file-
name.mp4]: [Failed probe/open: [Failed to read data: AssumeRole failed]]",
        "ErrorCode": 1434,
        "Role": "arn:aws:iam::012345678998:role/MediaConvertServiceRole",
        "Arn": "arn:aws:mediaconvert:us-
west-1:012345678998:jobs/1234567890987-1ab2c3",
        "UserMetadata": {},
        "Timing": {
            "FinishTime": 1517442131,
            "SubmitTime": 1517442103,
            "StartTime": 1517442104
        },
        "Id": "1234567890987-1ab2c3",
        "CreatedAt": 1517442103
    }
}
```

Per ulteriori informazioni, consulta Lavorare con AWS Elemental MediaConvert Jobs nella Guida per l' MediaConvert utente di AWS Elemental.

- Per i dettagli sull'API, consulta AWS CLI Command GetJobReference.

`get-preset`

Il seguente esempio di codice mostra come utilizzare`get-preset`.

AWS CLI

Come ottenere dettagli su una particolare preimpostazione

L'esempio `get-preset` seguente richiede la definizione JSON della preimpostazione personalizzata specificata.

```
aws mediaconvert get-preset \
    --name SimpleMP4 \
    --endpoint-url https://abcd1234.mediaconvert.us-west-2.amazonaws.com
```

Output:


```
{
    "Preset": {
        "Description": "Creates basic MP4 file. No filtering or preproccessing.",
        "Arn": "arn:aws:mediaconvert:us-west-2:123456789012:presets/SimpleMP4",
        "LastUpdated": 1568843141,
        "Name": "SimpleMP4",
        "Settings": {
            "ContainerSettings": {
                "Mp4Settings": {
                    "FreeSpaceBox": "EXCLUDE",
                    "CslgAtom": "INCLUDE",
                    "MoovPlacement": "PROGRESSIVE_DOWNLOAD"
                },
                "Container": "MP4"
            },
            "AudioDescriptions": [
                {
                    "LanguageCodeControl": "FOLLOW_INPUT",
                    "AudioTypeControl": "FOLLOW_INPUT",
                    "CodecSettings": {
                        "AacSettings": {
                            "RawFormat": "NONE",
                            "CodecProfile": "LC",
                            "AudioDescriptionBroadcasterMix": "NORMAL",
                            "SampleRate": 48000,
                            "Bitrate": 96000,
                            "RateControlMode": "CBR",
                            "Specification": "MPEG4",
                            "CodingMode": "CODING_MODE_2_0"
                        },
                        "Codec": "AAC"
                    }
                }
            ],
            "VideoDescription": {
                "RespondToAfd": "NONE",
                "TimecodeInsertion": "DISABLED",
                "Sharpness": 50,
                "ColorMetadata": "INSERT",
                "CodecSettings": {
                    "H264Settings": {
                        "FramerateControl": "INITIALIZE_FROM_SOURCE",
```

```
                        "SpatialAdaptiveQuantization": "ENABLED",
                        "Softness": 0,
                        "Telecine": "NONE",
                        "CodecLevel": "AUTO",
                        "QualityTuningLevel": "SINGLE_PASS",
                        "UnregisteredSeiTimecode": "DISABLED",
                        "Slices": 1,
                        "Syntax": "DEFAULT",
                        "GopClosedCadence": 1,
                        "AdaptiveQuantization": "HIGH",
                        "EntropyEncoding": "CABAC",
                        "InterlaceMode": "PROGRESSIVE",
                        "ParControl": "INITIALIZE_FROM_SOURCE",
                        "NumberBFramesBetweenReferenceFrames": 2,
                        "GopSizeUnits": "FRAMES",
                        "RepeatPps": "DISABLED",
                        "CodecProfile": "MAIN",
                        "FieldEncoding": "PAFF",
                        "GopSize": 90.0,
                        "SlowPal": "DISABLED",
                        "SceneChangeDetect": "ENABLED",
                        "GopBReference": "DISABLED",
                        "RateControlMode": "CBR",
                        "FramerateConversionAlgorithm": "DUPLICATE_DROP",
                        "FlickerAdaptiveQuantization": "DISABLED",
                        "DynamicSubGop": "STATIC",
                        "MinIInterval": 0,
                        "TemporalAdaptiveQuantization": "ENABLED",
                        "Bitrate": 400000,
                        "NumberReferenceFrames": 3
                    },
                    "Codec": "H_264"
                },
                "AfdSignaling": "NONE",
                "AntiAlias": "ENABLED",
                "ScalingBehavior": "DEFAULT",
                "DropFrameTimecode": "ENABLED"
            }
        },
        "Type": "CUSTOM",
        "CreatedAt": 1568841521
    }
}
```

Per ulteriori informazioni, consulta Lavorare con i preset di MediaConvert output AWS elementali nella Guida per l'utente di AWS MediaConvert Elemental.

- Per i dettagli sulle API, consultate Command GetPresetReference AWS CLI .

`get-queue`

Il seguente esempio di codice mostra come utilizzare`get-queue`.

AWS CLI

Come ottenere dettagli su una coda

L'esempio `get-queue` seguente recupera i dettagli della coda personalizzata specificata.

```
aws mediaconvert get-queue \
    --name Customer1 \
    --endpoint-url https://abcd1234.mediaconvert.us-west-2.amazonaws.com
```

Output:

```
{
    "Queue": {
        "LastUpdated": 1526428502,
        "Type": "CUSTOM",
        "SubmittedJobsCount": 0,
        "Status": "ACTIVE",
        "PricingPlan": "ON_DEMAND",
        "CreatedAt": 1526428502,
        "ProgressingJobsCount": 0,
        "Arn": "arn:aws:mediaconvert:us-west-2:123456789012:queues/Customer1",
        "Name": "Customer1"
    }
}
```

Per ulteriori informazioni, consulta Working with AWS Elemental MediaConvert Queues nella Guida per l'utente di AWS MediaConvert Elemental.

- Per i dettagli sull'API, consulta Command GetQueueReference AWS CLI .

`list-job-templates`

Il seguente esempio di codice mostra come utilizzare`list-job-templates`.

AWS CLI

Esempio 1: come elencare i modelli di processo personalizzati

L'esempio `list-job-templates` seguente elenca tutti i modelli di processo personalizzati nella Regione corrente. Per elencare i modelli di processo del sistema, consulta l'esempio successivo.

```
aws mediaconvert list-job-templates \
    --endpoint-url https://abcd1234.mediaconvert.us-west-2.amazonaws.com
```

Output:

```
{
    "JobTemplates": [
        {
            "Description": "Create a DASH streaming ABR stack",
            "Arn": "arn:aws:mediaconvert:us-west-2:123456789012:jobTemplates/DASH
 Streaming",
            "Name": "DASH Streaming",
            "LastUpdated": 1568653007,
            "Priority": 0,
            "Settings": {
                ...<truncatedforbrevity>...
            },
            "Type": "CUSTOM",
            "StatusUpdateInterval": "SECONDS_60",
            "CreatedAt": 1568653007
        },
        {
            "Description": "Create a high-res file",
            "Arn": "arn:aws:mediaconvert:us-west-2:123456789012:jobTemplates/File",
            "Name": "File",
            "LastUpdated": 1568653007,
            "Priority": 0,
            "Settings": {
                ...<truncatedforbrevity>...
            },
            "Type": "CUSTOM",
            "StatusUpdateInterval": "SECONDS_60",
```

```
            "CreatedAt": 1568653023
        }
    ]
}
```

Esempio 2: Per elencare i modelli MediaConvert di lavoro del sistema

L'esempio `list-job-templates` seguente elenca tutti i modelli di processo di sistema.

```
aws mediaconvert list-job-templates \
    --endpoint-url https://abcd1234.mediaconvert.us-east-1.amazonaws.com \
    --list-by SYSTEM
```

Output:

```
{
    "JobTemplates": [
        {
            "CreatedAt": 1568321779,
            "Arn": "arn:aws:mediaconvert:us-east-1:123456789012:jobTemplates/System-
Generic_Mp4_Hev1_Avc_Aac_Sdr_Qvbr",
            "Name": "System-Generic_Mp4_Hev1_Avc_Aac_Sdr_Qvbr",
            "Description": "GENERIC, MP4, AVC + HEV1(HEVC,SDR), AAC, SDR, QVBR",
            "Category": "GENERIC",
            "Settings": {
                "AdAvailOffset": 0,
                "OutputGroups": [
                    {
                        "Outputs": [
                            {
                                "Extension": "mp4",
                                "Preset": "System-
Generic_Hd_Mp4_Avc_Aac_16x9_Sdr_1280x720p_30Hz_5Mbps_Qvbr_Vq9",
                                "NameModifier":
 "_Generic_Hd_Mp4_Avc_Aac_16x9_Sdr_1280x720p_30Hz_5000Kbps_Qvbr_Vq9"
                            },
                            {
                                "Extension": "mp4",
                                "Preset": "System-
Generic_Hd_Mp4_Avc_Aac_16x9_Sdr_1920x1080p_30Hz_10Mbps_Qvbr_Vq9",
                                "NameModifier":
 "_Generic_Hd_Mp4_Avc_Aac_16x9_Sdr_1920x1080p_30Hz_10000Kbps_Qvbr_Vq9"
                            },
```

```
                            {
                                "Extension": "mp4",
                                "Preset": "System-
Generic_Sd_Mp4_Avc_Aac_16x9_Sdr_640x360p_30Hz_0.8Mbps_Qvbr_Vq7",
                                "NameModifier":
 "_Generic_Sd_Mp4_Avc_Aac_16x9_Sdr_640x360p_30Hz_800Kbps_Qvbr_Vq7"
                            },
                            {
                                "Extension": "mp4",
                                "Preset": "System-
Generic_Hd_Mp4_Hev1_Aac_16x9_Sdr_1280x720p_30Hz_4Mbps_Qvbr_Vq9",
                                "NameModifier":
 "_Generic_Hd_Mp4_Hev1_Aac_16x9_Sdr_1280x720p_30Hz_4000Kbps_Qvbr_Vq9"
                            },
                            {
                                "Extension": "mp4",
                                "Preset": "System-
Generic_Hd_Mp4_Hev1_Aac_16x9_Sdr_1920x1080p_30Hz_8Mbps_Qvbr_Vq9",
                                "NameModifier":
 "_Generic_Hd_Mp4_Hev1_Aac_16x9_Sdr_1920x1080p_30Hz_8000Kbps_Qvbr_Vq9"
                            },
                            {
                                "Extension": "mp4",
                                "Preset": "System-
Generic_Uhd_Mp4_Hev1_Aac_16x9_Sdr_3840x2160p_30Hz_12Mbps_Qvbr_Vq9",
                                "NameModifier":
 "_Generic_Uhd_Mp4_Hev1_Aac_16x9_Sdr_3840x2160p_30Hz_12000Kbps_Qvbr_Vq9"
                            }
                        ],
                        "OutputGroupSettings": {
                            "FileGroupSettings": {

                            },
                            "Type": "FILE_GROUP_SETTINGS"
                        },
                        "Name": "File Group"
                    }
                ]
            },
            "Type": "SYSTEM",
            "LastUpdated": 1568321779
        },
        ...<truncatedforbrevity>...
    ]
```

```
}
```

Per ulteriori informazioni, consulta Working with AWS Elemental MediaConvert Job Templates nella Guida per l' MediaConvert utente di AWS Elemental.

- Per i dettagli sulle API, consultate AWS CLI Command ListJobTemplatesReference.

`list-jobs`

Il seguente esempio di codice mostra come utilizzare`list-jobs`.

AWS CLI

Come ottenere dettagli su tutti i processi in una Regione

L'esempio seguente richiede le informazioni per tutti i processi nella Regione specificata.

```
aws mediaconvert list-jobs \
    --endpoint-url https://abcd1234.mediaconvert.region-name-1.amazonaws.com \
    --region region-name-1
```

Per ottenere un endpoint specifico per l'account, utilizza `describe-endpoints` o invia il comando senza l'endpoint. Il servizio restituisce un errore e il tuo endpoint.

Per ulteriori informazioni, consulta Lavorare con AWS Elemental MediaConvert Jobs nella Guida per l' MediaConvert utente di AWS Elemental.

- Per i dettagli sull'API, consulta AWS CLI Command ListJobsReference.

`list-presets`

Il seguente esempio di codice mostra come utilizzare`list-presets`.

AWS CLI

Esempio 1: come elencare le preimpostazioni di output personalizzate

L'esempio `list-presets` seguente elenca le preimpostazioni di output personalizzate. Per elencare le preimpostazioni del sistema, consulta l'esempio successivo.

```
aws mediaconvert list-presets \
    --endpoint-url https://abcd1234.mediaconvert.us-west-2.amazonaws.com
```

Output:

```
{
    "Presets": [
        {
            "Name": "SimpleMP4",
            "CreatedAt": 1568841521,
            "Settings": {
                ......
            },
            "Arn": "arn:aws:mediaconvert:us-east-1:003235472598:presets/SimpleMP4",
            "Type": "CUSTOM",
            "LastUpdated": 1568843141,
            "Description": "Creates basic MP4 file. No filtering or preproccessing."
        },
        {
            "Name": "SimpleTS",
            "CreatedAt": 1568843113,
            "Settings": {
                ... truncated for brevity ...
            },
            "Arn": "arn:aws:mediaconvert:us-east-1:003235472598:presets/SimpleTS",
            "Type": "CUSTOM",
            "LastUpdated": 1568843113,
            "Description": "Create a basic transport stream."
        }
    ]
}
```

Esempio 2: come elencare le preimpostazioni di output del sistema

L'`list-presets`esempio seguente elenca le preimpostazioni MediaConvert di sistema disponibili. Per elencare le preimpostazioni personalizzate, consulta l'esempio precedente.

```
aws mediaconvert list-presets \
    --list-by SYSTEM \
    --endpoint-url https://abcd1234.mediaconvert.us-west-2.amazonaws.com
```

Output:

```
{
    "Presets": [
        {
```

```
            "Arn": "arn:aws:mediaconvert:us-west-2:123456789012:presets/System-
Avc_16x9_1080p_29_97fps_8500kbps",
            "Name": "System-Avc_16x9_1080p_29_97fps_8500kbps",
            "CreatedAt": 1568321789,
            "Description": "Wifi, 1920x1080, 16:9, 29.97fps, 8500kbps",
            "LastUpdated": 1568321789,
            "Type": "SYSTEM",
            "Category": "HLS",
            "Settings": {
            ...<output settings removed for brevity>...
            }
        },

        ...<list of presets shortened for brevity>...

        {
            "Arn": "arn:aws:mediaconvert:us-east-1:123456789012:presets/System-
Xdcam_HD_1080i_29_97fps_35mpbs",
            "Name": "System-Xdcam_HD_1080i_29_97fps_35mpbs",
            "CreatedAt": 1568321790,
            "Description": "XDCAM MPEG HD, 1920x1080i, 29.97fps, 35mbps",
            "LastUpdated": 1568321790,
            "Type": "SYSTEM",
            "Category": "MXF",
            "Settings": {
            ...<output settings removed for brevity>...
            }
        }
    ]
}
```

Per ulteriori informazioni, vedete Lavorare con i preset di MediaConvert output AWS elementali nella Guida per l'utente di AWS MediaConvert Elemental.

- Per i dettagli sulle API, consultate Command ListPresetsReference AWS CLI .

`list-queues`

Il seguente esempio di codice mostra come utilizzare`list-queues`.

AWS CLI

Come elencare le code

L'`list-queues`esempio seguente elenca tutte le MediaConvert code.

```
aws mediaconvert list-queues \
    --endpoint-url https://abcd1234.mediaconvert.us-west-2.amazonaws.com
```

Output:

```
{
    "Queues": [
        {
            "PricingPlan": "ON_DEMAND",
            "Type": "SYSTEM",
            "Status": "ACTIVE",
            "CreatedAt": 1503451595,
            "Name": "Default",
            "SubmittedJobsCount": 0,
            "ProgressingJobsCount": 0,
            "Arn": "arn:aws:mediaconvert:us-west-2:123456789012:queues/Default",
            "LastUpdated": 1534549158
        },
        {
            "PricingPlan": "ON_DEMAND",
            "Type": "CUSTOM",
            "Status": "ACTIVE",
            "CreatedAt": 1537460025,
            "Name": "Customer1",
            "SubmittedJobsCount": 0,
            "Description": "Jobs we run for our cusotmer.",
            "ProgressingJobsCount": 0,
            "Arn": "arn:aws:mediaconvert:us-west-2:123456789012:queues/Customer1",
            "LastUpdated": 1537460025
        },
        {
            "ProgressingJobsCount": 0,
            "Status": "ACTIVE",
            "Name": "transcode-library",
            "SubmittedJobsCount": 0,
            "LastUpdated": 1564066204,
            "ReservationPlan": {
                "Status": "ACTIVE",
                "ReservedSlots": 1,
                "PurchasedAt": 1564066203,
                "Commitment": "ONE_YEAR",
```

```
                "ExpiresAt": 1595688603,
                "RenewalType": "EXPIRE"
            },
            "PricingPlan": "RESERVED",
            "Arn": "arn:aws:mediaconvert:us-west-2:123456789012:queues/transcode-
library",
            "Type": "CUSTOM",
            "CreatedAt": 1564066204
        }
    ]
}
```

Per ulteriori informazioni, consulta Working with AWS Elemental MediaConvert Queues nella Guida per l'utente di AWS MediaConvert Elemental.

- Per i dettagli sull'API, consulta Command ListQueuesReference AWS CLI .

`list-tags-for-resource`

Il seguente esempio di codice mostra come utilizzare`list-tags-for-resource`.

AWS CLI

Per elencare i tag su una MediaConvert coda, un modello di lavoro o una preimpostazione di output

L'esempio `list-tags-for-resource` seguente elenca i tag per la preimpostazione specificata.

```
aws mediaconvert list-tags-for-resource \
    --arn arn:aws:mediaconvert:us-west-2:123456789012:presets/SimpleMP4 \
    --endpoint-url https://abcd1234.mediaconvert.us-west-2.amazonaws.com
```

Output:

```
{
    "ResourceTags": {
        "Tags": {
            "customer": "zippyVideo"
        },
        "Arn": "arn:aws:mediaconvert:us-west-2:123456789012:presets/SimpleMP4"
    }
}
```

Per ulteriori informazioni, consulta Tagging AWS Elemental MediaConvert Queues, Job Templates e Output Preset nella Guida utente di Elemental AWS . MediaConvert

- Per i dettagli sulle API, consultate Command Reference. ListTagsForResourceAWS CLI

`update-job-template`

Il seguente esempio di codice mostra come utilizzare`update-job-template`.

AWS CLI

Come creare un modello di processo

L'esempio `update-job-template` seguente sostituisce la definizione JSON del modello di processo personalizzato specificato con la definizione JSON presente nel file fornito.

aws mediaconvert update-job-template --name File1 https://abcd1234.mediaconvert.us-west-2.amazonaws.com --endpoint-url -- file: //~/ .json cli-input-json job-template-update

Contenuto di `job-template-update.json`:

```
{
    "Description": "A simple job template that generates a single file output.",
    "Queue": "arn:aws:mediaconvert:us-east-1:012345678998:queues/Default",
    "Name": "SimpleFile",
    "Settings": {
      "OutputGroups": [
        {
          "Name": "File Group",
          "Outputs": [
            {
              "ContainerSettings": {
                "Container": "MP4",
                "Mp4Settings": {
                  "CslgAtom": "INCLUDE",
                  "FreeSpaceBox": "EXCLUDE",
                  "MoovPlacement": "PROGRESSIVE_DOWNLOAD"
                }
              },
              "VideoDescription": {
                "ScalingBehavior": "DEFAULT",
                "TimecodeInsertion": "DISABLED",
                "AntiAlias": "ENABLED",
```

```
                "Sharpness": 50,
                "CodecSettings": {
                  "Codec": "H_264",
                  "H264Settings": {
                    "InterlaceMode": "PROGRESSIVE",
                    "NumberReferenceFrames": 3,
                    "Syntax": "DEFAULT",
                    "Softness": 0,
                    "GopClosedCadence": 1,
                    "GopSize": 90,
                    "Slices": 1,
                    "GopBReference": "DISABLED",
                    "SlowPal": "DISABLED",
                    "SpatialAdaptiveQuantization": "ENABLED",
                    "TemporalAdaptiveQuantization": "ENABLED",
                    "FlickerAdaptiveQuantization": "DISABLED",
                    "EntropyEncoding": "CABAC",
                    "Bitrate": 400000,
                    "FramerateControl": "INITIALIZE_FROM_SOURCE",
                    "RateControlMode": "CBR",
                    "CodecProfile": "MAIN",
                    "Telecine": "NONE",
                    "MinIInterval": 0,
                    "AdaptiveQuantization": "HIGH",
                    "CodecLevel": "AUTO",
                    "FieldEncoding": "PAFF",
                    "SceneChangeDetect": "ENABLED",
                    "QualityTuningLevel": "SINGLE_PASS",
                    "FramerateConversionAlgorithm": "DUPLICATE_DROP",
                    "UnregisteredSeiTimecode": "DISABLED",
                    "GopSizeUnits": "FRAMES",
                    "ParControl": "INITIALIZE_FROM_SOURCE",
                    "NumberBFramesBetweenReferenceFrames": 2,
                    "RepeatPps": "DISABLED",
                    "DynamicSubGop": "STATIC"
                  }
                },
                "AfdSignaling": "NONE",
                "DropFrameTimecode": "ENABLED",
                "RespondToAfd": "NONE",
                "ColorMetadata": "INSERT"
              },
              "AudioDescriptions": [
                {
```

```
                  "AudioTypeControl": "FOLLOW_INPUT",
                  "CodecSettings": {
                    "Codec": "AAC",
                    "AacSettings": {
                      "AudioDescriptionBroadcasterMix": "NORMAL",
                      "Bitrate": 96000,
                      "RateControlMode": "CBR",
                      "CodecProfile": "LC",
                      "CodingMode": "CODING_MODE_2_0",
                      "RawFormat": "NONE",
                      "SampleRate": 48000,
                      "Specification": "MPEG4"
                    }
                  },
                  "LanguageCodeControl": "FOLLOW_INPUT"
                }
              ]
            }
          ],
          "OutputGroupSettings": {
            "Type": "FILE_GROUP_SETTINGS",
            "FileGroupSettings": {}
          }
        }
      ],
      "AdAvailOffset": 0
    },
    "StatusUpdateInterval": "SECONDS_60",
    "Priority": 0
}
```

Il sistema restituisce il payload JSON inviato con la richiesta, anche quando la richiesta genera un errore. Pertanto, il codice JSON restituito non è necessariamente la nuova definizione del modello di processo.

Poiché il payload JSON può essere lungo, potrebbe essere necessario scorrere verso l'alto per visualizzare eventuali messaggi di errore.

Per ulteriori informazioni, consulta Working with AWS Elemental MediaConvert Job Templates nella Guida per l' MediaConvert utente di AWS Elemental.

- Per i dettagli sulle API, consultate AWS CLI Command UpdateJobTemplateReference.

`update-preset`

Il seguente esempio di codice mostra come utilizzare`update-preset`.

AWS CLI

Come modificare una preimpostazione

L'esempio `update-preset` seguente sostituisce la descrizione per la preimpostazione specificata.

```
aws mediaconvert update-preset \
--name Customer1 \
--description "New description text."
--endpoint-url https://abcd1234.mediaconvert.us-west-2.amazonaws.com
```

Questo comando non produce alcun output. Output:

```
{
    "Preset": {
        "Arn": "arn:aws:mediaconvert:us-east-1:003235472598:presets/SimpleMP4",
        "Settings": {
        ...<output settings removed for brevity>...
        },
        "Type": "CUSTOM",
        "LastUpdated": 1568938411,
        "Description": "New description text.",
        "Name": "SimpleMP4",
        "CreatedAt": 1568938240
    }
}
```

Per ulteriori informazioni, consulta Lavorare con i preset di MediaConvert output AWS elementali nella Guida per l'utente di AWS MediaConvert Elemental.

- Per i dettagli sulle API, consultate Command UpdatePresetReference AWS CLI .

`update-queue`

Il seguente esempio di codice mostra come utilizzare`update-queue`.

AWS CLI

Come modificare una coda

L'esempio `update-queue` seguente mette in pausa la coda specificata, modificandone lo stato in `PAUSED`.

```
aws mediaconvert update-queue \
--name Customer1 \
--status PAUSED
--endpoint-url https://abcd1234.mediaconvert.us-west-2.amazonaws.com
```

Output:

```
{
    "Queue": {
        "LastUpdated": 1568839845,
        "Status": "PAUSED",
        "ProgressingJobsCount": 0,
        "CreatedAt": 1526428516,
        "Arn": "arn:aws:mediaconvert:us-west-1:123456789012:queues/Customer1",
        "Name": "Customer1",
        "SubmittedJobsCount": 0,
        "PricingPlan": "ON_DEMAND",
        "Type": "CUSTOM"
    }
}
```

Per ulteriori informazioni, consulta Working with AWS Elemental MediaConvert Queues nella Guida per l'utente di AWS MediaConvert Elemental.

- Per i dettagli sull'API, consulta Command UpdateQueueReference AWS CLI .

MediaLive esempi utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with MediaLive.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`create-channel`

Il seguente esempio di codice mostra come utilizzare`create-channel`.

AWS CLI

Come creare un canale

L'esempio `create-channel` seguente crea un canale passando un file JSON contenente i parametri da specificare.

Il canale in questo esempio inserisce un ingresso HLS PULL che si collega a una sorgente contenente video, audio e didascalie incorporate. Il canale crea un gruppo di output HLS con un server Akamai come destinazione. Il gruppo di output contiene due uscite: una per il video H.265 e l'audio AAC e una per i sottotitoli Web-VTT, solo in inglese.

Il codice JSON per questo canale di esempio include i parametri minimi richiesti per un canale che utilizza un input HLS PULL e che produce un gruppo di output HLS con Akamai come destinazione. Il JSON contiene le seguenti sezioni principali:

`InputAttachments`, che specifica una fonte per l'audio e una fonte per i sottotitoli. Non specifica un selettore video, il che significa che MediaLive estrae il primo video che trova nella sorgente. `Destinations`, che contiene i due indirizzi IP (URLs) per il singolo gruppo di output in questo canale. Questi indirizzi richiedono una password. `EncoderSettings`, che contiene sottosezioni. `AudioDescriptions`, che specifica che il canale contiene una risorsa di uscita audio, che utilizza la sorgente di InputAttachments e produce audio in formato AAC. `CaptionDescriptions`, che specifica che il canale contiene una risorsa di output per i sottotitoli, che utilizza la fonte da e produce sottotitoli in formato InputAttachments Web-VTT. `VideoDescriptions`, che specifica che il canale contiene una risorsa di uscita video con la risoluzione specificata. `OutputGroups`, che specifica i gruppi di output. In questo esempio è presente un solo gruppo denominato `Akamai`. La connessione viene effettuata utilizzando HLS PUT. Il gruppo di output contiene due output. Un'uscita è per un asset video (denominato

Video_high) e l'asset audio (denominatoAudio_EN). Un output è per l'asset didascalie (denominato WebVTT_EN).

In questo esempio, alcuni parametri non contengono alcun valore o contengono parametri vuoti annidati. Ad esempio, OutputSettings l'Video_and_audiooutput contiene diversi parametri annidati che terminano con un parametro vuoto M3U8Settings. Questo parametro deve essere incluso, ma è possibile omettere uno, più o tutti i relativi elementi secondari, e ciò significa che il parametro secondario assumerà il valore predefinito o sarà null.

Tutti i parametri che si applicano a questo canale di esempio ma che non sono specificati in questo file assumeranno il valore predefinito, saranno impostati su null o assumeranno un valore univoco generato da. MediaLive

```
aws medialive create-channel \
    --cli-input-json file://channel-in-hls-out-hls-akamai.json
```

Contenuto di channel-in-hls-out-hls-akamai.json:

```
{
    "Name": "News_West",
    "RoleArn": "arn:aws:iam::111122223333:role/MediaLiveAccessRole",
    "InputAttachments": [
        {
            "InputAttachmentName": "local_news",
            "InputId": "1234567",
            "InputSettings": {
                "AudioSelectors": [
                    {
                        "Name": "English-Audio",
                        "SelectorSettings": {
                            "AudioLanguageSelection": {
                                "LanguageCode": "EN"
                            }
                        }
                    }
                ],
                "CaptionSelectors": [
                    {
                        "LanguageCode": "ENE",
                        "Name": "English_embedded"
                    }
                ]
```

```
            }
        }
    ],
    "Destinations": [
        {
            "Id": "akamai-server-west",
            "Settings": [
                {
                    "PasswordParam": "/medialive/examplecorp1",
                    "Url": "http://203.0.113.55/news/news_west",
                    "Username": "examplecorp"
                },
                {
                    "PasswordParam": "/medialive/examplecorp2",
                    "Url": "http://203.0.113.82/news/news_west",
                    "Username": "examplecorp"
                }
            ]
        }
    ],
    "EncoderSettings": {
        "AudioDescriptions": [
            {
                "AudioSelectorName": "English-Audio",
                "CodecSettings": {
                    "AacSettings": {}
                },
                "Name": "Audio_EN"
            }
        ],
        "CaptionDescriptions": [
            {
                "CaptionSelectorName": "English_embedded",
                "DestinationSettings": {
                    "WebvttDestinationSettings": {}
                },
                "Name": "WebVTT_EN"
            }
        ],
        "VideoDescriptions": [
            {
                "Height": 720,
                "Name": "Video_high",
                "Width": 1280
```

```
            }
        ],
        "OutputGroups": [
            {
                "Name": "Akamai",
                "OutputGroupSettings": {
                    "HlsGroupSettings": {
                        "Destination": {
                            "DestinationRefId": "akamai-server-west"
                        },
                        "HlsCdnSettings": {
                            "HlsBasicPutSettings": {}
                        }
                    }
                },
                "Outputs": [
                    {
                        "AudioDescriptionNames": [
                            "Audio_EN"
                        ],
                        "OutputName": "Video_and_audio",
                        "OutputSettings": {
                            "HlsOutputSettings": {
                                "HlsSettings": {
                                    "StandardHlsSettings": {
                                        "M3u8Settings": {}
                                    }
                                },
                                "NameModifier": "_1"
                            }
                        },
                        "VideoDescriptionName": "Video_high"
                    },
                    {
                        "CaptionDescriptionNames": [
                            "WebVTT_EN"
                        ],
                        "OutputName": "Captions-WebVTT",
                        "OutputSettings": {
                            "HlsOutputSettings": {
                                "HlsSettings": {
                                    "StandardHlsSettings": {
                                        "M3u8Settings": {}
                                    }
```

```
                                },
                                "NameModifier": "_2"
                            }
                        }
                    }
                ]
            }
        ],
        "TimecodeConfig": {
            "Source": "EMBEDDED"
        }
    }
}
```

Output:

L'output ripete il contenuto del file JSON, più i seguenti valori. Tutti i parametri sono elencati in ordine alfabetico.

ARN per il canale. L'ultima parte dell'ARN è l'ID. del canale univoco.`EgressEndpoints` è vuoto in questo canale di esempio perché utilizzato solo per gli ingressi PUSH. Quando viene applicato, mostra gli indirizzi su MediaLive cui viene inviato il contenuto. `OutputGroups`,`Outputs`. Questi mostrano tutti i parametri per il gruppo di output e gli output, compresi quelli non inclusi ma che sono rilevanti per questo canale. I parametri potrebbero essere vuoti (ad esempio indicando che il parametro o la funzionalità sono disabilitati in questa configurazione del canale) o potrebbero mostrare il valore predefinito che verrà applicato.`LogLevel` è impostato sul valore predefinito (DISABLED).`Tags` è impostato sul valore predefinito (null).`PipelinesRunningCount` e `State` mostrano lo stato attuale del canale.

Per ulteriori informazioni, consulta Creazione di un canale da zero nella Guida per l' MediaLive utente di AWS Elemental.

- Per i dettagli sull'API, consulta CreateChannel AWS CLICommand Reference.

`create-input`

Il seguente esempio di codice mostra come utilizzare`create-input`.

AWS CLI

Come creare un input

L'esempio `create-input` seguente crea un input `HLS PULL` passando un file JSON contenente i parametri applicabili a questo tipo di input. Il codice JSON per questo esempio di input specifica due fonti (indirizzi) per l'input, in modo da supportare la ridondanza nell'inserimento. Questi indirizzi richiedono una password.

```
aws medialive create-input \
    --cli-input-json file://input-hls-pull-news.json
```

Contenuto di `input-hls-pull-news.json`:

```
{
    "Name": "local_news",
    "RequestId": "cli000059",
    "Sources": [
            {
                    "Url": "https://203.0.113.13/newschannel/anytownusa.m3u8",
                    "Username": "examplecorp",
                    "PasswordParam": "/medialive/examplecorp1"
            },
       {
                    "Url": "https://198.51.100.54/fillervideos/oceanwaves.mp4",
                    "Username": "examplecorp",
                    "PasswordParam": "examplecorp2"
            }
    ],
    "Type": "URL_PULL"
}
```

Output:

L'output ripete il contenuto del file JSON, più i seguenti valori. Tutti i parametri sono elencati in ordine alfabetico.

`Arn` per l'input. L'ultima parte dell'ARN è l'ID di input univoco. `Attached Channels`, che è sempre vuoto per un input appena creato. `Destinations`, che in questo esempio è vuoto perché viene utilizzato solo con un input PUSH. `Id`per l'input, lo stesso dell'ID nell'ARN. `MediaConnectFlows`, che in questo esempio è vuoto perché viene utilizzato solo con un input di tipo MediaConnect. `SecurityGroups`, che in questo esempio è vuoto perché viene utilizzato solo con un input PUSH. `State`di questo input. `Tags`, che è vuoto (impostazione predefinita per questo parametro).

Per ulteriori informazioni, consulta Creazione di un input nella Guida MediaLive utente di AWS Elemental.

- Per i dettagli sull'API, consulta CreateInput AWS CLICommand Reference.

MediaPackage esempi utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with MediaPackage.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`create-channel`

Il seguente esempio di codice mostra come utilizzare`create-channel`.

AWS CLI

Come creare un canale

Il comando `create-channel` seguente crea un canale denominato `sportschannel` nell'account corrente.

```
aws mediapackage create-channel --id sportschannel
```

Output:

```
{
   "Arn": "arn:aws:mediapackage:us-
west-2:111222333:channels/6d345804ec3f46c9b454a91d4a80d0e0",
   "HlsIngest": {
```

```
        "IngestEndpoints": [
            {
                "Id": "6d345804ec3f46c9b454a91d4a80d0e0",
                "Password": "generatedwebdavpassword1",
                "Url": "https://f31c86aed53b815a.mediapackage.us-
west-2.amazonaws.com/in/
v2/6d345804ec3f46c9b454a91d4a80d0e0/6d345804ec3f46c9b454a91d4a80d0e0/channel",
                "Username": "generatedwebdavusername1"
            },
            {
                "Id": "2daa32878af24803b24183727211b8ff",
                "Password": "generatedwebdavpassword2",
                "Url": "https://6ebbe7e04c4b0afa.mediapackage.us-
west-2.amazonaws.com/in/
v2/6d345804ec3f46c9b454a91d4a80d0e0/2daa32878af24803b24183727211b8ff/channel",
                "Username": "generatedwebdavusername2"
            }
        ]
    },
    "Id": "sportschannel",
    "Tags": {
        "region": "west"
    }
}
```

Per ulteriori informazioni, consulta Creazione di un canale nella Guida per l' MediaPackage utente di AWS Elemental.

- Per i dettagli sull'API, consulta CreateChannel AWS CLICommand Reference.

`create-origin-endpoint`

Il seguente esempio di codice mostra come utilizzare`create-origin-endpoint`.

AWS CLI

Come creare un endpoint di origine

Il comando `create-origin-endpoint` seguente crea un endpoint di origine denominato `cmafsports` con le impostazioni del pacchetto fornite in un file JSON e le impostazioni dell'endpoint specificate.

```
aws mediapackage create-origin-endpoint \
```

```
    --channel-id sportschannel \
    --id cmafsports \
    --cmaf-package file://file/path/cmafpkg.json --description "cmaf output of
 sports" \
    --id cmaf_sports \
    --manifest-name sports_channel \
    --startover-window-seconds 300 \
    --tags region=west,media=sports \
    --time-delay-seconds 10
```

Output:

```
{
    "Arn": "arn:aws:mediapackage:us-
west-2:111222333:origin_endpoints/1dc6718be36f4f34bb9cd86bc50925e6",
    "ChannelId": "sportschannel",
    "CmafPackage": {
        "HlsManifests": [
            {
                "AdMarkers": "PASSTHROUGH",
                "Id": "cmaf_sports_endpoint",
                "IncludeIframeOnlyStream": true,
                "ManifestName": "index",
                "PlaylistType": "EVENT",
                "PlaylistWindowSeconds": 300,
                "ProgramDateTimeIntervalSeconds": 300,
                "Url": "https://c4af3793bf76b33c.mediapackage.us-
west-2.amazonaws.com/out/v1/1dc6718be36f4f34bb9cd86bc50925e6/cmaf_sports_endpoint/
index.m3u8"
            }
        ],
        "SegmentDurationSeconds": 2,
        "SegmentPrefix": "sportschannel"
    },
    "Description": "cmaf output of sports",
    "Id": "cmaf_sports",
    "ManifestName": "sports_channel",
    "StartoverWindowSeconds": 300,
    "Tags": {
        "region": "west",
        "media": "sports"
    },
    "TimeDelaySeconds": 10,
```

```
    "Url": "",
    "Whitelist": []
}
```

Per ulteriori informazioni, consulta Creazione di un endpoint nella Guida per l' MediaPackage utente di AWS Elemental.

- Per i dettagli sull'API, consulta AWS CLI Command CreateOriginEndpointReference.

`delete-channel`

Il seguente esempio di codice mostra come utilizzare`delete-channel`.

AWS CLI

Come eliminare un canale

Il comando `delete-channel` seguente elimina il canale denominato `test`.

```
aws mediapackage delete-channel \
    --id test
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Eliminazione di un canale nella Guida per l' MediaPackage utente di AWS Elemental.

- Per i dettagli sull'API, consulta AWS CLI Command DeleteChannelReference.

`delete-origin-endpoint`

Il seguente esempio di codice mostra come utilizzare`delete-origin-endpoint`.

AWS CLI

Come eliminare un endpoint di origine

Il comando `delete-origin-endpoint` seguente elimina l'endpoint di origine denominato `tester2`.

```
aws mediapackage delete-origin-endpoint \
```

```
    --id tester2
```

Per ulteriori informazioni, consulta Eliminazione di un endpoint nella Guida per l'utente di AWS MediaPackage Elemental.

- Per i dettagli sull'API, consulta Command DeleteOriginEndpointReference AWS CLI .

`describe-channel`

Il seguente esempio di codice mostra come utilizzare`describe-channel`.

AWS CLI

Come descrivere un canale

Il comando `describe-channel` seguente visualizza tutti i dettagli del canale denominato `test`.

```
aws mediapackage describe-channel \
    --id test
```

Output:

```
{
    "Arn": "arn:aws:mediapackage:us-
west-2:111222333:channels/584797f1740548c389a273585dd22a63",
    "HlsIngest": {
        "IngestEndpoints": [
            {
                "Id": "584797f1740548c389a273585dd22a63",
                "Password": "webdavgeneratedpassword1",
                "Url": "https://9be9c4405c474882.mediapackage.us-
west-2.amazonaws.com/in/
v2/584797f1740548c389a273585dd22a63/584797f1740548c389a273585dd22a63/channel",
                "Username": "webdavgeneratedusername1"
            },
            {
                "Id": "7d187c8616fd455f88aaa5a9fcf74442",
                "Password": "webdavgeneratedpassword2",
                "Url": "https://7bf454c57220328d.mediapackage.us-
west-2.amazonaws.com/in/
v2/584797f1740548c389a273585dd22a63/7d187c8616fd455f88aaa5a9fcf74442/channel",
                "Username": "webdavgeneratedusername2"
```

```
            }
        ]
    },
    "Id": "test",
    "Tags": {}
}
```

Per ulteriori informazioni, consulta Visualizzazione dei dettagli dei canali< https:// docs.aws.amazon.com/mediapackage/ latest/ug/channels -view.html> nella Guida per l'utente di Elemental AWS MediaPackage

- Per i dettagli sull'API, consulta Command Reference. DescribeChannelAWS CLI

`describe-origin-endpoint`

Il seguente esempio di codice mostra come utilizzare`describe-origin-endpoint`.

AWS CLI

Come descrivere un endpoint di origine

Il comando `describe-origin-endpoint` seguente visualizza tutti i dettagli dell'endpoint di origine denominato `cmaf_sports`.

```
aws mediapackage describe-origin-endpoint \
    --id cmaf_sports
```

Output:

```
{
    "Arn": "arn:aws:mediapackage:us-
west-2:111222333:origin_endpoints/1dc6718be36f4f34bb9cd86bc50925e6",
    "ChannelId": "sportschannel",
    "CmafPackage": {
        "HlsManifests": [
            {
                "AdMarkers": "NONE",
                "Id": "cmaf_sports_endpoint",
                "IncludeIframeOnlyStream": false,
                "PlaylistType": "EVENT",
                "PlaylistWindowSeconds": 60,
                "ProgramDateTimeIntervalSeconds": 0,
```

```
                "Url": "https://c4af3793bf76b33c.mediapackage.us-
west-2.amazonaws.com/out/v1/1dc6718be36f4f34bb9cd86bc50925e6/cmaf_sports_endpoint/
index.m3u8"
            }
        ],
        "SegmentDurationSeconds": 2,
        "SegmentPrefix": "sportschannel"
    },
    "Id": "cmaf_sports",
    "ManifestName": "index",
    "StartoverWindowSeconds": 0,
    "Tags": {
        "region": "west",
        "media": "sports"
    },
    "TimeDelaySeconds": 0,
    "Url": "",
    "Whitelist": []
}
```

Per ulteriori informazioni, consulta Viewing a Single Endpoint nella Guida per l' MediaPackage utente di AWS Elemental.

- Per i dettagli sull'API, consulta AWS CLI Command DescribeOriginEndpointReference.

`list-channels`

Il seguente esempio di codice mostra come utilizzare`list-channels`.

AWS CLI

Come elencare tutti i canali

Il `list-channels` comando seguente elenca tutti i canali configurati nell' AWS account corrente.

```
aws mediapackage list-channels
```

Output:

```
{
    "Channels": [
        {
```

```
            "Arn": "arn:aws:mediapackage:us-
west-2:111222333:channels/584797f1740548c389a273585dd22a63",
            "HlsIngest": {
                "IngestEndpoints": [
                    {
                        "Id": "584797f1740548c389a273585dd22a63",
                        "Password": "webdavgeneratedpassword1",
                        "Url": "https://9be9c4405c474882.mediapackage.us-
west-2.amazonaws.com/in/
v2/584797f1740548c389a273585dd22a63/584797f1740548c389a273585dd22a63/channel",
                        "Username": "webdavgeneratedusername1"
                    },
                    {
                        "Id": "7d187c8616fd455f88aaa5a9fcf74442",
                        "Password": "webdavgeneratedpassword2",
                        "Url": "https://7bf454c57220328d.mediapackage.us-
west-2.amazonaws.com/in/
v2/584797f1740548c389a273585dd22a63/7d187c8616fd455f88aaa5a9fcf74442/channel",
                        "Username": "webdavgeneratedusername2"
                    }
                ]
            },
            "Id": "test",
            "Tags": {}
        }
    ]
}
```

Per ulteriori informazioni, consulta Visualizzazione dei dettagli dei canali nella Guida per l' MediaPackage utente di AWS Elemental.

- Per i dettagli sull'API, consulta ListChannels AWS CLICommand Reference.

`list-origin-endpoints`

Il seguente esempio di codice mostra come utilizzare`list-origin-endpoints`.

AWS CLI

Come elencare tutti gli endpoint di origine in un canale

Il comando `list-origin-endpoints` seguente elenca tutti gli endpoint di origine configurati nel canale denominato `test`.

```
aws mediapackage list-origin-endpoints \
    --channel-id test
```

Output:

```
{
    "OriginEndpoints": [
        {
            "Arn": "arn:aws:mediapackage:us-
west-2:111222333:origin_endpoints/247cff871f2845d3805129be22f2c0a2",
            "ChannelId": "test",
            "DashPackage": {
                "ManifestLayout": "FULL",
                "ManifestWindowSeconds": 60,
                "MinBufferTimeSeconds": 30,
                "MinUpdatePeriodSeconds": 15,
                "PeriodTriggers": [],
                "Profile": "NONE",
                "SegmentDurationSeconds": 2,
                "SegmentTemplateFormat": "NUMBER_WITH_TIMELINE",
                "StreamSelection": {
                    "MaxVideoBitsPerSecond": 2147483647,
                    "MinVideoBitsPerSecond": 0,
                    "StreamOrder": "ORIGINAL"
                },
                "SuggestedPresentationDelaySeconds": 25
            },
            "Id": "tester2",
            "ManifestName": "index",
            "StartoverWindowSeconds": 0,
            "Tags": {},
            "TimeDelaySeconds": 0,
            "Url": "https://8343f7014c0ea438.mediapackage.us-west-2.amazonaws.com/
out/v1/247cff871f2845d3805129be22f2c0a2/index.mpd",
            "Whitelist": []
        },
        {
            "Arn": "arn:aws:mediapackage:us-
west-2:111222333:origin_endpoints/869e237f851549e9bcf10e3bc2830839",
            "ChannelId": "test",
            "HlsPackage": {
                "AdMarkers": "NONE",
                "IncludeIframeOnlyStream": false,
```

```
                "PlaylistType": "EVENT",
                "PlaylistWindowSeconds": 60,
                "ProgramDateTimeIntervalSeconds": 0,
                "SegmentDurationSeconds": 6,
                "StreamSelection": {
                    "MaxVideoBitsPerSecond": 2147483647,
                    "MinVideoBitsPerSecond": 0,
                    "StreamOrder": "ORIGINAL"
                },
                "UseAudioRenditionGroup": false
            },
            "Id": "tester",
            "ManifestName": "index",
            "StartoverWindowSeconds": 0,
            "Tags": {},
            "TimeDelaySeconds": 0,
            "Url": "https://8343f7014c0ea438.mediapackage.us-west-2.amazonaws.com/
out/v1/869e237f851549e9bcf10e3bc2830839/index.m3u8",
            "Whitelist": []
        }
    ]
}
```

Per ulteriori informazioni, consulta Visualizzazione di tutti gli endpoint associati a un canale nella Guida per l' MediaPackage utente di AWS Elemental.

- Per i dettagli sull'API, consulta AWS CLI Command ListOriginEndpointsReference.

`list-tags-for-resource`

Il seguente esempio di codice mostra come utilizzare`list-tags-for-resource`.

AWS CLI

Come elencare i tag assegnati a una risorsa

Il comando `list-tags-for-resource` seguente elenca i tag assegnati alla risorsa specificata.

```
aws mediapackage list-tags-for-resource \
    --resource-arn arn:aws:mediapackage:us-
west-2:111222333:channels/6d345804ec3f46c9b454a91d4a80d0e0
```

Output:

```
{
    "Tags": {
        "region": "west"
    }
}
```

Per ulteriori informazioni, consulta Tagging Resources in AWS Elemental nella MediaPackage AWS MediaPackage Elemental User Guide.

- Per i dettagli sull'API, consulta Command ListTagsForResourceReference AWS CLI .

`rotate-ingest-endpoint-credentials`

Il seguente esempio di codice mostra come utilizzare`rotate-ingest-endpoint-credentials`.

AWS CLI

Come ruotare le credenziali di importazione

Il comando `rotate-ingest-endpoint-credentials` seguente ruota il nome utente e la password WebDAV per l'endpoint di importazione specificato.

```
aws mediapackage rotate-ingest-endpoint-credentials \
    --id test \
    --ingest-endpoint-id 584797f1740548c389a273585dd22a63
```

Output:

```
{
    "Arn": "arn:aws:mediapackage:us-
west-2:111222333:channels/584797f1740548c389a273585dd22a63",
    "HlsIngest": {
        "IngestEndpoints": [
            {
                "Id": "584797f1740548c389a273585dd22a63",
                "Password": "webdavregeneratedpassword1",
                "Url": "https://9be9c4405c474882.mediapackage.us-
west-2.amazonaws.com/in/
v2/584797f1740548c389a273585dd22a63/584797f1740548c389a273585dd22a63/channel",
                "Username": "webdavregeneratedusername1"
            },
```

```
            {
                "Id": "7d187c8616fd455f88aaa5a9fcf74442",
                "Password": "webdavgeneratedpassword2",
                "Url": "https://7bf454c57220328d.mediapackage.us-
west-2.amazonaws.com/in/
v2/584797f1740548c389a273585dd22a63/7d187c8616fd455f88aaa5a9fcf74442/channel",
                "Username": "webdavgeneratedusername2"
            }
        ]
    },
    "Id": "test",
    "Tags": {}
}
```

Per ulteriori informazioni, consulta Rotazione delle credenziali su un URL di input nella Guida per l'utente di AWS MediaPackage Elemental.

- Per i dettagli sull'API, consulta Command RotateIngestEndpointCredentialsReference AWS CLI .

`tag-resource`

Il seguente esempio di codice mostra come utilizzare`tag-resource`.

AWS CLI

Come aggiungere un tag a una risorsa

I comandi `tag-resource` seguenti aggiungono una coppia chiave/valore `region=west` alla risorsa specificata.

```
aws mediapackage tag-resource \
    --resource-arn arn:aws:mediapackage:us-
west-2:111222333:channels/6d345804ec3f46c9b454a91d4a80d0e0 \
    --tags region=west
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Tagging Resources in AWS Elemental nella MediaPackage AWS MediaPackage Elemental User Guide.

- Per i dettagli sull'API, consulta Command TagResourceReference AWS CLI .

`untag-resource`

Il seguente esempio di codice mostra come utilizzare`untag-resource`.

AWS CLI

Come rimuovere un tag da una risorsa

Il comando `untag-resource` seguente rimuove il tag con la chiave `region` dal canale specificato.

```
aws mediapackage untag-resource \
    --resource-arn arn:aws:mediapackage:us-
west-2:111222333:channels/6d345804ec3f46c9b454a91d4a80d0e0 \
    --tag-keys region
```

Per ulteriori informazioni, consulta Tagging Resources in AWS Elemental nella MediaPackage AWS MediaPackage Elemental User Guide.

- Per i dettagli sull'API, consulta Command UntagResourceReference AWS CLI .

`update-channel`

Il seguente esempio di codice mostra come utilizzare`update-channel`.

AWS CLI

Come modificare un canale

Il comando `update-channel` seguente aggiorna il canale denominato `sportschannel` per includere la descrizione `24x7 sports`.

```
aws mediapackage update-channel \
    --id sportschannel \
    --description "24x7 sports"
```

Output:

```
{
    "Arn": "arn:aws:mediapackage:us-
west-2:111222333:channels/6d345804ec3f46c9b454a91d4a80d0e0",
```

```
    "Description": "24x7 sports",
    "HlsIngest": {
        "IngestEndpoints": [
            {
                "Id": "6d345804ec3f46c9b454a91d4a80d0e0",
                "Password": "generatedwebdavpassword1",
                "Url": "https://f31c86aed53b815a.mediapackage.us-
west-2.amazonaws.com/in/
v2/6d345804ec3f46c9b454a91d4a80d0e0/6d345804ec3f46c9b454a91d4a80d0e0/channel",
                "Username": "generatedwebdavusername1"
            },
            {
                "Id": "2daa32878af24803b24183727211b8ff",
                "Password": "generatedwebdavpassword2",
                "Url": "https://6ebbe7e04c4b0afa.mediapackage.us-
west-2.amazonaws.com/in/
v2/6d345804ec3f46c9b454a91d4a80d0e0/2daa32878af24803b24183727211b8ff/channel",
                "Username": "generatedwebdavusername2"
            }
        ]
    },
    "Id": "sportschannel",
    "Tags": {}
}
```

Per ulteriori informazioni, consulta Modifica di un canale nella Guida per l' MediaPackage utente di AWS Elemental.

- Per i dettagli sull'API, consulta UpdateChannel AWS CLICommand Reference.

`update-origin-endpoint`

Il seguente esempio di codice mostra come utilizzare`update-origin-endpoint`.

AWS CLI

Come aggiornare un endpoint di origine

Il comando `update-origin-endpoint` seguente aggiorna l'endpoint di origine denominato `cmaf_sports`. Cambia il ritardo in `0` secondi.

```
aws mediapackage update-origin-endpoint \
```

```
    --id cmaf_sports \
    --time-delay-seconds 0
```

Output:

```
{
    "Arn": "arn:aws:mediapackage:us-
west-2:111222333:origin_endpoints/1dc6718be36f4f34bb9cd86bc50925e6",
    "ChannelId": "sportschannel",
    "CmafPackage": {
        "HlsManifests": [
            {
                "AdMarkers": "NONE",
                "Id": "cmaf_sports_endpoint",
                "IncludeIframeOnlyStream": false,
                "PlaylistType": "EVENT",
                "PlaylistWindowSeconds": 60,
                "ProgramDateTimeIntervalSeconds": 0,
                "Url": "https://c4af3793bf76b33c.mediapackage.us-
west-2.amazonaws.com/out/v1/1dc6718be36f4f34bb9cd86bc50925e6/cmaf_sports_endpoint/
index.m3u8"
            }
        ],
        "SegmentDurationSeconds": 2,
        "SegmentPrefix": "sportschannel"
    },
    "Id": "cmaf_sports",
    "ManifestName": "index",
    "StartoverWindowSeconds": 0,
    "Tags": {
        "region": "west",
        "media": "sports"
    },
    "TimeDelaySeconds": 0,
    "Url": "",
    "Whitelist": []
}
```

Per ulteriori informazioni, consulta Modifica di un endpoint nella Guida per l' MediaPackage utente di AWS Elemental.

- Per i dettagli sull'API, consulta AWS CLI Command UpdateOriginEndpointReference.

MediaPackage Esempi di VOD che utilizzano AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with MediaPackage VOD.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`create-asset`

Il seguente esempio di codice mostra come utilizzare`create-asset`.

AWS CLI

Come creare un asset

L'`create-asset`esempio seguente crea una risorsa denominata `Chicken_Asset` nell' AWS account corrente. La risorsa inserisce il file `30sec_chicken.smil` in. MediaPackage

```
aws mediapackage-vod create-asset \
    --id chicken_asset \
    --packaging-group-id hls_chicken_gp \
    --source-role-arn arn:aws:iam::111122223333:role/EMP_Vod \
    --source-arn arn:aws:s3::111122223333:video-bucket/A/30sec_chicken.smil
```

Output:

```
{
    "Arn":"arn:aws:mediapackage-vod:us-west-2:111122223333:assets/chicken_asset",
    "Id":"chicken_asset",
    "PackagingGroupId":"hls_chicken_gp",
```

```
    "SourceArn":"arn:aws:s3::111122223333:video-bucket/A/30sec_chicken.smil",
    "SourceRoleArn":"arn:aws:iam::111122223333:role/EMP_Vod",
    "EgressEndpoints":[
        {
            "PackagingConfigurationId":"New_config_1",
            "Url":"https://c75ea2668ab49d02bca7ae10ef31c59e.egress.mediapackage-
vod.us-west-2.amazonaws.com/out/
v1/6644b55df1744261ab3732a8e5cdaf07/904b06a58c7645e08d57d40d064216ac/
f5b2e633ff4942228095d164c10074f3/index.m3u8"
        },
        {
            "PackagingConfigurationId":"new_hls",
            "Url":" https://c75ea2668ab49d02bca7ae10ef31c59e.egress.mediapackage-
vod.us-west-2.amazonaws.com/out/v1/6644b55df1744261ab3732a8e5cdaf07/
fe8f1f00a80e424cb4f8da4095835e9e/7370ec57432343af816332356d2bd5c6/string.m3u8"
        }
    ]
}
```

Per ulteriori informazioni, consulta Ingest an Asset nella AWS MediaPackage Elemental User Guide.

- Per i dettagli sull'API, consulta AWS CLI Command CreateAssetReference.

`create-packaging-configuration`

Il seguente esempio di codice mostra come utilizzare`create-packaging-configuration`.

AWS CLI

Come creare una configurazione di creazione pacchetti

L'esempio `create-packaging-configuration` seguente crea una configurazione di creazione pacchetti denominata `new_hls` nel gruppo di pacchetti denominato `hls_chicken`. Questo esempio utilizza un file su disco denominato `hls_pc.json` per fornire i dettagli.

```
aws mediapackage-vod create-packaging-configuration \
    --id new_hls \
    --packaging-group-id hls_chicken \
    --hls-package file://hls_pc.json
```

Contenuto di `hls_pc.json`:

```
{
    "HlsManifests":[
        {
            "AdMarkers":"NONE",
            "IncludeIframeOnlyStream":false,
            "ManifestName":"string",
            "ProgramDateTimeIntervalSeconds":60,
            "RepeatExtXKey":true,
            "StreamSelection":{
                "MaxVideoBitsPerSecond":1000,
                "MinVideoBitsPerSecond":0,
                "StreamOrder":"ORIGINAL"
            }
        }
    ],
    "SegmentDurationSeconds":6,
    "UseAudioRenditionGroup":false
}
```

Output:

```
{
    "Arn":"arn:aws:mediapackage-vod:us-west-2:111122223333:packaging-configurations/
new_hls",
    "Id":"new_hls",
    "PackagingGroupId":"hls_chicken",
    "HlsManifests":{
        "SegmentDurationSeconds":6,
        "UseAudioRenditionGroup":false,
        "HlsMarkers":[
            {
                "AdMarkers":"NONE",
                "IncludeIframeOnlyStream":false,
                "ManifestName":"string",
                "ProgramDateTimeIntervalSeconds":60,
                "RepeatExtXKey":true,
                "StreamSelection":{
                    "MaxVideoBitsPerSecond":1000,
                    "MinVideoBitsPerSecond":0,
                    "StreamOrder":"ORIGINAL"
                }
            }
        ]
```

```
    }
}
```

Per ulteriori informazioni, consulta Creazione di una configurazione di packaging nella Guida per l' MediaPackage utente di AWS Elemental.

- Per i dettagli sull'API, consulta CreatePackagingConfiguration AWS CLICommand Reference.

`create-packaging-group`

Il seguente esempio di codice mostra come utilizzare`create-packaging-group`.

AWS CLI

Come creare un gruppo di creazione pacchetti

L'`create-packaging-group`esempio seguente elenca tutti i gruppi di pacchetti configurati nell' AWS account corrente.

```
aws mediapackage-vod create-packaging-group \
    --id hls_chicken
```

Output:

```
{
    "Arn": "arn:aws:mediapackage-vod:us-west-2:111122223333:packaging-groups/
hls_chicken",
    "Id": "hls_chicken"
}
```

Per ulteriori informazioni, consulta Creazione di un gruppo di pacchetti nella Guida per l' MediaPackage utente di AWS Elemental.

- Per i dettagli sull'API, consulta CreatePackagingGroup AWS CLICommand Reference.

`delete-asset`

Il seguente esempio di codice mostra come utilizzare`delete-asset`.

AWS CLI

Come eliminare un asset

L'esempio `delete-asset` seguente elimina l'asset denominato `30sec_chicken`.

```
aws mediapackage-vod delete-asset \
    --id 30sec_chicken
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Eliminazione di una risorsa nella Guida per l' MediaPackage utente di AWS Elemental.

- Per i dettagli sull'API, consultate AWS CLI Command DeleteAssetReference.

`delete-packaging-configuration`

Il seguente esempio di codice mostra come utilizzare`delete-packaging-configuration`.

AWS CLI

Come eliminare una configurazione di creazione pacchetti

L'esempio `delete-packaging-configuration` seguente elimina la configurazione di creazione pacchetti denominata `CMAF`.

```
aws mediapackage-vod delete-packaging-configuration \
    --id CMAF
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Eliminazione di una configurazione di packaging nella Guida per l' MediaPackage utente di AWS Elemental.

- Per i dettagli sull'API, consulta AWS CLI Command DeletePackagingConfigurationReference.

`delete-packaging-group`

Il seguente esempio di codice mostra come utilizzare`delete-packaging-group`.

AWS CLI

Come eliminare un gruppo di creazione pacchetti

L'esempio `delete-packaging-group` seguente elimina il gruppo di creazione pacchetti denominato `Dash_widevine`.

```
aws mediapackage-vod delete-packaging-group \
    --id Dash_widevine
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Eliminazione di un gruppo di pacchetti nella Guida per l' MediaPackage utente di AWS Elemental.

- Per i dettagli sull'API, consulta AWS CLI Command DeletePackagingGroupReference.

`describe-asset`

Il seguente esempio di codice mostra come utilizzare`describe-asset`.

AWS CLI

Come descrivere un asset

L'esempio `describe-asset` seguente mostra tutti i dettagli dell'asset denominato `30sec_chicken`.

```
aws mediapackage-vod describe-asset \
    --id 30sec_chicken
```

Output:

```
{
    "Arn":"arn:aws:mediapackage-vod:us-west-2:111122223333:assets/30sec_chicken",
    "Id":"30sec_chicken",
    "PackagingGroupId":"Packaging_group_1",
    "SourceArn":"arn:aws:s3::111122223333:video-bucket/A/30sec_chicken.smil",
    "SourceRoleArn":"arn:aws:iam::111122223333:role/EMP_Vod",
    "EgressEndpoints":[
        {
            "PackagingConfigurationId":"DASH",
            "Url":"https://a5f46a44118ba3e3724ef39ef532e701.egress.mediapackage-
vod.us-west-2.amazonaws.com/out/v1/
aad7962c569946119c2d5a691be5663c/66c25aff456d463aae0855172b3beb27/4ddfda6da17c4c279a1b8401cb
index.mpd"
        },
        {
            "PackagingConfigurationId":"HLS",
```

```
            "Url":"https://a5f46a44118ba3e3724ef39ef532e701.egress.mediapackage-
vod.us-west-2.amazonaws.com/out/v1/
aad7962c569946119c2d5a691be5663c/6e5bf286a3414254a2bf0d22ae148d7e/06b5875b4d004c3cbdc4da2dc4
index.m3u8"
        },
        {
            "PackagingConfigurationId":"CMAF",
            "Url":"https://a5f46a44118ba3e3724ef39ef532e701.egress.mediapackage-
vod.us-west-2.amazonaws.com/out/v1/
aad7962c569946119c2d5a691be5663c/628fb5d8d89e4702958b020af27fde0e/05eb062214064238ad6330a443
index.m3u8"
        }
    ]
}
```

Per ulteriori informazioni, consulta Visualizzazione dei dettagli degli asset nella Guida per l' MediaPackage utente di AWS Elemental.

- Per i dettagli sull'API, consultate DescribeAsset AWS CLICommand Reference.

`describe-packaging-configuration`

Il seguente esempio di codice mostra come utilizzare`describe-packaging-configuration`.

AWS CLI

Come descrivere una configurazione di creazione pacchetti

L'esempio `describe-packaging-configuration` seguente visualizza tutti i dettagli della configurazione di creazione pacchetti denominata `DASH`.

```
aws mediapackage-vod describe-packaging-configuration \
    --id DASH
```

Output:

```
{
    "Arn":"arn:aws:mediapackage-vod:us-west-2:111122223333:packaging-configurations/
DASH",
    "Id":"DASH",
    "PackagingGroupId":"Packaging_group_1",
    "DashPackage":[
        {
```

```
            "SegmentDurationSeconds":"2"
        },
        {
            "DashManifests":{
                "ManifestName":"index",
                "MinBufferTimeSeconds":"30",
                "Profile":"NONE"
            }
        }
    ]
}
```

Per ulteriori informazioni, consulta Visualizzazione dei dettagli della configurazione del pacchetto nella Guida per l' MediaPackage utente di AWS Elemental.

- Per i dettagli sull'API, consulta DescribePackagingConfiguration AWS CLICommand Reference.

`describe-packaging-group`

Il seguente esempio di codice mostra come utilizzare`describe-packaging-group`.

AWS CLI

Come descrivere un gruppo di creazione pacchetti

L'esempio `describe-packaging-group` seguente visualizza tutti i dettagli del gruppo di creazione pacchetti denominato `Packaging_group_1`.

```
aws mediapackage-vod describe-packaging-group \
    --id Packaging_group_1
```

Output:

```
{
    "Arn": "arn:aws:mediapackage-vod:us-west-2:111122223333:packaging-groups/
Packaging_group_1",
    "Id": "Packaging_group_1"
}
```

Per ulteriori informazioni, consulta Visualizzazione dei dettagli del gruppo di pacchetti nella Guida per l' MediaPackage utente di AWS Elemental.

- Per i dettagli sull'API, consulta DescribePackagingGroup AWS CLICommand Reference.

`list-assets`

Il seguente esempio di codice mostra come utilizzare`list-assets`.

AWS CLI

Come elencare tutti gli asset

L'`list-assets`esempio seguente elenca tutte le risorse configurate nell' AWS account corrente.

```
aws mediapackage-vod list-assets
```

Output:

```
{
    "Assets": [
        "Arn": "arn:aws:mediapackage-vod:us-
west-2:111122223333:assets/30sec_chicken",
        "Id": "30sec_chicken",
        "PackagingGroupId": "Packaging_group_1",
        "SourceArn": "arn:aws:s3::111122223333:video-bucket/A/30sec_chicken.smil",
        "SourceRoleArn": "arn:aws:iam::111122223333:role/EMP_Vod"
    ]
}
```

Per ulteriori informazioni, consultate Visualizzazione dei dettagli degli asset nella Guida per l' MediaPackage utente di AWS Elemental.

- Per i dettagli sull'API, consultate ListAssets AWS CLICommand Reference.

`list-packaging-configurations`

Il seguente esempio di codice mostra come utilizzare`list-packaging-configurations`.

AWS CLI

Come elencare tutte le configurazioni di creazione pacchetti

L'esempio `list-packaging-configurations` seguente elenca tutte le configurazioni di creazione pacchetti configurate nel gruppo di creazione pacchetti denominato `Packaging_group_1`.

```
aws mediapackage-vod list-packaging-configurations \
    --packaging-group-id Packaging_group_1
```

Output:

```
{
    "PackagingConfigurations":[
        {
            "Arn":"arn:aws:mediapackage-vod:us-west-2:111122223333:packaging-
configurations/CMAF",
            "Id":"CMAF",
            "PackagingGroupId":"Packaging_group_1",
            "CmafPackage":[
                {
                    "SegmentDurationSeconds":"2"
                },
                {
                    "HlsManifests":{
                        "AdMarkers":"NONE",
                        "RepeatExtXKey":"False",
                        "ManifestName":"index",
                        "ProgramDateTimeIntervalSeconds":"0",
                        "IncludeIframeOnlyStream":"False"
                    }
                }
            ]
        },
        {
            "Arn":"arn:aws:mediapackage-vod:us-west-2:111122223333:packaging-
configurations/DASH",
            "Id":"DASH",
            "PackagingGroupId":"Packaging_group_1",
            "DashPackage":[
                {
                    "SegmentDurationSeconds":"2"
                },
                {
                    "DashManifests":{
                        "ManifestName":"index",
                        "MinBufferTimeSeconds":"30",
                        "Profile":"NONE"
                    }
                }
```

```
            ]
        },
        {
            "Arn":"arn:aws:mediapackage-vod:us-west-2:111122223333:packaging-
configurations/HLS",
            "Id":"HLS",
            "PackagingGroupId":"Packaging_group_1",
            "HlsPackage":[
                {
                    "SegmentDurationSeconds":"6",
                    "UseAudioRenditionGroup":"False"
                },
                {
                    "HlsManifests":{
                        "AdMarkers":"NONE",
                        "RepeatExtXKey":"False",
                        "ManifestName":"index",
                        "ProgramDateTimeIntervalSeconds":"0",
                        "IncludeIframeOnlyStream":"False"
                    }
                }
            ]
        },
        {
            "Arn":"arn:aws:mediapackage-vod:us-west-2:111122223333:packaging-
configurations/New_config_0_copy",
            "Id":"New_config_0_copy",
            "PackagingGroupId":"Packaging_group_1",
            "HlsPackage":[
                {
                    "SegmentDurationSeconds":"6",
                    "UseAudioRenditionGroup":"False"
                },
                {
                    "Encryption":{
                        "EncryptionMethod":"AWS_128",
                        "SpekeKeyProvider":{
                           "RoleArn":"arn:aws:iam:111122223333::role/SPEKERole",
                            "Url":"https://lfgubdvs97.execute-api.us-
west-2.amazonaws.com/EkeStage/copyProtection/",
                            "SystemIds":[
                                "81376844-f976-481e-a84e-cc25d39b0b33"
                            ]
                        }
```

```
                    }
                },
                {
                    "HlsManifests":{
                        "AdMarkers":"NONE",
                        "RepeatExtXKey":"False",
                        "ManifestName":"index",
                        "ProgramDateTimeIntervalSeconds":"0",
                        "IncludeIframeOnlyStream":"False"
                    }
                }
            ]
        }
    ]
}
```

Per ulteriori informazioni, consulta Visualizzazione dei dettagli della configurazione del pacchetto nella Guida per l' MediaPackage utente di AWS Elemental.

- Per i dettagli sull'API, consulta ListPackagingConfigurations AWS CLICommand Reference.

`list-packaging-groups`

Il seguente esempio di codice mostra come utilizzare`list-packaging-groups`.

AWS CLI

Come elencare tutti i gruppi di creazione pacchetti

L'`list-packaging-groups`esempio seguente elenca tutti i gruppi di pacchetti configurati nell' AWS account corrente.

```
aws mediapackage-vod list-packaging-groups
```

Output:

```
{
    "PackagingGroups": [
        {
            "Arn": "arn:aws:mediapackage-vod:us-west-2:111122223333:packaging-
groups/Dash_widevine",
```

```
            "Id": "Dash_widevine"
        },
        {
            "Arn": "arn:aws:mediapackage-vod:us-west-2:111122223333:packaging-
groups/Encrypted_HLS",
            "Id": "Encrypted_HLS"
        },
        {
            "Arn": "arn:aws:mediapackage-vod:us-west-2:111122223333:packaging-
groups/Packaging_group_1",
            "Id": "Packaging_group_1"
        }
    ]
}
```

Per ulteriori informazioni, consulta Visualizzazione dei dettagli del gruppo di pacchetti nella Guida per l' MediaPackage utente di AWS Elemental.

- Per i dettagli sull'API, consulta ListPackagingGroups AWS CLICommand Reference.

MediaStore Esempi di piani dati che utilizzano AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with MediaStore Data Plane.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`delete-object`

Il seguente esempio di codice mostra come utilizzare`delete-object`.

AWS CLI

Come eliminare un oggetto

L'esempio `delete-object` seguente elimina l'oggetto specificato.

```
aws mediastore-data delete-object \
    --endpoint=https://aaabbbcccdddee.data.mediastore.us-west-2.amazonaws.com \
    --path=/folder_name/README.md
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Eliminazione di un oggetto nella Guida MediaStore utente di AWS Elemental.

- Per i dettagli sull'API, consulta AWS CLI Command DeleteObjectReference.

`describe-object`

Il seguente esempio di codice mostra come utilizzare`describe-object`.

AWS CLI

Come visualizzare le intestazioni di un oggetto

L'esempio `describe-object` seguente visualizza le intestazioni di un oggetto nel percorso specificato.

```
aws mediastore-data describe-object \
    --endpoint https://aaabbbcccdddee.data.mediastore.us-west-2.amazonaws.com \
    --path events/baseball/setup.jpg
```

Output:

```
{
    "LastModified": "Fri, 19 Jul 2019 21:50:31 GMT",
    "ContentType": "image/jpeg",
    "ContentLength": "3860266",
    "ETag": "2aa333bbcc8d8d22d777e999c88d4aa9eeeeee4dd89ff7f555555555555da6d3"
}
```

Per ulteriori informazioni, consulta Visualizzazione dei dettagli di un oggetto nella Guida per l' MediaStore utente di AWS Elemental.

- Per i dettagli sull'API, consulta DescribeObject AWS CLICommand Reference.

`get-object`

Il seguente esempio di codice mostra come utilizzare`get-object`.

AWS CLI

Esempio 1: come scaricare un intero oggetto

L'esempio `get-object` seguente scarica l'oggetto specificato.

```
aws mediastore-data get-object \
    --endpoint https://aaabbbcccdddee.data.mediastore.us-west-2.amazonaws.com \
    --path events/baseball/setup.jpg setup.jpg
```

Output:

```
{
    "ContentType": "image/jpeg",
    "StatusCode": 200,
    "ETag": "2aa333bbcc8d8d22d777e999c88d4aa9eeeeee4dd89ff7f555555555555da6d3",
    "ContentLength": "3860266",
    "LastModified": "Fri, 19 Jul 2019 21:50:31 GMT"
}
```

Esempio 2: come scaricare parte di un oggetto

L'esempio `get-object` seguente scarica la parte specificata di un oggetto.

```
aws mediastore-data get-object \
    --endpoint https://aaabbbcccdddee.data.mediastore.us-west-2.amazonaws.com \
    --path events/baseball/setup.jpg setup.jpg \
    --range "bytes=0-100"
```

Output:

```
{
    "StatusCode": 206,
```

```
    "LastModified": "Fri, 19 Jul 2019 21:50:31 GMT",
    "ContentType": "image/jpeg",
    "ContentRange": "bytes 0-100/3860266",
    "ETag": "2aa333bbcc8d8d22d777e999c88d4aa9eeeeee4dd89ff7f555555555555da6d3",
    "ContentLength": "101"
}
```

Per ulteriori informazioni, consulta Downloading an Object nella AWS Elemental MediaStore User Guide.

- Per i dettagli sull'API, consulta GetObject AWS CLICommand Reference.

`list-items`

Il seguente esempio di codice mostra come utilizzare`list-items`.

AWS CLI

Esempio 1: come visualizzare un elenco di elementi (oggetti e cartelle) memorizzati in un container

L'esempio `list-items` seguente visualizza un elenco di elementi (oggetti e cartelle) archiviati nel container specificato.

```
aws mediastore-data list-items \
    --endpoint https://aaabbbcccdddee.data.mediastore.us-west-2.amazonaws.com
```

Output:

```
{
    "Items": [
        {
            "Type": "OBJECT",
            "ContentLength": 3784,
            "Name": "setup.jpg",
            "ETag":
 "2aa333bbcc8d8d22d777e999c88d4aa9eeeeee4dd89ff7f555555555555da6d3",
            "ContentType": "image/jpeg",
            "LastModified": 1563571859.379
        },
        {
            "Type": "FOLDER",
```

```
            "Name": "events"
        }
    ]
}
```

Esempio 2: come visualizzare un elenco di elementi (oggetti e cartelle) memorizzati in una cartella

L'esempio `list-items` seguente visualizza un elenco di elementi (oggetti e cartelle) archiviati nella cartella specificata.

```
aws mediastore-data list-items \
    --endpoint https://aaabbbcccdddee.data.mediastore.us-west-2.amazonaws.com \
    --path events/baseball
```

Output:

```
{
    "Items": [
        {
            "ETag":
 "2aa333bbcc8d8d22d777e999c88d4aa9eeeeee4dd89ff7f555555555555da6d3",
            "ContentType": "image/jpeg",
            "Type": "OBJECT",
            "ContentLength": 3860266,
            "LastModified": 1563573031.872,
            "Name": "setup.jpg"
        }
    ]
}
```

Per ulteriori informazioni, consulta Visualizzazione di un elenco di oggetti nella Guida MediaStore utente di AWS Elemental.

- Per i dettagli sull'API, consulta ListItems AWS CLICommand Reference.

`put-object`

Il seguente esempio di codice mostra come utilizzare`put-object`.

AWS CLI

Esempio 1: come caricare un oggetto in un container

L'esempio `put-object` seguente carica un oggetto nel container specificato.

```
aws mediastore-data put-object \
    --endpoint https://aaabbbcccdddee.data.mediastore.us-west-2.amazonaws.com \
    --body ReadMe.md \
    --path ReadMe.md \
    --cache-control "max-age=6, public" \
    --content-type binary/octet-stream
```

Output:

```
{
    "ContentSHA256":
 "f29bc64a9d3732b4b9035125fdb3285f5b6455778edca72414671e0ca3b2e0de",
    "StorageClass": "TEMPORAL",
    "ETag": "2aa333bbcc8d8d22d777e999c88d4aa9eeeeee4dd89ff7f555555555555da6d3"
}
```

Esempio 2: come caricare un oggetto in una cartella all'interno di un container

L'esempio `put-object` seguente carica un oggetto nella cartella specificata all'interno di un container.

```
aws mediastore-data put-object \
    --endpoint https://aaabbbcccdddee.data.mediastore.us-west-2.amazonaws.com \
    --body ReadMe.md \
    --path /september-events/ReadMe.md \
    --cache-control "max-age=6, public" \
    --content-type binary/octet-stream
```

Output:

```
{
    "ETag": "2aa333bbcc8d8d22d777e999c88d4aa9eeeeee4dd89ff7f555555555555da6d3",
    "ContentSHA256":
 "f29bc64a9d3732b4b9035125fdb3285f5b6455778edca72414671e0ca3b2e0de",
    "StorageClass": "TEMPORAL"
}
```

Per ulteriori informazioni, consulta Uploading an Object nella AWS Elemental MediaStore User Guide.

- Per i dettagli sull'API, consulta AWS CLI Command PutObjectReference.

MediaTailor esempi utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with MediaTailor.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`delete-playback-configuration`

Il seguente esempio di codice mostra come utilizzare`delete-playback-configuration`.

AWS CLI

Come eliminare una configurazione

Il codice `delete-playback-configuration` seguente elimina la configurazione denominata `campaign_short`.

```
aws mediatailor delete-playback-configuration \
    --name campaign_short
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Eliminazione di una configurazione nella Guida per l' MediaTailor utente di AWS Elemental.

- Per i dettagli sull'API, consulta AWS CLI Command DeletePlaybackConfigurationReference.

`get-playback-configuration`

Il seguente esempio di codice mostra come utilizzare`get-playback-configuration`.

AWS CLI

Come descrivere una configurazione

Il codice `get-playback-configuration` seguente visualizza tutti i dettagli della configurazione denominata `west_campaign`.

```
aws mediatailor get-playback-configuration \
    --name west_campaign
```

Output:

```
{
     "AdDecisionServerUrl": "http://your.ads.url",
     "CdnConfiguration": {},
     "DashConfiguration": {
         "ManifestEndpointPrefix":
 "https://170c14299689462897d0cc45fc2000bb.mediatailor.us-west-2.amazonaws.com/v1/
dash/1cbfeaaecb69778e0c167d0505a2bc57da2b1754/west_campaign/",
         "MpdLocation": "EMT_DEFAULT",
         "OriginManifestType": "MULTI_PERIOD"
     },
     "HlsConfiguration": {
         "ManifestEndpointPrefix":
 "https://170c14299689462897d0cc45fc2000bb.mediatailor.us-west-2.amazonaws.com/v1/
master/1cbfeaaecb69778e0c167d0505a2bc57da2b1754/west_campaign/"
     },
     "Name": "west_campaign",
     "PlaybackConfigurationArn": "arn:aws:mediatailor:us-
west-2:123456789012:playbackConfiguration/west_campaign",
     "PlaybackEndpointPrefix":
 "https://170c14299689462897d0cc45fc2000bb.mediatailor.us-west-2.amazonaws.com",
     "SessionInitializationEndpointPrefix":
 "https://170c14299689462897d0cc45fc2000bb.mediatailor.us-west-2.amazonaws.com/v1/
session/1cbfeaaecb69778e0c167d0505a2bc57da2b1754/west_campaign/",
     "Tags": {},
     "VideoContentSourceUrl": "https://8343f7014c0ea438.mediapackage.us-
west-2.amazonaws.com/out/v1/683f0f2ff7cd43a48902e6dcd5e16dcf/index.m3u8"
}
```

Per ulteriori informazioni, consulta Visualizzazione di una configurazione nella Guida per l' MediaTailor utente di AWS Elemental.

- Per i dettagli sull'API, consulta GetPlaybackConfiguration AWS CLICommand Reference.

`list-playback-configurations`

Il seguente esempio di codice mostra come utilizzare`list-playback-configurations`.

AWS CLI

Come elencare tutte le configurazioni

Di seguito `list-playback-configurations` vengono visualizzati tutti i dettagli della configurazione dell' AWS account corrente.

```
aws mediatailor list-playback-configurations
```

Output:

```
{
    "Items": [
        {
            "AdDecisionServerUrl": "http://your.ads.url",
            "CdnConfiguration": {},
            "DashConfiguration": {
                "ManifestEndpointPrefix":
 "https://170c14299689462897d0cc45fc2000bb.mediatailor.us-west-2.amazonaws.com/v1/
dash/1cbfeaaecb69778e0c167d0505a2bc57da2b1754/west_campaign/",
                "MpdLocation": "EMT_DEFAULT",
                "OriginManifestType": "MULTI_PERIOD"
            },
            "HlsConfiguration": {
                "ManifestEndpointPrefix":
 "https://170c14299689462897d0cc45fc2000bb.mediatailor.us-west-2.amazonaws.com/v1/
master/1cbfeaaecb69778e0c167d0505a2bc57da2b1754/west_campaign/"
            },
            "Name": "west_campaign",
            "PlaybackConfigurationArn": "arn:aws:mediatailor:us-
west-2:123456789012:playbackConfiguration/west_campaign",
            "PlaybackEndpointPrefix":
 "https://170c14299689462897d0cc45fc2000bb.mediatailor.us-west-2.amazonaws.com",
```

```
            "SessionInitializationEndpointPrefix":
 "https://170c14299689462897d0cc45fc2000bb.mediatailor.us-west-2.amazonaws.com/v1/
session/1cbfeaaecb69778e0c167d0505a2bc57da2b1754/west_campaign/",
            "Tags": {},
            "VideoContentSourceUrl": "https://8343f7014c0ea438.mediapackage.us-
west-2.amazonaws.com/out/v1/683f0f2ff7cd43a48902e6dcd5e16dcf/index.m3u8"
        },
        {
            "AdDecisionServerUrl": "http://your.ads.url",
            "CdnConfiguration": {},
            "DashConfiguration": {
                "ManifestEndpointPrefix":
 "https://73511f91d6a24ca2b93f3cf1d7cedd67.mediatailor.us-west-2.amazonaws.com/v1/
dash/1cbfeaaecb69778e0c167d0505a2bc57da2b1754/sports_campaign/",
                "MpdLocation": "DISABLED",
                "OriginManifestType": "MULTI_PERIOD"
            },
            "HlsConfiguration": {
                "ManifestEndpointPrefix":
 "https://73511f91d6a24ca2b93f3cf1d7cedd67.mediatailor.us-west-2.amazonaws.com/v1/
master/1cbfeaaecb69778e0c167d0505a2bc57da2b1754/sports_campaign/"
            },
            "Name": "sports_campaign",
            "PlaybackConfigurationArn": "arn:aws:mediatailor:us-
west-2:123456789012:playbackConfiguration/sports_campaign",
            "PlaybackEndpointPrefix":
 "https://73511f91d6a24ca2b93f3cf1d7cedd67.mediatailor.us-west-2.amazonaws.com",
            "SessionInitializationEndpointPrefix":
 "https://73511f91d6a24ca2b93f3cf1d7cedd67.mediatailor.us-west-2.amazonaws.com/v1/
session/1cbfeaaecb69778e0c167d0505a2bc57da2b1754/sports_campaign/",
            "SlateAdUrl": "http://s3.bucket/slate_ad.mp4",
            "Tags": {},
            "VideoContentSourceUrl": "https://c4af3793bf76b33c.mediapackage.us-
west-2.amazonaws.com/out/v1/1dc6718be36f4f34bb9cd86bc50925e6/sports_endpoint/
index.m3u8"
        }
    ]
}
```

Per ulteriori informazioni, consulta Visualizzazione di una configurazione< https:// docs.aws.amazon.com/mediatailor/ latest/ug/configurations -view.html> nella Guida per l'utente di Elemental.AWS MediaTailor

- Per i dettagli sull'API, consulta Command Reference. ListPlaybackConfigurationsAWS CLI

`put-playback-configuration`

Il seguente esempio di codice mostra come utilizzare`put-playback-configuration`.

AWS CLI

Come creare una configurazione

Il codice `put-playback-configuration` seguente crea una configurazione denominata `campaign_short`.

```
aws mediatailor put-playback-configuration \
    --name campaign_short \
    --ad-decision-server-url http://your.ads.url \
    --video-content-source-url http://video.bucket/index.m3u8
```

Output:

```
{
    "AdDecisionServerUrl": "http://your.ads.url",
    "CdnConfiguration": {},
    "DashConfiguration": {
         "ManifestEndpointPrefix":
 "https://13484114d38f4383bc0d6a7cb879bd00.mediatailor.us-west-2.amazonaws.com/v1/
dash/1cbfeaaecb69778e0c167d0505a2bc57da2b1754/campaign_short/",
         "MpdLocation": "EMT_DEFAULT",
         "OriginManifestType": "MULTI_PERIOD"
    },
    "HlsConfiguration": {
        "ManifestEndpointPrefix":
 "https://13484114d38f4383bc0d6a7cb879bd00.mediatailor.us-west-2.amazonaws.com/v1/
master/1cbfeaaecb69778e0c167d0505a2bc57da2b1754/campaign_short/"
    },
    "Name": "campaign_short",
    "PlaybackConfigurationArn": "arn:aws:mediatailor:us-
west-2:123456789012:playbackConfiguration/campaign_short",
    "PlaybackEndpointPrefix":
 "https://13484114d38f4383bc0d6a7cb879bd00.mediatailor.us-west-2.amazonaws.com",
    "SessionInitializationEndpointPrefix":
 "https://13484114d38f4383bc0d6a7cb879bd00.mediatailor.us-west-2.amazonaws.com/v1/
session/1cbfeaaecb69778e0c167d0505a2bc57da2b1754/campaign_short/",
    "Tags": {},
    "VideoContentSourceUrl": "http://video.bucket/index.m3u8"
```

```
}
```

Per ulteriori informazioni, consulta Creazione di una configurazione nella Guida MediaTailor utente di AWS Elemental.

- Per i dettagli sull'API, consulta PutPlaybackConfiguration AWS CLICommand Reference.

Esempi di MemoryDB che utilizzano AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with MemoryDB.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`copy-snapshot`

Il seguente esempio di codice mostra come utilizzare. `copy-snapshot`

AWS CLI

Come copiare una snapshot

L'esempio `copy-snapshot` seguente crea una copia di uno snapshot.

```
aws memorydb copy-snapshot \
    --source-snapshot-name my-cluster-snapshot \
    --target-snapshot-name my-cluster-snapshot-copy
```

Output

```
{
```

```
    "Snapshot": {
        "Name": "my-cluster-snapshot-copy",
        "Status": "creating",
        "Source": "manual",
        "ARN": "arn:aws:memorydb:us-east-1:491658xxxxxx:snapshot/my-cluster-
snapshot-copy",
        "ClusterConfiguration": {
            "Name": "my-cluster",
            "Description": " ",
            "NodeType": "db.r6g.large",
            "EngineVersion": "6.2",
            "MaintenanceWindow": "wed:03:00-wed:04:00",
            "Port": 6379,
            "ParameterGroupName": "default.memorydb-redis6",
            "SubnetGroupName": "my-sg",
            "VpcId": "vpc-xx2574fc",
            "SnapshotRetentionLimit": 0,
            "SnapshotWindow": "04:30-05:30",
            "NumShards": 2
        }
    }
}
```

Per ulteriori informazioni, consulta Copia di uno snapshot nella Guida per l'utente di MemoryDB.

- Per i dettagli sull'API, consulta CopySnapshot AWS CLICommand Reference.

`create-acl`

Il seguente esempio di codice mostra come utilizzare`create-acl`.

AWS CLI

Come creare una lista di controllo degli accessi (ACL)

L'esempio `create-acl` seguente crea una nuova lista di controllo degli accessi (ACL).

```
aws memorydb create-acl \
    --acl-name "new-acl-1" \
    --user-names "my-user"
```

Output:

```
{
    "ACL": {
        "Name": "new-acl-1",
        "Status": "creating",
        "UserNames": [
            "my-user"
        ],
        "MinimumEngineVersion": "6.2",
        "Clusters": [],
        "ARN": "arn:aws:memorydb:us-east-1:491658xxxxxx:acl/new-acl-1"
    }
}
```

Per ulteriori informazioni, consulta Autenticazione degli utenti con le liste di controllo accessi nella Guida per l'utente di MemoryDB.

- Per i dettagli sull'API, consulta CreateAcl AWS CLICommand Reference.

`create-cluster`

Il seguente esempio di codice mostra come utilizzare`create-cluster`.

AWS CLI

Come creare un cluster

L'esempio `create-cluster` seguente crea un nuovo cluster.

```
aws memorydb create-cluster \
    --cluster-name my-new-cluster \
    --node-type db.r6g.large \
    --acl-name my-acl \
    --subnet-group my-sg
```

Output:

```
{
    "Cluster": {
        "Name": "my-new-cluster",
        "Status": "creating",
        "NumberOfShards": 1,
```

```
        "AvailabilityMode": "MultiAZ",
        "ClusterEndpoint": {
            "Port": 6379
        },
        "NodeType": "db.r6g.large",
        "EngineVersion": "6.2",
        "EnginePatchVersion": "6.2.6",
        "ParameterGroupName": "default.memorydb-redis6",
        "ParameterGroupStatus": "in-sync",
        "SubnetGroupName": "my-sg",
        "TLSEnabled": true,
        "ARN": "arn:aws:memorydb:us-east-1:49165xxxxxx:cluster/my-new-cluster",
        "SnapshotRetentionLimit": 0,
        "MaintenanceWindow": "sat:10:00-sat:11:00",
        "SnapshotWindow": "07:30-08:30",
        "ACLName": "my-acl",
        "AutoMinorVersionUpgrade": true
    }
}
```

Per ulteriori informazioni, consulta Gestione dei cluster nella Guida per l'utente di MemoryDB.

- Per i dettagli sull'API, consulta CreateCluster AWS CLICommand Reference.

`create-parameter-group`

Il seguente esempio di codice mostra come utilizzare`create-parameter-group`.

AWS CLI

Come creare un gruppo di parametri

L'esempio `create-parameter-group` seguente crea un gruppo di parametri.

```
aws memorydb create-parameter-group \
    --parameter-group-name myRedis6x \
    --family memorydb_redis6 \
    --description "my-parameter-group"
```

Output:

```
{
```

```
    "ParameterGroup": {
        "Name": "myredis6x",
        "Family": "memorydb_redis6",
        "Description": "my-parameter-group",
        "ARN": "arn:aws:memorydb:us-east-1:49165xxxxxx:parametergroup/myredis6x"
    }
}
```

Per ulteriori informazioni, consulta Creazione di un gruppo di parametri nella Guida per l'utente di MemoryDB.

- Per i dettagli sull'API, consulta CreateParameterGroup AWS CLICommand Reference.

`create-snapshot`

Il seguente esempio di codice mostra come utilizzare`create-snapshot`.

AWS CLI

Come creare uno snapshot

L'esempio `create-snapshot` seguente crea uno snapshot.

```
aws memorydb create-snapshot \
    --cluster-name my-cluster \
    --snapshot-name my-cluster-snapshot
```

Output:

```
{
    "Snapshot": {
        "Name": "my-cluster-snapshot1",
        "Status": "creating",
        "Source": "manual",
        "ARN": "arn:aws:memorydb:us-east-1:49165xxxxxx:snapshot/my-cluster-
snapshot",
        "ClusterConfiguration": {
            "Name": "my-cluster",
            "Description": "",
            "NodeType": "db.r6g.large",
            "EngineVersion": "6.2",
            "MaintenanceWindow": "wed:03:00-wed:04:00",
```

```
            "Port": 6379,
            "ParameterGroupName": "default.memorydb-redis6",
            "SubnetGroupName": "my-sg",
            "VpcId": "vpc-862xxxxc",
            "SnapshotRetentionLimit": 0,
            "SnapshotWindow": "04:30-05:30",
            "NumShards": 2
        }
    }
}
```

Per ulteriori informazioni, consulta Creazione manuale di snapshot nella Guida per l'utente di MemoryDB.

- Per i dettagli sull'API, consulta CreateSnapshot AWS CLICommand Reference.

`create-subnet-group`

Il seguente esempio di codice mostra come utilizzare`create-subnet-group`.

AWS CLI

Come creare un gruppo di sottoreti

L'esempio `create-subnet-group` seguente crea un gruppo di sottoreti.

```
aws memorydb create-subnet-group \
    --subnet-group-name mysubnetgroup \
    --description "my subnet group" \
    --subnet-ids subnet-5623xxxx
```

Output:

```
{
    "SubnetGroup": {
        "Name": "mysubnetgroup",
        "Description": "my subnet group",
        "VpcId": "vpc-86257xxx",
        "Subnets": [
            {
                "Identifier": "subnet-5623xxxx",
                "AvailabilityZone": {
```

```
                    "Name": "us-east-1a"
                }
            }
        ],
        "ARN": "arn:aws:memorydb:us-east-1:491658xxxxxx:subnetgroup/mysubnetgroup"
    }
}
```

Per ulteriori informazioni, consulta Creazione di un gruppo di sottoreti nella Guida per l'utente di MemoryDB.

- Per i dettagli sull'API, consulta CreateSubnetGroup AWS CLICommand Reference.

`create-user`

Il seguente esempio di codice mostra come utilizzare`create-user`.

AWS CLI

Come creare un utente

L'esempio `create-user` seguente crea un'attività di replica.

```
aws memorydb create-user \
    --user-name user-name-1 \
    --access-string "~objects:* ~items:* ~public:*" \
     --authentication-mode \
         Passwords="enterapasswordhere",Type=password
```

Output:

```
{
    "User": {
        "Name": "user-name-1",
        "Status": "active",
        "AccessString": "off ~objects:* ~items:* ~public:* resetchannels -@all",
        "ACLNames": [],
        "MinimumEngineVersion": "6.2",
        "Authentication": {
            "Type": "password",
            "PasswordCount": 1
        },
```

```
        "ARN": "arn:aws:memorydb:us-west-2:491658xxxxxx:user/user-name-1"
    }
}
```

Per ulteriori informazioni, consulta Autenticazione degli utenti con le liste di controllo accessi nella Guida per l'utente di MemoryDB.

- Per i dettagli sull'API, consulta CreateUser AWS CLICommand Reference.

`delete-acl`

Il seguente esempio di codice mostra come utilizzare`delete-acl`.

AWS CLI

Come eliminare una lista di controllo degli accessi (ACL)

L'esempio `delete-acl` seguente elimina una lista di controllo degli accessi (ACL).

```
aws memorydb delete-acl \
    --acl-name "new-acl-1"
```

Output:

```
{
    "ACL": {
        "Name": "new-acl-1",
        "Status": "deleting",
        "UserNames": [
            "pat"
        ],
        "MinimumEngineVersion": "6.2",
        "Clusters": [],
        "ARN": "arn:aws:memorydb:us-east-1:491658xxxxxx:acl/new-acl-1"
    }
}
```

Per ulteriori informazioni, consulta Autenticazione degli utenti con le liste di controllo accessi nella Guida per l'utente di MemoryDB.

- Per i dettagli sull'API, consulta DeleteAcl AWS CLICommand Reference.

`delete-cluster`

Il seguente esempio di codice mostra come utilizzare`delete-cluster`.

AWS CLI

Come eliminare un cluster

L'esempio `delete-cluster` seguente elimina un cluster.

```
aws memorydb delete-cluster \
    --cluster-name my-new-cluster
```

Output:

```
{
    "Cluster": {
        "Name": "my-new-cluster",
        "Status": "deleting",
        "NumberOfShards": 1,
        "ClusterEndpoint": {
            "Address": "clustercfg.my-new-cluster.xxxxx.memorydb.us-
east-1.amazonaws.com",
            "Port": 6379
        },
        "NodeType": "db.r6g.large",
        "EngineVersion": "6.2",
        "EnginePatchVersion": "6.2.6",
        "ParameterGroupName": "default.memorydb-redis6",
        "ParameterGroupStatus": "in-sync",
        "SubnetGroupName": "my-sg",
        "TLSEnabled": true,
        "ARN": "arn:aws:memorydb:us-east-1:491658xxxxxx:cluster/my-new-cluster",
        "SnapshotRetentionLimit": 0,
        "MaintenanceWindow": "sat:10:00-sat:11:00",
        "SnapshotWindow": "07:30-08:30",
        "AutoMinorVersionUpgrade": true
    }
}
```

Per ulteriori informazioni, consulta Eliminazione di un cluster nella Guida per l'utente di MemoryDB.

- Per i dettagli sull'API, consulta DeleteCluster AWS CLICommand Reference.

`delete-parameter-group`

Il seguente esempio di codice mostra come utilizzare`delete-parameter-group`.

AWS CLI

Come eliminare un gruppo di parametri

L'esempio `delete-parameter-group` seguente elimina un gruppo di parametri.

```
aws memorydb delete-parameter-group \
    --parameter-group-name myRedis6x
```

Output:

```
{
    "ParameterGroup": {
        "Name": "myredis6x",
        "Family": "memorydb_redis6",
        "Description": "my-parameter-group",
        "ARN": "arn:aws:memorydb:us-east-1:491658xxxxxx:parametergroup/myredis6x"
    }
}
```

Per ulteriori informazioni, consulta Eliminazione di un gruppo di parametri nella Guida per l'utente di MemoryDB.

- Per i dettagli sull'API, consulta DeleteParameterGroup AWS CLICommand Reference.

`delete-snapshot`

Il seguente esempio di codice mostra come utilizzare`delete-snapshot`.

AWS CLI

Come eliminare uno snapshot

L'esempio `delete-snapshot` seguente elimina uno snapshot.

```
aws memorydb delete-snapshot \
    --snapshot-name my-cluster-snapshot
```

Output:

```
{
    "Snapshot": {
        "Name": "my-cluster-snapshot",
        "Status": "deleting",
        "Source": "manual",
        "ARN": "arn:aws:memorydb:us-east-1:49165xxxxxx:snapshot/my-cluster-
snapshot",
        "ClusterConfiguration": {
            "Name": "my-cluster",
            "Description": "",
            "NodeType": "db.r6g.large",
            "EngineVersion": "6.2",
            "MaintenanceWindow": "wed:03:00-wed:04:00",
            "Port": 6379,
            "ParameterGroupName": "default.memorydb-redis6",
            "SubnetGroupName": "my-sg",
            "VpcId": "vpc-862xxxxc",
            "SnapshotRetentionLimit": 0,
            "SnapshotWindow": "04:30-05:30",
            "NumShards": 2
        }
    }
}
```

Per ulteriori informazioni, consulta Eliminazione di uno snapshot nella Guida per l'utente di MemoryDB.

- Per i dettagli sull'API, consulta DeleteSnapshot AWS CLICommand Reference.

`delete-subnet-group`

Il seguente esempio di codice mostra come utilizzare`delete-subnet-group`.

AWS CLI

Come eliminare un gruppo di sottoreti

L'esempio `delete-subnet-group` seguente elimina una sottorete.

```
aws memorydb delete-subnet-group \
    --subnet-group-name mysubnetgroup
```

Output:

```
{
    "SubnetGroup": {
        "Name": "mysubnetgroup",
        "Description": "my subnet group",
        "VpcId": "vpc-86xxxx4fc",
        "Subnets": [
            {
                "Identifier": "subnet-56xxx61b",
                "AvailabilityZone": {
                    "Name": "us-east-1a"
                }
            }
        ],
        "ARN": "arn:aws:memorydb:us-east-1:491658xxxxxx:subnetgroup/mysubnetgroup"
    }
}
```

Per ulteriori informazioni, consulta Eliminazione di un gruppo di sottoreti nella Guida per l'utente di MemoryDB.

- Per i dettagli sull'API, consulta DeleteSubnetGroup AWS CLICommand Reference.

`delete-user`

Il seguente esempio di codice mostra come utilizzare`delete-user`.

AWS CLI

Come eliminare un utente

L'esempio `delete-user` seguente elimina un sink.

```
aws memorydb delete-user \
    --user-name my-user
```

Output:

```
{
    "User": {
        "Name": "my-user",
        "Status": "deleting",
```

```
        "AccessString": "on ~app::* resetchannels -@all +@read",
        "ACLNames": [
            "my-acl"
        ],
        "MinimumEngineVersion": "6.2",
        "Authentication": {
            "Type": "password",
            "PasswordCount": 1
        },
        "ARN": "arn:aws:memorydb:us-east-1:491658xxxxxx:user/my-user"
    }
}
```

Per ulteriori informazioni, consulta Autenticazione degli utenti con le liste di controllo accessi nella Guida per l'utente di MemoryDB.

- Per i dettagli sull'API, consulta DeleteUser AWS CLICommand Reference.

`describe-acls`

Il seguente esempio di codice mostra come utilizzare`describe-acls`.

AWS CLI

Per restituire un elenco di ACLs

Il seguente describe-acls` restituisce un elenco di. ACLs

```
aws memorydb describe-acls
```

Output:

```
{
    "ACLs": [
        {
            "Name": "open-access",
            "Status": "active",
            "UserNames": [
                "default"
            ],
            "MinimumEngineVersion": "6.2",
            "Clusters": [],
            "ARN": "arn:aws:memorydb:us-east-1:491658xxxxxx:acl/open-access"
```

```
        },
        {
            "Name": my-acl",
            "Status": "active",
            "UserNames": [],
            "MinimumEngineVersion": "6.2",
            "Clusters": [
                "my-cluster"
            ],
            "ARN": "arn:aws:memorydb:us-east-1:49165xxxxxxx:acl/my-acl"
        }
    ]
}
```

Per ulteriori informazioni, consulta Autenticazione degli utenti con le liste di controllo accessi nella Guida per l'utente di MemoryDB.

- Per i dettagli sull'API, vedere in Command Reference. DescribeAclsAWS CLI

`describe-clusters`

Il seguente esempio di codice mostra come utilizzare`describe-clusters`.

AWS CLI

Come restituire un elenco di cluster

L'esempio describe-clusters seguente restituisce un elenco di cluster.

```
aws memorydb describe-clusters
```

Output:

```
{
    "Clusters": [
        {
                "Name": "my-cluster",
                "Status": "available",
                "NumberOfShards": 2,
                "ClusterEndpoint": {
                    "Address": "clustercfg.my-cluster.llru6f.memorydb.us-
east-1.amazonaws.com",
                    "Port": 6379
```

```
                },
                "NodeType": "db.r6g.large",
                "EngineVersion": "6.2",
                "EnginePatchVersion": "6.2.6",
                "ParameterGroupName": "default.memorydb-redis6",
                "ParameterGroupStatus": "in-sync",
                "SecurityGroups": [
                    {
                        "SecurityGroupId": "sg-0a1434xxxxxc9fae",
                        "Status": "active"
                    }
                ],
                "SubnetGroupName": "pat-sg",
                "TLSEnabled": true,
                "ARN": "arn:aws:memorydb:us-east-1:49165xxxxxx:cluster/my-cluster",
                "SnapshotRetentionLimit": 0,
                "MaintenanceWindow": "wed:03:00-wed:04:00",
                "SnapshotWindow": "04:30-05:30",
                "ACLName": "my-acl",
                "AutoMinorVersionUpgrade": true
        }
    ]
}
```

Per ulteriori informazioni, consulta Gestione dei cluster nella Guida per l'utente di MemoryDB.

- Per i dettagli sull'API, consulta DescribeClusters AWS CLICommand Reference.

`describe-engine-versions`

Il seguente esempio di codice mostra come utilizzare`describe-engine-versions`.

AWS CLI

Come restituire un elenco di versioni del motore

Il seguente describe-engine-versions `restituisce un elenco di versioni del motore.

```
aws memorydb describe-engine-versions
```

Output:

```
{
```

```
    "EngineVersions": [
        {
            "EngineVersion": "6.2",
            "EnginePatchVersion": "6.2.6",
            "ParameterGroupFamily": "memorydb_redis6"
        }
    ]
}
```

Per ulteriori informazioni, consulta Versioni e aggiornamenti del motore nella Guida per l'utente di MemoryDB.

- Per i dettagli sull'API, consulta DescribeEngineVersions AWS CLICommand Reference.

`describe-events`

Il seguente esempio di codice mostra come utilizzare`describe-events`.

AWS CLI

Come restituire un elenco di eventi

L'esempio describe-events seguente restituisce un elenco di eventi.

```
aws memorydb describe-events
```

Output:

```
{
    "Events": [
        {
            "SourceName": "my-cluster",
            "SourceType": "cluster",
            "Message": "Increase replica count started for replication group my-
cluster on 2022-07-22T14:09:01.440Z",
            "Date": "2022-07-22T07:09:01.443000-07:00"
        },
        {
            "SourceName": "my-user",
            "SourceType": "user",
            "Message": "Create user my-user operation completed.",
            "Date": "2022-07-22T07:00:02.975000-07:00"
        }
```

```
    ]
}
```

Per ulteriori informazioni, consulta Monitoraggio degli eventi nella Guida per l'utente di MemoryDB.

- Per i dettagli sull'API, consulta DescribeEvents AWS CLICommand Reference.

`describe-parameter-groups`

Il seguente esempio di codice mostra come utilizzare`describe-parameter-groups`.

AWS CLI

Come restituire un elenco di gruppi di parametri

Il seguente describe-parameter-groups `restituisce un elenco di gruppi di parametri.

```
aws memorydb describe-parameter-groups
```

Output:

```
{
    "ParameterGroups": [
        {
            "Name": "default.memorydb-redis6",
            "Family": "memorydb_redis6",
            "Description": "Default parameter group for memorydb_redis6",
            "ARN": "arn:aws:memorydb:us-east-1:491658xxxxxx:parametergroup/
default.memorydb-redis6"
        }
    ]
}
```

Per ulteriori informazioni, consulta Configurazione dei parametri di motore con i gruppi di parametri nella Guida per l'utente di MemoryDB.

- Per i dettagli sull'API, vedere DescribeParameterGroupsin AWS CLI Command Reference.

`describe-parameters`

Il seguente esempio di codice mostra come utilizzare`describe-parameters`.

AWS CLI

Come restituire un elenco di parametri

L'esempio describe-parameters seguente restituisce un elenco di parametri.

```
aws memorydb describe-parameters
```

Output:

```
{
    "Parameters": [
        {
            "Name": "acllog-max-len",
            "Value": "128",
            "Description": "The maximum length of the ACL Log",
            "DataType": "integer",
            "AllowedValues": "1-10000",
            "MinimumEngineVersion": "6.2.4"
        },
        {
            "Name": "activedefrag",
            "Value": "no",
            "Description": "Enabled active memory defragmentation",
            "DataType": "string",
            "AllowedValues": "yes,no",
            "MinimumEngineVersion": "6.2.4"
        },
        {
            "Name": "active-defrag-cycle-max",
            "Value": "75",
            "Description": "Maximal effort for defrag in CPU percentage",
            "DataType": "integer",
            "AllowedValues": "1-75",
            "MinimumEngineVersion": "6.2.4"
        },
        {
            "Name": "active-defrag-cycle-min",
            "Value": "5",
            "Description": "Minimal effort for defrag in CPU percentage",
            "DataType": "integer",
            "AllowedValues": "1-75",
            "MinimumEngineVersion": "6.2.4"
```

```
        },
        {
            "Name": "active-defrag-ignore-bytes",
            "Value": "104857600",
            "Description": "Minimum amount of fragmentation waste to start active
defrag",
            "DataType": "integer",
            "AllowedValues": "1048576-",
            "MinimumEngineVersion": "6.2.4"
        },
        {
            "Name": "active-defrag-max-scan-fields",
            "Value": "1000",
            "Description": "Maximum number of set/hash/zset/list fields that will be
processed from the main dictionary scan",
            "DataType": "integer",
            "AllowedValues": "1-1000000",
            "MinimumEngineVersion": "6.2.4"
        },
        {
            "Name": "active-defrag-threshold-lower",
            "Value": "10",
            "Description": "Minimum percentage of fragmentation to start active
defrag",
            "DataType": "integer",
            "AllowedValues": "1-100",
            "MinimumEngineVersion": "6.2.4"
        },
        {
            "Name": "active-defrag-threshold-upper",
            "Value": "100",
            "Description": "Maximum percentage of fragmentation at which we use
maximum effort",
            "DataType": "integer",
            "AllowedValues": "1-100",
            "MinimumEngineVersion": "6.2.4"
        },
        {
            "Name": "active-expire-effort",
            "Value": "1",
            "Description": "The amount of effort that redis uses to expire items in
the active expiration job",
            "DataType": "integer",
            "AllowedValues": "1-10",
```

```
            "MinimumEngineVersion": "6.2.4"
        },
        {
            "Name": "activerehashing",
            "Value": "yes",
            "Description": "Apply rehashing or not",
            "DataType": "string",
            "AllowedValues": "yes,no",
            "MinimumEngineVersion": "6.2.4"
        },
        {
            "Name": "client-output-buffer-limit-normal-hard-limit",
            "Value": "0",
            "Description": "Normal client output buffer hard limit in bytes",
            "DataType": "integer",
            "AllowedValues": "0-",
            "MinimumEngineVersion": "6.2.4"
        },
        {
            "Name": "client-output-buffer-limit-normal-soft-limit",
            "Value": "0",
            "Description": "Normal client output buffer soft limit in bytes",
            "DataType": "integer",
            "AllowedValues": "0-",
            "MinimumEngineVersion": "6.2.4"
        },
        {
            "Name": "client-output-buffer-limit-normal-soft-seconds",
            "Value": "0",
            "Description": "Normal client output buffer soft limit in seconds",
            "DataType": "integer",
            "AllowedValues": "0-",
            "MinimumEngineVersion": "6.2.4"
        },
        {
            "Name": "client-output-buffer-limit-pubsub-hard-limit",
            "Value": "33554432",
            "Description": "Pubsub client output buffer hard limit in bytes",
            "DataType": "integer",
            "AllowedValues": "0-",
            "MinimumEngineVersion": "6.2.4"
        },
        {
            "Name": "client-output-buffer-limit-pubsub-soft-limit",
```

```
            "Value": "8388608",
            "Description": "Pubsub client output buffer soft limit in bytes",
            "DataType": "integer",
            "AllowedValues": "0-",
            "MinimumEngineVersion": "6.2.4"
        },
        {
            "Name": "client-output-buffer-limit-pubsub-soft-seconds",
            "Value": "60",
            "Description": "Pubsub client output buffer soft limit in seconds",
            "DataType": "integer",
            "AllowedValues": "0-",
            "MinimumEngineVersion": "6.2.4"
        },
        {
            "Name": "hash-max-ziplist-entries",
            "Value": "512",
            "Description": "The maximum number of hash entries in order for the
dataset to be compressed",
            "DataType": "integer",
            "AllowedValues": "0-",
            "MinimumEngineVersion": "6.2.4"
        },
        {
            "Name": "hash-max-ziplist-value",
            "Value": "64",
            "Description": "The threshold of biggest hash entries in order for the
dataset to be compressed",
            "DataType": "integer",
            "AllowedValues": "0-",
            "MinimumEngineVersion": "6.2.4"
        },
        {
            "Name": "hll-sparse-max-bytes",
            "Value": "3000",
            "Description": "HyperLogLog sparse representation bytes limit",
            "DataType": "integer",
            "AllowedValues": "1-16000",
            "MinimumEngineVersion": "6.2.4"
        },
        {
            "Name": "lazyfree-lazy-eviction",
            "Value": "no",
            "Description": "Perform an asynchronous delete on evictions",
```

```
            "DataType": "string",
            "AllowedValues": "yes,no",
            "MinimumEngineVersion": "6.2.4"
        },
        {
            "Name": "lazyfree-lazy-expire",
            "Value": "no",
            "Description": "Perform an asynchronous delete on expired keys",
            "DataType": "string",
            "AllowedValues": "yes,no",
            "MinimumEngineVersion": "6.2.4"
        },
        {
            "Name": "lazyfree-lazy-server-del",
            "Value": "no",
            "Description": "Perform an asynchronous delete on key updates",
            "DataType": "string",
            "AllowedValues": "yes,no",
            "MinimumEngineVersion": "6.2.4"
        },
        {
            "Name": "lazyfree-lazy-user-del",
            "Value": "no",
            "Description": "Specifies whether the default behavior of DEL command
acts the same as UNLINK",
            "DataType": "string",
            "AllowedValues": "yes,no",
            "MinimumEngineVersion": "6.2.4"
        },
        {
            "Name": "lfu-decay-time",
            "Value": "1",
            "Description": "The amount of time in minutes to decrement the key
counter for LFU eviction policyd",
            "DataType": "integer",
            "AllowedValues": "0-",
            "MinimumEngineVersion": "6.2.4"
        },
        {
            "Name": "lfu-log-factor",
            "Value": "10",
            "Description": "The log factor for incrementing key counter for LFU
eviction policy",
            "DataType": "integer",
```

```
            "AllowedValues": "1-",
            "MinimumEngineVersion": "6.2.4"
        },
        {
            "Name": "list-compress-depth",
            "Value": "0",
            "Description": "Number of quicklist ziplist nodes from each side of
 the list to exclude from compression. The head and tail of the list are always
 uncompressed for fast push/pop operations",
            "DataType": "integer",
            "AllowedValues": "0-",
            "MinimumEngineVersion": "6.2.4"
        },
        {
            "Name": "maxmemory-policy",
            "Value": "noeviction",
            "Description": "Max memory policy",
            "DataType": "string",
            "AllowedValues": "volatile-lru,allkeys-lru,volatile-lfu,allkeys-
lfu,volatile-random,allkeys-random,volatile-ttl,noeviction",
            "MinimumEngineVersion": "6.2.4"
        },
        {
            "Name": "maxmemory-samples",
            "Value": "3",
            "Description": "Max memory samples",
            "DataType": "integer",
            "AllowedValues": "1-",
            "MinimumEngineVersion": "6.2.4"
        },
        {
            "Name": "notify-keyspace-events",
            "Description": "The keyspace events for Redis to notify Pub/Sub clients
 about. By default all notifications are disabled",
            "DataType": "string",
            "MinimumEngineVersion": "6.2.4"
        },
        {
            "Name": "set-max-intset-entries",
            "Value": "512",
            "Description": "The limit in the size of the set in order for the
 dataset to be compressed",
            "DataType": "integer",
            "AllowedValues": "0-",
```

```
            "MinimumEngineVersion": "6.2.4"
        },
        {
            "Name": "slowlog-log-slower-than",
            "Value": "10000",
            "Description": "The execution time, in microseconds, to exceed in order
for the command to get logged. Note that a negative number disables the slow log,
while a value of zero forces the logging of every command",
            "DataType": "integer",
            "AllowedValues": "-",
            "MinimumEngineVersion": "6.2.4"
        },
        {
            "Name": "slowlog-max-len",
            "Value": "128",
            "Description": "The length of the slow log. There is no limit to this
length. Just be aware that it will consume memory. You can reclaim memory used by
the slow log with SLOWLOG RESET.",
            "DataType": "integer",
            "AllowedValues": "0-",
            "MinimumEngineVersion": "6.2.4"
        },
        {
            "Name": "stream-node-max-bytes",
            "Value": "4096",
            "Description": "The maximum size of a single node in a stream in bytes",
            "DataType": "integer",
            "AllowedValues": "0-",
            "MinimumEngineVersion": "6.2.4"
        },
        {
            "Name": "stream-node-max-entries",
            "Value": "100",
            "Description": "The maximum number of items a single node in a stream
can contain",
            "DataType": "integer",
            "AllowedValues": "0-",
            "MinimumEngineVersion": "6.2.4"
        },
        {
            "Name": "tcp-keepalive",
            "Value": "300",
            "Description": "If non-zero, send ACKs every given number of seconds",
            "DataType": "integer",
```

```
            "AllowedValues": "0-",
            "MinimumEngineVersion": "6.2.4"
        },
        {
            "Name": "timeout",
            "Value": "0",
            "Description": "Close connection if client is idle for a given number of
 seconds, or never if 0",
            "DataType": "integer",
            "AllowedValues": "0,20-",
            "MinimumEngineVersion": "6.2.4"
        },
        {
            "Name": "tracking-table-max-keys",
            "Value": "1000000",
            "Description": "The maximum number of keys allowed for the tracking
 table for client side caching",
            "DataType": "integer",
            "AllowedValues": "1-100000000",
            "MinimumEngineVersion": "6.2.4"
        },
        {
            "Name": "zset-max-ziplist-entries",
            "Value": "128",
            "Description": "The maximum number of sorted set entries in order for
 the dataset to be compressed",
            "DataType": "integer",
            "AllowedValues": "0-",
            "MinimumEngineVersion": "6.2.4"
        },
        {
            "Name": "zset-max-ziplist-value",
            "Value": "64",
            "Description": "The threshold of biggest sorted set entries in order for
 the dataset to be compressed",
            "DataType": "integer",
            "AllowedValues": "0-",
            "MinimumEngineVersion": "6.2.4"
        }
    ]
}
```

Per ulteriori informazioni, consulta Configurazione dei parametri di motore con i gruppi di parametri nella Guida per l'utente di MemoryDB.

- Per i dettagli sull'API, consulta DescribeParameters AWS CLICommand Reference.

`describe-snapshots`

Il seguente esempio di codice mostra come utilizzare`describe-snapshots`.

AWS CLI

Come restituire un elenco di snapshot

L'esempio comando describe-snapshots seguente restituisce un elenco di istantanee.

```
aws memorydb describe-snapshots
```

Output:

```
{
"Snapshots": [
    {
        "Name": "my-cluster-snapshot",
        "Status": "available",
        "Source": "manual",
        "ARN": "arn:aws:memorydb:us-east-1:491658xxxxxx2:snapshot/my-cluster-
snapshot",
        "ClusterConfiguration": {
            "Name": "my-cluster",
            "Description": " ",
            "NodeType": "db.r6g.large",
            "EngineVersion": "6.2",
            "MaintenanceWindow": "wed:03:00-wed:04:00",
            "Port": 6379,
            "ParameterGroupName": "default.memorydb-redis6",
            "SubnetGroupName": "my-sg",
            "VpcId": "vpc-862574fc",
            "SnapshotRetentionLimit": 0,
            "SnapshotWindow": "04:30-05:30",
            "NumShards": 2
        }
    }
}
```

Per ulteriori informazioni, consulta Snapshot e ripristino nella Guida per l'utente di MemoryDB.

- Per i dettagli sull'API, consulta DescribeSnapshots AWS CLICommand Reference.

`describe-subnet-groups`

Il seguente esempio di codice mostra come utilizzare`describe-subnet-groups`.

AWS CLI

Come restituire un elenco di gruppi di sottoreti

Il seguente describe-subnet-groups `restituisce un elenco di gruppi di sottoreti.

```
aws memorydb describe-subnet-groups
```

Output

```
{
    "SubnetGroups": [
        {
            "Name": "my-sg",
            "Description": "pat-sg",
            "VpcId": "vpc-86xxx4fc",
            "Subnets": [
                {
                    "Identifier": "subnet-faxx84a6",
                    "AvailabilityZone": {
                        "Name": "us-east-1b"
                    }
                },
                {
                    "Identifier": "subnet-56xxf61b",
                    "AvailabilityZone": {
                        "Name": "us-east-1a"
                    }
                }
            ],
            "ARN": "arn:aws:memorydb:us-east-1:49165xxxxxx:subnetgroup/my-sg"
        }
    ]
}
```

Per ulteriori informazioni, consulta Sottoreti e gruppi di sottoreti nella Guida per l'utente di MemoryDB.

- Per i dettagli sull'API, vedere DescribeSubnetGroupsin AWS CLI Command Reference.

`describe-users`

Il seguente esempio di codice mostra come utilizzare`describe-users`.

AWS CLI

Come restituire un elenco di utenti

L'esempio describe-users seguente restituisce un elenco di utenti.

```
aws memorydb describe-users
```

Output

```
{
    "Users": [
        {
            "Name": "default",
            "Status": "active",
            "AccessString": "on ~* &* +@all",
            "ACLNames": [
                "open-access"
            ],
            "MinimumEngineVersion": "6.0",
            "Authentication": {
                "Type": "no-password"
            },
            "ARN": "arn:aws:memorydb:us-east-1:491658xxxxxx:user/default"
        },
        {
            "Name": "my-user",
            "Status": "active",
            "AccessString": "off ~objects:* ~items:* ~public:* resetchannels -@all",
            "ACLNames": [],
            "MinimumEngineVersion": "6.2",
            "Authentication": {
                "Type": "password",
                "PasswordCount": 2
```

```
            },
            "ARN": "arn:aws:memorydb:us-east-1:491658xxxxxx:user/my-user"
        }
    ]
}
```

Per ulteriori informazioni, consulta Autenticazione degli utenti con le liste di controllo accessi nella Guida per l'utente di MemoryDB.

- Per i dettagli sull'API, consulta DescribeUsers AWS CLICommand Reference.

`failover-shard`

Il seguente esempio di codice mostra come utilizzare`failover-shard`.

AWS CLI

Come eseguire il failover di uno shard

Il comando failover-shard seguente esegue il failover di uno shard.

```
aws memorydb failover-shard \
    --cluster-name my-cluster --shard-name 0001
```

Output:

```
{
    "Cluster": {
        "Name": "my-cluster",
        "Status": "available",
        "NumberOfShards": 2,
        "ClusterEndpoint": {
            "Address": "clustercfg.my-cluster.xxxxxx.memorydb.us-
east-1.amazonaws.com",
            "Port": 6379
        },
        "NodeType": "db.r6g.large",
        "EngineVersion": "6.2",
        "EnginePatchVersion": "6.2.6",
        "ParameterGroupName": "default.memorydb-redis6",
        "ParameterGroupStatus": "in-sync",
        "SecurityGroups": [
            {
```

```
                "SecurityGroupId": "sg-0a143xxxx45c9fae",
                "Status": "active"
            }
        ],
        "SubnetGroupName": "my-sg",
        "TLSEnabled": true,
        "ARN": "arn:aws:memorydb:us-east-1:491658xxxxxx:cluster/my-cluster",
        "SnapshotRetentionLimit": 0,
        "MaintenanceWindow": "wed:03:00-wed:04:00",
        "SnapshotWindow": "04:30-05:30",
        "AutoMinorVersionUpgrade": true
    }
}
```

Per ulteriori informazioni, consulta Ridurre al minimo i tempi di inattività con la funzione Multi-AZ nella Guida per l'utente di MemoryDB.

- Per i dettagli sull'API, consulta FailoverShard AWS CLICommand Reference.

`list-allowed-node-type-updates`

Il seguente esempio di codice mostra come utilizzare`list-allowed-node-type-updates`.

AWS CLI

Come restituire un elenco di aggiornamenti dei tipi di nodo consentiti

Il seguente list-allowed-node-type -updates restituisce un elenco di aggiornamenti disponibili per i tipi di nodo.

```
aws memorydb list-allowed-node-type-updates
```

Output:

```
{
    "Cluster": {
        "Name": "my-cluster",
        "Status": "available",
        "NumberOfShards": 2,
        "ClusterEndpoint": {
            "Address": "clustercfg.my-cluster.xxxxxx.memorydb.us-
east-1.amazonaws.com",
```

```
            "Port": 6379
        },
        "NodeType": "db.r6g.large",
        "EngineVersion": "6.2",
        "EnginePatchVersion": "6.2.6",
        "ParameterGroupName": "default.memorydb-redis6",
        "ParameterGroupStatus": "in-sync",
        "SecurityGroups": [
            {
                "SecurityGroupId": "sg-0a143xxxx45c9fae",
                "Status": "active"
            }
        ],
        "SubnetGroupName": "my-sg",
        "TLSEnabled": true,
        "ARN": "arn:aws:memorydb:us-east-1:491658xxxxxx:cluster/my-cluster",
        "SnapshotRetentionLimit": 0,
        "MaintenanceWindow": "wed:03:00-wed:04:00",
        "SnapshotWindow": "04:30-05:30",
        "AutoMinorVersionUpgrade": true
    }
}
```

Per ulteriori informazioni, consulta Dimensionamento nella Guida per l'utente di MemoryDB.

- Per i dettagli sull'API, vedere ListAllowedNodeTypeUpdatesin AWS CLI Command Reference.

`list-tags`

Il seguente esempio di codice mostra come utilizzare`list-tags`.

AWS CLI

Come restituire un elenco di tag

Il comando list-tags seguente restituisce un elenco di tag.

```
aws memorydb list-tags \
    --resource-arn arn:aws:memorydb:us-east-1:491658xxxxxx:cluster/my-cluster
```

Output:

```
{
```

```
    "TagList": [
        {
            "Key": "mytag",
            "Value": "myvalue"
        }
    ]
}
```

Per ulteriori informazioni, consulta Tagging di risorse nella Guida per l'utente di MemoryDB.

- Per i dettagli sull'API, consulta ListTags AWS CLICommand Reference.

`reset-parameter-group`

Il seguente esempio di codice mostra come utilizzare`reset-parameter-group`.

AWS CLI

Come reimpostare un gruppo di parametri

Il seguente reset-parameter-group `reimposta un gruppo di parametri.

```
aws memorydb reset-parameter-group \
    --parameter-group-name my-parameter-group \
    --all-parameters
```

Output:

```
{
    "ParameterGroup": {
        "Name": "my-parameter-group",
        "Family": "memorydb_redis6",
        "Description": "my parameter group",
        "ARN": "arn:aws:memorydb:us-east-1:491658xxxxxx:parametergroup/my-parameter-
group"
    }
}
```

Per ulteriori informazioni, consulta Configurazione dei parametri di motore con i gruppi di parametri nella Guida per l'utente di MemoryDB.

- Per i dettagli sull'API, vedere ResetParameterGroupin AWS CLI Command Reference.

`tag-resource`

Il seguente esempio di codice mostra come utilizzare`tag-resource`.

AWS CLI

Come taggare a una risorsa

Il comando tag-resource seguente aggiunge un tag a una risorsa.

```
aws memorydb tag-resource \
    --resource-arn arn:aws:memorydb:us-east-1:491658xxxxxx:cluster/my-cluster \
    --tags Key="mykey",Value="myvalue"
```

Output:

```
{
    "TagList": [
        {
            "Key": "mytag",
            "Value": "myvalue"
        },
        {
            "Key": "mykey",
            "Value": "myvalue"
        }
    ]
}
```

Per ulteriori informazioni, consulta Tagging di risorse nella Guida per l'utente di MemoryDB.

- Per i dettagli sull'API, consulta TagResource AWS CLICommand Reference.

`untag-resource`

Il seguente esempio di codice mostra come utilizzare`untag-resource`.

AWS CLI

Come aggiornare una lista di controllo degli accessi (ACL)

Il comando update-acl seguente aggiorna un ACL aggiungendo un utente.

```
aws memorydb untag-resource \
    --resource-arn arn:aws:memorydb:us-east-1:491658xxxxx:cluster/my-cluster \
    --tag-keys mykey
```

Output:

```
{
    "TagList": [
        {
            "Key": "mytag",
            "Value": "myvalue"
        }
    ]
}
```

Per ulteriori informazioni, consulta Tagging di risorse nella Guida per l'utente di MemoryDB.

- Per i dettagli sull'API, consulta UntagResource AWS CLICommand Reference.

`update-cluster`

Il seguente esempio di codice mostra come utilizzare`update-cluster`.

AWS CLI

Come aggiornare un cluster

Il seguente update-cluster`` aggiorna il gruppo di parametri di un cluster a. my-parameter-group

```
aws memorydb update-cluster \
    --cluster-name my-cluster \
    --parameter-group-name my-parameter-group
```

Output:

```
{
    "Cluster": {
        "Name": "my-cluster",
        "Status": "available",
        "NumberOfShards": 2,
        "AvailabilityMode": "MultiAZ",
        "ClusterEndpoint": {
```

```
            "Address": "clustercfg.my-cluster.llru6f.memorydb.us-
east-1.amazonaws.com",
            "Port": 6379
        },
        "NodeType": "db.r6g.large",
        "EngineVersion": "6.2",
        "EnginePatchVersion": "6.2.6",
        "ParameterGroupName": "my-parameter-group",
        "ParameterGroupStatus": "in-sync",
        "SecurityGroups": [
            {
                "SecurityGroupId": "sg-0a143xxxxxc9fae",
                "Status": "active"
            }
        ],
        "SubnetGroupName": "pat-sg",
        "TLSEnabled": true,
        "ARN": "arn:aws:memorydb:us-east-1:491658xxxxxx:cluster/my-cluster",
        "SnapshotRetentionLimit": 0,
        "MaintenanceWindow": "wed:03:00-wed:04:00",
        "SnapshotWindow": "04:30-05:30",
        "ACLName": "my-acl",
        "AutoMinorVersionUpgrade": true
    }
}
```

Per ulteriori informazioni, consulta Modifica di un cluster nella Guida per l'utente di MemoryDB.

- Per i dettagli sull'API, consulta Command Reference. UpdateClusterAWS CLI

`update-parameter-group`

Il seguente esempio di codice mostra come utilizzareu`pdate-parameter-group`.

AWS CLI

Come aggiornare un gruppo di parametri

Il seguente update-parameter-group ``aggiorna un gruppo di parametri.

```
aws memorydb update-parameter-group \
    --parameter-group-name my-parameter-group \
    --parameter-name-values "ParameterName=activedefrag, ParameterValue=no"
```

Output:

```
{
    "ParameterGroup": {
        "Name": "my-parameter-group",
        "Family": "memorydb_redis6",
        "Description": "my parameter group",
        "ARN": "arn:aws:memorydb:us-east-1:49165xxxxxx:parametergroup/my-parameter-
group"
    }
}
```

Per ulteriori informazioni, consulta Modifica di un gruppo di parametri nella Guida per l'utente di MemoryDB.

- Per i dettagli sull'API, consulta UpdateParameterGroup AWS CLICommand Reference.

`update-subnet-group`

Il seguente esempio di codice mostra come utilizzare`update-subnet-group`.

AWS CLI

Come aggiornare un gruppo di sottoreti

Il seguente update-subnet-group `aggiorna l'ID di sottorete di un gruppo di sottoreti.

```
aws memorydb update-subnet-group \
    --subnet-group-name my-sg \
    --subnet-ids subnet-01f29d458f3xxxxx
```

Output:

```
{
    "SubnetGroup": {
        "Name": "my-sg-1",
        "Description": "my-sg",
        "VpcId": "vpc-09d2cfc01xxxxxxx",
        "Subnets": [
            {
                "Identifier": "subnet-01f29d458fxxxxxx",
                "AvailabilityZone": {
```

```
                    "Name": "us-east-1a"
                }
            }
        ],
        "ARN": "arn:aws:memorydb:us-east-1:491658xxxxxx:subnetgroup/my-sg"
    }
}
```

Per ulteriori informazioni, consulta Sottoreti e gruppi di sottoreti nella Guida per l'utente di MemoryDB.

- Per i dettagli sull'API, vedere UpdateSubnetGroupin AWS CLI Command Reference.

`update-user`

Il seguente esempio di codice mostra come utilizzareupdate-user.

AWS CLI

Come aggiornare un utente

Il comando update-user seguente modifica la stringa di accesso di un utente.

```
aws memorydb update-user \
    --user-name my-user \
    --access-string "off ~objects:* ~items:* ~public:* resetchannels -@all"
```

Output:

```
{
    "User": {
        "Name": "my-user",
        "Status": "modifying",
        "AccessString": "off ~objects:* ~items:* ~public:* resetchannels -@all",
        "ACLNames": [
            "myt-acl"
        ],
        "MinimumEngineVersion": "6.2",
        "Authentication": {
            "Type": "password",
            "PasswordCount": 2
        },
        "ARN": "arn:aws:memorydb:us-east-1:491658xxxxxx:user/my-user"
```

```
    }
}
```

Per ulteriori informazioni, consulta Autenticazione degli utenti con le liste di controllo accessi nella Guida per l'utente di MemoryDB.

- Per i dettagli sull'API, consulta UpdateUser AWS CLICommand Reference.

Esempi di utilizzo di Amazon MSK AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface con Amazon MSK.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`create-cluster`

Il seguente esempio di codice mostra come usare`create-cluster`.

AWS CLI

Come creare un cluster Amazon MSK

L'esempio `create-cluster` seguente crea un cluster MSK denominato `MessagingCluster` con tre nodi broker. Un file JSON denominato `brokernodegroupinfo.json` specifica le tre sottoreti su cui desideri che Amazon MSK distribuisca i nodi del broker. Questo esempio non specifica il livello di monitoraggio, quindi il cluster ottiene il livello `DEFAULT`.

```
aws kafka create-cluster \
    --cluster-name "MessagingCluster" \
```

```
    --broker-node-group-info file://brokernodegroupinfo.json \
    --kafka-version "2.2.1" \
    --number-of-broker-nodes 3
```

Contenuto di `brokernodegroupinfo.json`:

```
{
    "InstanceType": "kafka.m5.xlarge",
    "BrokerAZDistribution": "DEFAULT",
    "ClientSubnets": [
        "subnet-0123456789111abcd",
        "subnet-0123456789222abcd",
        "subnet-0123456789333abcd"
    ]
}
```

Output:

```
{
    "ClusterArn": "arn:aws:kafka:us-west-2:123456789012:cluster/MessagingCluster/
a1b2c3d4-5678-90ab-cdef-11111EXAMPLE-2",
    "ClusterName": "MessagingCluster",
    "State": "CREATING"
}
```

Per ulteriori informazioni, consulta Creare un cluster Amazon MSK nello Streaming gestito da Amazon per Apache Kafka (Amazon MSK).

- Per i dettagli sull'API, consulta CreateCluster AWS CLICommand Reference.

`create-configuration`

Il seguente esempio di codice mostra come utilizzare`create-configuration`.

AWS CLI

Come creare una configurazione Amazon MSK personalizzata

L'esempio `create-configuration` seguente crea una configurazione MSK personalizzata con le proprietà del server specificate nel file di input.

```
aws kafka create-configuration \
```

```
    --name "CustomConfiguration" \
    --description "Topic autocreation enabled; Apache ZooKeeper timeout 2000 ms; Log
 rolling 604800000 ms." \
    --kafka-versions "2.2.1" \
    --server-properties file://configuration.txt
```

Contenuto di `configuration.txt`:

```
auto.create.topics.enable = true
zookeeper.connection.timeout.ms = 2000
log.roll.ms = 604800000
```

Questo comando non produce alcun output. Output:

```
{
    "Arn": "arn:aws:kafka:us-west-2:123456789012:configuration/CustomConfiguration/
a1b2c3d4-5678-90ab-cdef-11111EXAMPLE-2",
    "CreationTime": "2019-10-09T15:26:05.548Z",
    "LatestRevision":
        {
            "CreationTime": "2019-10-09T15:26:05.548Z",
            "Description": "Topic autocreation enabled; Apache ZooKeeper timeout
 2000 ms; Log rolling 604800000 ms.",
            "Revision": 1
        },
    "Name": "CustomConfiguration"
}
```

Per ulteriori informazioni, consulta Operazioni di configurazione di Amazon MSK nella Guida per gli sviluppatori dello Streaming gestito da Amazon per Apache Kafka (Amazon MSK).

- Per i dettagli sull'API, consulta CreateConfiguration AWS CLICommand Reference.

`describe-cluster`

Il seguente esempio di codice mostra come utilizzare`describe-cluster`.

AWS CLI

Come descrivere un cluster

L'esempio `describe-cluster` seguente descrive un cluster Amazon MSK.

```
aws kafka describe-cluster \
    --cluster-arn arn:aws:kafka:us-east-1:123456789012:cluster/demo-
cluster-1/6357e0b2-0e6a-4b86-a0b4-70df934c2e31-5
```

Output:

```
{
    "ClusterInfo": {
        "BrokerNodeGroupInfo": {
            "BrokerAZDistribution": "DEFAULT",
            "ClientSubnets": [
                "subnet-cbfff283",
                "subnet-6746046b"
            ],
            "InstanceType": "kafka.m5.large",
            "SecurityGroups": [
                "sg-f839b688"
            ],
            "StorageInfo": {
                "EbsStorageInfo": {
                    "VolumeSize": 100
                }
            }
        },
        "ClusterArn": "arn:aws:kafka:us-east-1:123456789012:cluster/demo-
cluster-1/6357e0b2-0e6a-4b86-a0b4-70df934c2e31-5",
        "ClusterName": "demo-cluster-1",
        "CreationTime": "2020-07-09T02:31:36.223000+00:00",
        "CurrentBrokerSoftwareInfo": {
            "KafkaVersion": "2.2.1"
        },
        "CurrentVersion": "K3AEGXETSR30VB",
        "EncryptionInfo": {
            "EncryptionAtRest": {
                "DataVolumeKMSKeyId": "arn:aws:kms:us-east-1:123456789012:key/
a7ca56d5-0768-4b64-a670-339a9fbef81c"
            },
            "EncryptionInTransit": {
                "ClientBroker": "TLS_PLAINTEXT",
                "InCluster": true
            }
        },
        "EnhancedMonitoring": "DEFAULT",
```

```
        "OpenMonitoring": {
            "Prometheus": {
                "JmxExporter": {
                    "EnabledInBroker": false
                },
                "NodeExporter": {
                    "EnabledInBroker": false
                }
            }
        },
        "NumberOfBrokerNodes": 2,
        "State": "ACTIVE",
        "Tags": {},
        "ZookeeperConnectString": "z-2.demo-cluster-1.xuy0sb.c5.kafka.us-
east-1.amazonaws.com:2181,z-1.demo-cluster-1.xuy0sb.c5.kafka.us-
east-1.amazonaws.com:2181,z-3.demo-cluster-1.xuy0sb.c5.kafka.us-
east-1.amazonaws.com:2181"
    }
}
```

Per ulteriori informazioni, consulta Elencazione dei cluster Amazon MSK nella Guida per gli sviluppatori dello Streaming gestito da Amazon per Apache Kafka (Amazon MSK).

- Per i dettagli sull'API, consulta DescribeCluster AWS CLICommand Reference.

`get-bootstrap-brokers`

Il seguente esempio di codice mostra come utilizzare`get-bootstrap-brokers`.

AWS CLI

Come ottenere i broker bootstrap

L'esempio `get-bootstrap-brokers` seguente recupera le informazioni del broker di bootstrap per un cluster Amazon MSK.

```
aws kafka get-bootstrap-brokers \
    --cluster-arn arn:aws:kafka:us-east-1:123456789012:cluster/demo-
cluster-1/6357e0b2-0e6a-4b86-a0b4-70df934c2e31-5
```

Output:

```
{
```

```
    "BootstrapBrokerString": "b-1.demo-cluster-1.xuy0sb.c5.kafka.us-
east-1.amazonaws.com:9092,b-2.demo-cluster-1.xuy0sb.c5.kafka.us-
east-1.amazonaws.com:9092",
    "BootstrapBrokerStringTls": "b-1.demo-cluster-1.xuy0sb.c5.kafka.us-
east-1.amazonaws.com:9094,b-2.demo-cluster-1.xuy0sb.c5.kafka.us-
east-1.amazonaws.com:9094"
}
```

Per ulteriori informazioni, consulta Ottenere i broker bootstrap nella Guida per gli sviluppatori dello Streaming gestito da Amazon per Apache Kafka (Amazon MSK).

- Per i dettagli sull'API, consulta GetBootstrapBrokers AWS CLICommand Reference.

`list-clusters`

Il seguente esempio di codice mostra come utilizzare`list-clusters`.

AWS CLI

Come elencare i cluster disponibili

L'`list-clusters`esempio seguente elenca i cluster Amazon MSK nel tuo AWS account.

```
aws kafka list-clusters
```

Output:

```
{
    "ClusterInfoList": [
        {
            "BrokerNodeGroupInfo": {
                "BrokerAZDistribution": "DEFAULT",
                "ClientSubnets": [
                    "subnet-cbfff283",
                    "subnet-6746046b"
                ],
                "InstanceType": "kafka.m5.large",
                "SecurityGroups": [
                    "sg-f839b688"
                ],
                "StorageInfo": {
                    "EbsStorageInfo": {
                        "VolumeSize": 100
```

```
                    }
                }
            },
            "ClusterArn": "arn:aws:kafka:us-east-1:123456789012:cluster/demo-
cluster-1/6357e0b2-0e6a-4b86-a0b4-70df934c2e31-5",
            "ClusterName": "demo-cluster-1",
            "CreationTime": "2020-07-09T02:31:36.223000+00:00",
            "CurrentBrokerSoftwareInfo": {
                "KafkaVersion": "2.2.1"
            },
            "CurrentVersion": "K3AEGXETSR30VB",
            "EncryptionInfo": {
                "EncryptionAtRest": {
                    "DataVolumeKMSKeyId": "arn:aws:kms:us-east-1:123456789012:key/
a7ca56d5-0768-4b64-a670-339a9fbef81c"
                },
                "EncryptionInTransit": {
                    "ClientBroker": "TLS_PLAINTEXT",
                    "InCluster": true
                }
            },
            "EnhancedMonitoring": "DEFAULT",
            "OpenMonitoring": {
                "Prometheus": {
                    "JmxExporter": {
                        "EnabledInBroker": false
                    },
                    "NodeExporter": {
                        "EnabledInBroker": false
                    }
                }
            },
            "NumberOfBrokerNodes": 2,
            "State": "ACTIVE",
            "Tags": {},
            "ZookeeperConnectString": "z-2.demo-cluster-1.xuy0sb.c5.kafka.us-
east-1.amazonaws.com:2181,z-1.demo-cluster-1.xuy0sb.c5.kafka.us-
east-1.amazonaws.com:2181,z-3.demo-cluster-1.xuy0sb.c5.kafka.us-
east-1.amazonaws.com:2181"
        }
    ]
}
```

Per ulteriori informazioni, consulta Elencazione dei cluster Amazon MSK nella Guida per gli sviluppatori dello Streaming gestito da Amazon per Apache Kafka (Amazon MSK).

- Per i dettagli sull'API, consulta AWS CLI Command ListClustersReference.

`update-broker-storage`

Il seguente esempio di codice mostra come utilizzare`update-broker-storage`.

AWS CLI

Come aggiornare l'archiviazione EBS per i broker

L'esempio `update-broker-storage` seguente aggiorna la quantità di archiviazione EBS per tutti i broker del cluster. Amazon MSK imposta la quantità di storage di destinazione per ogni broker sulla quantità specificata nell'esempio. Puoi ottenere la versione corrente del cluster descrivendo il cluster o elencando tutti i cluster.

```
aws kafka update-broker-storage \
    --cluster-arn "arn:aws:kafka:us-west-2:123456789012:cluster/MessagingCluster/
a1b2c3d4-5678-90ab-cdef-11111EXAMPLE-2" \
    --current-version "K21V3IB1VIZYYH" \
    --target-broker-ebs-volume-info "KafkaBrokerNodeId=ALL,VolumeSizeGB=1100"
```

L'output restituisce un ARN per questa operazione `update-broker-storage`. Per determinare se l'operazione è completa, utilizza il comando `describe-cluster-operation` con questo ARN come input.

```
{
    "ClusterArn": "arn:aws:kafka:us-west-2:123456789012:cluster/MessagingCluster/
a1b2c3d4-5678-90ab-cdef-11111EXAMPLE-2",
    "ClusterOperationArn": "arn:aws:kafka:us-west-2:123456789012:cluster-
operation/V123450123/a1b2c3d4-1234-abcd-cdef-22222EXAMPLE-2/a1b2c3d4-abcd-1234-
bcde-33333EXAMPLE"
}
```

Per ulteriori informazioni, consulta Aggiornamento dello storage EBS for broker nella Guida per gli sviluppatori dello Streaming gestito da Amazon per Apache Kafka (Amazon MSK).

- Per i dettagli sull'API, consulta UpdateBrokerStorage AWS CLICommand Reference.

`update-cluster-configuration`

Il seguente esempio di codice mostra come utilizzareu`pdate-cluster-configuration`.

AWS CLI

Come aggiornare la configurazione di un cluster Amazon MSK

L'esempio `update-cluster-configuration` seguente aggiorna la configurazione del cluster MSK esistente specificato. Utilizza una configurazione MSK personalizzata.

```
aws kafka update-cluster-configuration \
    --cluster-arn "arn:aws:kafka:us-west-2:123456789012:cluster/MessagingCluster/
a1b2c3d4-5678-90ab-cdef-11111EXAMPLE-2" \
    --configuration-info file://configuration-info.json \
    --current-version "K21V3IB1VIZYYH"
```

Contenuto di `configuration-info.json`:

```
{
    "Arn": "arn:aws:kafka:us-west-2:123456789012:configuration/CustomConfiguration/
a1b2c3d4-5678-90ab-cdef-11111EXAMPLE-2",
    "Revision": 1
}
```

L'output restituisce un ARN per questa operazione `update-cluster-configuration`. Per determinare se l'operazione è completa, utilizza il comando `describe-cluster-operation` con questo ARN come input.

```
{
    "ClusterArn": "arn:aws:kafka:us-west-2:123456789012:cluster/MessagingCluster/
a1b2c3d4-5678-90ab-cdef-11111EXAMPLE-2",
    "ClusterOperationArn": "arn:aws:kafka:us-west-2:123456789012:cluster-
operation/V123450123/a1b2c3d4-1234-abcd-cdef-22222EXAMPLE-2/a1b2c3d4-abcd-1234-
bcde-33333EXAMPLE"
}
```

Per ulteriori informazioni, consulta Aggiornare la configurazione di un cluster Amazon MSK nella Guida per gli sviluppatori dello Streaming gestito da Amazon per Apache Kafka (Amazon MSK).

- Per i dettagli sull'API, consulta UpdateClusterConfiguration AWS CLICommand Reference.

Esempi di Network Flow Monitor utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando Network Flow Monitor. AWS Command Line Interface

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`create-monitor`

Il seguente esempio di codice mostra come utilizzare`create-monitor`.

AWS CLI

Come creare un monitoraggio

L'esempio `create-monitor` seguente crea un monitoraggio denominato demo nell'account specificato.

```
aws networkflowmonitor create-monitor \
    --monitor-name demo \
    --local-resources type="AWS::EC2::VPC",identifier="arn:aws:ec2:us-
east-1:123456789012:vpc/vpc-03ea55eeda25adbb0"  \
    --scope-arn arn:aws:networkflowmonitor:us-east-1:123456789012:scope/
e21cda79-30a0-4c12-9299-d8629d76d8cf
```

Output:

```
{
    "monitorArn": "arn:aws:networkflowmonitor:us-east-1:123456789012:monitor/demo",
    "monitorName": "demo",
```

```
    "monitorStatus": "ACTIVE",
    "tags": {}
}
```

Per ulteriori informazioni, consulta Creare un monitor in Network Flow Monitor nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta CreateMonitor AWS CLICommand Reference.

`create-scope`

Il seguente esempio di codice mostra come utilizzare`create-scope`.

AWS CLI

Come creare un ambito

L'esempio `create-scope` seguente crea un ambito che include un set di risorse per le quali Network Flow Monitor genererà le metriche del traffico di rete.

```
aws networkflowmonitor create-scope \
    --targets '[{"targetIdentifier":{"targetId":
{"accountId":"123456789012"},"targetType":"ACCOUNT"},"region":"us-east-1"}]'
```

Output:

```
{
    "scopeId": "97626f8d-8a21-4b5d-813a-1a0962dd4615",
    "status": "IN_PROGRESS",
    "tags": {}
}
```

Per ulteriori informazioni, consulta Componenti e funzionalità di Network Flow Monitor nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta CreateScope AWS CLICommand Reference.

`delete-monitor`

Il seguente esempio di codice mostra come utilizzare`delete-monitor`.

AWS CLI

Come eliminare un monitoraggio

L'esempio `delete-monitor` seguente elimina un monitor denominato demo nell'account specificato.

```
aws networkflowmonitor delete-monitor \
    --monitor-name demo
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Eliminare un monitor in Network Flow Monitor nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta DeleteMonitor AWS CLICommand Reference.

`delete-scope`

Il seguente esempio di codice mostra come utilizzare`delete-scope`.

AWS CLI

Come eliminare un ambito

L'esempio `delete-scope` seguente elimina un ambito specificato.

```
aws networkflowmonitor delete-scope \
    --scope-id fdc20616-6bb4-4242-a24e-a748e65ca7ac
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Componenti e funzionalità di Network Flow Monitor nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta DeleteScope AWS CLICommand Reference.

`get-monitor`

Il seguente esempio di codice mostra come utilizzare`get-monitor`.

AWS CLI

Come recuperare le informazioni relative a un monitoraggio

L'esempio `get-monitor` seguente visualizza le informazioni sul monitoraggio denominato demo nell'account specificato.

```
aws networkflowmonitor get-monitor \
    --monitor-name Demo
```

Output:

```
{
    "monitorArn": "arn:aws:networkflowmonitor:us-east-1:123456789012:monitor/Demo",
    "monitorName": "Demo",
    "monitorStatus": "ACTIVE",
    "localResources": [
        {
            "type": "AWS::EC2::VPC",
            "identifier": "arn:aws:ec2:us-east-1:123456789012:vpc/
vpc-03ea55eeda25adbb0"
        }
    ],
    "remoteResources": [],
    "createdAt": "2024-12-09T12:21:51.616000-06:00",
    "modifiedAt": "2024-12-09T12:21:55.412000-06:00",
    "tags": {}
}
```

Per ulteriori informazioni, consulta Componenti e funzionalità di Network Flow Monitor nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta GetMonitor AWS CLICommand Reference.

`get-query-results-workload-insights-top-contributors-data`

Il seguente esempio di codice mostra come utilizzare`get-query-results-workload-insights-top-contributors-data`.

AWS CLI

Come recuperare i dati dei principali contributor sugli approfondimenti relativi al carico di lavoro

L'esempio `get-query-results-workload-insights-top-contributors-data` seguente restituisce i dati per la query specificata.

```
aws networkflowmonitor get-query-results-workload-insights-top-contributors-data \
    --scope-id e21cda79-30a0-4c12-9299-d8629d76d8cf \
    --query-id cc4f4ab3-3103-33b8-80ff-d6597a0c6cea
```

Output:

```
{
    "datapoints": [
        {
            "timestamps": [
                "2024-12-09T19:00:00+00:00",
                "2024-12-09T19:05:00+00:00",
                "2024-12-09T19:10:00+00:00"
            ],
            "values": [
                259943.0,
                194856.0,
                216432.0
            ],
            "label": "use1-az6"
        }
    ],
    "unit": "Bytes"
}
```

Per ulteriori informazioni, consulta Valuta i flussi di rete con informazioni sui carichi di lavoro nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta GetQueryResultsWorkloadInsightsTopContributorsData AWS CLICommand Reference.

`get-query-results-workload-insights-top-contributors`

Il seguente esempio di codice mostra come utilizzare`get-query-results-workload-insights-top-contributors`.

AWS CLI

Come recuperare i principali contributor sugli approfondimenti relativi al carico di lavoro

L'esempio `get-query-results-workload-insights-top-contributors` seguente restituisce i dati per la query specificata.

```
aws networkflowmonitor get-query-results-workload-insights-top-contributors \
    --scope-id e21cda79-30a0-4c12-9299-d8629d76d8cf \
    --query-id 1fc423d3-b144-37a6-80e6-e2c7d26eea0c
```

Output:

```
{
    "topContributors": [
        {
            "accountId": "123456789012",
            "localSubnetId": "subnet-0a5b30fb95dca2c14",
            "localAz": "use1-az6",
            "localVpcId": "vpc-03ea55eeda25adbb0",
            "localRegion": "us-east-1",
            "remoteIdentifier": "",
            "value": 908443,
            "localSubnetArn": "arn:aws:ec2:us-east-1:123456789012:subnet/
subnet-0a5b30fb95dca2c14",
            "localVpcArn": "arn:aws:ec2:us-east-1:123456789012:vpc/
vpc-03ea55eeda25adbb0"
        }
    ]
}
```

Per ulteriori informazioni, consulta Valuta i flussi di rete con informazioni sui carichi di lavoro nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta GetQueryResultsWorkloadInsightsTopContributors AWS CLICommand Reference.

`get-query-status-monitor-top-contributors`

Il seguente esempio di codice mostra come utilizzare`get-query-status-monitor-top-contributors`.

AWS CLI

Come recuperare lo stato della query

L'esempio `get-query-status-monitor-top-contributors` seguente visualizza lo stato corrente della query nell'account specificato.

```
aws networkflowmonitor get-query-status-monitor-top-contributors \
    --monitor-name Demo \
    --query-id 5398eabd-bc40-3f5f-aba3-bcb639d3c7ca
```

Output:

```
{
    "status": "SUCCEEDED"
}
```

Per ulteriori informazioni, consulta Valuta i flussi di rete con informazioni sui carichi di lavoro nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta GetQueryStatusMonitorTopContributors AWS CLICommand Reference.

`get-query-status-workload-insights-top-contributors-data`

Il seguente esempio di codice mostra come utilizzare`get-query-status-workload-insights-top-contributors-data`.

AWS CLI

Come recuperare lo stato della query

L'esempio `get-query-status-workload-insights-top-contributors-data` seguente visualizza lo stato corrente della query nell'account specificato.

```
aws networkflowmonitor get-query-status-workload-insights-top-contributors-data \
    --scope-id e21cda79-30a0-4c12-9299-d8629d76d8cf \
    --query-id 4333754d-8ae1-3f29-b6b7-c36db2e7f8ac
```

Output:

```
{
    "status": "SUCCEEDED"
}
```

Per ulteriori informazioni, consulta Valuta i flussi di rete con informazioni sui carichi di lavoro nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta GetQueryStatusWorkloadInsightsTopContributorsData AWS CLICommand Reference.

`get-query-status-workload-insights-top-contributors`

Il seguente esempio di codice mostra come utilizzare`get-query-status-workload-insights-top-contributors`.

AWS CLI

Come recuperare lo stato della query

L'esempio `get-query-status-workload-insights-top-contributors` seguente visualizza lo stato corrente della query nell'account specificato.

```
aws networkflowmonitor get-query-status-workload-insights-top-contributors \
    --scope-id e21cda79-30a0-4c12-9299-d8629d76d8cf \
    --query-id f2a87c70-3e5a-362e-8beb-4747d13d8419
```

Output:

```
{
    "status": "SUCCEEDED"
}
```

Per ulteriori informazioni, consulta Valuta i flussi di rete con informazioni sui carichi di lavoro nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta GetQueryStatusWorkloadInsightsTopContributors AWS CLICommand Reference.

`get-scope`

Il seguente esempio di codice mostra come utilizzare`get-scope`.

AWS CLI

Come recuperare le informazioni relative a un ambito

L'esempio `get-scope` seguente mostra le informazioni su un ambito, come lo stato, i tag, il nome e i dettagli di destinazione.

```
aws networkflowmonitor get-scope \
    --scope-id e21cda79-30a0-4c12-9299-d8629d76d8cf
```

Output:

```
{
    "scopeId": "e21cda79-30a0-4c12-9299-d8629d76d8cf",
    "status": "SUCCEEDED",
    "scopeArn": "arn:aws:networkflowmonitor:us-east-1:123456789012:scope/
e21cda79-30a0-4c12-9299-d8629d76d8cf",
    "targets": [
        {
            "targetIdentifier": {
                "targetId": {
                    "accountId": "123456789012"
                },
                "targetType": "ACCOUNT"
            },
            "region": "us-east-1"
        }
    ],
    "tags": {}
}
```

Per ulteriori informazioni, consulta Componenti e funzionalità di Network Flow Monitor nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta GetScope AWS CLICommand Reference.

`list-monitors`

Il seguente esempio di codice mostra come utilizzare`list-monitors`.

AWS CLI

Come recuperare un elenco di monitoraggi

L'elenco `list-monitors` seguente restituisce tutti i monitor dell'account specificato.

```
aws networkflowmonitor list-monitors
```

Output:

```
{
    "monitors": [
        {
            "monitorArn": "arn:aws:networkflowmonitor:us-
east-1:123456789012:monitor/Demo",
            "monitorName": "Demo",
            "monitorStatus": "ACTIVE"
        }
    ]
}
```

Per ulteriori informazioni, consulta Componenti e funzionalità di Network Flow Monitor nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta ListMonitors AWS CLICommand Reference.

`list-scopes`

Il seguente esempio di codice mostra come utilizzare`list-scopes`.

AWS CLI

Come recuperare un elenco di ambiti

L'esempio `list-scopes` seguente elenca tutti gli ambiti nell'account specificato.

```
aws networkflowmonitor list-scopes
```

Output:

```
{
    "scopes": [
        {
            "scopeId": "fdc20616-6bb4-4242-a24e-a748e65ca7ac",
            "status": "SUCCEEDED",
            "scopeArn": "arn:aws:networkflowmonitor:us-east-1:123456789012:scope/
fdc20616-6bb4-4242-a24e-a748e65ca7ac"
```

```
        }
    ]
}
```

Per ulteriori informazioni, consulta Componenti e funzionalità di Network Flow Monitor nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta ListScopes AWS CLICommand Reference.

`list-tags-for-resource`

Il seguente esempio di codice mostra come utilizzare`list-tags-for-resource`.

AWS CLI

Come elencare i tag

L'esempio `list-tags-for-resource` seguente restituisce tutti i tag associati alla risorsa specificata.

```
aws networkflowmonitor list-tags-for-resource \
    --resource-arn arn:aws:networkflowmonitor:us-east-1:123456789012:monitor/Demo
```

Output:

```
{
    "tags": {
        "Value": "Production",
        "Key": "stack"
    }
}
```

Per ulteriori informazioni, consulta Taggare le CloudWatch risorse Amazon nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta ListTagsForResource AWS CLICommand Reference.

`start-query-monitor-top-contributors`

Il seguente esempio di codice mostra come utilizzare`start-query-monitor-top-contributors`.

AWS CLI

Come avviare una query

L'esempio `start-query-monitor-top-contributors` seguente avvia la query che restituisce un ID di query per recuperare i principali contributor.

```
aws networkflowmonitor start-query-monitor-top-contributors \
    --monitor-name Demo \
    --start-time 2024-12-09T19:00:00Z \
    --end-time 2024-12-09T19:15:00Z \
    --metric-name DATA_TRANSFERRED \
    --destination-category UNCLASSIFIED
```

Output:

```
{
    "queryId": "aecd3a88-0283-35b0-a17d-6e944dc8531d"
}
```

Per ulteriori informazioni, consulta Valuta i flussi di rete con informazioni sui carichi di lavoro nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta StartQueryMonitorTopContributors AWS CLICommand Reference.

`start-query-workload-insights-top-contributors-data`

Il seguente esempio di codice mostra come utilizzare`start-query-workload-insights-top-contributors-data`.

AWS CLI

Come avviare una query

L'esempio `start-query-workload-insights-top-contributors-data` seguente avvia la query che restituisce un ID di query per recuperare i principali contributor.

```
aws networkflowmonitor start-query-workload-insights-top-contributors-data \
    --scope-id e21cda79-30a0-4c12-9299-d8629d76d8cf \
    --start-time 2024-12-09T19:00:00Z \
```

```
    --end-time 2024-12-09T19:15:00Z \
    --metric-name DATA_TRANSFERRED \
    --destination-category UNCLASSIFIED
```

Output:

```
{
    "queryId": "cc4f4ab3-3103-33b8-80ff-d6597a0c6cea"
}
```

Per ulteriori informazioni, consulta Valuta i flussi di rete con informazioni sui carichi di lavoro nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta StartQueryWorkloadInsightsTopContributorsData AWS CLICommand Reference.

`start-query-workload-insights-top-contributors`

Il seguente esempio di codice mostra come utilizzare`start-query-workload-insights-top-contributors`.

AWS CLI

Come avviare una query

L'esempio `start-query-workload-insights-top-contributors` seguente avvia la query che restituisce un ID di query per recuperare i principali contributor.

```
aws networkflowmonitor start-query-workload-insights-top-contributors \
    --scope-id e21cda79-30a0-4c12-9299-d8629d76d8cf \
    --start-time 2024-12-09T19:00:00Z \
    --end-time 2024-12-09T19:15:00Z \
    --metric-name DATA_TRANSFERRED \
    --destination-category UNCLASSIFIED
```

Output:

```
{
    "queryId": "1fc423d3-b144-37a6-80e6-e2c7d26eea0c"
}
```

Per ulteriori informazioni, consulta Valuta i flussi di rete con informazioni sui carichi di lavoro nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta StartQueryWorkloadInsightsTopContributors AWS CLICommand Reference.

`stop-query-monitor-top-contributors`

Il seguente esempio di codice mostra come utilizzare`stop-query-monitor-top-contributors`.

AWS CLI

Come arrestare una query

L'esempio `stop-query-monitor-top-contributors` seguente arresta la query nell'account specificato.

```
aws networkflowmonitor stop-query-monitor-top-contributors \
    --monitor-name Demo \
    --query-id aecd3a88-0283-35b0-a17d-6e944dc8531d
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Valuta i flussi di rete con informazioni sui carichi di lavoro nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta StopQueryMonitorTopContributors AWS CLICommand Reference.

`stop-query-workload-insights-top-contributors-data`

Il seguente esempio di codice mostra come utilizzare`stop-query-workload-insights-top-contributors-data`.

AWS CLI

Come arrestare una query

L'esempio `stop-query-workload-insights-top-contributors-data` seguente arresta la query nell'account specificato.

```
aws networkflowmonitor stop-query-workload-insights-top-contributors-data \
```

```
    --scope-id e21cda79-30a0-4c12-9299-d8629d76d8cf \
    --query-id cc4f4ab3-3103-33b8-80ff-d6597a0c6cea
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Valuta i flussi di rete con informazioni sui carichi di lavoro nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta StopQueryWorkloadInsightsTopContributorsData AWS CLICommand Reference.

`stop-query-workload-insights-top-contributors`

Il seguente esempio di codice mostra come utilizzare`stop-query-workload-insights-top-contributors`.

AWS CLI

Come arrestare una query

L'esempio `stop-query-workload-insights-top-contributors` seguente arresta la query nell'account specificato.

```
aws networkflowmonitor stop-query-workload-insights-top-contributors \
    --scope-id e21cda79-30a0-4c12-9299-d8629d76d8cf \
    --query-id 1fc423d3-b144-37a6-80e6-e2c7d26eea0c
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Valuta i flussi di rete con informazioni sui carichi di lavoro nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta StopQueryWorkloadInsightsTopContributors AWS CLICommand Reference.

`tag-resource`

Il seguente esempio di codice mostra come utilizzare`tag-resource`.

AWS CLI

Come aggiungere un tag alla risorsa specificata

L'esempio `tag-resource` seguente aggiunge un tag al monitor nell'account specificato.

```
aws networkflowmonitor tag-resource \
    --resource-arn arn:aws:networkflowmonitor:us-east-1:123456789012:monitor/Demo \
    --tags Key=stack,Value=Production
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Taggare le CloudWatch risorse Amazon nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta TagResource AWS CLICommand Reference.

`untag-resource`

Il seguente esempio di codice mostra come utilizzare`untag-resource`.

AWS CLI

Come rimuovere tutti i tag dalla risorsa specificata

L'esempio `untag-resource` seguente rimuove un tag dal monitor nell'account specificato.

```
aws networkflowmonitor untag-resource \
    --resource-arn arn:aws:networkflowmonitor:us-east-1:123456789012:monitor/Demo \
    --tag-keys stack
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Taggare le CloudWatch risorse Amazon nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta UntagResource AWS CLICommand Reference.

`update-monitor`

Il seguente esempio di codice mostra come utilizzare`update-monitor`.

AWS CLI

Come aggiornare un monitoraggio esistente

L'esempio `update-monitor` seguente aggiorna il monitor denominato `Demo` nell'account specificato.

```
aws networkflowmonitor update-monitor \
    --monitor-name Demo \
    --local-resources-to-add type="AWS::EC2::VPC",identifier="arn:aws:ec2:us-
east-1:123456789012:vpc/vpc-048d08dfbec623f94"
```

Output:

```
{
    "monitorArn": "arn:aws:networkflowmonitor:us-east-1:123456789012:monitor/Demo",
    "monitorName": "Demo",
    "monitorStatus": "ACTIVE",
    "tags": {
        "Value": "Production",
        "Key": "stack"
    }
}
```

Per ulteriori informazioni, consulta Componenti e funzionalità di Network Flow Monitor nella Amazon CloudWatch User Guide.

- Per i dettagli sull'API, consulta UpdateMonitor AWS CLICommand Reference.

Esempi di Network Manager che utilizzano AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with Network Manager.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`associate-customer-gateway`

Il seguente esempio di codice mostra come utilizzare`associate-customer-gateway`.

AWS CLI

Come associare un gateway del cliente

L'esempio `associate-customer-gateway` seguente associa il gateway del cliente `cgw-11223344556677889` nella rete globale specificata al dispositivo `device-07f6fd08867abc123`.

```
aws networkmanager associate-customer-gateway \
    --customer-gateway-arn arn:aws:ec2:us-west-2:123456789012:customer-gateway/
cgw-11223344556677889  \
    --global-network-id global-network-01231231231231231 \
    --device-id device-07f6fd08867abc123 \
    --region us-west-2
```

Output:

```
{
    "CustomerGatewayAssociation": {
        "CustomerGatewayArn": "arn:aws:ec2:us-west-2:123456789012:customer-gateway/
cgw-11223344556677889",
        "GlobalNetworkId": "global-network-01231231231231231",
        "DeviceId": "device-07f6fd08867abc123",
        "State": "PENDING"
    }
}
```

Per ulteriori informazioni, consulta Associazioni del gateway del cliente nella Guida di Transit Gateway network manager.

- Per i dettagli sull'API, consulta AssociateCustomerGateway AWS CLICommand Reference.

`associate-link`

Il seguente esempio di codice mostra come utilizzare`associate-link`.

AWS CLI

Come associare un link

L'esempio `associate-link` seguente associa il link `link-11112222aaaabbbb1` al dispositivo `device-07f6fd08867abc123`. Il link e il dispositivo si trovano nella rete globale specificata.

```
aws networkmanager associate-link \
    --global-network-id global-network-01231231231231231 \
    --device-id device-07f6fd08867abc123 \
    --link-id link-11112222aaaabbbb1 \
    --region us-west-2
```

Output:

```
{
    "LinkAssociation": {
        "GlobalNetworkId": "global-network-01231231231231231",
        "DeviceId": "device-07f6fd08867abc123",
        "LinkId": "link-11112222aaaabbbb1",
        "LinkAssociationState": "PENDING"
    }
}
```

Per ulteriori informazioni, consulta Associazioni di dispositivi e link nella Guida di Transit Gateway network manager.

- Per i dettagli sull'API, consulta AssociateLink AWS CLICommand Reference.

`create-core-network`

Il seguente esempio di codice mostra come utilizzare`create-core-network`.

AWS CLI

Come creare una rete principale

L'`create-core-network`esempio seguente crea una rete principale utilizzando una descrizione e tag opzionali all'interno di una rete globale AWS Cloud WAN.

```
aws networkmanager create-core-network \
    --global-network-id global-network-cdef-EXAMPLE22222 \
```

```
    --description "Main headquarters location" \
    --tags Key=Name,Value="New York City office"
```

Output:

```
{
    "CoreNetwork": {
        "GlobalNetworkId": "global-network-cdef-EXAMPLE22222",
        "CoreNetworkId": "core-network-cdef-EXAMPLE33333",
        "CoreNetworkArn": "arn:aws:networkmanager::987654321012:core-network/core-
network-cdef-EXAMPLE33333",
        "Description": "Main headquarters location",
        "CreatedAt": "2022-01-10T19:53:59+00:00",
        "State": "AVAILABLE",
        "Tags": [
            {
                "Key": "Name",
                "Value": "New York City office"
            }
        ]
    }
}
```

Per ulteriori informazioni, consulta Reti di base e globali nella Guida per l'utente di AWS Cloud WAN.

- Per i dettagli sull'API, consulta CreateCoreNetwork AWS CLICommand Reference.

`create-device`

Il seguente esempio di codice mostra come utilizzare`create-device`.

AWS CLI

Come creare un dispositivo

L'esempio `create-device` seguente crea un dispositivo nella rete globale specificata. I dettagli del dispositivo includono una descrizione, il tipo, il fornitore, il modello e il numero di serie.

```
aws networkmanager create-device
    --global-network-id global-network-01231231231231231 \
    --description "New York office device" \
    --type "office device" \
```

```
    --vendor "anycompany" \
    --model "abcabc" \
    --serial-number "1234" \
    --region us-west-2
```

Output:

```
{
    "Device": {
        "DeviceId": "device-07f6fd08867abc123",
        "DeviceArn": "arn:aws:networkmanager::123456789012:device/global-
network-0123123123123231/device-07f6fd08867abc123",
        "GlobalNetworkId": "global-network-0123123123123231",
        "Description": "New York office device",
        "Type": "office device",
        "Vendor": "anycompany",
        "Model": "abcabc",
        "SerialNumber": "1234",
        "CreatedAt": 1575554005.0,
        "State": "PENDING"
    }
}
```

Per ulteriori informazioni, consulta Utilizzo di dispositivi nella Guida di Transit Gateway network manager.

- Per i dettagli sull'API, consulta CreateDevice AWS CLICommand Reference.

`create-global-network`

Il seguente esempio di codice mostra come utilizzare`create-global-network`.

AWS CLI

Come creare una rete globale

L'esempio `create-global-network` seguente crea una nuova rete globale. Lo stato iniziale al momento della creazione è `PENDING`.

```
aws networkmanager create-global-network
```

Output:

```
{
    "GlobalNetwork": {
        "GlobalNetworkId": "global-network-00a77fc0f722dae74",
        "GlobalNetworkArn": "arn:aws:networkmanager::987654321012:global-network/
global-network-00a77fc0f722dae74",
        "CreatedAt": "2022-03-14T20:31:56+00:00",
        "State": "PENDING"
    }
}
```

- Per i dettagli sull'API, consulta CreateGlobalNetwork AWS CLICommand Reference.

`create-link`

Il seguente esempio di codice mostra come utilizzare`create-link`.

AWS CLI

Come creare un link

L'esempio `create-link` seguente crea un link nella rete globale specificata. Il link include una descrizione e i dettagli sul tipo di link, sulla larghezza di banda e sul provider. L'ID del sito indica il sito a cui è associato il link.

```
aws networkmanager create-link \
    --global-network-id global-network-01231231231231231 \
    --description "VPN Link" \
    --type "broadband" \
    --bandwidth UploadSpeed=10,DownloadSpeed=20 \
    --provider "AnyCompany" \
    --site-id site-444555aaabbb11223 \
    --region us-west-2
```

Output:

```
{
    "Link": {
        "LinkId": "link-11112222aaaabbbb1",
        "LinkArn": "arn:aws:networkmanager::123456789012:link/global-
network-01231231231231231/link-11112222aaaabbbb1",
        "GlobalNetworkId": "global-network-01231231231231231",
```

```
        "SiteId": "site-444555aaabbb11223",
        "Description": "VPN Link",
        "Type": "broadband",
        "Bandwidth": {
            "UploadSpeed": 10,
            "DownloadSpeed": 20
        },
        "Provider": "AnyCompany",
        "CreatedAt": 1575555811.0,
        "State": "PENDING"
    }
}
```

Per ulteriori informazioni, consulta Utilizzo di link nella Guida di Transit Gateway network manager.

- Per i dettagli sull'API, consulta CreateLink AWS CLICommand Reference.

`create-site`

Il seguente esempio di codice mostra come utilizzare`create-site`.

AWS CLI

Come creare un sito

L'esempio `create-site` seguente crea un sito nella rete globale specificata. I dettagli del sito includono una descrizione e informazioni sulla posizione.

```
aws networkmanager create-site \
    --global-network-id global-network-01231231231231231 \
    --description  "New York head office" \
    --location Latitude=40.7128,Longitude=-74.0060 \
    --region us-west-2
```

Output:

```
{
    "Site": {
        "SiteId": "site-444555aaabbb11223",
        "SiteArn": "arn:aws:networkmanager::123456789012:site/global-
network-01231231231231231/site-444555aaabbb11223",
```

```
        "GlobalNetworkId": "global-network-01231231231231231",
        "Description": "New York head office",
        "Location": {
            "Latitude": "40.7128",
            "Longitude": "-74.0060"
        },
        "CreatedAt": 1575554300.0,
        "State": "PENDING"
    }
}
```

Per ulteriori informazioni, consulta Utilizzo di siti nella Guida di Transit Gateway network manager.

- Per i dettagli sull'API, consulta CreateSite AWS CLICommand Reference.

`create-vpc-attachment`

Il seguente esempio di codice mostra come utilizzare`create-vpc-attachment`.

AWS CLI

Come creare un collegamento VPC

L'`create-vpc-attachment`esempio seguente crea un allegato VPC con IPv6 supporto in una rete centrale.

```
aws networkmanager create-vpc-attachment \
    --core-network-id core-network-0fab62fe438d94db6 \
    --vpc-arn arn:aws:ec2:us-east-1:987654321012:vpc/vpc-09f37f69e2786eeb8  \
    --subnet-arns arn:aws:ec2:us-east-1:987654321012:subnet/subnet-04ca4e010857e7bb7
 \
    --Ipv6Support=true
```

Output:

```
{
    "VpcAttachment": {
        "Attachment": {
            "CoreNetworkId": "core-network-0fab62fe438d94db6",
            "AttachmentId": "attachment-05e1da6eba87a06e6",
            "OwnerAccountId": "987654321012",
            "AttachmentType": "VPC",
            "State": "CREATING",
```

```
            "EdgeLocation": "us-east-1",
            "ResourceArn": "arn:aws:ec2:us-east-1:987654321012:vpc/
vpc-09f37f69e2786eeb8",
            "Tags": [],
            "CreatedAt": "2022-03-10T20:59:14+00:00",
            "UpdatedAt": "2022-03-10T20:59:14+00:00"
        },
        "SubnetArns": [
            "arn:aws:ec2:us-east-1:987654321012:subnet/subnet-04ca4e010857e7bb7"
        ],
        "Options": {
            "Ipv6Support": true
        }
    }
}
```

Per ulteriori informazioni, consulta Creare un collegameto nella Guida per l'utente di Cloud WAN.

- Per i dettagli sull'API, vedere CreateVpcAttachmentin AWS CLI Command Reference.

`delete-attachment`

Il seguente esempio di codice mostra come utilizzare`delete-attachment`.

AWS CLI

Come eliminare un collegamento

L'esempio `delete-attachment` seguente elimina un collegamento Connect.

```
aws networkmanager delete-attachment \
    --attachment-id attachment-01feddaeae26ab68c
```

Output:

```
{
    "Attachment": {
        "CoreNetworkId": "core-network-0f4b0a9d5ee7761d1",
        "AttachmentId": "attachment-01feddaeae26ab68c",
        "OwnerAccountId": "987654321012",
        "AttachmentType": "CONNECT",
        "State": "DELETING",
        "EdgeLocation": "us-east-1",
```

```
        "ResourceArn": "arn:aws:networkmanager::987654321012:attachment/
attachment-02c3964448fedf5aa",
        "CreatedAt": "2022-03-15T19:18:41+00:00",
        "UpdatedAt": "2022-03-15T19:28:59+00:00"
    }
}
```

Per ulteriori informazioni, consulta Eliminare i collegamenti nella Guida per l'utente di Cloud WAN.

- Per i dettagli sull'API, consulta DeleteAttachment AWS CLICommand Reference.

`delete-bucket-analytics-configuration`

Il seguente esempio di codice mostra come utilizzare`delete-bucket-analytics-configuration`.

AWS CLI

Come eliminare una configurazione di analisi per un bucket

L'esempio `delete-bucket-analytics-configuration` seguente rimuove la configurazione di analisi per il bucket e l'ID specificati.

```
aws s3api delete-bucket-analytics-configuration \
    --bucket amzn-s3-demo-bucket \
    --id 1
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta DeleteBucketAnalyticsConfiguration AWS CLICommand Reference.

`delete-bucket-metrics-configuration`

Il seguente esempio di codice mostra come utilizzare`delete-bucket-metrics-configuration`.

AWS CLI

Come eliminare una configurazione delle metriche per un bucket

L'esempio `delete-bucket-metrics-configuration` seguente rimuove la configurazione delle metriche per il bucket e l'ID specificati.

```
aws s3api delete-bucket-metrics-configuration \
    --bucket amzn-s3-demo-bucket \
    --id 123
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta DeleteBucketMetricsConfiguration AWS CLICommand Reference.

`delete-core-network`

Il seguente esempio di codice mostra come utilizzare`delete-core-network`.

AWS CLI

Come eliminare una rete principale

L'esempio `delete-core-network` seguente elimina una rete core da una rete globale Cloud WAN.

```
aws networkmanager delete-core-network \
    --core-network-id core-network-0fab62fe438d94db6
```

Output:

```
{
    "CoreNetwork": {
        "GlobalNetworkId": "global-network-0d59060f16a73bc41",
        "CoreNetworkId": "core-network-0fab62fe438d94db6",
        "Description": "Main headquarters location",
        "CreatedAt": "2021-12-09T18:31:11+00:00",
        "State": "DELETING",
        "Segments": [
            {
                "Name": "dev",
                "EdgeLocations": [
                    "us-east-1"
                ],
                "SharedSegments": []
            }
        ],
        "Edges": [
```

```
            {
                "EdgeLocation": "us-east-1",
                "Asn": 64512,
                "InsideCidrBlocks": []
            }
        ]
    }
}
```

Per ulteriori informazioni, consulta Reti di base nella Guida per l'utente di Cloud WAN.

- Per i dettagli sull'API, consulta DeleteCoreNetwork AWS CLICommand Reference.

`delete-device`

Il seguente esempio di codice mostra come utilizzare`delete-device`.

AWS CLI

Come eliminare un dispositivo

L'esempio `delete-device` seguente elimina il dispositivo specificato dalla rete globale specificata.

```
aws networkmanager delete-device \
    --global-network-id global-network-01231231231231231 \
    --device-id device-07f6fd08867abc123 \
    --region us-west-2
```

Output:

```
{
    "Device": {
        "DeviceId": "device-07f6fd08867abc123",
        "DeviceArn": "arn:aws:networkmanager::123456789012:device/global-
network-01231231231231231/device-07f6fd08867abc123",
        "GlobalNetworkId": "global-network-01231231231231231",
        "Description": "New York office device",
        "Type": "office device",
        "Vendor": "anycompany",
        "Model": "abcabc",
        "SerialNumber": "1234",
```

```
        "SiteId": "site-444555aaabbb11223",
        "CreatedAt": 1575554005.0,
        "State": "DELETING"
    }
}
```

Per ulteriori informazioni, consulta Utilizzo di dispositivi nella Guida di Transit Gateway network manager.

- Per i dettagli sull'API, consulta DeleteDevice AWS CLICommand Reference.

`delete-global-network`

Il seguente esempio di codice mostra come utilizzare`delete-global-network`.

AWS CLI

Come eliminare una rete globale

L'esempio `delete-global-network` seguente elimina una rete globale.

```
aws networkmanager delete-global-network \
    --global-network-id global-network-052bedddccb193b6b
```

Output:

```
{
    "GlobalNetwork": {
        "GlobalNetworkId": "global-network-052bedddccb193b6b",
        "GlobalNetworkArn": "arn:aws:networkmanager::987654321012:global-network/
global-network-052bedddccb193b6b",
        "CreatedAt": "2021-12-09T18:19:12+00:00",
        "State": "DELETING"
    }
}
```

- Per i dettagli sull'API, consulta DeleteGlobalNetwork AWS CLICommand Reference.

`delete-link`

Il seguente esempio di codice mostra come utilizzare`delete-link`.

AWS CLI

Come eliminare un link

L'esempio `delete-link` seguente elimina il link specificato dalla rete globale specificata.

```
aws networkmanager delete-link \
    --global-network-id global-network-01231231231231231  \
    --link-id link-11112222aaaabbbb1 \
    --region us-west-2
```

Output:

```
{
    "Link": {
        "LinkId": "link-11112222aaaabbbb1",
        "LinkArn": "arn:aws:networkmanager::123456789012:link/global-
network-01231231231231231/link-11112222aaaabbbb1",
        "GlobalNetworkId": "global-network-01231231231231231",
        "SiteId": "site-444555aaabbb11223",
        "Description": "VPN Link",
        "Type": "broadband",
        "Bandwidth": {
            "UploadSpeed": 20,
            "DownloadSpeed": 20
        },
        "Provider": "AnyCompany",
        "CreatedAt": 1575555811.0,
        "State": "DELETING"
    }
}
```

Per ulteriori informazioni, consulta Utilizzo di link nella Guida di Transit Gateway network manager.

- Per i dettagli sull'API, consulta DeleteLink AWS CLICommand Reference.

`delete-public-access-block`

Il seguente esempio di codice mostra come utilizzare`delete-public-access-block`.

AWS CLI

Come eliminare la configurazione di blocco dell'accesso pubblico per un bucket

L'esempio `delete-public-access-block` seguente rimuove la configurazione del blocco dell'accesso pubblico per il bucket e l'ID specificati.

```
aws s3api delete-public-access-block \
    --bucket amzn-s3-demo-bucket
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta DeletePublicAccessBlock AWS CLICommand Reference.

`delete-site`

Il seguente esempio di codice mostra come utilizzare`delete-site`.

AWS CLI

Come eliminare un sito

L'esempio `delete-site` seguente elimina il sito specificato (`site-444555aaabbb11223`) nella rete globale specificata.

```
aws networkmanager delete-site \
    --global-network-id global-network-01231231231231231  \
    --site-id site-444555aaabbb11223 \
    --region us-west-2
```

Output:

```
{
    "Site": {
        "SiteId": "site-444555aaabbb11223",
        "SiteArn": "arn:aws:networkmanager::123456789012:site/global-
network-01231231231231231/site-444555aaabbb11223",
        "GlobalNetworkId": "global-network-01231231231231231",
        "Description": "New York head office",
        "Location": {
            "Latitude": "40.7128",
            "Longitude": "-74.0060"
        },
```

```
        "CreatedAt": 1575554300.0,
        "State": "DELETING"
    }
}
```

Per ulteriori informazioni, consulta Utilizzo di siti nella Guida di Transit Gateway network manager.

- Per i dettagli sull'API, consulta DeleteSite AWS CLICommand Reference.

`deregister-transit-gateway`

Il seguente esempio di codice mostra come utilizzare`deregister-transit-gateway`.

AWS CLI

Come annullare la registrazione di un gateway di transito da una rete globale

L'esempio `deregister-transit-gateway` seguente annulla la registrazione del gateway di transito specificato nella rete globale specificata.

```
aws networkmanager deregister-transit-gateway \
    --global-network-id global-network-01231231231231231 \
    --transit-gateway-arn arn:aws:ec2:us-west-2:123456789012:transit-gateway/
tgw-123abc05e04123abc \
    --region us-west-2
```

Output:

```
{
    "TransitGatewayRegistration": {
        "GlobalNetworkId": "global-network-01231231231231231",
        "TransitGatewayArn": "arn:aws:ec2:us-west-2:123456789012:transit-gateway/
tgw-123abc05e04123abc",
        "State": {
            "Code": "DELETING"
        }
    }
}
```

Per ulteriori informazioni, consulta Registrazioni di Transit Gateway nella Guida di Transit Gateway network manager.

- Per i dettagli sull'API, consulta DeregisterTransitGateway AWS CLICommand Reference.

`describe-global-networks`

Il seguente esempio di codice mostra come utilizzare`describe-global-networks`.

AWS CLI

Come descrivere le reti globali

L'esempio `describe-global-networks` seguente descrive tutte le reti globali nell'account.

```
aws networkmanager describe-global-networks \
    --region us-west-2
```

Output:

```
{
    "GlobalNetworks": [
        {
            "GlobalNetworkId": "global-network-01231231231231231",
            "GlobalNetworkArn": "arn:aws:networkmanager::123456789012:global-
network/global-network-01231231231231231",
            "Description": "Company 1 global network",
            "CreatedAt": 1575553525.0,
            "State": "AVAILABLE"
        }
    ]
}
```

- Per i dettagli sull'API, consulta DescribeGlobalNetworks AWS CLICommand Reference.

`disassociate-customer-gateway`

Il seguente esempio di codice mostra come utilizzare`disassociate-customer-gateway`.

AWS CLI

Come dissociare un gateway del cliente

L'esempio `disassociate-customer-gateway` seguente dissocia il gateway del cliente specificato (`cgw-11223344556677889`) dalla rete globale specificata.

```
aws networkmanager disassociate-customer-gateway \
```

```
    --global-network-id global-network-01231231231231231 \
    --customer-gateway-arn arn:aws:ec2:us-west-2:123456789012:customer-gateway/
cgw-11223344556677889 \
    --region us-west-2
```

Output:

```
{
    "CustomerGatewayAssociation": {
        "CustomerGatewayArn": "arn:aws:ec2:us-west-2:123456789012:customer-gateway/
cgw-11223344556677889",
        "GlobalNetworkId": "global-network-01231231231231231",
        "DeviceId": "device-07f6fd08867abc123",
        "State": "DELETING"
    }
}
```

Per ulteriori informazioni, consulta Associazioni del gateway del cliente nella Guida di Transit Gateway network manager.

- Per i dettagli sull'API, consulta DisassociateCustomerGateway AWS CLICommand Reference.

`disassociate-link`

Il seguente esempio di codice mostra come utilizzare`disassociate-link`.

AWS CLI

Come dissociare un link

L'esempio `disassociate-link` seguente dissocia il link specificato dal dispositivo `device-07f6fd08867abc123` nella rete globale specificata.

```
aws networkmanager disassociate-link \
    --global-network-id global-network-01231231231231231 \
    --device-id device-07f6fd08867abc123 \
    --link-id link-11112222aaaabbbb1 \
    --region us-west-2
```

Output:

```
{
```

```
    "LinkAssociation": {
        "GlobalNetworkId": "global-network-01231231231231231",
        "DeviceId": "device-07f6fd08867abc123",
        "LinkId": "link-11112222aaaabbbb1",
        "LinkAssociationState": "DELETING"
    }
}
```

Per ulteriori informazioni, consulta Associazioni di dispositivi e link nella Guida di Transit Gateway network manager.

- Per i dettagli sull'API, consulta DisassociateLink AWS CLICommand Reference.

`get-bucket-analytics-configuration`

Il seguente esempio di codice mostra come utilizzare`get-bucket-analytics-configuration`.

AWS CLI

Come recuperare la configurazione di analisi per un bucket con un ID specifico

L'esempio `get-bucket-analytics-configuration` seguente visualizza la configurazione di analisi per il bucket e l'ID specificati.

```
aws s3api get-bucket-analytics-configuration \
    --bucket amzn-s3-demo-bucket \
    --id 1
```

Output:

```
{
    "AnalyticsConfiguration": {
        "StorageClassAnalysis": {},
        "Id": "1"
    }
}
```

- Per i dettagli sull'API, consulta GetBucketAnalyticsConfiguration AWS CLICommand Reference.

`get-bucket-metrics-configuration`

Il seguente esempio di codice mostra come utilizzare`get-bucket-metrics-configuration`.

AWS CLI

Come recuperare la configurazione delle metriche per un bucket con un ID specifico

L'esempio `get-bucket-metrics-configuration` seguente visualizza la configurazione delle metriche per il bucket e l'ID specificati.

```
aws s3api get-bucket-metrics-configuration \
    --bucket amzn-s3-demo-bucket \
    --id 123
```

Output:

```
{
    "MetricsConfiguration": {
        "Filter": {
            "Prefix": "logs"
        },
        "Id": "123"
    }
}
```

- Per i dettagli sull'API, consulta GetBucketMetricsConfiguration AWS CLICommand Reference.

`get-customer-gateway-associations`

Il seguente esempio di codice mostra come utilizzare`get-customer-gateway-associations`.

AWS CLI

Come ottenere le associazioni del gateway del cliente

L'esempio `get-customer-gateway-associations` seguente ottiene le associazioni del gateway del cliente per la rete globale specificata.

```
aws networkmanager get-customer-gateway-associations \
    --global-network-id global-network-01231231231231231 \
    --region us-west-2
```

Output:

```
{
```

```
    "CustomerGatewayAssociations": [
        {
            "CustomerGatewayArn": "arn:aws:ec2:us-west-2:123456789012:customer-
gateway/cgw-11223344556677889",
            "GlobalNetworkId": "global-network-01231231231231231",
            "DeviceId": "device-07f6fd08867abc123",
            "State": "AVAILABLE"
        }
    ]
}
```

- Per i dettagli sull'API, consulta GetCustomerGatewayAssociations AWS CLICommand Reference.

`get-devices`

Il seguente esempio di codice mostra come utilizzare`get-devices`.

AWS CLI

Come ottenere i tuoi dispositivi

L'esempio `get-devices` seguente ottiene i dispositivi nella rete globale specificata.

```
aws networkmanager get-devices \
    --global-network-id global-network-01231231231231231 \
    --region us-west-2
```

Output:

```
{
    "Devices": [
        {
            "DeviceId": "device-07f6fd08867abc123",
            "DeviceArn": "arn:aws:networkmanager::123456789012:device/global-
network-01231231231231231/device-07f6fd08867abc123",
            "GlobalNetworkId": "global-network-01231231231231231",
            "Description": "NY office device",
            "Type": "office device",
            "Vendor": "anycompany",
            "Model": "abcabc",
            "SerialNumber": "1234",
```

```
            "CreatedAt": 1575554005.0,
            "State": "AVAILABLE"
        }
    ]
}
```

- Per i dettagli sull'API, consulta GetDevices AWS CLICommand Reference.

`get-link-associations`

Il seguente esempio di codice mostra come utilizzare`get-link-associations`.

AWS CLI

Come ottenere le associazioni dei link

L'esempio `get-link-associations` seguente ottiene le associazioni dei link nella rete globale specificata.

```
aws networkmanager get-link-associations \
    --global-network-id global-network-01231231231231231 \
    --region us-west-2
```

Output:

```
{
    "LinkAssociations": [
        {
            "GlobalNetworkId": "global-network-01231231231231231",
            "DeviceId": "device-07f6fd08867abc123",
            "LinkId": "link-11112222aaaabbbb1",
            "LinkAssociationState": "AVAILABLE"
        }
    ]
}
```

- Per i dettagli sull'API, consulta GetLinkAssociations AWS CLICommand Reference.

`get-links`

Il seguente esempio di codice mostra come utilizzare`get-links`.

AWS CLI

Come ottenere i tuoi link

L'esempio `get-links` seguente ottiene i link nella rete globale specificata.

```
aws networkmanager get-links \
    --global-network-id global-network-01231231231231231 \
    --region us-west-2
```

Output:

```
{
    "Links": [
        {
            "LinkId": "link-11112222aaaabbbb1",
            "LinkArn": "arn:aws:networkmanager::123456789012:link/global-
network-01231231231231231/link-11112222aaaabbbb1",
            "GlobalNetworkId": "global-network-01231231231231231",
            "SiteId": "site-444555aaabbb11223",
            "Description": "VPN Link",
            "Type": "broadband",
            "Bandwidth": {
                "UploadSpeed": 10,
                "DownloadSpeed": 20
            },
            "Provider": "AnyCompany",
            "CreatedAt": 1575555811.0,
            "State": "AVAILABLE"
        }
    ]
}
```

- Per i dettagli sull'API, consulta GetLinks AWS CLICommand Reference.

`get-object-retention`

Il seguente esempio di codice mostra come utilizzare`get-object-retention`.

AWS CLI

Come recuperare la configurazione di conservazione degli oggetti per un oggetto

L'esempio `get-object-retention` seguente recupera la configurazione di conservazione degli oggetti per il bucket specificato.

```
aws s3api get-object-retention \
    --bucket amzn-s3-demo-bucket-with-object-lock \
    --key doc1.rtf
```

Output:

```
{
    "Retention": {
        "Mode": "GOVERNANCE",
        "RetainUntilDate": "2025-01-01T00:00:00.000Z"
    }
}
```

- Per i dettagli sull'API, consulta GetObjectRetention AWS CLICommand Reference.

`get-public-access-block`

Il seguente esempio di codice mostra come utilizzare`get-public-access-block`.

AWS CLI

Come impostare o modificare la configurazione di blocco dell'accesso pubblico per un bucket

L'esempio `get-public-access-block` seguente visualizza la configurazione del blocco dell'accesso pubblico per il bucket specificato.

```
aws s3api get-public-access-block --bucket amzn-s3-demo-bucket
```

Output:

```
{
    "PublicAccessBlockConfiguration": {
        "IgnorePublicAcls": true,
        "BlockPublicPolicy": true,
        "BlockPublicAcls": true,
        "RestrictPublicBuckets": true
    }
}
```

- Per i dettagli sull'API, consulta GetPublicAccessBlock AWS CLICommand Reference.

`get-sites`

Il seguente esempio di codice mostra come utilizzare`get-sites`.

AWS CLI

Come ottenere i tuoi siti

L'esempio `get-sites` seguente ottiene i siti nella rete globale specificata.

```
aws networkmanager get-sites \
    --global-network-id global-network-01231231231231231 \
    --region us-west-2
```

Output:

```
{
    "Sites": [
        {
            "SiteId": "site-444555aaabbb11223",
            "SiteArn": "arn:aws:networkmanager::123456789012:site/global-
network-01231231231231231/site-444555aaabbb11223",
            "GlobalNetworkId": "global-network-01231231231231231",
            "Description": "NY head office",
            "Location": {
                "Latitude": "40.7128",
                "Longitude": "-74.0060"
            },
            "CreatedAt": 1575554528.0,
            "State": "AVAILABLE"
        }
    ]
}
```

- Per i dettagli sull'API, consulta GetSites AWS CLICommand Reference.

`get-transit-gateway-registrations`

Il seguente esempio di codice mostra come utilizzare`get-transit-gateway-registrations`.

AWS CLI

Come ottenere le registrazioni dei gateway di transito

L'esempio `get-transit-gateway-registrations` seguente ottiene i gateway di transito registrati nella rete globale specificata.

```
aws networkmanager get-transit-gateway-registrations \
    --global-network-id global-network-01231231231231231 \
    --region us-west-2
```

Output:

```
{
    "TransitGatewayRegistrations": [
        {
            "GlobalNetworkId": "global-network-01231231231231231",
            "TransitGatewayArn": "arn:aws:ec2:us-west-2:123456789012:transit-
gateway/tgw-123abc05e04123abc",
            "State": {
                "Code": "AVAILABLE"
            }
        }
    ]
}
```

- Per i dettagli sull'API, consulta GetTransitGatewayRegistrations AWS CLICommand Reference.

`get-vpc-attachment`

Il seguente esempio di codice mostra come utilizzare`get-vpc-attachment`.

AWS CLI

Per ottenere un allegato VPC

L'esempio `get-vpc-attachment` seguente restituisce informazioni su un collegamento VPC.

```
aws networkmanager get-vpc-attachment \
    --attachment-id  attachment-03b7ea450134787da
```

Output:

```
{
    "VpcAttachment": {
        "Attachment": {
            "CoreNetworkId": "core-network-0522de1b226a5d7b3",
            "AttachmentId": "attachment-03b7ea450134787da",
            "OwnerAccountId": "987654321012",
            "AttachmentType": "VPC",
            "State": "CREATING",
            "EdgeLocation": "us-east-1",
            "ResourceArn": "arn:aws:ec2:us-east-1:987654321012:vpc/vpc-a7c4bbda",
            "Tags": [
                {
                    "Key": "Name",
                    "Value": "DevVPC"
                }
            ],
            "CreatedAt": "2022-03-11T17:48:58+00:00",
            "UpdatedAt": "2022-03-11T17:48:58+00:00"
        },
        "SubnetArns": [
            "arn:aws:ec2:us-east-1:987654321012:subnet/subnet-202cde6c",
            "arn:aws:ec2:us-east-1:987654321012:subnet/subnet-e5022dba",
            "arn:aws:ec2:us-east-1:987654321012:subnet/subnet-2387ae02",
            "arn:aws:ec2:us-east-1:987654321012:subnet/subnet-cda9dffc"
        ],
        "Options": {
            "Ipv6Support": false
        }
    }
}
```

Per ulteriori informazioni, consulta Collegamenti nella Guida per l'utente di Cloud WAN.

- Per i dettagli sull'API, consulta GetVpcAttachment AWS CLICommand Reference.

`list-bucket-analytics-configurations`

Il seguente esempio di codice mostra come utilizzare`list-bucket-analytics-configurations`.

AWS CLI

Come recuperare un elenco di configurazioni di analisi per un bucket

L'esempio `list-bucket-analytics-configurations` seguente recupera un elenco di configurazioni di analisi per il bucket specificato.

```
aws s3api list-bucket-analytics-configurations \
    --bucket amzn-s3-demo-bucket
```

Output:

```
{
    "AnalyticsConfigurationList": [
        {
            "StorageClassAnalysis": {},
            "Id": "1"
        }
    ],
    "IsTruncated": false
}
```

- Per i dettagli sull'API, consulta ListBucketAnalyticsConfigurations AWS CLICommand Reference.

`list-bucket-metrics-configurations`

Il seguente esempio di codice mostra come utilizzare`list-bucket-metrics-configurations`.

AWS CLI

Come recuperare un elenco di configurazioni di analisi per un bucket

L'esempio `list-bucket-metrics-configurations` seguente recupera un elenco di configurazioni di metriche per il bucket specificato.

```
aws s3api list-bucket-metrics-configurations \
    --bucket amzn-s3-demo-bucket
```

Output:

```
{
    "IsTruncated": false,
```

```
    "MetricsConfigurationList": [
        {
            "Filter": {
                "Prefix": "logs"
            },
            "Id": "123"
        },
        {
            "Filter": {
                "Prefix": "tmp"
            },
            "Id": "234"
        }
    ]
}
```

- Per i dettagli sull'API, consulta ListBucketMetricsConfigurations AWS CLICommand Reference.

`list-tags-for-resource`

Il seguente esempio di codice mostra come utilizzare`list-tags-for-resource`.

AWS CLI

Come elencare i tag per una risorsa

L'esempio `list-tags-for-resource` seguente elenca i tag per la risorsa dispositivo specificata (`device-07f6fd08867abc123`).

```
aws networkmanager list-tags-for-resource \
    --resource-arn arn:aws:networkmanager::123456789012:device/global-
network-01231231231231231/device-07f6fd08867abc123 \
    --region us-west-2
```

Output:

```
{
    "TagList": [
        {
            "Key": "Network",
            "Value": "Northeast"
```

```
        }
    ]
}
```

- Per i dettagli sull'API, consulta ListTagsForResource AWS CLICommand Reference.

`put-bucket-metrics-configuration`

Il seguente esempio di codice mostra come utilizzare`put-bucket-metrics-configuration`.

AWS CLI

Come eliminare una configurazione delle metriche per un bucket

L'esempio `put-bucket-metrics-configuration` seguente imposta una configurazione della metrica con ID 123 per il bucket specificato.

```
aws s3api put-bucket-metrics-configuration \
    --bucket amzn-s3-demo-bucket \
    --id 123 \
    --metrics-configuration '{"Id": "123", "Filter": {"Prefix": "logs"}}'
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta PutBucketMetricsConfiguration AWS CLICommand Reference.

`put-object-retention`

Il seguente esempio di codice mostra come utilizzare`put-object-retention`.

AWS CLI

Come impostare la configurazione di conservazione degli oggetti per un oggetto

L'esempio `put-object-retention` seguente imposta una configurazione di conservazione degli oggetti per l'oggetto specificato fino al giorno 01/01/2025.

```
aws s3api put-object-retention \
    --bucket amzn-s3-demo-bucket-with-object-lock \
    --key doc1.rtf \
```

```
    --retention '{ "Mode": "GOVERNANCE", "RetainUntilDate": "2025-01-01T00:00:00" }'
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta PutObjectRetention AWS CLICommand Reference.

`put-public-access-block`

Il seguente esempio di codice mostra come utilizzare`put-public-access-block`.

AWS CLI

Come impostare la configurazione di blocco dell'accesso pubblico per un bucket

L'esempio `put-public-access-block` seguente imposta una configurazione di blocco dell'accesso pubblico restrittiva per il bucket specificato.

```
aws s3api put-public-access-block \
    --bucket amzn-s3-demo-bucket \
    --public-access-block-
configuration "BlockPublicAcls=true,IgnorePublicAcls=true,BlockPublicPolicy=true,RestrictPub
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta PutPublicAccessBlock AWS CLICommand Reference.

`register-transit-gateway`

Il seguente esempio di codice mostra come utilizzare`register-transit-gateway`.

AWS CLI

Come registrare un gateway di transito in una rete globale

L'esempio `register-transit-gateway` seguente registra il gateway di transito `tgw-123abc05e04123abc` nella rete globale specificata.

```
aws networkmanager register-transit-gateway \
    --global-network-id global-network-01231231231231231 \
    --transit-gateway-arn arn:aws:ec2:us-west-2:123456789012:transit-gateway/
tgw-123abc05e04123abc \
```

```
    --region us-west-2
```

Output:

```
{
    "TransitGatewayRegistration": {
        "GlobalNetworkId": "global-network-01231231231231231",
        "TransitGatewayArn": "arn:aws:ec2:us-west-2:123456789012:transit-gateway/
tgw-123abc05e04123abc",
        "State": {
            "Code": "PENDING"
        }
    }
}
```

Per ulteriori informazioni, consulta Registrazioni di Transit Gateway nella Guida di Transit Gateway network manager.

- Per i dettagli sull'API, consulta RegisterTransitGateway AWS CLICommand Reference.

`reject-attachment`

Il seguente esempio di codice mostra come utilizzare`reject-attachment`.

AWS CLI

Come rifiutare un collegamento

L'esempio `reject-attachment` seguente rifiuta una richiesta di collegamento VPC.

```
aws networkmanager reject-attachment \
    --attachment-id  attachment-03b7ea450134787da
```

Output:

```
{
    "Attachment": {
        "CoreNetworkId": "core-network-0522de1b226a5d7b3",
        "AttachmentId": "attachment-03b7ea450134787da",
        "OwnerAccountId": "987654321012",
        "AttachmentType": "VPC",
```

```
        "State": "AVAILABLE",
        "EdgeLocation": "us-east-1",
        "ResourceArn": "arn:aws:ec2:us-east-1:987654321012:vpc/vpc-a7c4bbda",
        "CreatedAt": "2022-03-11T17:48:58+00:00",
        "UpdatedAt": "2022-03-11T17:51:25+00:00"
    }
}
```

Per ulteriori informazioni, consulta Accettazione dei collegamenti nella Guida per l'utente di Cloud WAN.

- Per i dettagli sull'API, consulta RejectAttachment AWS CLICommand Reference.

`start-route-analysis`

Il seguente esempio di codice mostra come utilizzare`start-route-analysis`.

AWS CLI

Come avviare l'analisi delle route

L'esempio `start-route-analysis` seguente avvia l'analisi tra un'origine e una destinazione, inclusa l'opzione `include-return-path` facoltativa.

```
aws networkmanager start-route-analysis \
    --global-network-id global-network-00aa0aaa0b0aaa000 \
    --source TransitGatewayAttachmentArn=arn:aws:ec2:us-east-1:503089527312:transit-gateway-attachment/tgw-attach-0d4a2d491bf68c093,IpAddress=10.0.0.0 \
    --destination TransitGatewayAttachmentArn=arn:aws:ec2:us-west-1:503089527312:transit-gateway-attachment/tgw-attach-002577f30bb181742,IpAddress=11.0.0.0 \
    --include-return-path
```

Output:

```
{
    "RouteAnalysis": {
        "GlobalNetworkId": "global-network-00aa0aaa0b0aaa000
        "OwnerAccountId": "1111222233333",
        "RouteAnalysisId": "a1873de1-273c-470c-1a2bc2345678",
        "StartTimestamp": 1695760154.0,
        "Status": "RUNNING",
```

```
        "Source": {
            "TransitGatewayAttachmentArn": "arn:aws:ec2:us-
east-1:111122223333:transit-gateway-attachment/tgw-attach-1234567890abcdef0,
            "TransitGatewayArn": "arn:aws:ec2:us-east-1:111122223333:transit-
gateway/tgw-abcdef01234567890",
            "IpAddress": "10.0.0.0"
        },
        "Destination": {
            "TransitGatewayAttachmentArn": "arn:aws:ec2:us-
west-1:555555555555:transit-gateway-attachment/tgw-attach-021345abcdef6789",
            "TransitGatewayArn": "arn:aws:ec2:us-west-1:111122223333:transit-
gateway/tgw-09876543210fedcba0",
            "IpAddress": "11.0.0.0"
        },
        "IncludeReturnPath": true,
        "UseMiddleboxes": false
    }
}
```

Per ulteriori informazioni, consulta Strumento di analisi delle route nella Guida per l'utente di Reti globali AWS per gateway di transito.

- Per i dettagli sull'API, consulta StartRouteAnalysis AWS CLICommand Reference.

`tag-resource`

Il seguente esempio di codice mostra come utilizzare`tag-resource`.

AWS CLI

Come applicare tag a una risorsa

L'esempio `tag-resource` seguente applica il tag `Network=Northeast` al dispositivo `device-07f6fd08867abc123`.

```
aws networkmanager tag-resource \
    --resource-arn arn:aws:networkmanager::123456789012:device/global-
network-0123123123123123/device-07f6fd08867abc123 \
    --tags Key=Network,Value=Northeast \
    --region us-west-2
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta TagResource AWS CLICommand Reference.

`untag-resource`

Il seguente esempio di codice mostra come utilizzare`untag-resource`.

AWS CLI

Come rimuovere tag da una risorsa

L'esempio `untag-resource` seguente rimuove il tag con la chiave `Network` dal dispositivo `device-07f6fd08867abc123`.

```
aws networkmanager untag-resource \
    --resource-arn arn:aws:networkmanager::123456789012:device/global-network-01231231231231231/device-07f6fd08867abc123 ]
    --tag-keys Network \
    --region us-west-2
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta UntagResource AWS CLICommand Reference.

`update-device`

Il seguente esempio di codice mostra come utilizzare`update-device`.

AWS CLI

Come aggiornare un dispositivo

L'esempio `update-device` seguente aggiorna il dispositivo `device-07f6fd08867abc123` specificando un ID del sito per il dispositivo.

```
aws networkmanager update-device \
    --global-network-id global-network-01231231231231231 \
    --device-id device-07f6fd08867abc123 \
    --site-id site-444555aaabbb11223 \
    --region us-west-2
```

Output:

```
{
    "Device": {
        "DeviceId": "device-07f6fd08867abc123",
        "DeviceArn": "arn:aws:networkmanager::123456789012:device/global-
network-01231231231231231/device-07f6fd08867abc123",
        "GlobalNetworkId": "global-network-01231231231231231",
        "Description": "NY office device",
        "Type": "Office device",
        "Vendor": "anycompany",
        "Model": "abcabc",
        "SerialNumber": "1234",
        "SiteId": "site-444555aaabbb11223",
        "CreatedAt": 1575554005.0,
        "State": "UPDATING"
    }
}
```

Per ulteriori informazioni, consulta Utilizzo di dispositivi nella Guida di Transit Gateway network manager.

- Per i dettagli sull'API, consulta UpdateDevice AWS CLICommand Reference.

`update-global-network`

Il seguente esempio di codice mostra come utilizzare`update-global-network`.

AWS CLI

Come aggiornare una rete globale

L'esempio `update-global-network` seguente aggiorna la descrizione per la rete globale `global-network-01231231231231231`.

```
aws networkmanager update-global-network \
    --global-network-id global-network-01231231231231231 \
    --description "Head offices" \
    --region us-west-2
```

Output:

```
{
```

```
    "GlobalNetwork": {
        "GlobalNetworkId": "global-network-01231231231231231",
        "GlobalNetworkArn": "arn:aws:networkmanager::123456789012:global-network/
global-network-01231231231231231",
        "Description": "Head offices",
        "CreatedAt": 1575553525.0,
        "State": "UPDATING"
    }
}
```

Per ulteriori informazioni, consulta Reti globali nella Guida di Transit Gateway network manager.

- Per i dettagli sull'API, consulta UpdateGlobalNetwork AWS CLICommand Reference.

`update-link`

Il seguente esempio di codice mostra come utilizzare`update-link`.

AWS CLI

Come aggiornare un link

L'esempio `update-link` seguente aggiorna le informazioni sulla larghezza di banda per il link `link-11112222aaaabbbb1`.

```
aws networkmanager update-link \
    --global-network-id global-network-01231231231231231 \
    --link-id link-11112222aaaabbbb1 \
    --bandwidth  UploadSpeed=20,DownloadSpeed=20 \
    --region us-west-2
```

Output:

```
{
    "Link": {
        "LinkId": "link-11112222aaaabbbb1",
        "LinkArn": "arn:aws:networkmanager::123456789012:link/global-
network-01231231231231231/link-11112222aaaabbbb1",
        "GlobalNetworkId": "global-network-01231231231231231",
        "SiteId": "site-444555aaabbb11223",
        "Description": "VPN Link",
        "Type": "broadband",
```

```
        "Bandwidth": {
            "UploadSpeed": 20,
            "DownloadSpeed": 20
        },
        "Provider": "AnyCompany",
        "CreatedAt": 1575555811.0,
        "State": "UPDATING"
    }
}
```

Per ulteriori informazioni, consulta Utilizzo di link nella Guida di Transit Gateway network manager.

- Per i dettagli sull'API, consulta UpdateLink AWS CLICommand Reference.

`update-site`

Il seguente esempio di codice mostra come utilizzare`update-site`.

AWS CLI

Come aggiornare un sito

L'esempio `update-site` seguente aggiorna la descrizione per il sito `site-444555aaabbb11223` nella rete globale specificata.

```
aws networkmanager update-site \
    --global-network-id global-network-01231231231231231 \
    --site-id site-444555aaabbb11223 \
    --description "New York Office site" \
    --region us-west-2
```

Output:

```
{
    "Site": {
        "SiteId": "site-444555aaabbb11223",
        "SiteArn": "arn:aws:networkmanager::123456789012:site/global-
network-01231231231231231/site-444555aaabbb11223",
        "GlobalNetworkId": "global-network-01231231231231231",
        "Description": "New York Office site",
        "Location": {
```

```
            "Latitude": "40.7128",
            "Longitude": "-74.0060"
        },
        "CreatedAt": 1575554528.0,
        "State": "UPDATING"
    }
}
```

Per ulteriori informazioni, consulta Utilizzo di siti nella Guida di Transit Gateway network manager.

- Per i dettagli sull'API, consulta UpdateSite AWS CLICommand Reference.

OpenSearch Esempi di servizi che utilizzano AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with OpenSearch Service.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`create-elasticsearch-domain`

Il seguente esempio di codice mostra come utilizzare`create-elasticsearch-domain`.

AWS CLI

Come creare un dominio Amazon Elasticsearch Service

Il comando `create-elasticsearch-domain` seguente crea un nuovo dominio Amazon Elasticsearch Service all'interno di un VPC e limita l'accesso a un singolo utente. Amazon ES deduce l'ID VPC dalla sottorete e dal gruppo di sicurezza specificati. IDs

```
aws es create-elasticsearch-domain \
    --domain-name vpc-cli-example \
    --elasticsearch-version 6.2 \
    --elasticsearch-cluster-
config InstanceType=m4.large.elasticsearch,InstanceCount=1 \
    --ebs-options EBSEnabled=true,VolumeType=standard,VolumeSize=10 \
    --access-policies '{"Version":"2012-10-17",        "Statement": [ { "Effect":
 "Allow", "Principal": {"AWS": "arn:aws:iam::123456789012:root" }, "Action":"es:*",
 "Resource": "arn:aws:es:us-west-1:123456789012:domain/vpc-cli-example/*" } ] }' \
    --vpc-options SubnetIds=subnet-1a2a3a4a,SecurityGroupIds=sg-2a3a4a5a
```

Output:

```
{
    "DomainStatus": {
        "ElasticsearchClusterConfig": {
            "DedicatedMasterEnabled": false,
            "InstanceCount": 1,
            "ZoneAwarenessEnabled": false,
            "InstanceType": "m4.large.elasticsearch"
        },
        "DomainId": "123456789012/vpc-cli-example",
        "CognitoOptions": {
            "Enabled": false
        },
        "VPCOptions": {
            "SubnetIds": [
                "subnet-1a2a3a4a"
            ],
            "VPCId": "vpc-3a4a5a6a",
            "SecurityGroupIds": [
                "sg-2a3a4a5a"
            ],
            "AvailabilityZones": [
                "us-west-1c"
            ]
        },
        "Created": true,
        "Deleted": false,
        "EBSOptions": {
            "VolumeSize": 10,
            "VolumeType": "standard",
            "EBSEnabled": true
```

```
        },
        "Processing": true,
        "DomainName": "vpc-cli-example",
        "SnapshotOptions": {
            "AutomatedSnapshotStartHour": 0
        },
        "ElasticsearchVersion": "6.2",
        "AccessPolicies": "{\"Version\":\"2012-10-17\",\"Statement\":[{\"Effect\":
\"Allow\",\"Principal\":{\"AWS\":\"arn:aws:iam::123456789012:root\"},\"Action\":
\"es:*\",\"Resource\":\"arn:aws:es:us-west-1:123456789012:domain/vpc-cli-example/*
\"}]}",
        "AdvancedOptions": {
            "rest.action.multi.allow_explicit_index": "true"
        },
        "EncryptionAtRestOptions": {
            "Enabled": false
        },
        "ARN": "arn:aws:es:us-west-1:123456789012:domain/vpc-cli-example"
    }
}
```

Per ulteriori informazioni, consulta Creazione e gestione dei domini di Amazon Elasticsearch Service nella Guida per gli sviluppatori di Amazon Elasticsearch Service.

- Per i dettagli sull'API, consulta Command CreateElasticsearchDomainReference AWS CLI .

`describe-elasticsearch-domain-config`

Il seguente esempio di codice mostra come utilizzare`describe-elasticsearch-domain-config`.

AWS CLI

Come ottenere i dettagli sulla configurazione del dominio

L'esempio `describe-elasticsearch-domain-config` seguente fornisce i dettagli di configurazione per un determinato dominio, insieme alle informazioni sullo stato per ogni singolo componente del dominio.

```
aws es describe-elasticsearch-domain-config \
    --domain-name cli-example
```

Output:


```
{
    "DomainConfig": {
        "ElasticsearchVersion": {
            "Options": "7.4",
            "Status": {
                "CreationDate": 1589395034.946,
                "UpdateDate": 1589395827.325,
                "UpdateVersion": 8,
                "State": "Active",
                "PendingDeletion": false
            }
        },
        "ElasticsearchClusterConfig": {
            "Options": {
                "InstanceType": "c5.large.elasticsearch",
                "InstanceCount": 1,
                "DedicatedMasterEnabled": true,
                "ZoneAwarenessEnabled": false,
                "DedicatedMasterType": "c5.large.elasticsearch",
                "DedicatedMasterCount": 3,
                "WarmEnabled": true,
                "WarmType": "ultrawarm1.medium.elasticsearch",
                "WarmCount": 2
            },
            "Status": {
                "CreationDate": 1589395034.946,
                "UpdateDate": 1589395827.325,
                "UpdateVersion": 8,
                "State": "Active",
                "PendingDeletion": false
            }
        },
        "EBSOptions": {
            "Options": {
                "EBSEnabled": true,
                "VolumeType": "gp2",
                "VolumeSize": 10
            },
            "Status": {
                "CreationDate": 1589395034.946,
                "UpdateDate": 1589395827.325,
                "UpdateVersion": 8,
```

```
                "State": "Active",
                "PendingDeletion": false
            }
        },
        "AccessPolicies": {
            "Options": "{\"Version\":\"2012-10-17\",\"Statement\":[{\"Effect
\":\"Allow\",\"Principal\":{\"AWS\":\"*\"},\"Action\":\"es:*\",\"Resource\":
\"arn:aws:es:us-east-1:123456789012:domain/cli-example/*\"}]}",
            "Status": {
                "CreationDate": 1589395034.946,
                "UpdateDate": 1589395827.325,
                "UpdateVersion": 8,
                "State": "Active",
                "PendingDeletion": false
            }
        },
        "SnapshotOptions": {
            "Options": {
                "AutomatedSnapshotStartHour": 0
            },
            "Status": {
                "CreationDate": 1589395034.946,
                "UpdateDate": 1589395827.325,
                "UpdateVersion": 8,
                "State": "Active",
                "PendingDeletion": false
            }
        },
        "VPCOptions": {
            "Options": {},
            "Status": {
                "CreationDate": 1591210426.162,
                "UpdateDate": 1591210426.162,
                "UpdateVersion": 18,
                "State": "Active",
                "PendingDeletion": false
            }
        },
        "CognitoOptions": {
            "Options": {
                "Enabled": false
            },
            "Status": {
                "CreationDate": 1591210426.163,
```

```
                "UpdateDate": 1591210426.163,
                "UpdateVersion": 18,
                "State": "Active",
                "PendingDeletion": false
            }
        },
        "EncryptionAtRestOptions": {
            "Options": {
                "Enabled": true,
                "KmsKeyId": "arn:aws:kms:us-
east-1:123456789012:key/1a2a3a4a-1a2a-1a2a-1a2a-1a2a3a4a5a6a"
            },
            "Status": {
                "CreationDate": 1589395034.946,
                "UpdateDate": 1589395827.325,
                "UpdateVersion": 8,
                "State": "Active",
                "PendingDeletion": false
            }
        },
        "NodeToNodeEncryptionOptions": {
            "Options": {
                "Enabled": true
            },
            "Status": {
                "CreationDate": 1589395034.946,
                "UpdateDate": 1589395827.325,
                "UpdateVersion": 8,
                "State": "Active",
                "PendingDeletion": false
            }
        },
        "AdvancedOptions": {
            "Options": {
                "rest.action.multi.allow_explicit_index": "true"
            },
            "Status": {
                "CreationDate": 1589395034.946,
                "UpdateDate": 1589395827.325,
                "UpdateVersion": 8,
                "State": "Active",
                "PendingDeletion": false
            }
        },
```

```
        "LogPublishingOptions": {
            "Options": {},
            "Status": {
                "CreationDate": 1591210426.164,
                "UpdateDate": 1591210426.164,
                "UpdateVersion": 18,
                "State": "Active",
                "PendingDeletion": false
            }
        },
        "DomainEndpointOptions": {
            "Options": {
                "EnforceHTTPS": true,
                "TLSSecurityPolicy": "Policy-Min-TLS-1-0-2019-07"
            },
            "Status": {
                "CreationDate": 1589395034.946,
                "UpdateDate": 1589395827.325,
                "UpdateVersion": 8,
                "State": "Active",
                "PendingDeletion": false
            }
        },
        "AdvancedSecurityOptions": {
            "Options": {
                "Enabled": true,
                "InternalUserDatabaseEnabled": true
            },
            "Status": {
                "CreationDate": 1589395034.946,
                "UpdateDate": 1589827485.577,
                "UpdateVersion": 14,
                "State": "Active",
                "PendingDeletion": false
            }
        }
    }
}
```


Per ulteriori informazioni, consulta Creazione e gestione dei domini di Amazon Elasticsearch Service nella Guida per gli sviluppatori di Amazon Elasticsearch Service.

- Per i dettagli sull'API, consulta DescribeElasticsearchDomainConfig AWS CLICommand Reference.

`describe-elasticsearch-domain`

Il seguente esempio di codice mostra come utilizzare`describe-elasticsearch-domain`.

AWS CLI

Come ottenere i dettagli per un singolo dominio

L'esempio `describe-elasticsearch-domain` seguente fornisce i dettagli di configurazione per un determinato dominio.

```
aws es describe-elasticsearch-domain \
    --domain-name cli-example
```

Output:

```
{
    "DomainStatus": {
        "DomainId": "123456789012/cli-example",
        "DomainName": "cli-example",
        "ARN": "arn:aws:es:us-east-1:123456789012:domain/cli-example",
        "Created": true,
        "Deleted": false,
        "Endpoint": "search-cli-example-1a2a3a4a5a6a7a8a9a0a.us-
east-1.es.amazonaws.com",
        "Processing": false,
        "UpgradeProcessing": false,
        "ElasticsearchVersion": "7.4",
        "ElasticsearchClusterConfig": {
            "InstanceType": "c5.large.elasticsearch",
            "InstanceCount": 1,
            "DedicatedMasterEnabled": true,
            "ZoneAwarenessEnabled": false,
            "DedicatedMasterType": "c5.large.elasticsearch",
            "DedicatedMasterCount": 3,
            "WarmEnabled": true,
            "WarmType": "ultrawarm1.medium.elasticsearch",
            "WarmCount": 2
        },
        "EBSOptions": {
            "EBSEnabled": true,
            "VolumeType": "gp2",
            "VolumeSize": 10
```

```
        },
        "AccessPolicies": "{\"Version\":\"2012-10-17\",\"Statement\":[{\"Effect
\":\"Allow\",\"Principal\":{\"AWS\":\"*\"},\"Action\":\"es:*\",\"Resource\":
\"arn:aws:es:us-east-1:123456789012:domain/cli-example/*\"}]}",
        "SnapshotOptions": {
            "AutomatedSnapshotStartHour": 0
        },
        "CognitoOptions": {
            "Enabled": false
        },
        "EncryptionAtRestOptions": {
            "Enabled": true,
            "KmsKeyId": "arn:aws:kms:us-
east-1:123456789012:key/1a2a3a4a-1a2a-1a2a-1a2a-1a2a3a4a5a6a"
        },
        "NodeToNodeEncryptionOptions": {
            "Enabled": true
        },
        "AdvancedOptions": {
            "rest.action.multi.allow_explicit_index": "true"
        },
        "ServiceSoftwareOptions": {
            "CurrentVersion": "R20200522",
            "NewVersion": "",
            "UpdateAvailable": false,
            "Cancellable": false,
            "UpdateStatus": "COMPLETED",
            "Description": "There is no software update available for this domain.",
            "AutomatedUpdateDate": 0.0
        },
        "DomainEndpointOptions": {
            "EnforceHTTPS": true,
            "TLSSecurityPolicy": "Policy-Min-TLS-1-0-2019-07"
        },
        "AdvancedSecurityOptions": {
            "Enabled": true,
            "InternalUserDatabaseEnabled": true
        }
    }
}
```

Per ulteriori informazioni, consulta Creazione e gestione dei domini di Amazon Elasticsearch Service nella Guida per gli sviluppatori di Amazon Elasticsearch Service.

- Per i dettagli sull'API, consulta DescribeElasticsearchDomain AWS CLICommand Reference.

`describe-elasticsearch-domains`

Il seguente esempio di codice mostra come utilizzare`describe-elasticsearch-domains`.

AWS CLI

Come ottenere dettagli per uno o più domini

L'esempio `describe-elasticsearch-domains` seguente fornisce i dettagli di configurazione per uno o più domini.

```
aws es describe-elasticsearch-domains \
    --domain-names cli-example-1 cli-example-2
```

Output:

```
{
    "DomainStatusList": [{
            "DomainId": "123456789012/cli-example-1",
            "DomainName": "cli-example-1",
            "ARN": "arn:aws:es:us-east-1:123456789012:domain/cli-example-1",
            "Created": true,
            "Deleted": false,
            "Endpoint": "search-cli-example-1-1a2a3a4a5a6a7a8a9a0a.us-
east-1.es.amazonaws.com",
            "Processing": false,
            "UpgradeProcessing": false,
            "ElasticsearchVersion": "7.4",
            "ElasticsearchClusterConfig": {
                "InstanceType": "c5.large.elasticsearch",
                "InstanceCount": 1,
                 "DedicatedMasterEnabled": true,
                "ZoneAwarenessEnabled": false,
                "DedicatedMasterType": "c5.large.elasticsearch",
                "DedicatedMasterCount": 3,
                "WarmEnabled": true,
                "WarmType": "ultrawarm1.medium.elasticsearch",
                "WarmCount": 2
            },
            "EBSOptions": {
                "EBSEnabled": true,
```

```
                "VolumeType": "gp2",
                "VolumeSize": 10
            },
            "AccessPolicies": "{\"Version\":\"2012-10-17\",\"Statement\":[{\"Effect
\":\"Allow\",\"Principal\":{\"AWS\":\"*\"},\"Action\":\"es:*\",\"Resource\":
\"arn:aws:es:us-east-1:123456789012:domain/cli-example-1/*\"}]}",
            "SnapshotOptions": {
                "AutomatedSnapshotStartHour": 0
            },
            "CognitoOptions": {
                "Enabled": false
            },
            "EncryptionAtRestOptions": {
                "Enabled": true,
                "KmsKeyId": "arn:aws:kms:us-
east-1:123456789012:key/1a2a3a4a-1a2a-1a2a-1a2a-1a2a3a4a5a6a"
            },
            "NodeToNodeEncryptionOptions": {
                "Enabled": true
            },
            "AdvancedOptions": {
                "rest.action.multi.allow_explicit_index": "true"
            },
            "ServiceSoftwareOptions": {
                "CurrentVersion": "R20200522",
                "NewVersion": "",
                "UpdateAvailable": false,
                "Cancellable": false,
                "UpdateStatus": "COMPLETED",
                "Description": "There is no software update available for this
 domain.",
                "AutomatedUpdateDate": 0.0
            },
            "DomainEndpointOptions": {
                "EnforceHTTPS": true,
                "TLSSecurityPolicy": "Policy-Min-TLS-1-0-2019-07"
            },
            "AdvancedSecurityOptions": {
                "Enabled": true,
                "InternalUserDatabaseEnabled": true
            }
        },
        {
            "DomainId": "123456789012/cli-example-2",
```

```
            "DomainName": "cli-example-2",
            "ARN": "arn:aws:es:us-east-1:123456789012:domain/cli-example-2",
            "Created": true,
            "Deleted": false,
            "Processing": true,
            "UpgradeProcessing": false,
            "ElasticsearchVersion": "7.4",
            "ElasticsearchClusterConfig": {
                "InstanceType": "r5.large.elasticsearch",
                "InstanceCount": 1,
                "DedicatedMasterEnabled": false,
                "ZoneAwarenessEnabled": false,
                "WarmEnabled": false
            },
            "EBSOptions": {
                "EBSEnabled": true,
                "VolumeType": "gp2",
                "VolumeSize": 10
            },
            "AccessPolicies": "{\"Version\":\"2012-10-17\",\"Statement\":[{\"Effect
\":\"Deny\",\"Principal\":{\"AWS\":\"*\"},\"Action\":\"es:*\",\"Resource\":
\"arn:aws:es:us-east-1:123456789012:domain/cli-example-2/*\"}]}",
            "SnapshotOptions": {
                "AutomatedSnapshotStartHour": 0
            },
            "CognitoOptions": {
                "Enabled": false
            },
            "EncryptionAtRestOptions": {
                "Enabled": false
            },
            "NodeToNodeEncryptionOptions": {
                "Enabled": false
            },
            "AdvancedOptions": {
                "rest.action.multi.allow_explicit_index": "true"
            },
            "ServiceSoftwareOptions": {
                "CurrentVersion": "",
                "NewVersion": "",
                "UpdateAvailable": false,
                "Cancellable": false,
                "UpdateStatus": "COMPLETED",
```

```
                "Description": "There is no software update available for this
 domain.",
                "AutomatedUpdateDate": 0.0
            },
            "DomainEndpointOptions": {
                "EnforceHTTPS": false,
                "TLSSecurityPolicy": "Policy-Min-TLS-1-0-2019-07"
            },
            "AdvancedSecurityOptions": {
                "Enabled": false,
                "InternalUserDatabaseEnabled": false
            }
        }
    ]
}
```

Per ulteriori informazioni, consulta Creazione e gestione dei domini di Amazon Elasticsearch Service nella Guida per gli sviluppatori di Amazon Elasticsearch Service.

- Per i dettagli sull'API, consulta DescribeElasticsearchDomains AWS CLICommand Reference.

`describe-reserved-elasticsearch-instances`

Il seguente esempio di codice mostra come utilizzare`describe-reserved-elasticsearch-instances`.

AWS CLI

Come visualizzare tutte le istanze riservate

L'esempio `describe-elasticsearch-domains` seguente fornisce un riepilogo di tutte le istanze prenotate in una Regione.

```
aws es describe-reserved-elasticsearch-instances
```

Output:

```
{
    "ReservedElasticsearchInstances": [{
        "FixedPrice": 100.0,
        "ReservedElasticsearchInstanceOfferingId":
 "1a2a3a4a5-1a2a-3a4a-5a6a-1a2a3a4a5a6a",
```

```
        "ReservationName": "my-reservation",
        "PaymentOption": "PARTIAL_UPFRONT",
        "UsagePrice": 0.0,
        "ReservedElasticsearchInstanceId": "9a8a7a6a-5a4a-3a2a-1a0a-9a8a7a6a5a4a",
        "RecurringCharges": [{
            "RecurringChargeAmount": 0.603,
            "RecurringChargeFrequency": "Hourly"
        }],
        "State": "payment-pending",
        "StartTime": 1522872571.229,
        "ElasticsearchInstanceCount": 3,
        "Duration": 31536000,
        "ElasticsearchInstanceType": "m4.2xlarge.elasticsearch",
        "CurrencyCode": "USD"
    }]
}
```

Per ulteriori informazioni, consulta Istanze riservate nella Guida per gli sviluppatori di Amazon Elasticsearch Service.

- Per i dettagli sull'API, consulta DescribeReservedElasticsearchInstances AWS CLICommand Reference.

`list-domain-names`

Il seguente esempio di codice mostra come utilizzare`list-domain-names`.

AWS CLI

Come elencare tutti i domini

L'esempio `list-domain-names` seguente fornisce un breve riepilogo di tutti i domini della Regione.

```
aws es list-domain-names
```

Output:

```
{
    "DomainNames": [{
            "DomainName": "cli-example-1"
        },
        {
```

```
            "DomainName": "cli-example-2"
        }
    ]
}
```

Per ulteriori informazioni, consulta Creazione e gestione dei domini di Amazon Elasticsearch Service nella Guida per gli sviluppatori di Amazon Elasticsearch Service.

- Per i dettagli sull'API, consulta ListDomainNames AWS CLICommand Reference.

Organizations: esempi che utilizzano AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with Organizations.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`accept-handshake`

Il seguente esempio di codice mostra come utilizzare`accept-handshake`.

AWS CLI

Come accettare un handshake da un altro account

Bill, il proprietario di un'organizzazione, ha invitato l'account di Juan a partecipare alla sua organizzazione. L'esempio seguente mostra l'account di Juan che accetta l'handshake e quindi l'invito.

```
aws organizations accept-handshake --handshake-id h-examplehandshakeid111
```

L'output mostra quanto segue:

```
{
        "Handshake": {
                "Action": "INVITE",
                "Arn": "arn:aws:organizations::111111111111:handshake/o-
exampleorgid/invite/h-examplehandshakeid111",
                "RequestedTimestamp": 1481656459.257,
                "ExpirationTimestamp": 1482952459.257,
                "Id": "h-examplehandshakeid111",
                "Parties": [
                        {
                                "Id": "o-exampleorgid",
                                "Type": "ORGANIZATION"
                        },
                        {
                                "Id": "juan@example.com",
                                "Type": "EMAIL"
                        }
                ],
                "Resources": [
                        {
                                "Resources": [
                                        {
                                                "Type": "MASTER_EMAIL",
                                                "Value": "bill@amazon.com"
                                        },
                                        {
                                                "Type": "MASTER_NAME",
                                                "Value": "Org Master Account"
                                        },
                                        {
                                                "Type": "ORGANIZATION_FEATURE_SET",
                                                "Value": "ALL"
                                        }
                                ],
                                "Type": "ORGANIZATION",
                                "Value": "o-exampleorgid"
                        },
                        {
                                "Type": "EMAIL",
                                "Value": "juan@example.com"
                        }
                ],
```

```
                "State": "ACCEPTED"
        }
}
```

- Per i dettagli sull'API, consulta AcceptHandshake AWS CLICommand Reference.

`attach-policy`

Il seguente esempio di codice mostra come utilizzare`attach-policy`.

AWS CLI

Come associare una policy a una root, un'unità organizzativa (UO) o un account

Esempio 1

L'esempio seguente mostra come collegare una policy di controllo dei servizi a un'unità organizzativa (UO):

```
aws organizations attach-policy
                --policy-id p-examplepolicyid111
                --target-id ou-examplerootid111-exampleouid111
```

Esempio 2

L'esempio seguente mostra come collegare una policy di controllo dei servizi direttamente a un account:

```
aws organizations attach-policy
                --policy-id p-examplepolicyid111
                --target-id 333333333333
```

- Per i dettagli sull'API, consulta AttachPolicy AWS CLICommand Reference.

`cancel-handshake`

Il seguente esempio di codice mostra come utilizzare`cancel-handshake`.

AWS CLI

Come annullare un handshake inviato da un altro account

Bill ha inviato un invito all'account di Susan per partecipare alla sua organizzazione. Cambia idea e decide di annullare l'invito prima che Susan lo accetti. L'esempio seguente mostra l'annullamento di Bill:

```
aws organizations cancel-handshake --handshake-id h-examplehandshakeid111
```

L'output include un oggetto handshake che mostra che lo stato è CANCELED:

```
{
        "Handshake": {
                "Id": "h-examplehandshakeid111",
                "State":"CANCELED",
                "Action": "INVITE",
                "Arn": "arn:aws:organizations::111111111111:handshake/o-
exampleorgid/invite/h-examplehandshakeid111",
                "Parties": [
                        {
                                "Id": "o-exampleorgid",
                                "Type": "ORGANIZATION"
                        },
                        {
                                "Id": "susan@example.com",
                                "Type": "EMAIL"
                        }
                ],
                "Resources": [
                        {
                                "Type": "ORGANIZATION",
                                "Value": "o-exampleorgid",
                                "Resources": [
                                        {
                                                "Type": "MASTER_EMAIL",
                                                "Value": "bill@example.com"
                                        },
                                        {
                                                "Type": "MASTER_NAME",
                                                "Value": "Master Account"
                                        },
                                        {
                                                "Type": "ORGANIZATION_FEATURE_SET",
                                                "Value": "CONSOLIDATED_BILLING"
                                        }
                                ]
```

```
                        },
                        {
                                "Type": "EMAIL",
                                "Value": "anika@example.com"
                        },
                        {
                                "Type": "NOTES",
                                "Value": "This is a request for Susan's account to
 join Bob's organization."
                        }
                ],
                "RequestedTimestamp": 1.47008383521E9,
                "ExpirationTimestamp": 1.47137983521E9
        }
}
```

- Per i dettagli sull'API, consulta CancelHandshake AWS CLICommand Reference.

`create-account`

Il seguente esempio di codice mostra come utilizzare`create-account`.

AWS CLI

Come creare un account membro automaticamente appartenente all'organizzazione

L'esempio seguente mostra come creare un account membro in un'organizzazione. L'account membro è configurato con il nome Production Account e l'indirizzo e-mail susan@example.com. Organizations crea automaticamente un ruolo IAM utilizzando il nome predefinito di OrganizationAccountAccessRole perché il parametro RoleName non è specificato. Inoltre, l'impostazione che consente agli utenti o ai ruoli IAM con autorizzazioni sufficienti di accedere ai dati di fatturazione dell'account è impostata sul valore predefinito di ALLOW perché il IamUserAccessToBilling parametro non è specificato. Organizations invia automaticamente a Susan un'e-mail di «Benvenuto a AWS»:

```
aws organizations create-account --email susan@example.com --account-
name "Production Account"
```

L'output include un oggetto richiesta che mostra che lo stato è `IN_PROGRESS`:

```
{
```

```
        "CreateAccountStatus": {
                "State": "IN_PROGRESS",
                "Id": "car-examplecreateaccountrequestid111"
        }
}
```

Successivamente è possibile interrogare lo stato corrente della richiesta fornendo il valore di risposta Id al describe-create-account-status comando come valore per il create-account-request-id parametro.

Per ulteriori informazioni, consulta Creare un AWS account nella tua organizzazione nella AWS Organizations Users Guide.

- Per i dettagli sull'API, consulta CreateAccount AWS CLICommand Reference.

`create-organization`

Il seguente esempio di codice mostra come utilizzare`create-organization`.

AWS CLI

Esempio 1: come creare una nuova organizzazione

Bill desidera creare un'organizzazione utilizzando le credenziali dell'account 111111111111. L'esempio seguente mostra che l'account diventa l'account principale nella nuova organizzazione. Poiché non specifica un set di funzionalità, per impostazione predefinita la nuova organizzazione prevede che tutte le funzionalità siano abilitate e le policy di controllo del servizio siano abilitate nella root.

```
aws organizations create-organization
```

L'output include un oggetto organizzazione con dettagli sulla nuova organizzazione:

```
{
        "Organization": {
                "AvailablePolicyTypes": [
                        {
                                "Status": "ENABLED",
                                "Type": "SERVICE_CONTROL_POLICY"
                        }
                ],
                "MasterAccountId": "111111111111",
```

```
                "MasterAccountArn": "arn:aws:organizations::111111111111:account/o-
exampleorgid/111111111111",
                "MasterAccountEmail": "bill@example.com",
                "FeatureSet": "ALL",
                "Id": "o-exampleorgid",
                "Arn": "arn:aws:organizations::111111111111:organization/o-
exampleorgid"
        }
}
```

Esempio 2: come creare una nuova organizzazione solo con le funzionalità di fatturazione consolidata abilitate

L'esempio seguente crea un'organizzazione che supporta solo le funzionalità di fatturazione consolidata:

```
aws organizations create-organization --feature-set CONSOLIDATED_BILLING
```

L'output include un oggetto organizzazione con dettagli sulla nuova organizzazione:

```
{
        "Organization": {
                "Arn": "arn:aws:organizations::111111111111:organization/o-
exampleorgid",
                "AvailablePolicyTypes": [],
                "Id": "o-exampleorgid",
                "MasterAccountArn": "arn:aws:organizations::111111111111:account/o-
exampleorgid/111111111111",
                "MasterAccountEmail": "bill@example.com",
                "MasterAccountId": "111111111111",
                "FeatureSet": "CONSOLIDATED_BILLING"
        }
}
```

Per ulteriori informazioni, consulta Creazione di un account nell'organizzazione nella Guida per l'utente di AWS Organizations.

- Per i dettagli sull'API, consulta CreateOrganization AWS CLICommand Reference.

`create-organizational-unit`

Il seguente esempio di codice mostra come utilizzare`create-organizational-unit`.

AWS CLI

Come creare un'unità organizzativa in un'unità organizzativa root o principale

L'esempio seguente mostra come creare un'unità organizzativa denominata AccountingOU:

```
aws organizations create-organizational-unit --parent-id r-examplerootid111 --
name AccountingOU
```

L'output include un oggetto organizationalUnit con dettagli sulla nuova unità organizzativa:

```
{
        "OrganizationalUnit": {
                "Id": "ou-examplerootid111-exampleouid111",
                "Arn": "arn:aws:organizations::111111111111:ou/o-exampleorgid/ou-
examplerootid111-exampleouid111",
                "Name": "AccountingOU"
        }
}
```

- Per i dettagli sull'API, consulta CreateOrganizationalUnit AWS CLICommand Reference.

`create-policy`

Il seguente esempio di codice mostra come utilizzare`create-policy`.

AWS CLI

Esempio 1: come creare una policy con un file di sorgente del testo per la policy JSON

L'esempio seguente mostra come creare una policy di controllo dei servizi denominata `AllowAllS3Actions`. Il contenuto della policy è tratto da un file presente sul computer locale chiamato `policy.json`.

```
aws organizations create-policy --content file://policy.json --
name AllowAllS3Actions, --type SERVICE_CONTROL_POLICY --description "Allows
 delegation of all S3 actions"
```

L'output include un oggetto policy con dettagli sulla nuova policy:

```
{
        "Policy": {
```

```
                "Content": "{\"Version\":\"2012-10-17\",\"Statement\":[{\"Effect\":
\"Allow\",\"Action\":[\"s3:*\"],\"Resource\":[\"*\"]}]}",
                "PolicySummary": {
                        "Arn": "arn:aws:organizations::o-exampleorgid:policy/
service_control_policy/p-examplepolicyid111",
                        "Description": "Allows delegation of all S3 actions",
                        "Name": "AllowAllS3Actions",
                        "Type":"SERVICE_CONTROL_POLICY"
                }
        }
}
```

Esempio 2: come creare una policy con una policy JSON come parametro

L'esempio seguente mostra come creare la stessa policy di controllo dei servizi, questa volta incorporando il contenuto della policy come stringa JSON nel parametro. La stringa deve essere preceduta da barre rovesciate come caratteri di escape prima delle virgolette doppie affinché queste ultime vengano considerate valori letterali del parametro, che a sua volta dovrà essere racchiuso tra virgolette doppie:

```
aws organizations create-policy --content "{\"Version\":\"2012-10-17\",\"Statement
\":[{\"Effect\":\"Allow\",\"Action\":[\"s3:*\"],\"Resource\":[\"*\"]}]}" --
name AllowAllS3Actions --type SERVICE_CONTROL_POLICY --description "Allows
 delegation of all S3 actions"
```

Per ulteriori informazioni sulla creazione e l'utilizzo delle policy nell'organizzazione, consulta Gestione delle policy dell'organizzazione nella Guida per l'utente di AWS Organizations.

- Per i dettagli sull'API, consulta CreatePolicy AWS CLICommand Reference.

`decline-handshake`

Il seguente esempio di codice mostra come utilizzare`decline-handshake`.

AWS CLI

Come annullare un handshake inviato da un altro account

L'esempio seguente mostra che Susan, un'amministratrice proprietaria dell'account 222222222222, rifiuta un invito a partecipare all'organizzazione di Bill. L' DeclineHandshake operazione restituisce un oggetto handshake, a dimostrazione del fatto che lo stato è ora RIFIUTATO:

```
aws organizations decline-handshake --handshake-id h-examplehandshakeid111
```

L'output include un oggetto handshake che mostra il nuovo stato DECLINED:

```
{
        "Handshake": {
                "Id": "h-examplehandshakeid111",
                "State": "DECLINED",
                "Resources": [
                        {
                                "Type": "ORGANIZATION",
                                "Value": "o-exampleorgid",
                                "Resources": [
                                        {
                                                "Type": "MASTER_EMAIL",
                                                "Value": "bill@example.com"
                                        },
                                        {
                                                "Type": "MASTER_NAME",
                                                "Value": "Master Account"
                                        }
                                ]
                        },
                        {
                                "Type": "EMAIL",
                                "Value": "susan@example.com"
                        },
                        {
                                "Type": "NOTES",
                                "Value": "This is an invitation to Susan's account
 to join the Bill's organization."
                        }
                ],
                "Parties": [
                        {
                                "Type": "EMAIL",
                                "Id": "susan@example.com"
                        },
                        {
                                "Type": "ORGANIZATION",
                                "Id": "o-exampleorgid"
                        }
                ],
```

```
                "Action": "INVITE",
                "RequestedTimestamp": 1470684478.687,
                "ExpirationTimestamp": 1471980478.687,
                "Arn": "arn:aws:organizations::111111111111:handshake/o-
exampleorgid/invite/h-examplehandshakeid111"
        }
}
```

- Per i dettagli sull'API, consulta AWS CLI Command DeclineHandshakeReference.

`delete-organization`

Il seguente esempio di codice mostra come utilizzare`delete-organization`.

AWS CLI

Come eliminare un'organizzazione

L'esempio seguente mostra come eliminare un'organizzazione. Per eseguire questa operazione, devi essere un amministratore dell'account principale dell'organizzazione. L'esempio presuppone che in precedenza siano stati rimossi tutti gli account dei membri e le politiche dall'organizzazione: OUs

```
aws organizations delete-organization
```

- Per i dettagli sull'API, consulta DeleteOrganization AWS CLICommand Reference.

`delete-organizational-unit`

Il seguente esempio di codice mostra come utilizzare`delete-organizational-unit`.

AWS CLI

Come eliminare un'UO

L'esempio seguente mostra come eliminare un'UO. L'esempio presuppone che in precedenza siano stati rimossi tutti gli account e altri account OUs dall'unità organizzativa:

```
aws organizations delete-organizational-unit --organizational-unit-id ou-
examplerootid111-exampleouid111
```

- Per i dettagli sull'API, vedere DeleteOrganizationalUnitin AWS CLI Command Reference.

`delete-policy`

Il seguente esempio di codice mostra come utilizzare`delete-policy`.

AWS CLI

Come eliminare una policy

L'esempio seguente mostra come eliminare una policy da un'organizzazione. L'esempio presuppone che in precedenza la policy sia stata scollegata da tutte le entità:

```
aws organizations delete-policy --policy-id p-examplepolicyid111
```

- Per i dettagli sull'API, consulta DeletePolicy AWS CLICommand Reference.

`describe-account`

Il seguente esempio di codice mostra come utilizzare`describe-account`.

AWS CLI

Come ottenere i dettagli su un account

L'esempio seguente mostra come richiedere i dettagli relativi a un account:

```
aws organizations describe-account --account-id 555555555555
```

L'output mostra un oggetto account con i dettagli sull'account:

```
{
        "Account": {
                "Id": "555555555555",
                "Arn": "arn:aws:organizations::111111111111:account/o-
exampleorgid/555555555555",
                "Name": "Beta account",
                "Email": "anika@example.com",
                "JoinedMethod": "INVITED",
                "JoinedTimeStamp": 1481756563.134,
                "Status": "ACTIVE"
```

```
        }
}
```

- Per i dettagli sull'API, consulta DescribeAccount AWS CLICommand Reference.

`describe-create-account-status`

Il seguente esempio di codice mostra come utilizzare`describe-create-account-status`.

AWS CLI

Come ottenere lo stato più recente di una richiesta per creare un account

L'esempio seguente mostra come richiedere lo stato più recente di una precedente richiesta per creare un account in un'organizzazione. L'identificatore --request-id specificato proviene dalla risposta della chiamata originale a create-account. La richiesta di creazione dell'account indica nel campo Stato che Organizations ha completato la creazione dell'account.

Comando:

```
aws organizations describe-create-account-status --create-account-request-id car-examplecreateaccountrequestid111
```

Output:

```
{
  "CreateAccountStatus": {
    "State": "SUCCEEDED",
    "AccountId": "555555555555",
    "AccountName": "Beta account",
    "RequestedTimestamp": 1470684478.687,
    "CompletedTimestamp": 1470684532.472,
    "Id": "car-examplecreateaccountrequestid111"
  }
}
```

- Per i dettagli sull'API, consulta DescribeCreateAccountStatus AWS CLICommand Reference.

`describe-handshake`

Il seguente esempio di codice mostra come utilizzare`describe-handshake`.

AWS CLI

Come ottenere informazioni su un handshake

L'esempio seguente mostra come richiedere i dettagli relativi a un handshake. L'ID dell'handshake proviene dalla chiamata originale a `InviteAccountToOrganization` o da una chiamata a `ListHandshakesForAccount` o `ListHandshakesForOrganization`:

```
aws organizations describe-handshake --handshake-id h-examplehandshakeid111
```

L'output include un oggetto handshake che contiene tutti i dettagli sull'handshake richiesto:

```
{
        "Handshake": {
                "Id": "h-examplehandshakeid111",
                "State": "OPEN",
                "Resources": [
                        {
                                "Type": "ORGANIZATION",
                                "Value": "o-exampleorgid",
                                "Resources": [
                                        {
                                                "Type": "MASTER_EMAIL",
                                                "Value": "bill@example.com"
                                        },
                                        {
                                                "Type": "MASTER_NAME",
                                                "Value": "Master Account"
                                        }
                                ]
                        },
                        {
                                "Type": "EMAIL",
                                "Value": "anika@example.com"
                        }
                ],
                "Parties": [
                        {
                                "Type": "ORGANIZATION",
                                "Id": "o-exampleorgid"
                        },
                        {
                                "Type": "EMAIL",
```

```
                                "Id": "anika@example.com"
                        }
                ],
                "Action": "INVITE",
                "RequestedTimestamp": 1470158698.046,
                "ExpirationTimestamp": 1471454698.046,
                "Arn": "arn:aws:organizations::111111111111:handshake/o-
exampleorgid/invite/h-examplehandshakeid111"
        }
}
```

- Per i dettagli sull'API, consulta DescribeHandshake AWS CLICommand Reference.

`describe-organization`

Il seguente esempio di codice mostra come utilizzare`describe-organization`.

AWS CLI

Come ottenere informazioni sull'organizzazione corrente

L'esempio seguente mostra come richiedere i dettagli relativi a un'organizzazione:

```
aws organizations describe-organization
```

L'output include un oggetto organizzazione con dettagli sull'organizzazione:

```
{
        "Organization": {
                "MasterAccountArn": "arn:aws:organizations::111111111111:account/o-
exampleorgid/111111111111",
                "MasterAccountEmail": "bill@example.com",
                "MasterAccountId": "111111111111",
                "Id": "o-exampleorgid",
                "FeatureSet": "ALL",
                "Arn": "arn:aws:organizations::111111111111:organization/o-
exampleorgid",
                "AvailablePolicyTypes": [
                        {
                                "Status": "ENABLED",
                                "Type": "SERVICE_CONTROL_POLICY"
                        }
```

```
                ]
        }
}
```

- Per i dettagli sull'API, consulta DescribeOrganization AWS CLICommand Reference.

`describe-organizational-unit`

Il seguente esempio di codice mostra come utilizzare`describe-organizational-unit`.

AWS CLI

Come ottenere informazioni su un'unità organizzativa

L'esempio `describe-organizational-unit` seguente richiede dettagli su un'unità organizzativa.

```
aws organizations describe-organizational-unit \
    --organizational-unit-id ou-examplerootid111-exampleouid111
```

Output:

```
{
    "OrganizationalUnit": {
        "Name": "Accounting Group",
        "Arn": "arn:aws:organizations::123456789012:ou/o-exampleorgid/ou-
examplerootid111-exampleouid111",
        "Id": "ou-examplerootid111-exampleouid111"
    }
}
```

- Per i dettagli sull'API, consulta DescribeOrganizationalUnit AWS CLICommand Reference.

`describe-policy`

Il seguente esempio di codice mostra come utilizzare`describe-policy`.

AWS CLI

Come ottenere informazioni su una policy

L'esempio seguente mostra come richiedere informazioni su una policy:

```
aws organizations describe-policy --policy-id p-examplepolicyid111
```

L'output include un oggetto policy contenente i dettagli sulla policy:

```
{
        "Policy": {
                "Content": "{\n  \"Version\": \"2012-10-17\",\n  \"Statement\": [\n
    {\n      \"Effect\": \"Allow\",\n      \"Action\": \"*\",\n      \"Resource\":
 \"*\"\n    }\n  ]\n}",
                "PolicySummary": {
                        "Arn": "arn:aws:organizations::111111111111:policy/o-
exampleorgid/service_control_policy/p-examplepolicyid111",
                        "Type": "SERVICE_CONTROL_POLICY",
                        "Id": "p-examplepolicyid111",
                        "AwsManaged": false,
                        "Name": "AllowAllS3Actions",
                        "Description": "Enables admins to delegate S3 permissions"
                }
        }
}
```

- Per i dettagli sull'API, consulta DescribePolicy AWS CLICommand Reference.

`detach-policy`

Il seguente esempio di codice mostra come utilizzare`detach-policy`.

AWS CLI

Come scollegare una policy da una root, una UO o un account

L'esempio seguente mostra come scollegare una policy da un'unità organizzativa.

```
aws organizations  detach-policy  --target-id ou-examplerootid111-exampleouid111 --
policy-id p-examplepolicyid111
```

- Per i dettagli sull'API, consulta DetachPolicy AWS CLICommand Reference.

`disable-policy-type`

Il seguente esempio di codice mostra come utilizzare`disable-policy-type`.

AWS CLI

Come disabilitare un tipo di policy in una root

L'esempio seguente mostra come disabilitare il tipo di policy di controllo dei servizi in una root:

```
aws organizations disable-policy-type --root-id r-examplerootid111 --policy-
type SERVICE_CONTROL_POLICY
```

L'output mostra che l'elemento di PolicyTypes risposta non include più SERVICE_CONTROL_POLICY:

```
{
        "Root": {
                "PolicyTypes": [],
                "Name": "Root",
                "Id": "r-examplerootid111",
                "Arn": "arn:aws:organizations::111111111111:root/o-exampleorgid/r-
examplerootid111"
        }
}
```

- Per i dettagli sull'API, vedere in Command Reference. DisablePolicyTypeAWS CLI

`enable-all-features`

Il seguente esempio di codice mostra come utilizzare`enable-all-features`.

AWS CLI

Come abilitare tutte le funzionalità in un'organizzazione

Questo esempio mostra l'amministratore che chiede a tutti gli account invitati dell'organizzazione di approvare l'attivazione di tutte le funzionalità dell'organizzazione. AWS Organizations invia un'e-mail all'indirizzo registrato con ogni account membro invitato chiedendo al proprietario di approvare la modifica a tutte le funzionalità accettando l'handshake che viene inviato. Dopo che tutti gli account dei membri invitati hanno accettato l'handshake, l'amministratore dell'organizzazione può finalizzare la modifica a tutte le funzionalità e chi dispone delle autorizzazioni appropriate può creare politiche e applicarle a root e account: OUs

```
aws organizations enable-all-features
```

L'output è un oggetto handshake che viene inviato a tutti gli account membro invitati per l'approvazione:

```
{
        "Handshake": {
                "Action": "ENABLE_ALL_FEATURES",
                "Arn":"arn:aws:organizations::111111111111:handshake/o-exampleorgid/
enable_all_features/h-examplehandshakeid111",
                "ExpirationTimestamp":1.483127868609E9,
                "Id":"h-examplehandshakeid111",
                "Parties": [
                        {
                                "id":"o-exampleorgid",
                                "type":"ORGANIZATION"
                        }
                ],
                "requestedTimestamp":1.481831868609E9,
                "resources": [
                        {
                                "type":"ORGANIZATION",
                                "value":"o-exampleorgid"
                        }
                ],
                "state":"REQUESTED"
        }
}
```

- Per i dettagli sull'API, consulta Command EnableAllFeaturesReference AWS CLI .

`enable-policy-type`

Il seguente esempio di codice mostra come utilizzare`enable-policy-type`.

AWS CLI

Come abilitare l'uso di un tipo di policy in una root

L'esempio seguente mostra come abilitare il tipo di policy di controllo dei servizi in una root:

```
aws organizations enable-policy-type --root-id r-examplerootid111 --policy-
type SERVICE_CONTROL_POLICY
```

L'output mostra un oggetto root con un elemento di risposta PolicyTypes che mostra che ora SCPs sono abilitati:

```
{
        "Root": {
                "PolicyTypes": [
                        {
                                "Status":"ENABLED",
                                "Type":"SERVICE_CONTROL_POLICY"
                        }
                ],
                "Id": "r-examplerootid111",
                "Name": "Root",
                "Arn": "arn:aws:organizations::111111111111:root/o-exampleorgid/r-
examplerootid111"
        }
}
```

- Per i dettagli sull'API, consultate AWS CLI Command EnablePolicyTypeReference.

`invite-account-to-organization`

Il seguente esempio di codice mostra come utilizzare`invite-account-to-organization`.

AWS CLI

Come invitare un account a partecipare a un'organizzazione

L'esempio seguente mostra l'account master di proprietà di bill@example.com che invita l'account di proprietà di juan@example.com a partecipare a un'organizzazione:

```
aws organizations invite-account-to-organization --target '{"Type": "EMAIL", "Id":
 "juan@example.com"}' --notes "This is a request for Juan's account to join Bill's
 organization."
```

L'output include una struttura handshake che mostra ciò che viene inviato all'account invitato:

```
{
        "Handshake": {
                "Action": "INVITE",
                "Arn": "arn:aws:organizations::111111111111:handshake/o-
exampleorgid/invite/h-examplehandshakeid111",
```

```
                "ExpirationTimestamp": 1482952459.257,
                "Id": "h-examplehandshakeid111",
                "Parties": [
                        {
                                "Id": "o-exampleorgid",
                                "Type": "ORGANIZATION"
                        },
                        {
                                "Id": "juan@example.com",
                                "Type": "EMAIL"
                        }
                ],
                "RequestedTimestamp": 1481656459.257,
                "Resources": [
                        {
                                "Resources": [
                                        {
                                                "Type": "MASTER_EMAIL",
                                                "Value": "bill@amazon.com"
                                        },
                                        {
                                                "Type": "MASTER_NAME",
                                                "Value": "Org Master Account"
                                        },
                                        {
                                                "Type": "ORGANIZATION_FEATURE_SET",
                                                "Value": "FULL"
                                        }
                                ],
                                "Type": "ORGANIZATION",
                                "Value": "o-exampleorgid"
                        },
                        {
                                "Type": "EMAIL",
                                "Value": "juan@example.com"
                        }
                ],
                "State": "OPEN"
        }
}
```

- Per i dettagli sull'API, consulta [InviteAccountToOrganization AWS CLI](#)Command Reference.

`leave-organization`

Il seguente esempio di codice mostra come utilizzare`leave-organization`.

AWS CLI

Come lasciare un'organizzazione come account membro

L'esempio seguente mostra l'amministratore di un account membro che richiede di lasciare l'organizzazione di cui è attualmente membro:

```
aws organizations leave-organization
```

- Per i dettagli sull'API, consulta LeaveOrganization AWS CLICommand Reference.

`list-accounts-for-parent`

Il seguente esempio di codice mostra come utilizzare`list-accounts-for-parent`.

AWS CLI

Come recuperare un elenco di tutti gli account in una root o un'unità organizzativa principale specificata

L'esempio seguente mostra come richiedere un elenco degli account in un'unità organizzativa:

```
aws organizations list-accounts-for-parent --parent-id ou-examplerootid111-
exampleouid111
```

L'output include un elenco di oggetti di riepilogo degli account.

```
{
        "Accounts": [
                {
                        "Arn": "arn:aws:organizations::111111111111:account/o-
exampleorgid/333333333333",
                        "JoinedMethod": "INVITED",
                        "JoinedTimestamp": 1481835795.536,
                        "Id": "333333333333",
                        "Name": "Development Account",
                        "Email": "juan@example.com",
```

```
                        "Status": "ACTIVE"
                },
                {
                        "Arn": "arn:aws:organizations::111111111111:account/o-
exampleorgid/444444444444",
                        "JoinedMethod": "INVITED",
                        "JoinedTimestamp": 1481835812.143,
                        "Id": "444444444444",
                        "Name": "Test Account",
                        "Email": "anika@example.com",
                        "Status": "ACTIVE"
                }
        ]
}
```

- Per i dettagli sull'API, consulta ListAccountsForParent AWS CLICommand Reference.

`list-accounts`

Il seguente esempio di codice mostra come utilizzare`list-accounts`.

AWS CLI

Come recuperare un elenco di tutti gli account di un'organizzazione

L'esempio seguente mostra come richiedere un elenco degli account in un'organizzazione:

```
aws organizations list-accounts
```

L'output include un elenco di oggetti di riepilogo degli account.

```
{
        "Accounts": [
                {
                        "Arn": "arn:aws:organizations::111111111111:account/o-
exampleorgid/111111111111",
                        "JoinedMethod": "INVITED",
                        "JoinedTimestamp": 1481830215.45,
                        "Id": "111111111111",
                        "Name": "Master Account",
                        "Email": "bill@example.com",
                        "Status": "ACTIVE"
```

```
                },
                {
                        "Arn": "arn:aws:organizations::111111111111:account/o-
exampleorgid/222222222222",
                        "JoinedMethod": "INVITED",
                        "JoinedTimestamp": 1481835741.044,
                        "Id": "222222222222",
                        "Name": "Production Account",
                        "Email": "alice@example.com",
                        "Status": "ACTIVE"
                },
                {
                        "Arn": "arn:aws:organizations::111111111111:account/o-
exampleorgid/333333333333",
                        "JoinedMethod": "INVITED",
                        "JoinedTimestamp": 1481835795.536,
                        "Id": "333333333333",
                        "Name": "Development Account",
                        "Email": "juan@example.com",
                        "Status": "ACTIVE"
                },
                {
                        "Arn": "arn:aws:organizations::111111111111:account/o-
exampleorgid/444444444444",
                        "JoinedMethod": "INVITED",
                        "JoinedTimestamp": 1481835812.143,
                        "Id": "444444444444",
                        "Name": "Test Account",
                        "Email": "anika@example.com",
                        "Status": "ACTIVE"
                }
        ]
}
```

- Per i dettagli sull'API, consulta ListAccounts AWS CLICommand Reference.

`list-children`

Il seguente esempio di codice mostra come utilizzare`list-children`.

AWS CLI

Per recuperare gli account secondari e quelli OUs di un'unità organizzativa o root principale

L'esempio seguente illustra come elencare la root o l'unità organizzativa che contiene l'account 444444444444:

```
aws organizations list-children --child-type ORGANIZATIONAL_UNIT --parent-id ou-examplerootid111-exampleouid111
```

L'output mostra i due figli OUs contenuti dal genitore:

```
{
        "Children": [
                {
                        "Id": "ou-examplerootid111-exampleouid111",
                        "Type":"ORGANIZATIONAL_UNIT"
                },
                {
                        "Id":"ou-examplerootid111-exampleouid222",
                        "Type":"ORGANIZATIONAL_UNIT"
                }
        ]
}
```

- Per i dettagli sull'API, consulta ListChildren AWS CLICommand Reference.

`list-create-account-status`

Il seguente esempio di codice mostra come utilizzare`list-create-account-status`.

AWS CLI

Esempio 1: come recuperare un elenco delle richieste di creazione di account effettuate nell'organizzazione corrente

L'esempio seguente mostra come richiedere un elenco di richieste di creazione di account che sono state completate per un'organizzazione:

```
aws organizations list-create-account-status --states SUCCEEDED
```

L'output include una serie di oggetti con informazioni su ogni richiesta.

```
{
        "CreateAccountStatuses": [
```

```
                {
                        "AccountId": "444444444444",
                        "AccountName": "Developer Test Account",
                        "CompletedTimeStamp": 1481835812.143,
                        "Id": "car-examplecreateaccountrequestid111",
                        "RequestedTimeStamp": 1481829432.531,
                        "State": "SUCCEEDED"
                }
        ]
}
```

Esempio 2: come recuperare un elenco delle richieste di creazione di account in corso nell'organizzazione corrente

L'esempio seguente ottiene un elenco di richieste di creazione di account in corso per un'organizzazione:

```
aws organizations list-create-account-status --states IN_PROGRESS
```

L'output include una serie di oggetti con informazioni su ogni richiesta.

```
{
        "CreateAccountStatuses": [
                {
                  "State": "IN_PROGRESS",
                  "Id": "car-examplecreateaccountrequestid111",
                  "RequestedTimeStamp": 1481829432.531,
                  "AccountName": "Production Account"
                }
        ]
}
```

- Per i dettagli sull'API, consulta ListCreateAccountStatus AWS CLICommand Reference.

`list-handshakes-for-account`

Il seguente esempio di codice mostra come utilizzare`list-handshakes-for-account`.

AWS CLI

Come recuperare un elenco di handshake inviati a un account

L'esempio seguente mostra come ottenere un elenco di tutti gli handshake associati all'account delle credenziali utilizzate per chiamare l'operazione:

```
aws organizations list-handshakes-for-account
```

L'output include un elenco di strutture handshake con informazioni su ogni handshake, incluso lo stato corrente:

```
{
        "Handshake": {
                "Action": "INVITE",
                "Arn": "arn:aws:organizations::111111111111:handshake/o-
exampleorgid/invite/h-examplehandshakeid111",
                "ExpirationTimestamp": 1482952459.257,
                "Id": "h-examplehandshakeid111",
                "Parties": [
                        {
                                "Id": "o-exampleorgid",
                                "Type": "ORGANIZATION"
                        },
                        {
                                "Id": "juan@example.com",
                                "Type": "EMAIL"
                        }
                ],
                "RequestedTimestamp": 1481656459.257,
                "Resources": [
                        {
                                "Resources": [
                                        {
                                                "Type": "MASTER_EMAIL",
                                                "Value": "bill@amazon.com"
                                        },
                                        {
                                                "Type": "MASTER_NAME",
                                                "Value": "Org Master Account"
                                        },
                                        {
                                                "Type": "ORGANIZATION_FEATURE_SET",
                                                "Value": "FULL"
                                        }
                                ],
                                "Type": "ORGANIZATION",
```

```
                                "Value": "o-exampleorgid"
                        },
                        {
                                "Type": "EMAIL",
                                "Value": "juan@example.com"
                        }
                ],
                "State": "OPEN"
        }
}
```

- Per i dettagli sull'API, consulta ListHandshakesForAccount AWS CLICommand Reference.

`list-handshakes-for-organization`

Il seguente esempio di codice mostra come utilizzare`list-handshakes-for-organization`.

AWS CLI

Come recuperare un elenco di handshake associati a un'organizzazione

L'esempio seguente mostra come ottenere un elenco di handshake associati all'organizzazione corrente:

```
aws organizations list-handshakes-for-organization
```

L'output mostra due handshake. Il primo è un invito all'account di Juan e mostra lo stato OPEN. Il secondo è un invito all'account di Anika e mostra lo stato ACCEPTED:

```
{
        "Handshakes": [
                {
                        "Action": "INVITE",
                        "Arn": "arn:aws:organizations::111111111111:handshake/o-
exampleorgid/invite/h-examplehandshakeid111",
                        "ExpirationTimestamp": 1482952459.257,
                        "Id": "h-examplehandshakeid111",
                        "Parties": [
                                {
                                        "Id": "o-exampleorgid",
                                        "Type": "ORGANIZATION"
                                },
```

```
                                {
                                        "Id": "juan@example.com",
                                        "Type": "EMAIL"
                                }
                        ],
                        "RequestedTimestamp": 1481656459.257,
                        "Resources": [
                                {
                                        "Resources": [
                                                {
                                                        "Type": "MASTER_EMAIL",
                                                        "Value": "bill@amazon.com"
                                                },
                                                {
                                                        "Type": "MASTER_NAME",
                                                        "Value": "Org Master
 Account"
                                                },
                                                {
                                                        "Type":
 "ORGANIZATION_FEATURE_SET",
                                                        "Value": "FULL"
                                                }
                                        ],
                                        "Type": "ORGANIZATION",
                                        "Value": "o-exampleorgid"
                                },
                                {
                                        "Type": "EMAIL",
                                        "Value": "juan@example.com"
                                },
                                {
                                        "Type":"NOTES",
                                        "Value":"This is an invitation to Juan's
 account to join Bill's organization."
                                }
                        ],
                        "State": "OPEN"
                },
                {
                        "Action": "INVITE",
                        "State":"ACCEPTED",
                        "Arn": "arn:aws:organizations::111111111111:handshake/o-
exampleorgid/invite/h-examplehandshakeid111",
```

```
                        "ExpirationTimestamp": 1.471797437427E9,
                        "Id": "h-examplehandshakeid222",
                        "Parties": [
                                {
                                        "Id": "o-exampleorgid",
                                        "Type": "ORGANIZATION"
                                },
                                {
                                        "Id": "anika@example.com",
                                        "Type": "EMAIL"
                                }
                        ],
                        "RequestedTimestamp": 1.469205437427E9,
                        "Resources": [
                                {
                                        "Resources": [
                                                {
                                                        "Type":"MASTER_EMAIL",
                                                        "Value":"bill@example.com"
                                                },
                                                {
                                                        "Type":"MASTER_NAME",
                                                        "Value":"Master Account"
                                                }
                                        ],
                                        "Type":"ORGANIZATION",
                                        "Value":"o-exampleorgid"
                                },
                                {
                                        "Type":"EMAIL",
                                        "Value":"anika@example.com"
                                },
                                {
                                        "Type":"NOTES",
                                        "Value":"This is an invitation to Anika's
 account to join Bill's organization."
                                }
                        ]
                }
        ]
}
```

- Per i dettagli sull'API, consulta ListHandshakesForOrganization AWS CLICommand Reference.

`list-organizational-units-for-parent`

Il seguente esempio di codice mostra come utilizzare`list-organizational-units-for-parent`.

AWS CLI

Per recuperare un elenco di file OUs in un'unità organizzativa o root principale

L'esempio seguente mostra come ottenere un elenco di OUs in una radice specificata:

```
aws organizations list-organizational-units-for-parent --parent-id r-
examplerootid111
```

L'output mostra che la radice specificata ne contiene due OUs e mostra i dettagli di ciascuna:

```
{
        "OrganizationalUnits": [
                {
                        "Name": "AccountingDepartment",
                        "Arn": "arn:aws:organizations::o-exampleorgid:ou/r-
examplerootid111/ou-examplerootid111-exampleouid111"
                },
                {
                        "Name": "ProductionDepartment",
                        "Arn": "arn:aws:organizations::o-exampleorgid:ou/r-
examplerootid111/ou-examplerootid111-exampleouid222"
                }
        ]
}
```

- Per i dettagli sull'API, vedere ListOrganizationalUnitsForParentin AWS CLI Command Reference.

`list-parents`

Il seguente esempio di codice mostra come utilizzare`list-parents`.

AWS CLI

Per elencare il genitore OUs o i root di un account o di un'unità organizzativa secondaria

L'esempio seguente illustra come elencare la root o l'unità organizzativa principale che contiene l'account 444444444444:

```
aws organizations list-parents --child-id 444444444444
```

L'output mostra che l'account specificato si trova nell'unità organizzativa con l'ID specificato:

```
{
  "Parents": [
        {
          "Id": "ou-examplerootid111-exampleouid111",
          "Type": "ORGANIZATIONAL_UNIT"
        }
  ]
}
```

- Per i dettagli sull'API, consulta ListParents AWS CLICommand Reference.

`list-policies-for-target`

Il seguente esempio di codice mostra come utilizzare`list-policies-for-target`.

AWS CLI

Per recuperare un elenco degli SCPs allegati direttamente a un account

L'esempio seguente mostra come ottenere un elenco di tutte le politiche di controllo del servizio (SCPs), come specificato dal parametro Filter, che sono direttamente collegate a un account:

```
aws organizations list-policies-for-target --filter SERVICE_CONTROL_POLICY --target-
id 444444444444
```

L'output include un elenco di strutture policy con informazioni di riepilogo sulle policy. L'elenco non include le policy che si applicano all'account a causa dell'ereditarietà dalla posizione in una gerarchia di unità organizzative:

```
{
        "Policies": [
                {
                        "Type": "SERVICE_CONTROL_POLICY",
                        "Name": "AllowAllEC2Actions",
```

```
                        "AwsManaged", false,
                        "Id": "p-examplepolicyid222",
                        "Arn": "arn:aws:organizations::o-exampleorgid:policy/
service_control_policy/p-examplepolicyid222",
                        "Description": "Enables account admins to delegate
 permissions for any EC2 actions to users and roles in their accounts."
                }
        ]
}
```

- Per i dettagli sull'API, vedere ListPoliciesForTargetin AWS CLI Command Reference.

`list-policies`

Il seguente esempio di codice mostra come utilizzare`list-policies`.

AWS CLI

Come recuperare un elenco di tutte le policy di un'organizzazione di un determinato tipo

L'esempio seguente mostra come ottenere un elenco di SCPs, come specificato dal parametro filter:

```
aws organizations list-policies --filter SERVICE_CONTROL_POLICY
```

L'output include un elenco di policy con informazioni di riepilogo:

```
{
        "Policies": [
                {
                        "Type": "SERVICE_CONTROL_POLICY",
                        "Name": "AllowAllS3Actions",
                        "AwsManaged": false,
                        "Id": "p-examplepolicyid111",
                        "Arn": "arn:aws:organizations::111111111111:policy/
service_control_policy/p-examplepolicyid111",
                        "Description": "Enables account admins to delegate
 permissions for any S3 actions to users and roles in their accounts."
                },
                {
                        "Type": "SERVICE_CONTROL_POLICY",
                        "Name": "AllowAllEC2Actions",
```

```
                        "AwsManaged": false,
                        "Id": "p-examplepolicyid222",
                        "Arn": "arn:aws:organizations::111111111111:policy/
service_control_policy/p-examplepolicyid222",
                        "Description": "Enables account admins to delegate
 permissions for any EC2 actions to users and roles in their accounts."
                },
                {
                        "AwsManaged": true,
                        "Description": "Allows access to every operation",
                        "Type": "SERVICE_CONTROL_POLICY",
                        "Id": "p-FullAWSAccess",
                        "Arn": "arn:aws:organizations::aws:policy/
service_control_policy/p-FullAWSAccess",
                        "Name": "FullAWSAccess"
                }
        ]
}
```

- Per i dettagli sull'API, consulta ListPolicies AWS CLICommand Reference.

`list-roots`

Il seguente esempio di codice mostra come utilizzare`list-roots`.

AWS CLI

Come recuperare un elenco delle root di un'organizzazione

L'esempio seguente mostra come ottenere l'elenco delle root per un'organizzazione:

```
aws organizations list-roots
```

L'output include un elenco di strutture root con informazioni di riepilogo:

```
{
        "Roots": [
                {
                        "Name": "Root",
                        "Arn": "arn:aws:organizations::111111111111:root/o-
exampleorgid/r-examplerootid111",
                        "Id": "r-examplerootid111",
```

```
                        "PolicyTypes": [
                                {
                                        "Status":"ENABLED",
                                        "Type":"SERVICE_CONTROL_POLICY"
                                }
                        ]
                }
        ]
}
```

- Per i dettagli sull'API, consulta ListRoots AWS CLICommand Reference.

`list-targets-for-policy`

Il seguente esempio di codice mostra come utilizzare`list-targets-for-policy`.

AWS CLI

Per recuperare un elenco delle radici e degli account a cui è associata una politica OUs

L'esempio seguente mostra come ottenere un elenco delle radici e degli account a cui è associata la politica specificata: OUs

```
aws organizations list-targets-for-policy --policy-id p-FullAWSAccess
```

L'output include un elenco di oggetti allegati con informazioni di riepilogo sulle radici e gli account a cui è associata la politica: OUs

```
{
        "Targets": [
                {
                        "Arn": "arn:aws:organizations::111111111111:root/o-
exampleorgid/r-examplerootid111",
                        "Name": "Root",
                        "TargetId":"r-examplerootid111",
                        "Type":"ROOT"
                },
                {
                        "Arn": "arn:aws:organizations::111111111111:account/o-
exampleorgid/333333333333;",
                        "Name": "Developer Test Account",
                        "TargetId": "333333333333",
```

```
                        "Type": "ACCOUNT"
                },
                {
                        "Arn":"arn:aws:organizations::111111111111:ou/o-
exampleorgid/ou-examplerootid111-exampleouid111",
                        "Name":"Accounting",
                        "TargetId":"ou-examplerootid111-exampleouid111",
                        "Type":"ORGANIZATIONAL_UNIT"
                }
        ]
}
```

- Per i dettagli sull'API, consulta ListTargetsForPolicy AWS CLICommand Reference.

`move-account`

Il seguente esempio di codice mostra come utilizzare`move-account`.

AWS CLI

Per spostare un account tra root o OUs

L'esempio seguente mostra come spostare l'account master dell'organizzazione dalla root a un'unità organizzativa:

```
aws organizations move-account --account-id 333333333333 --source-parent-id r-
examplerootid111 --destination-parent-id ou-examplerootid111-exampleouid111
```

- Per i dettagli sull'API, consulta MoveAccount AWS CLICommand Reference.

`remove-account-from-organization`

Il seguente esempio di codice mostra come utilizzare`remove-account-from-organization`.

AWS CLI

Come rimuovere un account da un'organizzazione come account master

L'esempio seguente mostra come rimuovere un account da un'organizzazione:

```
aws organizations remove-account-from-organization --account-id 333333333333
```

- Per i dettagli sull'API, consulta RemoveAccountFromOrganization AWS CLICommand Reference.

`update-organizational-unit`

Il seguente esempio di codice mostra come utilizzare`update-organizational-unit`.

AWS CLI

Come rinominare un'unità organizzativa

Questo esempio mostra come rinominare un'unità organizzativa in "AccountingOU":

```
aws organizations update-organizational-unit --organizational-unit-id ou-
examplerootid111-exampleouid111 --name AccountingOU
```

L'output mostra il nuovo nome:

```
{
        "OrganizationalUnit": {
                "Id": "ou-examplerootid111-exampleouid111"
                "Name": "AccountingOU",
                "Arn": "arn:aws:organizations::111111111111:ou/o-exampleorgid/ou-
examplerootid111-exampleouid111""
        }
}
```

- Per i dettagli sull'API, consulta UpdateOrganizationalUnit AWS CLICommand Reference.

`update-policy`

Il seguente esempio di codice mostra come utilizzare`update-policy`.

AWS CLI

Esempio 1: come rinominare una policy

L'esempio `update-policy` seguente rinomina una policy e le fornisce una nuova descrizione.

```
aws organizations update-policy \
```

```
    --policy-id p-examplepolicyid111 \
    --name Renamed-Policy \
    --description "This description replaces the original."
```

L'output mostra il nuovo nome e la descrizione.

```
{
    "Policy": {
        "Content": "{\n  \"Version\":\"2012-10-17\",\n  \"Statement\":{\n
  \"Effect\":\"Allow\",\n    \"Action\":\"ec2:*\",\n    \"Resource\":\"*\"\n
  }\n}\n",
        "PolicySummary": {
            "Id": "p-examplepolicyid111",
            "AwsManaged": false,
            "Arn":"arn:aws:organizations::111111111111:policy/o-exampleorgid/
service_control_policy/p-examplepolicyid111",
            "Description": "This description replaces the original.",
            "Name": "Renamed-Policy",
            "Type": "SERVICE_CONTROL_POLICY"
        }
    }
}
```

Esempio 2: come sostituire il contenuto di testo JSON di una policy

L'esempio seguente mostra come sostituire il testo JSON di SCP nell'esempio precedente con una nuova stringa di testo della policy JSON che consente S3 anziché: EC2

```
aws organizations update-policy \
    --policy-id p-examplepolicyid111 \
    --content "{\"Version\":\"2012-10-17\",\"Statement\":{\"Effect\":\"Allow\",
\"Action\":\"s3:*\",\"Resource\":\"*\"}}"
```

L'output mostra il nuovo contenuto:

```
{
    "Policy": {
        "Content": "{ \"Version\": \"2012-10-17\", \"Statement\": { \"Effect\":
 \"Allow\", \"Action\": \"s3:*\", \"Resource\": \"*\" } }",
        "PolicySummary": {
            "Arn": "arn:aws:organizations::111111111111:policy/o-exampleorgid/
service_control_policy/p-examplepolicyid111",
```

```
            "AwsManaged": false;
            "Description": "This description replaces the original.",
            "Id": "p-examplepolicyid111",
            "Name": "Renamed-Policy",
            "Type": "SERVICE_CONTROL_POLICY"
        }
    }
}
```

- Per i dettagli sull'API, consulta UpdatePolicyCommand Reference.AWS CLI

AWS Outposts esempi utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with AWS Outposts.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`get-outpost-instance-types`

Il seguente esempio di codice mostra come utilizzare`get-outpost-instance-types`.

AWS CLI

Come ottenere i tipi di istanza sul tuo Outpost

L'esempio `get-outpost-instance-types` seguente ottiene i tipi di istanza per l'Outpost specificato.

```
aws outposts get-outpost-instance-types \
```

```
    --outpost-id op-0ab23c4567EXAMPLE
```

Output:

```
{
    "InstanceTypes": [
        {
            "InstanceType": "c5d.large"
        },
        {
            "InstanceType": "i3en.24xlarge"
        },
        {
            "InstanceType": "m5d.large"
        },
        {
            "InstanceType": "r5d.large"
        }
    ],
    "OutpostId": "op-0ab23c4567EXAMPLE",
    "OutpostArn": "arn:aws:outposts:us-west-2:123456789012:outpost/
op-0ab23c4567EXAMPLE"
}
```

Per ulteriori informazioni, consulta Avvio di un'istanza sull'Outpost nella AWS Guida per l'utente di Outposts.

- Per i dettagli sull'API, consulta GetOutpostInstanceTypes AWS CLICommand Reference.

`get-outpost`

Il seguente esempio di codice mostra come utilizzare`get-outpost`.

AWS CLI

Come ottenere i dettagli di Outpost

L'esempio `get-outpost` seguente mostra i dettagli dell'Outpost specificato.

```
aws outposts get-outpost \
    --outpost-id op-0ab23c4567EXAMPLE
```

Output:

```
{
    "Outpost": {
        "OutpostId": "op-0ab23c4567EXAMPLE",
        "OwnerId": "123456789012",
        "OutpostArn": "arn:aws:outposts:us-west-2:123456789012:outpost/
op-0ab23c4567EXAMPLE",
        "SiteId": "os-0ab12c3456EXAMPLE",
        "Name": "EXAMPLE",
        "LifeCycleStatus": "ACTIVE",
        "AvailabilityZone": "us-west-2a",
        "AvailabilityZoneId": "usw2-az1",
        "Tags": {}
    }
}
```

Per ulteriori informazioni, consulta Utilizzo di Outposts nella Guida per l'utente di AWS Outposts.

- Per i dettagli sull'API, consulta GetOutpost AWS CLICommand Reference.

`list-outposts`

Il seguente esempio di codice mostra come utilizzare`list-outposts`.

AWS CLI

Come elencare gli Outpost

L'`list-outposts`esempio seguente elenca gli Outposts presenti nel tuo AWS account.

```
aws outposts list-outposts
```

Output:

```
{
    "Outposts": [
        {
            "OutpostId": "op-0ab23c4567EXAMPLE",
            "OwnerId": "123456789012",
            "OutpostArn": "arn:aws:outposts:us-west-2:123456789012:outpost/
op-0ab23c4567EXAMPLE",
            "SiteId": "os-0ab12c3456EXAMPLE",
```

```
            "Name": "EXAMPLE",
            "Description": "example",
            "LifeCycleStatus": "ACTIVE",
            "AvailabilityZone": "us-west-2a",
            "AvailabilityZoneId": "usw2-az1",
            "Tags": {
                "Name": "EXAMPLE"
            }
        },
        {
            "OutpostId": "op-4fe3dc21baEXAMPLE",
            "OwnerId": "123456789012",
            "OutpostArn": "arn:aws:outposts:us-west-2:123456789012:outpost/
op-4fe3dc21baEXAMPLE",
            "SiteId": "os-0ab12c3456EXAMPLE",
            "Name": "EXAMPLE2",
            "LifeCycleStatus": "ACTIVE",
            "AvailabilityZone": "us-west-2a",
            "AvailabilityZoneId": "usw2-az1",
            "Tags": {}
        }
    ]
}
```

Per ulteriori informazioni, consulta Utilizzo di Outposts nella Guida per l'utente di AWS Outposts.

- Per i dettagli sull'API, consulta ListOutposts AWS CLICommand Reference.

`list-sites`

Il seguente esempio di codice mostra come utilizzare`list-sites`.

AWS CLI

Come elencare i siti

L'`list-sites`esempio seguente elenca i siti Outpost disponibili nel tuo AWS account.

```
aws outposts list-sites
```

Output:

```
{
```

```
    "Sites": [
        {
            "SiteId": "os-0ab12c3456EXAMPLE",
            "AccountId": "123456789012",
            "Name": "EXAMPLE",
            "Description": "example",
            "Tags": {}
        }
    ]
}
```

Per ulteriori informazioni, consulta Utilizzo di Outposts nella Guida per l'utente di AWS Outposts.

- Per i dettagli sull'API, consulta ListSites AWS CLICommand Reference.

AWS Payment Cryptography esempi utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with AWS Payment Cryptography.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`create-alias`

Il seguente esempio di codice mostra come utilizzare`create-alias`.

AWS CLI

Come creare un alias per una chiave

L'esempio `create-alias` seguente crea un alias per una chiave.

```
aws payment-cryptography create-alias \
    --alias-name alias/sampleAlias1 \
    --key-arn arn:aws:payment-cryptography:us-east-2:123456789012:key/
kwapwa6qaifllw2h
```

Output:

```
{
    "Alias": {
        "AliasName": "alias/sampleAlias1",
        "KeyArn": "arn:aws:payment-cryptography:us-west-2:123456789012:key/
kwapwa6qaifllw2h"
    }
}
```

Per ulteriori informazioni, consulta Informazioni sugli alias nella Guida per l'utente di AWS Payment Cryptography (APC).

- Per i dettagli sull'API, consulta CreateAlias AWS CLICommand Reference.

`create-key`

Il seguente esempio di codice mostra come utilizzare`create-key`.

AWS CLI

Come creare una chiave

L'`create-key`esempio seguente genera una chiave TDES 2KEY che è possibile utilizzare per generare e verificare i valori CVV2 CVV/.

```
aws payment-cryptography create-key \
    --exportable \
    --key-
attributes KeyAlgorithm=TDES_2KEY, KeyUsage=TR31_C0_CARD_VERIFICATION_KEY,KeyClass=SYMMETRIC
```

Output:

```
{
    "Key": {
```

```
        "CreateTimestamp": "1686800690",
        "Enabled": true,
        "Exportable": true,
        "KeyArn": "arn:aws:payment-cryptography:us-west-2:123456789012:key/
kwapwa6qaifllw2h",
        "KeyAttributes": {
            "KeyAlgorithm": "TDES_2KEY",
            "KeyClass": "SYMMETRIC_KEY",
            "KeyModesOfUse": {
                "Decrypt": false,
                "DeriveKey": false,
                "Encrypt": false,
                "Generate": true,
                "NoRestrictions": false,
                "Sign": false,
                "Unwrap": false,
                "Verify": true,
                "Wrap": false
            },
            "KeyUsage": "TR31_C0_CARD_VERIFICATION_KEY"
        },
        "KeyCheckValue": "F2E50F",
        "KeyCheckValueAlgorithm": "ANSI_X9_24",
        "KeyOrigin": "AWS_PAYMENT_CRYPTOGRAPHY",
        "KeyState": "CREATE_COMPLETE",
        "UsageStartTimestamp": "1686800690"
    }
}
```

Per ulteriori informazioni, consulta Generazione delle chiavi nella Guida per l'utente di AWS Payment Cryptography (APC).

- Per i dettagli sull'API, consulta Command Reference. CreateKeyAWS CLI

`delete-alias`

Il seguente esempio di codice mostra come utilizzare`delete-alias`.

AWS CLI

Come eliminare un alias

L'esempio `delete-alias` seguente elimina un alias. L'operazione non interessa la chiave.

```
aws payment-cryptography delete-alias \
    --alias-name alias/sampleAlias1
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Informazioni sugli alias nella Guida per l'utente di AWS Payment Cryptography (APC).

- Per i dettagli sull'API, consulta DeleteAlias AWS CLICommand Reference.

`delete-key`

Il seguente esempio di codice mostra come utilizzare`delete-key`.

AWS CLI

Come eliminare una chiave

L'esempio `delete-key` seguente pianifica l'eliminazione di una chiave dopo 7 giorni, il periodo di attesa predefinito.

```
aws payment-cryptography delete-key \
    --key-identifier arn:aws:payment-cryptography:us-west-2:123456789012:key/
kwapwa6qaifllw2h
```

Output:

```
{
    "Key": {
        "CreateTimestamp": "1686801198",
        "DeletePendingTimestamp": "1687405998",
        "Enabled": true,
        "Exportable": true,
        "KeyArn": "arn:aws:payment-cryptography:us-west-2:123456789012:key/
kwapwa6qaifllw2h",
        "KeyAttributes": {
            "KeyAlgorithm": "TDES_2KEY",
            "KeyClass": "SYMMETRIC_KEY",
            "KeyModesOfUse": {
                "Decrypt": false,
                "DeriveKey": false,
```

```
                "Encrypt": false,
                "Generate": true,
                "NoRestrictions": false,
                "Sign": false,
                "Unwrap": false,
                "Verify": true,
                "Wrap": false
            },
            "KeyUsage": "TR31_C0_CARD_VERIFICATION_KEY"
        },
        "KeyCheckValue": "F2E50F",
        "KeyCheckValueAlgorithm": "ANSI_X9_24",
        "KeyOrigin": "AWS_PAYMENT_CRYPTOGRAPHY",
        "KeyState": "DELETE_PENDING",
        "UsageStartTimestamp": "1686801190"
    }
}
```

Per ulteriori informazioni, consulta Eliminazione delle chiavi nella Guida per l'utente di AWS Payment Cryptography (APC).

- Per i dettagli sull'API, consulta DeleteKey AWS CLICommand Reference.

`export-key`

Il seguente esempio di codice mostra come utilizzare`export-key`.

AWS CLI

Come esportare una chiave

L'esempio `export-key` seguente esporta una chiave.

```
aws payment-cryptography export-key \
    --export-key-identifier arn:aws:payment-cryptography:us-west-2:123456789012:key/lco3w6agsk7zgu2l \
    --key-material '{"Tr34KeyBlock": { \
        "CertificateAuthorityPublicKeyIdentifier": "arn:aws:payment-cryptography:us-west-2:123456789012:key/ftobshq7pvioc5fx", \
        "ExportToken": "export-token-cu4lg26ofcziixny", \
        "KeyBlockFormat": "X9_TR34_2012", \
        "WrappingKeyCertificate": file://wrapping-key-certificate.pem }}'
```

Contenuto di `wrapping-key-certificate.pem`:

```
LS0tLS1CRUdJTiBDRVJUSUZJQ0FURS0tLS0tCk1JSUV2VENDQXFXZ0F3SUJBZ0lSQU1ZZS8xMXFUK2svVzlRUDJQOElv
```

Output:

```
{
    "WrappedKey": {
        "KeyMaterial":
 "308205A106092A864886F70D010702A08205923082058E020101310D300B0609608648016503040201308203F
        "WrappedKeyMaterialFormat": "TR34_KEY_BLOCK"
    }
}
```

Per ulteriori informazioni, consulta Esportare le chiavi nella Guida per l'utente di AWS Payment Cryptography (APC).

- Per i dettagli sull'API, consulta ExportKey AWS CLICommand Reference.

`get-alias`

Il seguente esempio di codice mostra come utilizzare`get-alias`.

AWS CLI

Come ottenere un alias

L'esempio `get-alias` seguente restituisce l'ARN della chiave associata all'alias.

```
aws payment-cryptography get-alias \
    --alias-name alias/sampleAlias1
```

Output:

```
{
    "Alias": {
        "AliasName": "alias/sampleAlias1",
        "KeyArn": "arn:aws:payment-cryptography:us-west-2:123456789012:key/
kwapwa6qaifllw2h"
    }
```

```
}
```

Per ulteriori informazioni, consulta Informazioni sugli alias nella Guida per l'utente di AWS Payment Cryptography (APC).

- Per i dettagli sull'API, consulta GetAlias AWS CLICommand Reference.

`get-key`

Il seguente esempio di codice mostra come utilizzare`get-key`.

AWS CLI

Come ottenere i metadati di una chiave

L'esempio `get-key` seguente restituisce i metadati della chiave associata all'alias. Questa operazione non restituisce materiale crittografico.

```
aws payment-cryptography get-key \
    --key-identifier alias/sampleAlias1
```

Output:

```
{
    "Key": {
        "CreateTimestamp": "1686800690",
        "DeletePendingTimestamp": "1687405998",
        "Enabled": true,
        "Exportable": true,
        "KeyArn": "arn:aws:payment-cryptography:us-west-2:123456789012:key/
kwapwa6qaifllw2h",
        "KeyAttributes": {
            "KeyAlgorithm": "TDES_2KEY",
            "KeyClass": "SYMMETRIC_KEY",
            "KeyModesOfUse": {
                "Decrypt": false,
                "DeriveKey": false,
                "Encrypt": false,
                "Generate": true,
                "NoRestrictions": false,
                "Sign": false,
```

```
                "Unwrap": false,
                "Verify": true,
                "Wrap": false
            },
            "KeyUsage": "TR31_C0_CARD_VERIFICATION_KEY"
        },
        "KeyCheckValue": "F2E50F",
        "KeyCheckValueAlgorithm": "ANSI_X9_24",
        "KeyOrigin": "AWS_PAYMENT_CRYPTOGRAPHY",
        "KeyState": "DELETE_PENDING",
        "UsageStartTimestamp": "1686801190"
    }
}
```

Per ulteriori informazioni, consulta Ottenere le chiavi nella Guida per l'utente di AWS Payment Cryptography (APC).

- Per i dettagli sull'API, consulta GetKey AWS CLICommand Reference.

`get-parameters-for-export`

Il seguente esempio di codice mostra come utilizzare`get-parameters-for-export`.

AWS CLI

Come inizializzare il processo di esportazione

L'esempio `get-parameters-for-export` seguente genera una coppia di chiavi, firma la chiave e quindi restituisce il certificato e la root del certificato.

```
aws payment-cryptography get-parameters-for-export \
    --signing-key-algorithm RSA_2048 \
    --key-material-type TR34_KEY_BLOCK
```

Output:

```
{
    "ExportToken": "export-token-ep5cwyzune7oya53",
    "ParametersValidUntilTimestamp": "1687415640",
    "SigningKeyAlgorithm": "RSA_2048",
    "SigningKeyCertificate":
```

```
    "MIICiTCCAfICCQD6m7oRw0uXOjANBgkqhkiG9w0BAQUFADCBiDELMAkGA1UEBhMC
    VVMxCzAJBgNVBAgTAldBMRAwDgYDVQQHEwdTZWF0dGxlMQ8wDQYDVQQKEwZBbWF6
    b24xFDASBgNVBAsTC0lBTSBDb25zb2xlMRIwEAYDVQQDEwlUZXN0Q2lsYWMxHzAd
    BgkqhkiG9w0BCQEWEG5vb25lQGFtYXpvbi5jb20wHhcNMTEwNDI1MjA0NTIxWhcN
    MTIwNDI0MjA0NTIxWjCBiDELMAkGA1UEBhMCVVMxCzAJBgNVBAgTAldBMRAwDgYD
    VQQHEwdTZWF0dGxlMQ8wDQYDVQQKEwZBbWF6b24xFDASBgNVBAsTC0lBTSBDb25z
    b2xlMRIwEAYDVQQDEwlUZXN0Q2lsYWMxHzAdBgkqhkiG9w0BCQEWEG5vb25lQGFt
    YXpvbi5jb20wgZ8wDQYJKoZIhvcNAQEBBQADgY0AMIGJAoGBAMaK0dn+a4GmWIWJ
    21uUSfwfEvySWtC2XADZ4nB+BLYgVIk60CpiwsZ3G93vUEIO3IyNoH/f0wYK8m9T
    rDHudUZg3qX4waLG5M43q7Wgc/MbQITxOUSQv7c7ugFFDzQGBzZswY6786m86gpE
    Ibb3OhjZnzcvQAaRHhdlQWIMm2nrAgMBAAEwDQYJKoZIhvcNAQEFBQADgYEAtCu4
    nUhVVxYUntneD9+h8Mg9q6q+auNKyExzyLwaxlAoo7TJHidbtS4J5iNmZgXL0Fkb
    FFBjvSfpJIlJ00zbhNYS5f6GuoEDmFJl0ZxBHjJnyp378OD8uTs7fLvjx79LjSTb
    NYiytVbZPQUQ5Yaxu2jXnimvw3rrszlaEXAMPLE=",
    "SigningKeyCertificateChain":
    "NIICiTCCAfICCQD6m7oRw0uXOjANBgkqhkiG9w0BAQUFADCBiDELMAkGA1UEBhMC
    VVMxCzAJBgNVBAgTAldBMRAwDgYDVQQHEwdTZWF0dGxlMQ8wDQYDVQQKEwZBbWF6
    b24xFDASBgNVBAsTC0lBTSBDb25zb2xlMRIwEAYDVQQDEwlUZXN0Q2lsYWMxHzAd
    BgkqhkiG9w0BCQEWEG5vb25lQGFtYXpvbi5jb20wHhcNMTEwNDI1MjA0NTIxWhcN
    MTIwNDI0MjA0NTIxWjCBiDELMAkGA1UEBhMCVVMxCzAJBgNVBAgTAldBMRAwDgYD
    VQQHEwdTZWF0dGxlMQ8wDQYDVQQKEwZBbWF6b24xFDASBgNVBAsTC0lBTSBDb25z
    b2xlMRIwEAYDVQQDEwlUZXN0Q2lsYWMxHzAdBgkqhkiG9w0BCQEWEG5vb25lQGFt
    YXpvbi5jb20wgZ8wDQYJKoZIhvcNAQEBBQADgY0AMIGJAoGBAMaK0dn+a4GmWIWJ
    21uUSfwfEvySWtC2XADZ4nB+BLYgVIk60CpiwsZ3G93vUEIO3IyNoH/f0wYK8m9T
    rDHudUZg3qX4waLG5M43q7Wgc/MbQITxOUSQv7c7ugFFDzQGBzZswY6786m86gpE
    Ibb3OhjZnzcvQAaRHhdlQWIMm2nrAgMBAAEwDQYJKoZIhvcNAQEFBQADgYEAtCu4
    nUhVVxYUntneD9+h8Mg9q6q+auNKyExzyLwaxlAoo7TJHidbtS4J5iNmZgXL0Fkb
    FFBjvSfpJIlJ00zbhNYS5f6GuoEDmFJl0ZxBHjJnyp378OD8uTs7fLvjx79LjSTb
    NYiytVbZPQUQ5Yaxu2jXnimvw3rrszlaEXAMPLE="
}
```

Per ulteriori informazioni, consulta Esportare le chiavi nella Guida per l'utente di AWS Payment Cryptography (APC).

- Per i dettagli sull'API, consulta GetParametersForExport AWS CLICommand Reference.

`get-parameters-for-import`

Il seguente esempio di codice mostra come utilizzare`get-parameters-for-import`.

AWS CLI

Come inizializzare il processo di importazione

L'esempio `get-parameters-for-import` seguente genera una coppia di chiavi, firma la chiave e quindi restituisce il certificato e la root del certificato.

```
aws payment-cryptography get-parameters-for-import \
    --key-material-type TR34_KEY_BLOCK \
    --wrapping-key-algorithm RSA_2048
```

Output:

```
{
    "ImportToken": "import-token-qgmafpaa7nt2kfbb",
    "ParametersValidUntilTimestamp": "1687415640",
    "WrappingKeyAlgorithm": "RSA_2048",
    "WrappingKeyCertificate":
    "MIICiTCCAfICCQD6m7oRw0uXOjANBgkqhkiG9w0BAQUFADCBiDELMAkGA1UEBhMC
    VVMxCzAJBgNVBAgTAldBMRAwDgYDVQQHEwdTZWF0dGxlMQ8wDQYDVQQKEwZBbWF6
    b24xFDASBgNVBAsTC0lBTSBDb25zb2xlMRIwEAYDVQQDEwlUZXN0Q2lsYWMxHzAd
    BgkqhkiG9w0BCQEWEG5vb25lQGFtYXpvbi5jb20wHhcNMTEwNDI1MjA0NTIxWhcN
    MTIwNDI0MjA0NTIxWjCBiDELMAkGA1UEBhMCVVMxCzAJBgNVBAgTAldBMRAwDgYD
    VQQHEwdTZWF0dGxlMQ8wDQYDVQQKEwZBbWF6b24xFDASBgNVBAsTC0lBTSBDb25z
    b2xlMRIwEAYDVQQDEwlUZXN0Q2lsYWMxHzAdBgkqhkiG9w0BCQEWEG5vb25lQGFt
    YXpvbi5jb20wgZ8wDQYJKoZIhvcNAQEBBQADgY0AMIGJAoGBAMaK0dn+a4GmWIWJ
    21uUSfwfEvySWtC2XADZ4nB+BLYgVIk60CpiwsZ3G93vUEIO3IyNoH/f0wYK8m9T
    rDHudUZg3qX4waLG5M43q7Wgc/MbQITxOUSQv7c7ugFFDzQGBzZswY6786m86gpE
    Ibb3OhjZnzcvQAaRHhdlQWIMm2nrAgMBAAEwDQYJKoZIhvcNAQEFBQADgYEAtCu4
    nUhVVxYUntneD9+h8Mg9q6q+auNKyExzyLwaxlAoo7TJHidbtS4J5iNmZgXL0Fkb
    FFBjvSfpJIlJ00zbhNYS5f6GuoEDmFJl0ZxBHjJnyp378OD8uTs7fLvjx79LjSTb
    NYiytVbZPQUQ5Yaxu2jXnimvw3rrszlaEXAMPLE=",
    "WrappingKeyCertificateChain":
    "NIICiTCCAfICCQD6m7oRw0uXOjANBgkqhkiG9w0BAQUFADCBiDELMAkGA1UEBhMC
    VVMxCzAJBgNVBAgTAldBMRAwDgYDVQQHEwdTZWF0dGxlMQ8wDQYDVQQKEwZBbWF6
    b24xFDASBgNVBAsTC0lBTSBDb25zb2xlMRIwEAYDVQQDEwlUZXN0Q2lsYWMxHzAd
    BgkqhkiG9w0BCQEWEG5vb25lQGFtYXpvbi5jb20wHhcNMTEwNDI1MjA0NTIxWhcN
    MTIwNDI0MjA0NTIxWjCBiDELMAkGA1UEBhMCVVMxCzAJBgNVBAgTAldBMRAwDgYD
    VQQHEwdTZWF0dGxlMQ8wDQYDVQQKEwZBbWF6b24xFDASBgNVBAsTC0lBTSBDb25z
    b2xlMRIwEAYDVQQDEwlUZXN0Q2lsYWMxHzAdBgkqhkiG9w0BCQEWEG5vb25lQGFt
    YXpvbi5jb20wgZ8wDQYJKoZIhvcNAQEBBQADgY0AMIGJAoGBAMaK0dn+a4GmWIWJ
    21uUSfwfEvySWtC2XADZ4nB+BLYgVIk60CpiwsZ3G93vUEIO3IyNoH/f0wYK8m9T
    rDHudUZg3qX4waLG5M43q7Wgc/MbQITxOUSQv7c7ugFFDzQGBzZswY6786m86gpE
    Ibb3OhjZnzcvQAaRHhdlQWIMm2nrAgMBAAEwDQYJKoZIhvcNAQEFBQADgYEAtCu4
    nUhVVxYUntneD9+h8Mg9q6q+auNKyExzyLwaxlAoo7TJHidbtS4J5iNmZgXL0Fkb
    FFBjvSfpJIlJ00zbhNYS5f6GuoEDmFJl0ZxBHjJnyp378OD8uTs7fLvjx79LjSTb
    NYiytVbZPQUQ5Yaxu2jXnimvw3rrszlaEXAMPLE="
```

```
}
```

Per ulteriori informazioni, consulta Importare le chiavi nella Guida per l'utente di AWS Payment Cryptography (APC).

- Per i dettagli sull'API, consulta GetParametersForImport AWS CLICommand Reference.

`get-public-key-certificate`

Il seguente esempio di codice mostra come utilizzare`get-public-key-certificate`.

AWS CLI

Come restituire la chiave pubblica

L'esempio `get-public-key-certificate` seguente restituisce la parte di chiave pubblica di una coppia di chiavi.

```
aws payment-cryptography get-public-key-certificate \
    --key-identifier arn:aws:payment-cryptography:us-east-2:123456789012:key/
kwapwa6qaifllw2h
```

Output:

```
{
    "KeyCertificate":
    "MIICiTCCAfICCQD6m7oRw0uXOjANBgkqhkiG9w0BAQUFADCBiDELMAkGA1UEBhMC
    VVMxCzAJBgNVBAgTAldBMRAwDgYDVQQHEwdTZWF0dGxlMQ8wDQYDVQQKEwZBbWF6
    b24xFDASBgNVBAsTC0lBTSBDb25zb2xlMRIwEAYDVQQDEwlUZXN0Q2lsYWMxHzAd
    BgkqhkiG9w0BCQEWEG5vb25lQGFtYXpvbi5jb20wHhcNMTEwNDI1MjA0NTIxWhcN
    MTIwNDI0MjA0NTIxWjCBiDELMAkGA1UEBhMCVVMxCzAJBgNVBAgTAldBMRAwDgYD
    VQQHEwdTZWF0dGxlMQ8wDQYDVQQKEwZBbWF6b24xFDASBgNVBAsTC0lBTSBDb25z
    b2xlMRIwEAYDVQQDEwlUZXN0Q2lsYWMxHzAdBgkqhkiG9w0BCQEWEG5vb25lQGFt
    YXpvbi5jb20wgZ8wDQYJKoZIhvcNAQEBBQADgY0AMIGJAoGBAMaK0dn+a4GmWIWJ
    21uUSfwfEvySWtC2XADZ4nB+BLYgVIk60CpiwsZ3G93vUEIO3IyNoH/f0wYK8m9T
    rDHudUZg3qX4waLG5M43q7Wgc/MbQITxOUSQv7c7ugFFDzQGBzZswY6786m86gpE
    Ibb3OhjZnzcvQAaRHhdlQWIMm2nrAgMBAAEwDQYJKoZIhvcNAQEFBQADgYEAtCu4
    nUhVVxYUntneD9+h8Mg9q6q+auNKyExzyLwaxlAoo7TJHidbtS4J5iNmZgXL0Fkb
    FFBjvSfpJIlJ00zbhNYS5f6GuoEDmFJl0ZxBHjJnyp378OD8uTs7fLvjx79LjSTb
    NYiytVbZPQUQ5Yaxu2jXnimvw3rrszlaEXAMPLE=",
    "KeyCertificateChain":
    "NIICiTCCAfICCQD6m7oRw0uXOjANBgkqhkiG9w0BAQUFADCBiDELMAkGA1UEBhMC
    VVMxCzAJBgNVBAgTAldBMRAwDgYDVQQHEwdTZWF0dGxlMQ8wDQYDVQQKEwZBbWF6
```

```
    b24xFDASBgNVBAsTC0lBTSBDb25zb2xlMRIwEAYDVQQDEwlUZXN0Q2lsYWMxHzAd
    BgkqhkiG9w0BCQEWEG5vb25lQGFtYXpvbi5jb20wHhcNMTEwNDI1MjA0NTIxWhcN
    MTIwNDI0MjA0NTIxWjCBiDELMAkGA1UEBhMCVVMxCzAJBgNVBAgTAldBMRAwDgYD
    VQQHEwdTZWF0dGxlMQ8wDQYDVQQKEwZBbWF6b24xFDASBgNVBAsTC0lBTSBDb25z
    b2xlMRIwEAYDVQQDEwlUZXN0Q2lsYWMxHzAdBgkqhkiG9w0BCQEWEG5vb25lQGFt
    YXpvbi5jb20wgZ8wDQYJKoZIhvcNAQEBBQADgY0AMIGJAoGBAMaK0dn+a4GmWIWJ
    21uUSfwfEvySWtC2XADZ4nB+BLYgVIk60CpiwsZ3G93vUEIO3IyNoH/f0wYK8m9T
    rDHudUZg3qX4waLG5M43q7Wgc/MbQITxOUSQv7c7ugFFDzQGBzZswY6786m86gpE
    Ibb3OhjZnzcvQAaRHhdlQWIMm2nrAgMBAAEwDQYJKoZIhvcNAQEFBQADgYEAtCu4
    nUhVVxYUntneD9+h8Mg9q6q+auNKyExzyLwaxlAoo7TJHidbtS4J5iNmZgXL0Fkb
    FFBjvSfpJIlJ00zbhNYS5f6GuoEDmFJl0ZxBHjJnyp378OD8uTs7fLvjx79LjSTb
    NYiytVbZPQUQ5Yaxu2jXnimvw3rrszlaEXAMPLE="
}
```

Per ulteriori informazioni, consulta key/certificate Associare il pubblico a una coppia di chiavi nella AWS Payment Cryptography User Guide.

- Per i dettagli sull'API, consulta GetPublicKeyCertificate AWS CLICommand Reference.

`import-key`

Il seguente esempio di codice mostra come utilizzare`import-key`.

AWS CLI

Come importare una chiave TR-34

L'esempio `import-key` seguente importa una chiave TR-34.

```
aws payment-cryptography import-key \
        --key-material='{ "Tr34KeyBlock": {" \
            CertificateAuthorityPublicKeyIdentifier": "arn:aws:payment-
cryptography:us-west-2:123456789012:key/rmm5wn2q564njnjm", \
            "ImportToken": "import-token-5ott6ho5nts7bbcg", \
            "KeyBlockFormat": "X9_TR34_2012", \
            "SigningKeyCertificate": file://signing-key-certificate.pem, \
            "WrappedKeyBlock": file://wrapped-key-block.pem }}'
```

Contenuto di `signing-key-certificate.pem`:

```
LS0tLS1CRUdJTiBDRVJUSUZJQ0FURS0tLS0tCk1JSUV2RENDQXFTZ0F3SUJBZ0lRYWVCK25IbE1WZU1PR1ZiNjU1Q2Jz
```

Contenuto di `wrapped-key-block.pem`:

```
3082059806092A864886F70D010702A0820589308205850201013 10D300B06096086480165030402013082031606
```

Output:

```
{
    "Key": {
        "CreateTimestamp": "2023-06-09T16:56:27.621000-07:00",
        "Enabled": true,
        "KeyArn": "arn:aws:payment-cryptography:us-west-2:123456789012:key/
bzmvgyxdg3sktwxd",
        "KeyAttributes": {
            "KeyAlgorithm": "TDES_2KEY",
            "KeyClass": "SYMMETRIC_KEY",
            "KeyModesOfUse": {
                "Decrypt": false,
                "DeriveKey": false,
                "Encrypt": false,
                "Generate": true,
                "NoRestrictions": false,
                "Sign": false,
                "Unwrap": false,
                "Verify": true,
                "Wrap": false
            },
            "KeyUsage": "TR31_C0_CARD_VERIFICATION_KEY"
        },
        "KeyCheckValue": "D9B20E",
        "KeyCheckValueAlgorithm": "ANSI_X9_24",
        "KeyOrigin": "EXTERNAL",
        "KeyState": "CREATE_COMPLETE",
        "UsageStartTimestamp": "2023-06-09T16:56:27.621000-07:00"
    }
}
```

Per ulteriori informazioni, consulta Importare le chiavi nella Guida per l'utente di AWS Payment Cryptography (APC).

- Per i dettagli sull'API, consulta ImportKey AWS CLICommand Reference.

`list-aliases`

Il seguente esempio di codice mostra come utilizzare`list-aliases`.

AWS CLI

Come ottenere un elenco di alias

L'esempio `list-aliases` seguente mostra tutti gli alias del tuo account in questa Regione.

```
aws payment-cryptography list-aliases
```

Output:

```
{
    "Aliases": [
        {
            "AliasName": "alias/sampleAlias1",
            "KeyArn": "arn:aws:payment-cryptography:us-east-2:123456789012:key/
kwapwa6qaifllw2h"
        },
        {
            "AliasName": "alias/sampleAlias2",
            "KeyArn": "arn:aws:payment-cryptography:us-east-2:123456789012:key/
kwapwa6qaifllw2h"
        }
    ]
}
```

Per ulteriori informazioni, consulta Informazioni sugli alias nella Guida per l'utente di AWS Payment Cryptography (APC).

- Per i dettagli sull'API, consulta ListAliases AWS CLICommand Reference.

`list-keys`

Il seguente esempio di codice mostra come utilizzare`list-keys`.

AWS CLI

Come ottenere un elenco di chiavi

L'esempio `list-keys` seguente mostra tutte le chiavi del tuo account in questa Regione.

```
aws payment-cryptography list-keys
```

Output:

```
{
    "Keys": [
    {
        "CreateTimestamp": "1666506840",
        "Enabled": false,
        "Exportable": true,
        "KeyArn": "arn:aws:payment-cryptography:us-east-2:123456789012:key/
kwapwa6qaifllw2h",
        "KeyAttributes": {
            "KeyAlgorithm": "TDES_3KEY",
            "KeyClass": "SYMMETRIC_KEY",
            "KeyModesOfUse": {
                "Decrypt": true,
                "DeriveKey": false,
                "Encrypt": true,
                "Generate": false,
                "NoRestrictions": false,
                "Sign": false,
                "Unwrap": true,
                "Verify": false,
                "Wrap": true
            },
            "KeyUsage": "TR31_P1_PIN_GENERATION_KEY"
        },
        "KeyCheckValue": "369D",
        "KeyCheckValueAlgorithm": "ANSI_X9_24",
        "KeyOrigin": "AWS_PAYMENT_CRYPTOGRAPHY",
        "KeyState": "CREATE_COMPLETE",
        "UsageStopTimestamp": "1666938840"
    }
    ]
}
```

Per ulteriori informazioni, consulta Elencare le chiavi nella Guida per l'utente di AWS Payment Cryptography (APC).

- Per i dettagli sull'API, consulta ListKeys AWS CLICommand Reference.

`list-tags-for-resource`

Il seguente esempio di codice mostra come utilizzare`list-tags-for-resource`.

AWS CLI

Come ottenere l'elenco dei tag relativi a una chiave

L'esempio `list-tags-for-resource` seguente ottiene i tag per una chiave.

```
aws payment-cryptography list-tags-for-resource \
    --resource-arn arn:aws:payment-cryptography:us-east-2:123456789012:key/
kwapwa6qaifllw2h
```

Output:

```
{
    "Tags": [
        {
            "Key": "BIN",
            "Value": "20151120"
        },
        {
            "Key": "Project",
            "Value": "Production"
        }
    ]
}
```

Per ulteriori informazioni, consulta Gestione dei tag delle chiavi con operazioni API nella Guida per l'utente di AWS Payment Cryptography (APC).

- Per i dettagli sull'API, consulta ListTagsForResource AWS CLICommand Reference.

`restore-key`

Il seguente esempio di codice mostra come utilizzare`restore-key`.

AWS CLI

Come ripristinare una chiave pianificata per l'eliminazione

L'esempio `restore-key` seguente annulla l'eliminazione di una chiave.

```
aws payment-cryptography restore-key \
```

```
    --key-identifier arn:aws:payment-cryptography:us-east-2:123456789012:key/
kwapwa6qaifllw2h
```

Output:

```
{
    "Key": {
        "KeyArn": "arn:aws:payment-cryptography:us-east-2:123456789012:key/
kwapwa6qaifllw2h",
        "KeyAttributes": {
            "KeyUsage": "TR31_V2_VISA_PIN_VERIFICATION_KEY",
            "KeyClass": "SYMMETRIC_KEY",
            "KeyAlgorithm": "TDES_3KEY",
            "KeyModesOfUse": {
                "Encrypt": false,
                "Decrypt": false,
                "Wrap": false,
                "Unwrap": false,
                "Generate": true,
                "Sign": false,
                "Verify": true,
                "DeriveKey": false,
                "NoRestrictions": false
            }
        },
        "KeyCheckValue": "",
        "KeyCheckValueAlgorithm": "ANSI_X9_24",
        "Enabled": false,
        "Exportable": true,
        "KeyState": "CREATE_COMPLETE",
        "KeyOrigin": "AWS_PAYMENT_CRYPTOGRAPHY",
        "CreateTimestamp": "1686800690",
        "UsageStopTimestamp": "1687405998"
    }
}
```

Per ulteriori informazioni, consulta Eliminazione delle chiavi nella Guida per l'utente di AWS Payment Cryptography (APC).

- Per i dettagli sull'API, consulta RestoreKey AWS CLICommand Reference.

`start-key-usage`

Il seguente esempio di codice mostra come utilizzare`start-key-usage`.

AWS CLI

Come abilitare una chiave

L'esempio `start-key-usage` seguente consente di utilizzare una chiave.

```
aws payment-cryptography start-key-usage \
    --key-identifier arn:aws:payment-cryptography:us-east-2:123456789012:key/
kwapwa6qaifllw2h
```

Output:

```
{
    "Key": {
        "CreateTimestamp": "1686800690",
        "Enabled": true,
        "Exportable": true,
        "KeyArn": "arn:aws:payment-cryptography:us-east-2:111122223333:key/
alsuwfxug3pgy6xh",
        "KeyAttributes": {
            "KeyAlgorithm": "TDES_3KEY",
            "KeyClass": "SYMMETRIC_KEY",
            "KeyModesOfUse": {
                "Decrypt": true,
                "DeriveKey": false,
                "Encrypt": true,
                "Generate": false,
                "NoRestrictions": false,
                "Sign": false,
                "Unwrap": true,
                "Verify": false,
                "Wrap": true
            },
            "KeyUsage": "TR31_P1_PIN_GENERATION_KEY"
        },
        "KeyCheckValue": "369D",
        "KeyCheckValueAlgorithm": "ANSI_X9_24",
        "KeyOrigin": "AWS_PAYMENT_CRYPTOGRAPHY",
        "KeyState": "CREATE_COMPLETE",
```

```
        "UsageStartTimestamp": "1686800690"
    }
}
```

Per ulteriori informazioni, consulta Abilitazione e disabilitazione delle chiavi nella Guida per l'utente di AWS Payment Cryptography (APC).

- Per i dettagli sull'API, consulta StartKeyUsage AWS CLICommand Reference.

`stop-key-usage`

Il seguente esempio di codice mostra come utilizzare`stop-key-usage`.

AWS CLI

Come disabilitare una chiave

L'esempio `stop-key-usage` seguente disabilita una chiave.

```
aws payment-cryptography stop-key-usage \
    --key-identifier arn:aws:payment-cryptography:us-east-2:123456789012:key/
kwapwa6qaifllw2h
```

Output:

```
{
    "Key": {
        "CreateTimestamp": "1686800690",
        "Enabled": true,
        "Exportable": true,
        "KeyArn": "arn:aws:payment-cryptography:us-east-2:111122223333:key/
alsuwfxug3pgy6xh",
        "KeyAttributes": {
            "KeyAlgorithm": "TDES_3KEY",
            "KeyClass": "SYMMETRIC_KEY",
            "KeyModesOfUse": {
                "Decrypt": true,
                "DeriveKey": false,
                "Encrypt": true,
                "Generate": false,
                "NoRestrictions": false,
                "Sign": false,
                "Unwrap": true,
```

```
                "Verify": false,
                "Wrap": true
            },
            "KeyUsage": "TR31_P1_PIN_GENERATION_KEY"
        },
        "KeyCheckValue": "369D",
        "KeyCheckValueAlgorithm": "ANSI_X9_24",
        "KeyOrigin": "AWS_PAYMENT_CRYPTOGRAPHY",
        "KeyState": "CREATE_COMPLETE",
        "UsageStartTimestamp": "1686800690"
    }
}
```

Per ulteriori informazioni, consulta Abilitazione e disabilitazione delle chiavi nella Guida per l'utente di AWS Payment Cryptography (APC).

- Per i dettagli sull'API, consulta StopKeyUsage AWS CLICommand Reference.

`tag-resource`

Il seguente esempio di codice mostra come utilizzare`tag-resource`.

AWS CLI

Come associare un tag a una chiave

L'esempio `tag-resource` seguente tagga una chiave.

```
aws payment-cryptography tag-resource \
    --resource-arn arn:aws:payment-cryptography:us-east-2:123456789012:key/
kwapwa6qaifllw2h \
    --tags Key=sampleTag,Value=sampleValue
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Gestione dei tag delle chiavi nella Guida per l'utente di AWS Payment Cryptography (APC).

- Per i dettagli sull'API, consulta TagResource AWS CLICommand Reference.

`untag-resource`

Il seguente esempio di codice mostra come utilizzare`untag-resource`.

AWS CLI

Come rimuovere un tag da una chiave

L'esempio `untag-resource` seguente rimuove un tag da una chiave.

```
aws payment-cryptography untag-resource \
    --resource-arn arn:aws:payment-cryptography:us-east-2:123456789012:key/
kwapwa6qaifllw2h \
    --tag-keys sampleTag
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Gestione dei tag delle chiavi nella Guida per l'utente di AWS Payment Cryptography (APC).

- Per i dettagli sull'API, consulta UntagResource AWS CLICommand Reference.

`update-alias`

Il seguente esempio di codice mostra come utilizzare`update-alias`.

AWS CLI

Come aggiornare un alias

L'esempio `update-alias` seguente associa l'alias a un'altra chiave.

```
aws payment-cryptography update-alias \
    --alias-name alias/sampleAlias1 \
    --key-arn arn:aws:payment-cryptography:us-east-2:123456789012:key/
tqv5yij6wtxx64pi
```

Output:

```
{
    "Alias": {
        "AliasName": "alias/sampleAlias1",
        "KeyArn": "arn:aws:payment-cryptography:us-west-2:123456789012:key/
tqv5yij6wtxx64pi "
    }
}
```

Per ulteriori informazioni, consulta Informazioni sugli alias nella Guida per l'utente di AWS Payment Cryptography (APC).

- Per i dettagli sull'API, consulta UpdateAlias AWS CLICommand Reference.

AWS Payment Cryptography Esempi di piani dati che utilizzano AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with AWS Payment Cryptography Data Plane.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`decrypt-data`

Il seguente esempio di codice mostra come utilizzare`decrypt-data`.

AWS CLI

Come decrittografare il testo criptato

L'esempio `decrypt-data` seguente decrittografa i dati di testo criptato utilizzando una chiave simmetrica. Per questa operazione, la chiave deve avere `KeyModesOfUse` impostato su `Decrypt` e `KeyUsage` impostato su `TR31_D0_SYMMETRIC_DATA_ENCRYPTION_KEY`.

```
aws payment-cryptography-data decrypt-data \
    --key-identifier arn:aws:payment-cryptography:us-east-2:123456789012:key/
kwapwa6qaifllw2h \
    --cipher-text 33612AB9D6929C3A828EB6030082B2BD \
    --decryption-attributes 'Symmetric={Mode=CBC}'
```

Output:

```
{
    "KeyArn": "arn:aws:payment-cryptography:us-east-2:123456789012:key/
kwapwa6qaifllw2h",
    "KeyCheckValue": "71D7AE",
    "PlainText": "31323334313233343132333431323334"
}
```

Per ulteriori informazioni, consulta Decriptare i dati nella Guida per l'utente di AWS Payment Cryptography (APC).

- Per i dettagli sull'API, consulta DecryptData AWS CLICommand Reference.

`encrypt-data`

Il seguente esempio di codice mostra come utilizzare`encrypt-data`.

AWS CLI

Come crittografare i dati

L'esempio `encrypt-data` seguente crittografa i dati in chiaro utilizzando una chiave simmetrica. Per questa operazione, la chiave deve avere `KeyModesOfUse` impostato su `Encrypt` e `KeyUsage` impostato su `TR31_D0_SYMMETRIC_DATA_ENCRYPTION_KEY`.

```
aws payment-cryptography-data encrypt-data \
    --key-identifier arn:aws:payment-cryptography:us-east-2:123456789012:key/
kwapwa6qaifllw2h \
    --plain-text 31323334313233343132333431323334 \
    --encryption-attributes 'Symmetric={Mode=CBC}'
```

Output:

```
{
    "KeyArn": "arn:aws:payment-cryptography:us-east-2:123456789012:key/
kwapwa6qaifllw2h",
    "KeyCheckValue": "71D7AE",
    "CipherText": "33612AB9D6929C3A828EB6030082B2BD"
}
```

Per ulteriori informazioni, consulta Crittografare i dati nella Guida per l'utente di AWS Payment Cryptography (APC).

- Per i dettagli sull'API, consulta EncryptData AWS CLICommand Reference.

`generate-card-validation-data`

Il seguente esempio di codice mostra come utilizzare`generate-card-validation-data`.

AWS CLI

Come generare un CVV

L'`generate-card-validation-data`esempio seguente genera un CVV/CVV2.

```
aws payment-cryptography-data generate-card-validation-data \
    --key-identifier arn:aws:payment-cryptography:us-east-2:123456789012:key/
kwapwa6qaifllw2h \
    --primary-account-number=171234567890123 \
    --generation-attributes CardVerificationValue2={CardExpiryDate=0123}
```

Output:

```
{
    "KeyArn": "arn:aws:payment-cryptography:us-east-2:123456789012:key/
kwapwa6qaifllw2h",
    "KeyCheckValue": "CADDA1",
    "ValidationData": "801"
}
```

Per ulteriori informazioni, consulta Generare i dati della carta nella Guida per l'utente di AWS Payment Cryptography (APC).

- Per i dettagli sull'API, vedere GenerateCardValidationDatain AWS CLI Command Reference.

`generate-mac`

Il seguente esempio di codice mostra come utilizzare`generate-mac`.

AWS CLI

Come generare un MAC

L'`generate-card-validation-data`esempio seguente genera un codice di autenticazione dei messaggi basato su hash (HMAC) per l'autenticazione dei dati delle carte utilizzando l'algoritmo HMAC_ SHA256 e una chiave di crittografia HMAC. La chiave deve avere `KeyUsage` impostato su `TR31_M7_HMAC_KEY` e `KeyModesOfUse` su `Generate`.

```
aws payment-cryptography-data generate-mac \
    --key-identifier arn:aws:payment-cryptography:us-east-2:123456789012:key/
kwapwa6qaifllw2h \
    --message-
data "3b31303838343930303130373333934313533d323430383232363030303730303033f33" \
    --generation-attributes Algorithm=HMAC_SHA256
```

Output:

```
{
    "KeyArn": "arn:aws:payment-cryptography:us-east-2:123456789012:key/
kwapwa6qaifllw2h,
    "KeyCheckValue": "2976E7",
    "Mac": "ED87F26E961C6D0DDB78DA5038AA2BDDEA0DCE03E5B5E96BDDD494F4A7AA470C"
}
```

Per ulteriori informazioni, consulta Generare l'indirizzo MAC nella Guida per l'utente di AWS Payment Cryptography (APC).

- Per i dettagli sull'API, vedere in Command Reference. GenerateMacAWS CLI

`generate-pin-data`

Il seguente esempio di codice mostra come utilizzare`generate-pin-data`.

AWS CLI

Come generare un PIN

L'esempio `generate-card-validation-data` seguente genera un nuovo PIN casuale utilizzando lo schema PIN Visa.

```
aws payment-cryptography-data generate-pin-data \
    --generation-key-identifier arn:aws:payment-cryptography:us-
east-2:111122223333:key/37y2tsl45p5zjbh2 \
```

```
    --encryption-key-identifier arn:aws:payment-cryptography:us-
east-2:111122223333:key/ivi5ksfsuplneuyt \
    --primary-account-number 171234567890123 \
    --pin-block-format ISO_FORMAT_0 \
    --generation-attributes VisaPin={PinVerificationKeyIndex=1}
```

Output:

```
{
    "GenerationKeyArn": "arn:aws:payment-cryptography:us-
east-2:111122223333:key/37y2tsl45p5zjbh2",
    "GenerationKeyCheckValue": "7F2363",
    "EncryptionKeyArn": "arn:aws:payment-cryptography:us-east-2:111122223333:key/
ivi5ksfsuplneuyt",
    "EncryptionKeyCheckValue": "7CC9E2",
    "EncryptedPinBlock": "AC17DC148BDA645E",
    "PinData": {
        "VerificationValue": "5507"
    }
}
```

Per ulteriori informazioni, consulta Generare i dati del PIN nella Guida per l'utente di AWS Payment Cryptography (APC).

- Per i dettagli sull'API, consulta GeneratePinData AWS CLICommand Reference.

re-encrypt-data

Il seguente esempio di codice mostra come utilizzare`re-encrypt-data`.

AWS CLI

Come crittografare nuovamente i dati con una chiave diversa

L'esempio `re-encrypt-data` seguente decrittografa il testo cifrato che è stato crittografato utilizzando una chiave simmetrica AES e lo cripta nuovamente utilizzando una chiave Derived Unique Key Per Transaction (DUKPT).

```
aws payment-cryptography-data re-encrypt-data \
    --incoming-key-identifier arn:aws:payment-cryptography:us-
west-2:111122223333:key/hyvv7ymboitd4vfy \
```

```
    --outgoing-key-identifier arn:aws:payment-cryptography:us-
west-2:111122223333:key/jl6ythkcvzesbxen \
    --cipher-
text 4D2B0BDBA192D5AEFEAA5B3EC28E4A65383C313FFA25140101560F75FE1B99F27192A90980AB9334
 \
    --incoming-encryption-
attributes "Dukpt={Mode=ECB,KeySerialNumber=0123456789111111}" \
    --outgoing-encryption-attributes '{"Symmetric": {"Mode": "ECB"}}'
```

Output:

```
{
    "CipherText":
 "F94959DA30EEFF0C035483C6067667CF6796E3C1AD28C2B61F9CFEB772A8DD41C0D6822931E0D3B1",
    "KeyArn": "arn:aws:payment-cryptography:us-west-2:111122223333:key/
jl6ythkcvzesbxen",
    "KeyCheckValue": "2E8CD9"
}
```

Per ulteriori informazioni, consulta Crittografia e decrittografia dei dati nella Guida per l'utente di AWS Payment Cryptography (APC).

- Per i dettagli sull'API, consulta ReEncryptData AWS CLICommand Reference.

`translate-pin-data`

Il seguente esempio di codice mostra come utilizzare`translate-pin-data`.

AWS CLI

Come tradurre i dati PIN

L'esempio `translate-pin-data` seguente traduce un PIN dalla crittografia PEK TDES utilizzando un blocco PIN ISO 0 in un blocco PIN AES ISO 4 utilizzando l'algoritmo DUKPT.

```
aws payment-cryptography-data translate-pin-data \
    --encrypted-pin-block "AC17DC148BDA645E" \
    --incoming-translation-
attributes=IsoFormat0='{PrimaryAccountNumber=171234567890123}' \
    --incoming-key-identifier arn:aws:payment-cryptography:us-
east-2:111122223333:key/ivi5ksfsuplneuyt \
    --outgoing-key-identifier arn:aws:payment-cryptography:us-
east-2:111122223333:key/4pmyquwjs3yj4vwe \
```

```
    --outgoing-translation-attributes
 IsoFormat4="{PrimaryAccountNumber=171234567890123}" \
    --outgoing-dukpt-attributes KeySerialNumber="FFFF9876543210E00008"
```

Output:

```
{
    "PinBlock": "1F4209C670E49F83E75CC72E81B787D9",
    "KeyArn": "arn:aws:payment-cryptography:us-east-2:111122223333:key/
ivi5ksfsuplneuyt
    "KeyCheckValue": "7CC9E2"
}
```

Per ulteriori informazioni, consulta Tradurre i dati del PIN nella Guida per l'utente di AWS Payment Cryptography (APC).

- Per i dettagli sull'API, consulta TranslatePinData AWS CLICommand Reference.

`verify-auth-request-cryptogram`

Il seguente esempio di codice mostra come utilizzare`verify-auth-request-cryptogram`.

AWS CLI

Come verificare una richiesta di autenticazione

L'esempio `verify-auth-request-cryptogram` seguente verifica un Authorization Request Cryptogram (ARQC).

```
aws payment-cryptography-data verify-auth-request-cryptogram \
    --auth-request-cryptogram F6E1BD1E6037FB3E \
    --auth-response-attributes '{"ArpcMethod1": {"AuthResponseCode": "1111"}}' \
    --key-identifier arn:aws:payment-cryptography:us-west-2:111122223333:key/
pboipdfzd4mdklya \
    --major-key-derivation-mode "EMV_OPTION_A" \
    --session-key-derivation-attributes '{"EmvCommon":
 {"ApplicationTransactionCounter": "1234","PanSequenceNumber":
 "01","PrimaryAccountNumber": "471234567890123"}}' \
    --transaction-data "123456789ABCDEF"
```

Output:

```
{
    "AuthResponseValue": "D899B8C6FBF971AA",
    "KeyArn": "arn:aws:payment-cryptography:us-west-2:111122223333:key/
pboipdfzd4mdklya",
    "KeyCheckValue": "985792"
}
```

Per ulteriori informazioni, consulta Crittogramma di verifica della richiesta di autorizzazione (ARQC) nella Guida per l'utente di AWS Payment Cryptography (APC).

- Per i dettagli sull'API, consulta VerifyAuthRequestCryptogram AWS CLICommand Reference.

`verify-card-validation-data`

Il seguente esempio di codice mostra come utilizzare`verify-card-validation-data`.

AWS CLI

Come convalidare un CVV

L'`verify-card-validation-data`esempio seguente convalida un CVV/ CVV2 per un PAN.

```
aws payment-cryptography-data verify-card-validation-data \
    --key-identifier arn:aws:payment-cryptography:us-east-2:111122223333:key/
tqv5yij6wtxx64pi \
    --primary-account-number=171234567890123 \
    --verification-attributes CardVerificationValue2={CardExpiryDate=0123} \
    --validation-data 801
```

Output:

```
{
    "KeyArn": "arn:aws:payment-cryptography:us-east-2:111122223333:key/
tqv5yij6wtxx64pi",
    "KeyCheckValue": "CADDA1"
}
```

Per ulteriori informazioni, consulta Verificare i dati della carta nella Guida per l'utente di AWS Payment Cryptography (APC).

- Per i dettagli sull'API, vedere VerifyCardValidationDatain Command Reference.AWS CLI

`verify-mac`

Il seguente esempio di codice mostra come utilizzare`verify-mac`.

AWS CLI

Come verificare un MAC

L'`verify-mac`esempio seguente verifica un codice di autenticazione dei messaggi basato su hash (HMAC) per l'autenticazione dei dati delle carte utilizzando l'algoritmo HMAC_ SHA256 e una chiave di crittografia HMAC.

```
aws payment-cryptography-data verify-mac \
    --key-identifier arn:aws:payment-cryptography:us-east-2:111122223333:key/
qnobl5lghrzunce6 \
    --message-
data "3b343038383439303031303733393431353d323430383232363030303730303f33" \
    --verification-attributes='Algorithm=HMAC_SHA256' \
    --mac ED87F26E961C6D0DDB78DA5038AA2BDDEA0DCE03E5B5E96BDDD494F4A7AA470C
```

Output:

```
{
    "KeyArn": "arn:aws:payment-cryptography:us-east-2:111122223333:key/
qnobl5lghrzunce6,
    "KeyCheckValue": "2976E7",
}
```

Per ulteriori informazioni, consulta Verificare l'indirizzo MAC nella Guida per l'utente di AWS Payment Cryptography (APC).

- Per i dettagli sull'API, vedere in Command Reference. VerifyMacAWS CLI

`verify-pin-data`

Il seguente esempio di codice mostra come utilizzare`verify-pin-data`.

AWS CLI

Come verificare un PIN

L'esempio `verify-pin-data` seguente convalida un PIN per un PAN.

```
aws payment-cryptography-data verify-pin-data \
    --verification-key-identifier arn:aws:payment-cryptography:us-
east-2:111122223333:key/37y2tsl45p5zjbh2 \
    --encryption-key-identifier arn:aws:payment-cryptography:us-
east-2:111122223333:key/ivi5ksfsuplneuyt \
    --primary-account-number 171234567890123 \
    --pin-block-format ISO_FORMAT_0 \
    --verification-attributes
 VisaPin="{PinVerificationKeyIndex=1,VerificationValue=5507}" \
    --encrypted-pin-block AC17DC148BDA645E
```

Output:

```
{
    "VerificationKeyArn": "arn:aws:payment-cryptography:us-
east-2:111122223333:key/37y2tsl45p5zjbh2",
    "VerificationKeyCheckValue": "7F2363",
    "EncryptionKeyArn": "arn:aws:payment-cryptography:us-east-2:111122223333:key/
ivi5ksfsuplneuyt",
    "EncryptionKeyCheckValue": "7CC9E2",
}
```

Per ulteriori informazioni, consulta Verificare i dati del PIN nella Guida per l'utente di AWS Payment Cryptography.

- Per i dettagli sull'API, consulta VerifyPinData AWS CLICommand Reference.

Esempi di utilizzo di Amazon Pinpoint AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando Amazon Pinpoint. AWS Command Line Interface

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`create-app`

Il seguente esempio di codice mostra come usare`create-app`.

AWS CLI

Esempio 1: creazione di un'applicazione

L'esempio `create-app` seguente crea una nuova applicazione (progetto).

```
aws pinpoint create-app \
    --create-application-request Name=ExampleCorp
```

Output:

```
{
    "ApplicationResponse": {
        "Arn": "arn:aws:mobiletargeting:us-
west-2:AIDACKCEVSQ6C2EXAMPLE:apps/810c7aab86d42fb2b56c8c966example",
        "Id": "810c7aab86d42fb2b56c8c966example",
        "Name": "ExampleCorp",
        "tags": {}
    }
}
```

Esempio 2: creazione di un'applicazione con tag

L'esempio `create-app` seguente crea una nuova applicazione (progetto) e associa un tag (chiave e valore) all'applicazione.

```
aws pinpoint create-app \
    --create-application-request Name=ExampleCorp,tags={"Stack"="Test"}
```

Output:

```
{
    "ApplicationResponse": {
        "Arn": "arn:aws:mobiletargeting:us-
west-2:AIDACKCEVSQ6C2EXAMPLE:apps/810c7aab86d42fb2b56c8c966example",
        "Id": "810c7aab86d42fb2b56c8c966example",
```

```
        "Name": "ExampleCorp",
        "tags": {
            "Stack": "Test"
        }
    }
}
```

- Per i dettagli sull'API, consulta CreateApp AWS CLICommand Reference.

`create-sms-template`

Il seguente esempio di codice mostra come utilizzare`create-sms-template`.

AWS CLI

Crea un modello di messaggio per i messaggi inviati attraverso il canale SMS

L'esempio `create-sms-template` seguente crea un modello di messaggio SMS.

```
aws pinpoint create-sms-template \
    --template-name TestTemplate \
    --sms-template-request file://myfile.json \
    --region us-east-1
```

Contenuto di `myfile.json`:

```
{
    "Body": "hello\n how are you?\n food is good",
    "TemplateDescription": "Test SMS Template"
}
```

Output:

```
{
    "CreateTemplateMessageBody": {
        "Arn": "arn:aws:mobiletargeting:us-east-1:AIDACKCEVSQ6C2EXAMPLE:templates/
TestTemplate/SMS",
        "Message": "Created",
        "RequestID": "8c36b17f-a0b0-400f-ac21-29e9b62a975d"
    }
}
```

Per ulteriori informazioni, consulta Modelli di messaggio di Amazon Pinpoint nella Guida per l'utente di Amazon Pinpoint.

- Per i dettagli sull'API, consulta CreateSmsTemplate AWS CLICommand Reference.

`delete-app`

Il seguente esempio di codice mostra come utilizzare`delete-app`.

AWS CLI

Come eliminare un'applicazione

L'esempio `delete-app` seguente elimina un'applicazione (progetto).

```
aws pinpoint delete-app \
    --application-id 810c7aab86d42fb2b56c8c966example
```

Output:

```
{
    "ApplicationResponse": {
        "Arn": "arn:aws:mobiletargeting:us-
west-2:AIDACKCEVSQ6C2EXAMPLE:apps/810c7aab86d42fb2b56c8c966example",
        "Id": "810c7aab86d42fb2b56c8c966example",
        "Name": "ExampleCorp",
        "tags": {}
    }
}
```

- Per i dettagli sull'API, consulta DeleteApp AWS CLICommand Reference.

`get-apns-channel`

Il seguente esempio di codice mostra come utilizzare`get-apns-channel`.

AWS CLI

Per recuperare informazioni sullo stato e sulle impostazioni del APNs canale per un'applicazione

L'`get-apns-channel`esempio seguente recupera informazioni sullo stato e le impostazioni del APNs canale per un'applicazione.

```
aws pinpoint get-apns-channel \
    --application-id 9ab1068eb0a6461c86cce7f27ce0efd7 \
    --region us-east-1
```

Output:

```
{
    "APNSChannelResponse": {
        "ApplicationId": "9ab1068eb0a6461c86cce7f27ce0efd7",
        "CreationDate": "2019-05-09T21:54:45.082Z",
        "DefaultAuthenticationMethod": "CERTIFICATE",
        "Enabled": true,
        "HasCredential": true,
        "HasTokenKey": false,
        "Id": "apns",
        "IsArchived": false,
        "LastModifiedDate": "2019-05-09T22:04:01.067Z",
        "Platform": "APNS",
        "Version": 2
    }
}
```

- Per i dettagli sull'API, vedere GetApnsChannelin AWS CLI Command Reference.

`get-app`

Il seguente esempio di codice mostra come utilizzare`get-app`.

AWS CLI

Come recuperare informazioni su un'applicazione (progetto)

L'esempio `get-app` seguente recupera informazioni su un'applicazione (progetto).

```
aws pinpoint get-app \
    --application-id 810c7aab86d42fb2b56c8c966example \
    --region us-east-1
```

Output:

```
{
```

```
    "ApplicationResponse": {
        "Arn": "arn:aws:mobiletargeting:us-
east-1:AIDACKCEVSQ6C2EXAMPLE:apps/810c7aab86d42fb2b56c8c966example",
        "Id": "810c7aab86d42fb2b56c8c966example",
        "Name": "ExampleCorp",
        "tags": {
                "Year": "2019",
                "Stack": "Production"
            }
    }
}
```

- Per i dettagli sull'API, consulta GetApp AWS CLICommand Reference.

`get-apps`

Il seguente esempio di codice mostra come utilizzare`get-apps`.

AWS CLI

Come recuperare informazioni su tutte le applicazioni

L'esempio `get-apps` seguente recupera informazioni su tutte le tue applicazioni (progetti).

```
aws pinpoint get-apps
```

Output:

```
{
    "ApplicationsResponse": {
        "Item": [
            {
                "Arn": "arn:aws:mobiletargeting:us-
west-2:AIDACKCEVSQ6C2EXAMPLE:apps/810c7aab86d42fb2b56c8c966example",
                "Id": "810c7aab86d42fb2b56c8c966example",
                "Name": "ExampleCorp",
                "tags": {
                    "Year": "2019",
                    "Stack": "Production"
                }
            },
            {
```

```
                "Arn": "arn:aws:mobiletargeting:us-
west-2:AIDACKCEVSQ6C2EXAMPLE:apps/42d8c7eb0990a57ba1d5476a3example",
                "Id": "42d8c7eb0990a57ba1d5476a3example",
                "Name": "AnyCompany",
                "tags": {}
            },
            {
                "Arn": "arn:aws:mobiletargeting:us-
west-2:AIDACKCEVSQ6C2EXAMPLE:apps/80f5c382b638ffe5ad12376bbexample",
                "Id": "80f5c382b638ffe5ad12376bbexample",
                "Name": "ExampleCorp_Test",
                "tags": {
                    "Year": "2019",
                    "Stack": "Test"
                }
            }
        ],
        "NextToken":
 "eyJDcmVhdGlvbkRhdGUiOiIyMDE5LTA3LTE2VDE0OjM4OjUzLjkwM1oiLCJBY2NvdW50SWQiOiI1MTIzOTcxODM4Nz
    }
}
```


La presenza del valore di risposta `NextToken` indica che è disponibile una maggiore quantità di output. Chiama il comando e fornisci quel valore come parametro di input di `NextToken`.

- Per i dettagli sull'API, consulta GetApps AWS CLICommand Reference.

`get-campaign`

Il seguente esempio di codice mostra come utilizzare`get-campaign`.

AWS CLI

Come recuperare informazioni sullo stato, la configurazione e altre impostazioni di una campagna

L'esempio `get-campaign` seguente recupera informazioni sullo stato, sulla configurazione e su altre impostazioni per una campagna.

```
aws pinpoint get-campaign \
    --application-id 6e0b7591a90841d2b5d93fa11143e5a7 \
    --campaign-id a1e63c6cc0eb43ed826ffcc3cc90b30d \
    --region us-east-1
```

Output:


```
{
    "CampaignResponse": {
        "AdditionalTreatments": [],
        "ApplicationId": "6e0b7591a90841d2b5d93fa11143e5a7",
        "Arn": "arn:aws:mobiletargeting:us-
east-1:AIDACKCEVSQ6C2EXAMPLE:apps/6e0b7591a90841d2b5d93fa11143e5a7/campaigns/
a1e63c6cc0eb43ed826ffcc3cc90b30d",
        "CreationDate": "2019-10-08T18:40:16.581Z",
        "Description": " ",
        "HoldoutPercent": 0,
        "Id": "a1e63c6cc0eb43ed826ffcc3cc90b30d",
        "IsPaused": false,
        "LastModifiedDate": "2019-10-08T18:40:16.581Z",
        "Limits": {
            "Daily": 0,
            "MaximumDuration": 60,
            "MessagesPerSecond": 50,
            "Total": 0
        },
        "MessageConfiguration": {
            "EmailMessage": {
                "FromAddress": "sender@example.com",
                "HtmlBody": "<!DOCTYPE html>\n  <html lang=\"en\">\n    <head>\n
 <meta http-equiv=\"Content-Type\" content=\"text/html; charset=utf-8\" />\n</head>
\n<body>Hello</body>\n</html>",
                "Title": "PinpointDemo"
            }
        },
        "Name": "MyCampaign",
        "Schedule": {
            "IsLocalTime": false,
            "StartTime": "IMMEDIATE",
            "Timezone": "utc"
        },
        "SegmentId": "b66c9e42f71444b2aa2e0ffc1df28f60",
        "SegmentVersion": 1,
        "State": {
            "CampaignStatus": "COMPLETED"
        },
        "tags": {},
        "TemplateConfiguration": {},
        "Version": 1
```

```
    }
}
```

- Per i dettagli sull'API, consulta GetCampaign AWS CLICommand Reference.

`get-campaigns`

Il seguente esempio di codice mostra come utilizzare`get-campaigns`.

AWS CLI

Come recuperare informazioni sullo stato, la configurazione e altre impostazioni per tutte le campagne associate a un'applicazione

L'esempio `get-campaigns` seguente recupera informazioni sullo stato, la configurazione e altre impostazioni per tutte le campagne associate a un'applicazione.

```
aws pinpoint get-campaigns \
    --application-id 6e0b7591a90841d2b5d93fa11143e5a7 \
    --region us-east-1
```

Output:

```
{
    "CampaignsResponse": {
        "Item": [
            {
                "AdditionalTreatments": [],
                "ApplicationId": "6e0b7591a90841d2b5d93fa11143e5a7",
                "Arn": "arn:aws:mobiletargeting:us-
east-1:AIDACKCEVSQ6C2EXAMPLE:apps/6e0b7591a90841d2b5d93fa11143e5a7/
campaigns/7e1280344c8f4a9aa40a00b006fe44f1",
                "CreationDate": "2019-10-08T18:40:22.905Z",
                "Description": " ",
                "HoldoutPercent": 0,
                "Id": "7e1280344c8f4a9aa40a00b006fe44f1",
                "IsPaused": false,
                "LastModifiedDate": "2019-10-08T18:40:22.905Z",
                "Limits": {},
                "MessageConfiguration": {
                    "EmailMessage": {
                        "FromAddress": "sender@example.com",
```

```
                        "HtmlBody": "<!DOCTYPE html>\n    <html lang=\"en
\">\n    <head>\n    <meta http-equiv=\"Content-Type\" content=\"text/html;
 charset=utf-8\" />\n</head>\n<body>Hello</body>\n</html>",
                        "Title": "PInpointDemo Test"
                    }
                },
                "Name": "MyCampaign1",
                "Schedule": {
                    "IsLocalTime": false,
                    "QuietTime": {},
                    "StartTime": "IMMEDIATE",
                    "Timezone": "UTC"
                },
                "SegmentId": "b66c9e42f71444b2aa2e0ffc1df28f60",
                "SegmentVersion": 1,
                "State": {
                    "CampaignStatus": "COMPLETED"
                },
                "tags": {},
                "TemplateConfiguration": {},
                "Version": 1
            },
            {
                "AdditionalTreatments": [],
                "ApplicationId": "6e0b7591a90841d2b5d93fa11143e5a7",
                "Arn": "arn:aws:mobiletargeting:us-
east-1:AIDACKCEVSQ6C2EXAMPLE:apps/6e0b7591a90841d2b5d93fa11143e5a7/campaigns/
a1e63c6cc0eb43ed826ffcc3cc90b30d",
                "CreationDate": "2019-10-08T18:40:16.581Z",
                "Description": " ",
                "HoldoutPercent": 0,
                "Id": "a1e63c6cc0eb43ed826ffcc3cc90b30d",
                "IsPaused": false,
                "LastModifiedDate": "2019-10-08T18:40:16.581Z",
                "Limits": {
                    "Daily": 0,
                    "MaximumDuration": 60,
                    "MessagesPerSecond": 50,
                    "Total": 0
                },
                "MessageConfiguration": {
                    "EmailMessage": {
                        "FromAddress": "sender@example.com",
```

```
                        "HtmlBody": "<!DOCTYPE html>\n    <html lang=\"en
\">\n    <head>\n    <meta http-equiv=\"Content-Type\" content=\"text/html;
 charset=utf-8\" />\n</head>\n<body>Demo</body>\n</html>",
                        "Title": "PinpointDemo"
                    }
                },
                "Name": "MyCampaign2",
                "Schedule": {
                    "IsLocalTime": false,
                    "StartTime": "IMMEDIATE",
                    "Timezone": "utc"
                },
                "SegmentId": "b66c9e42f71444b2aa2e0ffc1df28f60",
                "SegmentVersion": 1,
                "State": {
                    "CampaignStatus": "COMPLETED"
                },
                "tags": {},
                "TemplateConfiguration": {},
                "Version": 1
            }
        ]
    }
}
```

- Per i dettagli sull'API, consulta GetCampaigns AWS CLICommand Reference.

`get-channels`

Il seguente esempio di codice mostra come utilizzare`get-channels`.

AWS CLI

Come recuperare le informazioni sulla cronologia e lo stato di ciascun canale per un'applicazione

L'esempio `get-channels` seguente recupera le informazioni sulla cronologia e lo stato di ciascun canale per un'applicazione.

```
aws pinpoint get-channels \
    --application-id 6e0b7591a90841d2b5d93fa11143e5a7 \
    --region us-east-1
```

Output:

```
{
    "ChannelsResponse": {
        "Channels": {
            "GCM": {
                "ApplicationId": "6e0b7591a90841d2b5d93fa11143e5a7",
                "CreationDate": "2019-10-08T18:28:23.182Z",
                "Enabled": true,
                "HasCredential": true,
                "Id": "gcm",
                "IsArchived": false,
                "LastModifiedDate": "2019-10-08T18:28:23.182Z",
                "Version": 1
            },
            "SMS": {
                "ApplicationId": "6e0b7591a90841d2b5d93fa11143e5a7",
                "CreationDate": "2019-10-08T18:39:18.511Z",
                "Enabled": true,
                "Id": "sms",
                "IsArchived": false,
                "LastModifiedDate": "2019-10-08T18:39:18.511Z",
                "Version": 1
            },
            "EMAIL": {
                "ApplicationId": "6e0b7591a90841d2b5d93fa11143e5a7",
                "CreationDate": "2019-10-08T18:27:23.990Z",
                "Enabled": true,
                "Id": "email",
                "IsArchived": false,
                "LastModifiedDate": "2019-10-08T18:27:23.990Z",
                "Version": 1
            },
            "IN_APP": {
                "Enabled": true,
                "IsArchived": false,
                "Version": 0
            }
        }
    }
}
```

- Per i dettagli sull'API, consulta GetChannels AWS CLICommand Reference.

`get-email-channel`

Il seguente esempio di codice mostra come utilizzare`get-email-channel`.

AWS CLI

Come recuperare le informazioni sullo stato e sulle impostazioni del canale di e-mail per un'applicazione

L'esempio `get-email-channel` seguente recupera lo stato e le impostazioni del canale di e-mail per un'applicazione.

```
aws pinpoint get-email-channel \
    --application-id 6e0b7591a90841d2b5d93fa11143e5a7 \
    --region us-east-1
```

Output:

```
{
    "EmailChannelResponse": {
        "ApplicationId": "6e0b7591a90841d2b5d93fa11143e5a7",
        "CreationDate": "2019-10-08T18:27:23.990Z",
        "Enabled": true,
        "FromAddress": "sender@example.com",
        "Id": "email",
        "Identity": "arn:aws:ses:us-east-1:AIDACKCEVSQ6C2EXAMPLE:identity/
sender@example.com",
        "IsArchived": false,
        "LastModifiedDate": "2019-10-08T18:27:23.990Z",
        "MessagesPerSecond": 1,
        "Platform": "EMAIL",
        "RoleArn": "arn:aws:iam::AIDACKCEVSQ6C2EXAMPLE:role/pinpoint-events",
        "Version": 1
    }
}
```

- Per i dettagli sull'API, consulta GetEmailChannel AWS CLICommand Reference.

`get-endpoint`

Il seguente esempio di codice mostra come utilizzare`get-endpoint`.

AWS CLI

Recupero di informazioni sulle impostazioni e gli attributi di un endpoint specifico per un'applicazione

L'esempio `get-endpoint` seguente recupera le informazioni sulle impostazioni e gli attributi di un endpoint specifico per un'applicazione.

```
aws pinpoint get-endpoint \
    --application-id 611e3e3cdd47474c9c1399a505665b91 \
    --endpoint-id testendpoint \
    --region us-east-1
```

Output:

```
{
    "EndpointResponse": {
        "Address": "+11234567890",
        "ApplicationId": "611e3e3cdd47474c9c1399a505665b91",
        "Attributes": {},
        "ChannelType": "SMS",
        "CohortId": "63",
        "CreationDate": "2019-01-28T23:55:11.534Z",
        "EffectiveDate": "2021-08-06T00:04:51.763Z",
        "EndpointStatus": "ACTIVE",
        "Id": "testendpoint",
        "Location": {
            "Country": "USA"
        },
        "Metrics": {
            "SmsDelivered": 1.0
        },
        "OptOut": "ALL",
        "RequestId": "a204b1f2-7e26-48a7-9c80-b49a2143489d",
        "User": {
            "UserAttributes": {
                "Age": [
                    "24"
                ]
            },
        "UserId": "testuser"
        }
    }
```

```
}
```

- Per i dettagli sull'API, consulta GetEndpoint AWS CLICommand Reference.

`get-gcm-channel`

Il seguente esempio di codice mostra come utilizzare`get-gcm-channel`.

AWS CLI

Come recuperare le informazioni sullo stato e sulle impostazioni del canale GCM per un'applicazione

L'esempio `get-gcm-channel` seguente recupera le informazioni sullo stato e le impostazioni sul canale GCM per un'applicazione.

```
aws pinpoint get-gcm-channel \
    --application-id 6e0b7591a90841d2b5d93fa11143e5a7 \
    --region us-east-1
```

Output:

```
{
    "GCMChannelResponse": {
        "ApplicationId": "6e0b7591a90841d2b5d93fa11143e5a7",
        "CreationDate": "2019-10-08T18:28:23.182Z",
        "Enabled": true,
        "HasCredential": true,
        "Id": "gcm",
        "IsArchived": false,
        "LastModifiedDate": "2019-10-08T18:28:23.182Z",
        "Platform": "GCM",
        "Version": 1
    }
}
```

- Per i dettagli sull'API, consulta GetGcmChannel AWS CLICommand Reference.

`get-sms-channel`

Il seguente esempio di codice mostra come utilizzare`get-sms-channel`.

AWS CLI

Recupero delle informazioni sullo stato e sulle impostazioni del canale SMS per un'applicazione

L'esempio `get-sms-channel` seguente recupera lo stato e le impostazioni del canale di sms per un'applicazione.

```
aws pinpoint get-sms-channel \
    --application-id 6e0b7591a90841d2b5d93fa11143e5a7 \
    --region us-east-1
```

Output:

```
{
    "SMSChannelResponse": {
        "ApplicationId": "6e0b7591a90841d2b5d93fa11143e5a7",
        "CreationDate": "2019-10-08T18:39:18.511Z",
        "Enabled": true,
        "Id": "sms",
        "IsArchived": false,
        "LastModifiedDate": "2019-10-08T18:39:18.511Z",
        "Platform": "SMS",
        "PromotionalMessagesPerSecond": 20,
        "TransactionalMessagesPerSecond": 20,
        "Version": 1
    }
}
```

- Per i dettagli sull'API, consulta GetSmsChannel AWS CLICommand Reference.

`get-sms-template`

Il seguente esempio di codice mostra come utilizzare`get-sms-template`.

AWS CLI

Recupera il contenuto e le impostazioni di un modello di messaggio per i messaggi inviati attraverso il canale SMS

L'esempio `get-sms-template` seguente recupera il contenuto e le impostazioni di un modello di messaggio SMS.

```
aws pinpoint get-sms-template \
    --template-name TestTemplate \
    --region us-east-1
```

Output:

```
{
    "SMSTemplateResponse": {
        "Arn": "arn:aws:mobiletargeting:us-east-1:AIDACKCEVSQ6C2EXAMPLE:templates/
TestTemplate/SMS",
        "Body": "hello\n how are you?\n food is good",
        "CreationDate": "2023-06-20T21:37:30.124Z",
        "LastModifiedDate": "2023-06-20T21:37:30.124Z",
        "tags": {},
        "TemplateDescription": "Test SMS Template",
        "TemplateName": "TestTemplate",
        "TemplateType": "SMS",
        "Version": "1"
    }
}
```

Per ulteriori informazioni, consulta Modelli di messaggio di Amazon Pinpoint nella Guida per l'utente di Amazon Pinpoint.

- Per i dettagli sull'API, consulta GetSmsTemplate AWS CLICommand Reference.

`get-voice-channel`

Il seguente esempio di codice mostra come utilizzare`get-voice-channel`.

AWS CLI

Come recuperare le informazioni sullo stato e sulle impostazioni del canale vocale per un'applicazione

L'esempio `get-voice-channel` seguente recupera lo stato e le impostazioni del canale vocale per un'applicazione.

```
aws pinpoint get-voice-channel \
    --application-id 6e0b7591a90841d2b5d93fa11143e5a7 \
    --region us-east-1
```

Output:

```
{
    "VoiceChannelResponse": {
        "ApplicationId": "6e0b7591a90841d2b5d93fa11143e5a7",
        "CreationDate": "2022-04-28T00:17:03.836Z",
        "Enabled": true,
        "Id": "voice",
        "IsArchived": false,
        "LastModifiedDate": "2022-04-28T00:17:03.836Z",
        "Platform": "VOICE",
        "Version": 1
    }
}
```

- Per i dettagli sull'API, consulta GetVoiceChannel AWS CLICommand Reference.

`list-tags-for-resource`

Il seguente esempio di codice mostra come utilizzare`list-tags-for-resource`.

AWS CLI

Come recuperare un elenco di tag per una risorsa

L'esempio `list-tags-for-resource` seguente recupera tutti i tag (nomi e valori delle chiavi) associati alla risorsa specificata.

```
aws pinpoint list-tags-for-resource \
    --resource-arn arn:aws:mobiletargeting:us-
west-2:AIDACKCEVSQ6C2EXAMPLE:apps/810c7aab86d42fb2b56c8c966example
```

Output:

```
{
    "TagsModel": {
        "tags": {
            "Year": "2019",
            "Stack": "Production"
        }
    }
```

```
}
```

Per ulteriori informazioni, consulta 'Tagging Amazon Pinpoint Resources https://docs.aws.amazon.com/pinpoint/ latest/developerguide/tagging < -resources.html>'__ nella Amazon Pinpoint Developer Guide.

- Per i ListTagsForResourcedettagli sull'AWS CLI API, consulta Command Reference.

`phone-number-validate`

Il seguente esempio di codice mostra come utilizzare`phone-number-validate`.

AWS CLI

Recupera informazioni su un numero di telefono

L'esempio `phone-number-validate` seguente recupera le informazioni relative a un numero di telefono.

```
aws pinpoint phone-number-validate \
    --number-validate-request PhoneNumber="+12065550142" \
    --region us-east-1
```

Output:

```
{
    "NumberValidateResponse": {
        "Carrier": "ExampleCorp Mobile",
        "City": "Seattle",
        "CleansedPhoneNumberE164": "+12065550142",
        "CleansedPhoneNumberNational": "2065550142",
        "Country": "United States",
        "CountryCodeIso2": "US",
        "CountryCodeNumeric": "1",
        "OriginalPhoneNumber": "+12065550142",
        "PhoneType": "MOBILE",
        "PhoneTypeCode": 0,
        "Timezone": "America/Los_Angeles",
        "ZipCode": "98101"
    }
}
```

Per ulteriori informazioni, consulta Canale SMS di Amazon Pinpoint nella Guida per l'utente di Amazon Pinpoint.

- Per i dettagli sull'API, consulta PhoneNumberValidate AWS CLICommand Reference.

`send-messages`

Il seguente esempio di codice mostra come utilizzare`send-messages`.

AWS CLI

Invio di un messaggio SMS utilizzando l'endpoint di un'applicazione

L'esempio `send-messages` seguente invia un messaggio diretto per un'applicazione con un endpoint.

```
aws pinpoint send-messages \
    --application-id 611e3e3cdd47474c9c1399a505665b91 \
    --message-request file://myfile.json \
    --region us-west-2
```

Contenuto di `myfile.json`:

```
{
    "MessageConfiguration": {
        "SMSMessage": {
            "Body": "hello, how are you?"
        }
    },
    "Endpoints": {
        "testendpoint": {}
    }
}
```

Output:

```
{
    "MessageResponse": {
        "ApplicationId": "611e3e3cdd47474c9c1399a505665b91",
        "EndpointResult": {
            "testendpoint": {
```

```
                "Address": "+12345678900",
                "DeliveryStatus": "SUCCESSFUL",
                "MessageId": "itnuqhai5alf1n6ahv3udc05n7hhddr6gb3lq6g0",
                "StatusCode": 200,
                "StatusMessage": "MessageId:
 itnuqhai5alf1n6ahv3udc05n7hhddr6gb3lq6g0"
            }
        },
        "RequestId": "c7e23264-04b2-4a46-b800-d24923f74753"
    }
}
```

Per ulteriori informazioni, consulta Canale SMS di Amazon Pinpoint nella Guida per l'utente di Amazon Pinpoint.

- Per i dettagli sull'API, consulta SendMessages AWS CLICommand Reference.

`send-users-messages`

Il seguente esempio di codice mostra come utilizzare`send-users-messages`.

AWS CLI

Come inviare un messaggio SMS a un utente di un'applicazione

L'esempio `send-users-messages` seguente invia un messaggio diretto per un utente di un'applicazione.

```
aws pinpoint send-users-messages \
    --application-id 611e3e3cdd47474c9c1399a505665b91 \
    --send-users-message-request file://myfile.json \
    --region us-west-2
```

Contenuto di `myfile.json`:

```
{
    "MessageConfiguration": {
        "SMSMessage": {
            "Body": "hello, how are you?"
        }
    },
```

```
    "Users": {
        "testuser": {}
    }
}
```

Output:

```
{
    "SendUsersMessageResponse": {
        "ApplicationId": "611e3e3cdd47474c9c1399a505665b91",
        "RequestId": "e0b12cf5-2359-11e9-bb0b-d5fb91876b25",
        "Result": {
            "testuser": {
                "testuserendpoint": {
                    "DeliveryStatus": "SUCCESSFUL",
                    "MessageId": "7qu4hk5bqhda3i7i2n4pjf98qcuh8b7p45ifsmo0",
                    "StatusCode": 200,
                    "StatusMessage": "MessageId:
 7qu4hk5bqhda3i7i2n4pjf98qcuh8b7p45ifsmo0",
                    "Address": "+12345678900"
                }
            }
        }
    }
}
```

Per ulteriori informazioni, consulta Canale SMS di Amazon Pinpoint nella Guida per l'utente di Amazon Pinpoint.

- Per i dettagli sull'API, consulta SendUsersMessages AWS CLICommand Reference.

`tag-resource`

Il seguente esempio di codice mostra come utilizzare`tag-resource`.

AWS CLI

Come aggiungere tag a una risorsa

L'esempio seguente aggiunge due tag (nomi e valori di chiave) a una risorsa.

```
aws pinpoint list-tags-for-resource \
```

```
    --resource-arn arn:aws:mobiletargeting:us-
east-1:AIDACKCEVSQ6C2EXAMPLE:apps/810c7aab86d42fb2b56c8c966example \
    --tags-model tags={Stack=Production,Year=2019}
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta 'Tagging Amazon Pinpoint Resources https:// docs.aws.amazon.com/pinpoint/ latest/developerguide/tagging < -resources.html>'__ nella Amazon Pinpoint Developer Guide.

- Per i TagResourcedettagli sull'AWS CLI API, consulta Command Reference.

`untag-resource`

Il seguente esempio di codice mostra come utilizzare`untag-resource`.

AWS CLI

Esempio 1: come rimuovere un tag da una risorsa

L'esempio `untag-resource` seguente rimuove il tag (nome e valore di chiave) dalla risorsa specificata.

```
aws pinpoint untag-resource \
    --resource-arn arn:aws:mobiletargeting:us-
west-2:AIDACKCEVSQ6C2EXAMPLE:apps/810c7aab86d42fb2b56c8c966example \
    --tag-keys Year
```

Questo comando non produce alcun output.

Esempio 2: come rimuovere più tag da una risorsa

L'esempio `untag-resource` seguente rimuove i tag specificati (nomi e valori di chiave) da una risorsa.

```
aws pinpoint untag-resource \
    --resource-arn arn:aws:mobiletargeting:us-
east-1:AIDACKCEVSQ6C2EXAMPLE:apps/810c7aab86d42fb2b56c8c966example \
    --tag-keys Year Stack
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta 'Tagging Amazon Pinpoint Resources https://docs.aws.amazon.com/pinpoint/ latest/developerguide/tagging < -resources.html>'__ nella Amazon Pinpoint Developer Guide.

- Per i UntagResourcedettagli sull'AWS CLI API, consulta Command Reference.

`update-sms-channel`

Il seguente esempio di codice mostra come utilizzare`update-sms-channel`.

AWS CLI

Come abilitare il canale SMS o aggiornare lo stato e le impostazioni del canale SMS per un'applicazione.

L'esempio `update-sms-channel` seguente abilita il canale SMS per un canale SMS per un'applicazione.

```
aws pinpoint update-sms-channel \
    --application-id 611e3e3cdd47474c9c1399a505665b91 \
    --sms-channel-request Enabled=true \
    --region us-west-2
```

Output:

```
{
    "SMSChannelResponse": {
        "ApplicationId": "611e3e3cdd47474c9c1399a505665b91",
        "CreationDate": "2019-01-28T23:25:25.224Z",
        "Enabled": true,
        "Id": "sms",
        "IsArchived": false,
        "LastModifiedDate": "2023-05-18T23:22:50.977Z",
        "Platform": "SMS",
        "PromotionalMessagesPerSecond": 20,
        "TransactionalMessagesPerSecond": 20,
        "Version": 3
    }
}
```

Per ulteriori informazioni, consulta Canale SMS di Amazon Pinpoint nella Guida per l'utente di Amazon Pinpoint.

- Per i dettagli sull'API, consulta UpdateSmsChannel AWS CLICommand Reference.

Esempi di utilizzo di Amazon Polly AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando Amazon Polly. AWS Command Line Interface

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`delete-lexicon`

Il seguente esempio di codice mostra come usare`delete-lexicon`.

AWS CLI

Come eliminare un lessico

L'esempio `delete-lexicon` seguente elimina il lessico specificato.

```
aws polly delete-lexicon \
    --name w3c
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Utilizzo dell' DeleteLexicon operazione nella Amazon Polly Developer Guide.

- Per i dettagli sull'API, consulta DeleteLexicon AWS CLICommand Reference.

`get-lexicon`

Il seguente esempio di codice mostra come utilizzare`get-lexicon`.

AWS CLI

Come recuperare il contenuto di un lessico

L'esempio `get-lexicon` seguente recupera il contenuto del lessico di pronuncia specificato.

```
aws polly get-lexicon \
    --name w3c
```

Output:

```
{
    "Lexicon": {
        "Content": "<?xml version=\"1.0\" encoding=\"UTF-8\"?>\n<lexicon version=
\"1.0\" \n      xmlns=    \"http://www.w3.org/2005/01/pronunciation-lexicon
\"\n      xmlns:xsi=\"http://www.w3.org/2001/XMLSchema-instance\" \n
 xsi:schemaLocation=\"http://www.w3.org/2005/01/pronunciation-lexicon \n
 http://www.w3.org/TR/2007/CR-pronunciation-    lexicon-20071212/pls.xsd\"\n
   alphabet=\"ipa\" \n      xml:lang=\"en-US\">\n  <lexeme>\n    <grapheme>W3C</
grapheme>\n        <alias>World Wide Web Consortium</alias>\n  </lexeme>\n</lexicon>
\n",
        "Name": "w3c"
    },
    "LexiconAttributes": {
        "Alphabet": "ipa",
        "LanguageCode": "en-US",
        "LastModified": 1603908910.99,
        "LexiconArn": "arn:aws:polly:us-west-2:880185128111:lexicon/w3c",
        "LexemesCount": 1,
        "Size": 492
    }
}
```

Per ulteriori informazioni, consulta Utilizzo dell' GetLexicon operazione nella Amazon Polly Developer Guide.

- Per i dettagli sull'API, consulta GetLexicon AWS CLICommand Reference.

`get-speech-synthesis-task`

Il seguente esempio di codice mostra come utilizzare`get-speech-synthesis-task`.

AWS CLI

Come ottenere informazioni su un'attività di sintesi vocale

L'esempio `get-speech-synthesis-task` seguente recupera le informazioni sull'attività di sintesi vocale specificata.

```
aws polly get-speech-synthesis-task \
    --task-id 70b61c0f-57ce-4715-a247-cae8729dcce9
```

Output:

```
{
    "SynthesisTask": {
        "TaskId": "70b61c0f-57ce-4715-a247-cae8729dcce9",
        "TaskStatus": "completed",
        "OutputUri": "https://s3.us-west-2.amazonaws.com/amzn-s3-demo-
bucket/70b61c0f-57ce-4715-a247-cae8729dcce9.mp3",
        "CreationTime": 1603911042.689,
        "RequestCharacters": 1311,
        "OutputFormat": "mp3",
        "TextType": "text",
        "VoiceId": "Joanna"
    }
}
```

Per ulteriori informazioni, consulta Creazione di file audio di grandi dimensioni nella Guida per gli sviluppatori di Amazon Polly.

- Per i dettagli sull'API, consulta GetSpeechSynthesisTask AWS CLICommand Reference.

`list-lexicons`

Il seguente esempio di codice mostra come utilizzare`list-lexicons`.

AWS CLI

Come elencare i lessici correnti

L'esempio `list-lexicons` seguente elenca i lessici di pronuncia in uso.

```
aws polly list-lexicons
```

Output:

```
{
    "Lexicons": [
        {
            "Name": "w3c",
            "Attributes": {
                "Alphabet": "ipa",
                "LanguageCode": "en-US",
                "LastModified": 1603908910.99,
                "LexiconArn": "arn:aws:polly:us-east-2:123456789012:lexicon/w3c",
                "LexemesCount": 1,
                "Size": 492
            }
        }
    ]
}
```

Per ulteriori informazioni, consulta Utilizzo dell' ListLexicons operazione nella Amazon Polly Developer Guide.

- Per i dettagli sull'API, consulta ListLexicons AWS CLICommand Reference.

`list-speech-synthesis-tasks`

Il seguente esempio di codice mostra come utilizzare`list-speech-synthesis-tasks`.

AWS CLI

Come elencare le tue attività di sintesi vocale

L'esempio `list-speech-synthesis-tasks` seguente elenca le attività di sintesi vocale.

```
aws polly list-speech-synthesis-tasks
```

Output:

```
{
```

```
    "SynthesisTasks": [
        {
            "TaskId": "70b61c0f-57ce-4715-a247-cae8729dcce9",
            "TaskStatus": "completed",
            "OutputUri": "https://s3.us-west-2.amazonaws.com/amzn-s3-demo-
bucket/70b61c0f-57ce-4715-a247-cae8729dcce9.mp3",
            "CreationTime": 1603911042.689,
            "RequestCharacters": 1311,
            "OutputFormat": "mp3",
            "TextType": "text",
            "VoiceId": "Joanna"
        }
    ]
}
```

Per ulteriori informazioni, consulta Creazione di file audio di grandi dimensioni nella Guida per gli sviluppatori di Amazon Polly.

- Per i dettagli sull'API, consulta ListSpeechSynthesisTasks AWS CLICommand Reference.

`put-lexicon`

Il seguente esempio di codice mostra come utilizzare`put-lexicon`.

AWS CLI

Come archiviare un lessico

L'esempio `put-lexicon` seguente archivia il lessico di pronuncia specificato. Il file `example.pls` specifica un lessico conforme a W3C PLS.

```
aws polly put-lexicon \
    --name w3c \
    --content file://example.pls
```

Contenuto di `example.pls`

```
{
    <?xml version="1.0" encoding="UTF-8"?>
    <lexicon version="1.0"
        xmlns="http://www.w3.org/2005/01/pronunciation-lexicon"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
```

```
        xsi:schemaLocation="http://www.w3.org/2005/01/pronunciation-lexicon
            http://www.w3.org/TR/2007/CR-pronunciation-lexicon-20071212/pls.xsd"
        alphabet="ipa"
        xml:lang="en-US">
        <lexeme>
            <grapheme>W3C</grapheme>
            <alias>World Wide Web Consortium</alias>
        </lexeme>
    </lexicon>
}
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Utilizzo dell' PutLexicon operazione nella Amazon Polly Developer Guide.

- Per i dettagli sull'API, consulta PutLexicon AWS CLICommand Reference.

`start-speech-synthesis-task`

Il seguente esempio di codice mostra come utilizzare`start-speech-synthesis-task`.

AWS CLI

Come sintetizzare il testo

L'`start-speech-synthesis-task`esempio seguente sintetizza il testo `text_file.txt` e memorizza il MP3 file risultante nel bucket specificato.

```
aws polly start-speech-synthesis-task \
    --output-format mp3 \
    --output-s3-bucket-name amzn-s3-demo-bucket \
    --text  file://text_file.txt \
    --voice-id Joanna
```

Output:

```
{
    "SynthesisTask": {
        "TaskId": "70b61c0f-57ce-4715-a247-cae8729dcce9",
        "TaskStatus": "scheduled",
        "OutputUri": "https://s3.us-east-2.amazonaws.com/amzn-s3-demo-
bucket/70b61c0f-57ce-4715-a247-cae8729dcce9.mp3",
```

```
        "CreationTime": 1603911042.689,
        "RequestCharacters": 1311,
        "OutputFormat": "mp3",
        "TextType": "text",
        "VoiceId": "Joanna"
    }
}
```

Per ulteriori informazioni, consulta Creazione di file audio di grandi dimensioni nella Guida per gli sviluppatori di Amazon Polly.

- Per i dettagli sull'API, vedere StartSpeechSynthesisTaskin AWS CLI Command Reference.

Listino prezzi AWS esempi utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with Listino prezzi AWS.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`describe-services`

Il seguente esempio di codice mostra come utilizzare`describe-services`.

AWS CLI

Come recuperare i metadati del servizio

Questo esempio recupera i metadati per il codice di EC2 servizio Amazon.

Comando:

```
aws pricing describe-services --service-code AmazonEC2 --format-version aws_v1 --
max-items 1
```

Output:


```
{
  "Services": [
      {
          "ServiceCode": "AmazonEC2",
          "AttributeNames": [
              "volumeType",
              "maxIopsvolume",
              "instance",
              "instanceCapacity10xlarge",
              "locationType",
              "instanceFamily",
              "operatingSystem",
              "clockSpeed",
              "LeaseContractLength",
              "ecu",
              "networkPerformance",
              "instanceCapacity8xlarge",
              "group",
              "maxThroughputvolume",
              "gpuMemory",
              "ebsOptimized",
              "elasticGpuType",
              "maxVolumeSize",
              "gpu",
              "processorFeatures",
              "intelAvxAvailable",
              "instanceCapacity4xlarge",
              "servicecode",
              "groupDescription",
              "processorArchitecture",
              "physicalCores",
              "productFamily",
              "enhancedNetworkingSupported",
              "intelTurboAvailable",
              "memory",
              "dedicatedEbsThroughput",
              "vcpu",
              "OfferingClass",
```

```
                "instanceCapacityLarge",
                "capacitystatus",
                "termType",
                "storage",
                "intelAvx2Available",
                "storageMedia",
                "physicalProcessor",
                "provisioned",
                "servicename",
                "PurchaseOption",
                "instanceCapacity18xlarge",
                "instanceType",
                "tenancy",
                "usagetype",
                "normalizationSizeFactor",
                "instanceCapacity2xlarge",
                "instanceCapacity16xlarge",
                "maxIopsBurstPerformance",
                "instanceCapacity12xlarge",
                "instanceCapacity32xlarge",
                "instanceCapacityXlarge",
                "licenseModel",
                "currentGeneration",
                "preInstalledSw",
                "location",
                "instanceCapacity24xlarge",
                "instanceCapacity9xlarge",
                "instanceCapacityMedium",
                "operation"
            ]
        }
    ],
    "FormatVersion": "aws_v1"
}
```

- Per i dettagli sull'API, consulta AWS CLI Command DescribeServicesReference.

`get-attribute-values`

Il seguente esempio di codice mostra come utilizzare`get-attribute-values`.

AWS CLI

Come recuperare un elenco dei valori degli attributi

L'esempio `get-attribute-values` seguente recupera un elenco di valori disponibili per l'attributo specificato.

```
aws pricing get-attribute-values \
    --service-code AmazonEC2 \
    --attribute-name volumeType \
    --max-items 2
```

Output:

```
{
    "NextToken": "eyJOZXh0VG9rZW4iOiBudWxsLCAiYm90b190cnVuY2F0ZV9hbW91bnQiOiAyfQ==",
    "AttributeValues": [
        {
            "Value": "Cold HDD"
        },
        {
            "Value": "General Purpose"
        }
    ]
}
```

- Per i dettagli sull'API, consulta GetAttributeValues AWS CLICommand Reference.

`get-products`

Il seguente esempio di codice mostra come utilizzare`get-products`.

AWS CLI

Come recuperare un elenco di prodotti

Questo esempio recupera un elenco di prodotti corrispondenti ai criteri indicati.

Comando:

```
aws pricing get-products --filters file://filters.json --format-version aws_v1 --
max-results 1 --service-code AmazonEC2
```

filters.json:

```
        [
  {
    "Type": "TERM_MATCH",
    "Field": "ServiceCode",
    "Value": "AmazonEC2"
  },
  {
    "Type": "TERM_MATCH",
    "Field": "volumeType",
    "Value": "Provisioned IOPS"
  }
]
```

Output:


```
{
  "FormatVersion": "aws_v1",
  "NextToken": "WGDY7ko8fQXdlaUZVdasFQ==:RVSagyIFn770XQOzdUIcO9BY6ucBG9itXAZGZF/
zioUzOsUKh6PCcPWaOyPZRiMePb986TeoKYB9l55fw/
CyoMq5ymnGmT1Vj39TljbbAlhcqnVfTmPIilx8Uy5bdDaBYy/e/2Ofw9EdzsykbS8LTBuNbiDQ
+BBds5yeI9AQkUepruKk3aEahFPxJ55kx/zk",
  "PriceList": [
      "{\"product\":{\"productFamily\":\"Storage\",\"attributes\":{\"storageMedia\":
\"SSD-backed\",\"maxThroughputvolume\":\"320 MB/sec\",\"volumeType\":\"Provisioned
 IOPS\",\"maxIopsvolume\":\"20000\",\"servicecode\":\"AmazonEC2\",\"usagetype
\":\"APS1-EBS:VolumeUsage.piops\",\"locationType\":\"AWS Region\",\"location\":
\"Asia Pacific (Singapore)\",\"servicename\":\"Amazon Elastic Compute Cloud\",
\"maxVolumeSize\":\"16 TiB\",\"operation\":\"\"},\"sku\":\"3MKHN58N7RDDVGKJ\"},
\"serviceCode\":\"AmazonEC2\",\"terms\":{\"OnDemand\":{\"3MKHN58N7RDDVGKJ.JRTCKXETXF
\":{\"priceDimensions\":{\"3MKHN58N7RDDVGKJ.JRTCKXETXF.6YS6EN2CT7\":{\"unit\":\"GB-
Mo\",\"endRange\":\"Inf\",\"description\":\"$0.138 per GB-month of Provisioned
 IOPS SSD (io1)  provisioned storage - Asia Pacific (Singapore)\",\"appliesTo
\":[],\"rateCode\":\"3MKHN58N7RDDVGKJ.JRTCKXETXF.6YS6EN2CT7\",\"beginRange\":
\"0\",\"pricePerUnit\":{\"USD\":\"0.1380000000\"}}},\"sku\":\"3MKHN58N7RDDVGKJ
\",\"effectiveDate\":\"2018-08-01T00:00:00Z\",\"offerTermCode\":\"JRTCKXETXF
\",\"termAttributes\":{}}}},\"version\":\"20180808005701\",\"publicationDate\":
\"2018-08-08T00:57:01Z\"}"
  ]
}
```


- Per i dettagli sull'API, consulta GetProducts AWS CLICommand Reference.

AWS Private CA esempi utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with AWS Private CA.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`create-certificate-authority-audit-report`

Il seguente esempio di codice mostra come utilizzare`create-certificate-authority-audit-report`.

AWS CLI

Come creare un report di audit dell'autorità di certificazione

Il comando `create-certificate-authority-audit-report` seguente crea un report di controllo per la CA privata identificata dall'ARN.

```
aws acm-pca create-certificate-authority-audit-report --certificate-authority-arn arn:aws:acm-pca:us-east-1:accountid:certificate-authority/12345678-1234-1234-1234-123456789012 --s3-bucket-name your-bucket-name --audit-report-response-format JSON
```

- Per i dettagli sull'API, consulta CreateCertificateAuthorityAuditReport AWS CLICommand Reference.

`create-certificate-authority`

Il seguente esempio di codice mostra come utilizzare`create-certificate-authority`.

AWS CLI

Come creare un'autorità di certificazione privata

Il `create-certificate-authority` comando seguente crea un'autorità di certificazione privata nel tuo AWS account.

```
aws acm-pca create-certificate-authority --certificate-authority-configuration
 file://C:\ca_config.txt --revocation-configuration file://C:\revoke_config.txt --
certificate-authority-type "SUBORDINATE" --idempotency-token 98256344
```

- Per i dettagli sull'API, consulta CreateCertificateAuthority AWS CLICommand Reference.

`delete-certificate-authority`

Il seguente esempio di codice mostra come utilizzare`delete-certificate-authority`.

AWS CLI

Come eliminare un'autorità di certificazione privata

Il comando `delete-certificate-authority` seguente elimina l'autorità di certificazione identificata dall'ARN.

```
aws acm-pca delete-certificate-authority --certificate-
authority-arn arn:aws:acm-pca:us-west-2:123456789012:certificate-
authority/12345678-1234-1234-1234-123456789012
```

- Per i dettagli sull'API, consulta DeleteCertificateAuthority AWS CLICommand Reference.

`describe-certificate-authority-audit-report`

Il seguente esempio di codice mostra come utilizzare`describe-certificate-authority-audit-report`.

AWS CLI

Come descrivere un report di audit per un'autorità di certificazione (CA)

Il comando `describe-certificate-authority-audit-report` seguente elenca le informazioni sul report di audit specificato per la CA identificata dall'ARN.

```
aws acm-pca describe-certificate-authority-audit-report --certificate-
authority-arn arn:aws:acm-pca:us-west-2:123456789012:certificate-
authority/99999999-8888-7777-6666-555555555555 --audit-report-
id 11111111-2222-3333-4444-555555555555
```

- Per i dettagli sull'API, consulta DescribeCertificateAuthorityAuditReport AWS CLICommand Reference.

`describe-certificate-authority`

Il seguente esempio di codice mostra come utilizzare`describe-certificate-authority`.

AWS CLI

Come descrivere un'autorità di certificazione privata

Il comando `describe-certificate-authority` seguente elenca le informazioni sulla CA privata identificata dall'ARN.

```
aws acm-pca describe-certificate-authority --certificate-
authority-arn arn:aws:acm-pca:us-west-2:123456789012:certificate-
authority/12345678-1234-1234-1234-123456789012
```

- Per i dettagli sull'API, consulta DescribeCertificateAuthority AWS CLICommand Reference.

`get-certificate-authority-certificate`

Il seguente esempio di codice mostra come utilizzare`get-certificate-authority-certificate`.

AWS CLI

Come recuperare un certificato dell'autorità di certificazione (CA)

Il comando `get-certificate-authority-certificate` seguente recupera il certificato e la catena di certificati per la CA privata specificata dall'ARN.

```
aws acm-pca get-certificate-authority-certificate --certificate-
authority-arn arn:aws:acm-pca:us-west-2:123456789012:certificate-
authority/12345678-1234-1234-1234-123456789012 --output text
```

- Per i dettagli sull'API, consulta GetCertificateAuthorityCertificate AWS CLICommand Reference.

`get-certificate-authority-csr`

Il seguente esempio di codice mostra come utilizzare`get-certificate-authority-csr`.

AWS CLI

Come recuperare la richiesta di firma del certificato per un'autorità di certificazione (CA)

Il comando `get-certificate-authority-csr` seguente recupera la CSR per la CA privata specificata dall'ARN.

```
aws acm-pca get-certificate-authority-csr --certificate-
authority-arn arn:aws:acm-pca:us-west-2:123456789012:certificate-
authority/12345678-1234-1234-1234-123456789012 --output text
```

- Per i dettagli sull'API, consulta GetCertificateAuthorityCsr AWS CLICommand Reference.

`get-certificate`

Il seguente esempio di codice mostra come utilizzare`get-certificate`.

AWS CLI

Come recuperare un certificato emesso

L'esempio `get-certificate` seguente recupera un certificato dalla CA privata specificata.

```
aws acm-pca get-certificate \
    --certificate-authority-arn arn:aws:acm-pca:us-west-2:123456789012:certificate-
authority/12345678-1234-1234-1234-123456789012 \
    --certificate-arn arn:aws:acm-pca:us-west-2:123456789012:certificate-
authority/12345678-1234-1234-1234-123456789012/
certificate/6707447683a9b7f4055627ffd55cebcc \
    --output text
```

Output:

```
-----BEGIN CERTIFICATE-----
```

```
MIIEDzCCAvegAwIBAgIRAJuJ8f6ZVYL7gG/rS3qvrZMwDQYJKoZIhvcNAQELBQAw
cTELMAkGA1UEBhMCVVMxEzARBgNVBAgMCldhc2hpbmd0b24xEDAOBgNVBAcMB1Nl
    ....certificate body truncated for brevity....
tKCSglgZZrd4FdLw1EkGm+UVXnodwMtJEQyy3oTfZjURPIyyaqskTu/KSS7YDjK0
KQNy73D6LtmdOEbAyq10XiDxqY41lvKHJ1eZrPaBmYNABxU=
-----END CERTIFICATE---- -----BEGIN CERTIFICATE-----
MIIDrzCCApegAwIBAgIRAOskdzLvcj1eShkoyEE693AwDQYJKoZIhvcNAQELBQAw
cTELMAkGA1UEBhMCVVMxEzARBgNVBAgMCldhc2hpbmd0b24xEDAOBgNVBAcMB1Nl
    ...certificate body truncated for brevity....
kdRGB6P2hpxstDOUIwAoCbhoaWwfA4ybJznf+jOQhAziNlRdKQRR8nODWpKt7H9w
dJ5nxsTk/fniJz86Ddtp6n8s82wYdkN3cVffeK72A9aTCOU=
-----END CERTIFICATE-----
```

La prima parte dell'output è il certificato stesso. La seconda parte è la catena di certificati che si collega al certificato CA root. Si noti che quando si utilizza l'opzione `--output text`, viene inserito un carattere TAB tra i due elementi del certificato (questa è la causa del testo con rientro). Se intendi accettare questo risultato e analizzare i certificati con altri strumenti, potresti dover rimuovere il carattere TAB in modo che venga elaborato correttamente.

- Per i dettagli sull'API, consulta GetCertificate AWS CLICommand Reference.

`import-certificate-authority-certificate`

Il seguente esempio di codice mostra come utilizzare`import-certificate-authority-certificate`.

AWS CLI

Come importare il certificato dell'autorità di certificazione (CA) in ACM PCA

Il comando `import-certificate-authority-certificate` seguente importa il certificato CA privato firmato per la CA specificata dall'ARN in ACM PCA.

```
aws acm-pca import-certificate-authority-certificate --certificate-
authority-arn arn:aws:acm-pca:us-west-2:123456789012:certificate-
authority/12345678-1234-1234-1234-123456789012 --certificate file://C:\ca_cert.pem
 --certificate-chain file://C:\ca_cert_chain.pem
```

- Per i dettagli sull'API, consulta ImportCertificateAuthorityCertificate AWS CLICommand Reference.

`issue-certificate`

Il seguente esempio di codice mostra come utilizzare`issue-certificate`.

AWS CLI

Come emettere un certificato privato

Il comando `issue-certificate` seguente utilizza la CA privata specificata dall'ARN per emettere un certificato privato.

```
aws acm-pca issue-certificate --certificate-authority-arn arn:aws:acm-pca:us-
west-2:123456789012:certificate-authority/12345678-1234-1234-1234-123456789012
 --csr file://C:\cert_1.csr --signing-algorithm "SHA256WITHRSA" --validity
 Value=365,Type="DAYS" --idempotency-token 1234
```

- Per i dettagli sull'API, consulta IssueCertificate AWS CLICommand Reference.

`list-certificate-authorities`

Il seguente esempio di codice mostra come utilizzare`list-certificate-authorities`.

AWS CLI

Come elencare le autorità di certificazione privata

Il `list-certificate-authorities` comando seguente elenca le informazioni su tutti i dati privati CAs presenti nell'account.

```
aws acm-pca list-certificate-authorities --max-results 10
```

- Per i dettagli sull'API, consulta ListCertificateAuthorities AWS CLICommand Reference.

`list-tags`

Il seguente esempio di codice mostra come utilizzare`list-tags`.

AWS CLI

Come elencare i tag dell'autorità di certificazione (CA)

Il comando `list-tags` seguente elenca i tag associati alla CA privata specificata dall'ARN.

```
aws acm-pca list-tags --certificate-authority-arn arn:aws:acm-pca:us-
west-2:123456789012:certificate-authority/123455678-1234-1234-1234-123456789012 --
max-results 10
```

- Per i dettagli sull'API, consulta ListTags AWS CLICommand Reference.

`revoke-certificate`

Il seguente esempio di codice mostra come utilizzare`revoke-certificate`.

AWS CLI

Come revocare un certificato privato

Il comando `revoke-certificate` seguente revoca un certificato privato dalla CA identificata dall'ARN.

```
aws acm-pca revoke-certificate --certificate-authority-arn arn:aws:acm-pca:us-
west-2:1234567890:certificate-authority/12345678-1234-1234-1234-123456789012 --
certificate-serial 67:07:44:76:83:a9:b7:f4:05:56:27:ff:d5:5c:eb:cc --revocation-
reason "KEY_COMPROMISE"
```

- Per i dettagli sull'API, consulta RevokeCertificate AWS CLICommand Reference.

`tag-certificate-authority`

Il seguente esempio di codice mostra come utilizzare`tag-certificate-authority`.

AWS CLI

Come aggiungere tag all'autorità di certificazione privata

Il comando `tag-certificate-authority` seguente applica uno o più tag alla CA privata.

```
aws acm-pca tag-certificate-authority --certificate-authority-
arn arn:aws:acm-pca:us-west-2:123456789012:certificate-
authority/12345678-1234-1234-1234-123456789012 --tags Key=Admin,Value=Alice
```

- Per i dettagli sull'API, consulta TagCertificateAuthority AWS CLICommand Reference.

`untag-certificate-authority`

Il seguente esempio di codice mostra come utilizzare`untag-certificate-authority`.

AWS CLI

Come rimuovere uno o più tag dall'autorità di certificazione privata

Il comando `untag-certificate-authority` seguente rimuove i tag dalla CA privata identificata dall'ARN.

```
aws acm-pca untag-certificate-authority --certificate-authority-
arn arn:aws:acm-pca:us-west-2:123456789012:certificate-
authority/12345678-1234-1234-1234-123456789012 --tags Key=Purpose,Value=Website
```

- Per i dettagli sull'API, consulta UntagCertificateAuthority AWS CLICommand Reference.

`update-certificate-authority`

Il seguente esempio di codice mostra come utilizzare`update-certificate-authority`.

AWS CLI

Come aggiornare la configurazione dell'autorità di certificazione privata

Il comando `update-certificate-authority` seguente aggiorna lo stato e la configurazione della CA privata identificata dall'ARN.

```
aws acm-pca update-certificate-authority --certificate-
authority-arn arn:aws:acm-pca:us-west-2:123456789012:certificate-
authority/12345678-1234-1234-1234-1232456789012 --revocation-configuration file://C:
\revoke_config.txt --status "DISABLED"
```

- Per i dettagli sull'API, consulta UpdateCertificateAuthority AWS CLICommand Reference.

AWS Proton esempi utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with AWS Proton.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`cancel-service-instance-deployment`

Il seguente esempio di codice mostra come utilizzare`cancel-service-instance-deployment`.

AWS CLI

Come annullare l'implementazione di un'istanza di servizio

L'esempio `cancel-service-instance-deployment` seguente annulla l'implementazione di un'istanza di servizio.

```
aws proton cancel-service-instance-deployment \
    --service-instance-name "instance-one" \
    --service-name "simple-svc"
```

Output:

```
{
    "serviceInstance": {
        "arn": "arn:aws:proton:region-id:123456789012:service/simple-svc/service-
instance/instance-one",
        "createdAt": "2021-04-02T21:29:59.962000+00:00",
        "deploymentStatus": "CANCELLING",
        "environmentName": "simple-env",
        "lastDeploymentAttemptedAt": "2021-04-02T21:45:15.406000+00:00",
        "lastDeploymentSucceededAt": "2021-04-02T21:38:00.823000+00:00",
        "name": "instance-one",
        "serviceName": "simple-svc",
```

```
        "spec": "proton: ServiceSpec\npipeline:\n
 my_sample_pipeline_optional_input: abc\n  my_sample_pipeline_required_input:
 '123'\ninstances:\n- name: my-instance\n  environment: MySimpleEnv
\n  spec:\n    my_sample_service_instance_optional_input: def\n
 my_sample_service_instance_required_input: '456'\n- name: my-other-instance\n
 environment: MySimpleEnv\n  spec:\n    my_sample_service_instance_required_input:
 '789'\n",
        "templateMajorVersion": "1",
        "templateMinorVersion": "1",
        "templateName": "svc-simple"
    }
}
```

Per ulteriori informazioni, vedere Aggiornare un'istanza di servizio nella Guida per l'amministratore di AWS Proton o Aggiornare un'istanza di servizio nella Guida per l'utente di AWS Proton.

- Per i dettagli sull'API, vedere CancelServiceInstanceDeploymentin AWS CLI Command Reference.

`cancel-service-pipeline-deployment`

Il seguente esempio di codice mostra come utilizzare`cancel-service-pipeline-deployment`.

AWS CLI

Come annullare l'implementazione di una pipeline di servizio

L'esempio `cancel-service-pipeline-deployment` seguente annulla l'implementazione di una pipeline di servizio.

```
aws proton cancel-service-pipeline-deployment \
    --service-name "simple-svc"
```

Output:

```
{
    "pipeline": {
        "arn": "arn:aws:proton:region-id:123456789012:service/simple-svc/pipeline",
        "createdAt": "2021-04-02T21:29:59.962000+00:00",
        "deploymentStatus": "CANCELLING",
        "lastDeploymentAttemptedAt": "2021-04-02T22:02:45.095000+00:00",
        "lastDeploymentSucceededAt": "2021-04-02T21:39:28.991000+00:00",
```

```
        "templateMajorVersion": "1",
        "templateMinorVersion": "1",
        "templateName": "svc-simple"
    }
}
```

Per ulteriori informazioni, vedere Aggiornare una pipeline di servizi nella The AWS Proton Administrator Guide o Aggiornare una pipeline di servizi nella The AWS Proton User Guide.

- Per i dettagli sull'API, vedere CancelServicePipelineDeploymentin Command Reference.AWS CLI

`create-service`

Il seguente esempio di codice mostra come utilizzare`create-service`.

AWS CLI

Come creare un servizio

L'esempio `create-service` seguente crea un servizio con una pipeline di servizi.

```
aws proton create-service \
    --name "MySimpleService" \
    --template-name "fargate-service" \
    --template-major-version "1" \
    --branch-name "mainline" \
    --repository-connection-arn "arn:aws:codestar-connections:region-id:account-
id:connection/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111" \
    --repository-id "myorg/myapp" \
    --spec file://spec.yaml
```

Contenuto di `spec.yaml`:

```
proton: ServiceSpec

pipeline:
    my_sample_pipeline_required_input: "hello"
    my_sample_pipeline_optional_input: "bye"

instances:
    - name: "acme-network-dev"
```

```
        environment: "ENV_NAME"
        spec:
            my_sample_service_instance_required_input: "hi"
            my_sample_service_instance_optional_input: "ho"
```

Output:

```
{
    "service": {
        "arn": "arn:aws:proton:region-id:123456789012:service/MySimpleService",
        "createdAt": "2020-11-18T19:50:27.460000+00:00",
        "lastModifiedAt": "2020-11-18T19:50:27.460000+00:00",
        "name": "MySimpleService",
        "repositoryConnectionArn": "arn:aws:codestar-connections:region-
id:123456789012connection/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
        "repositoryId": "myorg/myapp",
        "status": "CREATE_IN_PROGRESS",
        "templateName": "fargate-service"
    }
}
```

Per ulteriori informazioni, vedere Creare un servizio nella Guida per l'amministratore di AWS Proton e Creare un servizio nella Guida per l'utente di AWS Proton.

- Per i dettagli sull'API, vedere CreateServicein AWS CLI Command Reference.

`delete-service`

Il seguente esempio di codice mostra come utilizzare`delete-service`.

AWS CLI

Come eliminare un servizio

L'esempio `delete-service` seguente elimina un segreto.

```
aws proton delete-service \
    --name "simple-svc"
```

Output:

```
{
```

```
    "service": {
        "arn": "arn:aws:proton:region-id:123456789012:service/simple-svc",
        "branchName": "mainline",
        "createdAt": "2020-11-28T22:40:50.512000+00:00",
        "description": "Edit by updating description",
        "lastModifiedAt": "2020-11-29T00:30:39.248000+00:00",
        "name": "simple-svc",
        "repositoryConnectionArn": "arn:aws:codestar-connections:region-
id:123456789012:connection/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
        "repositoryId": "myorg/myapp",
        "status": "DELETE_IN_PROGRESS",
        "templateName": "fargate-service"
    }
}
```

Per ulteriori informazioni, vedere Eliminare un servizio nella The AWS Proton Administrator Guide.

- Per i dettagli sull'API, vedere DeleteServicein AWS CLI Command Reference.

`get-service-instance`

Il seguente esempio di codice mostra come utilizzare`get-service-instance`.

AWS CLI

Come ottenere dettagli sull'istanza del servizio

L'esempio `get-service-instance` seguente ottiene dati di dettaglio per un'istanza di servizio.

```
aws proton get-service-instance \
    --name "instance-one" \
    --service-name "simple-svc"
```

Output:

```
{
    "serviceInstance": {
        "arn": "arn:aws:proton:region-id:123456789012:service/simple-svc/service-
instance/instance-one",
        "createdAt": "2020-11-28T22:40:50.512000+00:00",
        "deploymentStatus": "SUCCEEDED",
        "environmentName": "simple-env",
```

```
        "lastDeploymentAttemptedAt": "2020-11-28T22:40:50.512000+00:00",
        "lastDeploymentSucceededAt": "2020-11-28T22:40:50.512000+00:00",
        "name": "instance-one",
        "serviceName": "simple-svc",
        "spec": "proton: ServiceSpec\npipeline:\n
 my_sample_pipeline_optional_input: hello world\n
 my_sample_pipeline_required_input: pipeline up\ninstances:\n- name: instance-one\n
 environment: my-simple-env\n  spec:\n    my_sample_service_instance_optional_input:
 Ola\n    my_sample_service_instance_required_input: Ciao\n",
        "templateMajorVersion": "1",
        "templateMinorVersion": "0",
        "templateName": "svc-simple"
    }
}
```

Per ulteriori informazioni, vedere Visualizza i dati di servizio nella The AWS Proton Administrator Guide o Visualizza i dati di servizio nella The AWS Proton User Guide.

- Per i dettagli sull'API, vedere GetServiceInstancein AWS CLI Command Reference.

`get-service`

Il seguente esempio di codice mostra come utilizzare`get-service`.

AWS CLI

Come ottenere i dettagli del servizio

L'esempio `get-service` seguente ottiene dati di dettaglio per un servizio.

```
aws proton get-service \
    --name "simple-svc"
```

Output:

```
{
    "service": {
        "arn": "arn:aws:proton:region-id:123456789012:service/simple-svc",
        "branchName": "mainline",
        "createdAt": "2020-11-28T22:40:50.512000+00:00",
        "lastModifiedAt": "2020-11-28T22:44:51.207000+00:00",
        "name": "simple-svc",
```

```
        "pipeline": {
            "arn": "arn:aws:proton:region-id:123456789012:service/simple-svc/
pipeline/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
            "createdAt": "2020-11-28T22:40:50.512000+00:00",
            "deploymentStatus": "SUCCEEDED",
            "lastDeploymentAttemptedAt": "2020-11-28T22:40:50.512000+00:00",
            "lastDeploymentSucceededAt": "2020-11-28T22:40:50.512000+00:00",
            "spec": "proton: ServiceSpec\npipeline:\n
 my_sample_pipeline_required_input: hello\n  my_sample_pipeline_optional_input:
 bye\ninstances:\n- name: instance-svc-simple\n  environment: my-simple-
env\n  spec:\n    my_sample_service_instance_required_input: hi\n
 my_sample_service_instance_optional_input: ho\n",
            "templateMajorVersion": "1",
            "templateMinorVersion": "1",
            "templateName": "svc-simple"
        },
        "repositoryConnectionArn": "arn:aws:codestar-connections:region-
id:123456789012:connection/a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
        "repositoryId": "myorg/myapp",
        "spec": "proton: ServiceSpec\npipeline:\n
 my_sample_pipeline_required_input: hello\n  my_sample_pipeline_optional_input:
 bye\ninstances:\n- name: instance-svc-simple\n  environment: my-simple-
env\n  spec:\n    my_sample_service_instance_required_input: hi\n
 my_sample_service_instance_optional_input: ho\n",
        "status": "ACTIVE",
        "templateName": "svc-simple"
    }
}
```

Per ulteriori informazioni, vedere Visualizza i dati di servizio nella The AWS Proton Administrator Guide o Visualizza i dati di servizio nella The AWS Proton User Guide.

- Per i dettagli sull'API, vedere GetServicein AWS CLI Command Reference.

`list-service-instances`

Il seguente esempio di codice mostra come utilizzare`list-service-instances`.

AWS CLI

Esempio 1: come elencare tutte le istanze del servizio

L'esempio `list-service-instances` seguente elenca le istanze del servizio.

```
aws proton list-service-instances
```

Output:

```
{
    "serviceInstances": [
        {
            "arn": "arn:aws:proton:region-id:123456789012:service/simple-svc/
service-instance/instance-one",
            "createdAt": "2020-11-28T22:40:50.512000+00:00",
            "deploymentStatus": "SUCCEEDED",
            "environmentArn": "arn:aws:proton:region-id:123456789012:environment/
simple-env",
            "lastDeploymentAttemptedAt": "2020-11-28T22:40:50.512000+00:00",
            "lastDeploymentSucceededAt": "2020-11-28T22:40:50.512000+00:00",
            "name": "instance-one",
            "serviceName": "simple-svc",
            "templateMajorVersion": "1",
            "templateMinorVersion": "0",
            "templateName": "fargate-service"
        }
    ]
}
```

Per ulteriori informazioni, vedere Visualizza i dati dell'istanza del servizio nella Guida per l'amministratore di AWS Proton o Visualizza i dati dell'istanza del servizio nella Guida per l'utente di AWS Proton.

Esempio 2: come elencare l'istanza del servizio specificata

L'esempio `get-service-instance` seguente ottiene un'istanza di servizio.

```
aws proton get-service-instance \
    --name "instance-one" \
    --service-name "simple-svc"
```

Output:

```
{
    "serviceInstance": {
```

```
        "arn": "arn:aws:proton:region-id:123456789012:service/simple-svc/service-
instance/instance-one",
        "createdAt": "2020-11-28T22:40:50.512000+00:00",
        "deploymentStatus": "SUCCEEDED",
        "environmentName": "simple-env",
        "lastDeploymentAttemptedAt": "2020-11-28T22:40:50.512000+00:00",
        "lastDeploymentSucceededAt": "2020-11-28T22:40:50.512000+00:00",
        "name": "instance-one",
        "serviceName": "simple-svc",
        "spec": "proton: ServiceSpec\npipeline:\n
 my_sample_pipeline_optional_input: hello world\n
 my_sample_pipeline_required_input: pipeline up\ninstances:\n- name: instance-one\n
 environment: my-simple-env\n  spec:\n    my_sample_service_instance_optional_input:
 Ola\n    my_sample_service_instance_required_input: Ciao\n",
        "templateMajorVersion": "1",
        "templateMinorVersion": "0",
        "templateName": "svc-simple"
    }
}
```

Per ulteriori informazioni, vedere Visualizza i dati dell'istanza del servizio nella Guida per l'amministratore di AWS Proton o Visualizza i dati dell'istanza del servizio nella Guida per l'utente di AWS Proton.

- Per i dettagli sull'API, vedere ListServiceInstancesin AWS CLI Command Reference.

`update-service-instance`

Il seguente esempio di codice mostra come utilizzare`update-service-instance`.

AWS CLI

Come aggiornare un'istanza del servizio a una nuova versione secondaria

L'`update-service-instance`esempio seguente aggiorna un'istanza di servizio con una nuova versione secondaria del relativo modello di servizio che aggiunge una nuova istanza denominata my-other-instance "" con un nuovo input richiesto.

```
aws proton update-service-instance \
    --service-name "simple-svc" \
    --spec "file://service-spec.yaml " \
    --template-major-version "1" \
    --template-minor-version "1" \
```

```
    --deployment-type "MINOR_VERSION" \
    --name "instance-one"
```

Contenuto di `service-spec.yaml`:

```
proton: ServiceSpec
pipeline:
    my_sample_pipeline_optional_input: "abc"
    my_sample_pipeline_required_input: "123"
instances:
    - name: "instance-one"
        environment: "simple-env"
        spec:
            my_sample_service_instance_optional_input: "def"
            my_sample_service_instance_required_input: "456"
    - name: "my-other-instance"
        environment: "simple-env"
        spec:
            my_sample_service_instance_required_input: "789"
```

Output:

```
{
    "serviceInstance": {
        "arn": "arn:aws:proton:region-id:123456789012:service/simple-svc/service-
instance/instance-one",
        "createdAt": "2021-04-02T21:29:59.962000+00:00",
        "deploymentStatus": "IN_PROGRESS",
        "environmentName": "arn:aws:proton:region-id:123456789012:environment/
simple-env",
        "lastDeploymentAttemptedAt": "2021-04-02T21:38:00.823000+00:00",
        "lastDeploymentSucceededAt": "2021-04-02T21:29:59.962000+00:00",
        "name": "instance-one",
        "serviceName": "simple-svc",
        "templateMajorVersion": "1",
        "templateMinorVersion": "0",
        "templateName": "svc-simple"
    }
}
```

Per ulteriori informazioni, vedere Aggiornare un'istanza di servizio nella Guida per l'amministratore di AWS Proton o Aggiornare un'istanza di servizio nella Guida per l'utente di AWS Proton.

- Per i dettagli sull'API, vedere UpdateServiceInstancein AWS CLI Command Reference.

`update-service-pipeline`

Il seguente esempio di codice mostra come utilizzareu`pdate-service-pipeline`.

AWS CLI

Come aggiornare una pipeline di servizi

L'esempio `update-service-pipeline` seguente aggiorna una pipeline di servizi con una nuova versione secondaria del relativo modello di servizio.

```
aws proton update-service-pipeline \
    --service-name "simple-svc" \
    --spec "file://service-spec.yaml" \
    --template-major-version "1" \
    --template-minor-version "1" \
    --deployment-type "MINOR_VERSION"
```

Output:

```
{
    "pipeline": {
        "arn": "arn:aws:proton:region-id:123456789012:service/simple-svc/pipeline/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
        "createdAt": "2021-04-02T21:29:59.962000+00:00",
        "deploymentStatus": "IN_PROGRESS",
        "lastDeploymentAttemptedAt": "2021-04-02T21:39:28.991000+00:00",
        "lastDeploymentSucceededAt": "2021-04-02T21:29:59.962000+00:00",
        "spec": "proton: ServiceSpec\n\npipeline:\n
 my_sample_pipeline_optional_input: \"abc\"\n  my_sample_pipeline_required_input:
 \"123\"\n\ninstances:\n  - name: \"my-instance\"\n    environment: \"MySimpleEnv
\"\n    spec:\n      my_sample_service_instance_optional_input: \"def
\"\n      my_sample_service_instance_required_input: \"456\"\n  - name:
 \"my-other-instance\"\n    environment: \"MySimpleEnv\"\n    spec:\n
 my_sample_service_instance_required_input: \"789\"\n",
        "templateMajorVersion": "1",
        "templateMinorVersion": "0",
        "templateName": "svc-simple"
    }
}
```

Per ulteriori informazioni, vedere Aggiornare una pipeline di servizi nella The AWS Proton Administrator Guide o Aggiornare una pipeline di servizi nella The AWS Proton User Guide.

- Per i dettagli sull'API, vedere UpdateServicePipelinein Command Reference.AWS CLI

`update-service`

Il seguente esempio di codice mostra come utilizzare`update-service`.

AWS CLI

Come aggiornare un servizio

L'esempio `update-service` seguente modifica una descrizione del servizio.

```
aws proton update-service \
    --name "MySimpleService" \
    --description "Edit by updating description"
```

Output:

```
{
    "service": {
        "arn": "arn:aws:proton:region-id:123456789012:service/MySimpleService",
        "branchName": "mainline",
        "createdAt": "2021-03-12T22:39:42.318000+00:00",
        "description": "Edit by updating description",
        "lastModifiedAt": "2021-03-12T22:44:21.975000+00:00",
        "name": "MySimpleService",
        "repositoryConnectionArn": "arn:aws:codestar-connections:region-
id:123456789012:connection/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
        "repositoryId": "myorg/myapp",
        "status": "ACTIVE",
        "templateName": "fargate-service"
    }
}
```

Per ulteriori informazioni, vedere Modificare un servizio nella The AWS Proton Administrator Guide o Modificare un servizio nella The AWS Proton User Guide.

- Per i dettagli sull'API, vedere UpdateServicein AWS CLI Command Reference.

Esempi di utilizzo di Amazon RDS AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando Amazon RDS. AWS Command Line Interface

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`add-option-to-option-group`

Il seguente esempio di codice mostra come usare`add-option-to-option-group`.

AWS CLI

Come aggiungere un'opzione a un gruppo di opzioni

L'esempio `add-option-to-option-group` seguente aggiunge un'opzione al gruppo di opzioni specificato.

```
aws rds add-option-to-option-group \
    --option-group-name myoptiongroup \
    --options OptionName=OEM,Port=5500,DBSecurityGroupMemberships=default \
    --apply-immediately
```

Output:

```
{
    "OptionGroup": {
        "OptionGroupName": "myoptiongroup",
        "OptionGroupDescription": "Test Option Group",
        "EngineName": "oracle-ee",
        "MajorEngineVersion": "12.1",
```

```
        "Options": [
            {
                "OptionName": "Timezone",
                "OptionDescription": "Change time zone",
                "Persistent": true,
                "Permanent": false,
                "OptionSettings": [
                    {
                        "Name": "TIME_ZONE",
                        "Value": "Australia/Sydney",
                        "DefaultValue": "UTC",
                        "Description": "Specifies the timezone the user wants to
 change the system time to",
                        "ApplyType": "DYNAMIC",
                        "DataType": "STRING",
                        "AllowedValues": "Africa/Cairo,Africa/Casablanca,Africa/
Harare,Africa/Lagos,Africa/Luanda,Africa/Monrovia,Africa/Nairobi,Africa/
Tripoli,Africa/Windhoek,America/Araguaina,America/Argentina/Buenos_Aires,America/
Asuncion,America/Bogota,America/Caracas,America/Chicago,America/Chihuahua,America/
Cuiaba,America/Denver,America/Detroit,America/Fortaleza,America/Godthab,America/
Guatemala,America/Halifax,America/Lima,America/Los_Angeles,America/Manaus,America/
Matamoros,America/Mexico_City,America/Monterrey,America/Montevideo,America/
New_York,America/Phoenix,America/Santiago,America/Sao_Paulo,America/Tijuana,America/
Toronto,Asia/Amman,Asia/Ashgabat,Asia/Baghdad,Asia/Baku,Asia/Bangkok,Asia/
Beirut,Asia/Calcutta,Asia/Damascus,Asia/Dhaka,Asia/Hong_Kong,Asia/Irkutsk,Asia/
Jakarta,Asia/Jerusalem,Asia/Kabul,Asia/Karachi,Asia/Kathmandu,Asia/Kolkata,Asia/
Krasnoyarsk,Asia/Magadan,Asia/Manila,Asia/Muscat,Asia/Novosibirsk,Asia/Rangoon,Asia/
Riyadh,Asia/Seoul,Asia/Shanghai,Asia/Singapore,Asia/Taipei,Asia/Tehran,Asia/
Tokyo,Asia/Ulaanbaatar,Asia/Vladivostok,Asia/Yakutsk,Asia/Yerevan,Atlantic/
Azores,Atlantic/Cape_Verde,Australia/Adelaide,Australia/Brisbane,Australia/
Darwin,Australia/Eucla,Australia/Hobart,Australia/Lord_Howe,Australia/
Perth,Australia/Sydney,Brazil/DeNoronha,Brazil/East,Canada/Newfoundland,Canada/
Saskatchewan,Etc/GMT-3,Europe/Amsterdam,Europe/Athens,Europe/Berlin,Europe/
Dublin,Europe/Helsinki,Europe/Kaliningrad,Europe/London,Europe/Madrid,Europe/
Moscow,Europe/Paris,Europe/Prague,Europe/Rome,Europe/Sarajevo,Pacific/Apia,Pacific/
Auckland,Pacific/Chatham,Pacific/Fiji,Pacific/Guam,Pacific/Honolulu,Pacific/
Kiritimati,Pacific/Marquesas,Pacific/Samoa,Pacific/Tongatapu,Pacific/Wake,US/
Alaska,US/Central,US/East-Indiana,US/Eastern,US/Pacific,UTC",
                        "IsModifiable": true,
                        "IsCollection": false
                    }
                ],
                "DBSecurityGroupMemberships": [],
                "VpcSecurityGroupMemberships": []
```

```
            },
            {
                "OptionName": "OEM",
                "OptionDescription": "Oracle 12c EM Express",
                "Persistent": false,
                "Permanent": false,
                "Port": 5500,
                "OptionSettings": [],
                "DBSecurityGroupMemberships": [
                    {
                        "DBSecurityGroupName": "default",
                        "Status": "authorized"
                    }
                ],
                "VpcSecurityGroupMemberships": []
            }
        ],
        "AllowsVpcAndNonVpcInstanceMemberships": false,
        "OptionGroupArn": "arn:aws:rds:us-east-1:123456789012:og:myoptiongroup"
    }
}
```

Per ulteriori informazioni, consulta Aggiunta di un'opzione a un gruppo di opzioni nella Guida per l'utente di Amazon RDS.

- Per i dettagli sull'API, consulta AddOptionToOptionGroup AWS CLICommand Reference.

`add-role-to-db-cluster`

Il seguente esempio di codice mostra come utilizzare`add-role-to-db-cluster`.

AWS CLI

Per associare un ruolo AWS Identity and Access Management (IAM) a un cluster DB

L'esempio `add-role-to-db-cluster` seguente associa un ruolo a un cluster di database.

```
aws rds add-role-to-db-cluster \
    --db-cluster-identifier mydbcluster \
    --role-arn arn:aws:iam::123456789012:role/RDSLoadFromS3
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Associazione di un ruolo IAM a un cluster DB Amazon Aurora MySQL nella Guida per l'utente di Amazon Aurora.

- Per i dettagli sull'API, consulta AddRoleToDbCluster AWS CLICommand Reference.

`add-role-to-db-instance`

Il seguente esempio di codice mostra come utilizzare`add-role-to-db-instance`.

AWS CLI

Per associare un ruolo AWS Identity and Access Management (IAM) a un'istanza DB

L'esempio `add-role-to-db-instance` seguente aggiunge il ruolo a un'istanza database Oracle denominata `test-instance`.

```
aws rds add-role-to-db-instance \
    --db-instance-identifier test-instance \
    --feature-name S3_INTEGRATION \
    --role-arn arn:aws:iam::111122223333:role/rds-s3-integration-role
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Prerequisiti per l'integrazione di Amazon RDS per Oracle con Amazon S3 nella Guida per l'utente di Amazon RDS.

- Per i dettagli sull'API, consulta AddRoleToDbInstance AWS CLICommand Reference.

`add-source-identifier-to-subscription`

Il seguente esempio di codice mostra come utilizzare`add-source-identifier-to-subscription`.

AWS CLI

Come aggiungere un identificatore di origine a un'abbonamento

L'esempio `add-source-identifier` seguente aggiunge l'identificatore di origine all'abbonamento esistente.

```
aws rds add-source-identifier-to-subscription \
```

```
    --subscription-name my-instance-events \
    --source-identifier test-instance-repl
```

Output:

```
{
    "EventSubscription": {
        "SubscriptionCreationTime": "Tue Jul 31 23:22:01 UTC 2018",
        "CustSubscriptionId": "my-instance-events",
        "EventSubscriptionArn": "arn:aws:rds:us-east-1:123456789012:es:my-instance-
events",
        "Enabled": false,
        "Status": "modifying",
        "EventCategoriesList": [
            "backup",
            "recovery"
        ],
        "CustomerAwsId": "123456789012",
        "SnsTopicArn": "arn:aws:sns:us-east-1:123456789012:interesting-events",
        "SourceType": "db-instance",
        "SourceIdsList": [
            "test-instance",
            "test-instance-repl"
        ]
    }
}
```

- Per i dettagli sull'API, consulta AddSourceIdentifierToSubscription AWS CLICommand Reference.

`add-tags-to-resource`

Il seguente esempio di codice mostra come utilizzare`add-tags-to-resource`.

AWS CLI

Come aggiungere tag a una risorsa

L'esempio `add-tags-to-resource` seguente aggiunge tag a un database RDS.

```
aws rds add-tags-to-resource \
    --resource-name arn:aws:rds:us-east-1:123456789012:db:database-mysql \
```

```
    --tags "[{\"Key\": \"Name\",\"Value\": \"MyDatabase\"},{\"Key\": \"Environment
\",\"Value\": \"test\"}]"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta la pagina Tagging delle risorse Amazon RDS nella Guida per l'utente di Amazon RDS.

- Per i dettagli sull'API, consulta AddTagsToResource AWS CLICommand Reference.

`apply-pending-maintenance-action`

Il seguente esempio di codice mostra come utilizzare`apply-pending-maintenance-action`.

AWS CLI

Esecuzione di azioni di manutenzione in attesa

L'esempio `apply-pending-maintenance-action` seguente applica le azioni di manutenzione in sospeso per i cluster di database.

```
aws rds apply-pending-maintenance-action \
    --resource-identifier arn:aws:rds:us-east-1:123456789012:cluster:my-db-cluster \
    --apply-action system-update \
    --opt-in-type immediate
```

Output:

```
{
    "ResourcePendingMaintenanceActions": {
        "ResourceIdentifier": "arn:aws:rds:us-east-1:123456789012:cluster:my-db-
cluster",
        "PendingMaintenanceActionDetails": [
            {
                "Action": "system-update",
                "OptInStatus": "immediate",
                "CurrentApplyDate": "2021-01-23T01:07:36.100Z",
                "Description": "Upgrade to Aurora PostgreSQL 3.3.2"
            }
        ]
    }
}
```

Per ulteriori informazioni, consulta Manutenzione di un'istanza database nella Guida per l'utente di Amazon RDS e Manutenzione di un cluster database Amazon Aurora nella Guida per l'utente di Amazon Aurora.

- Per i dettagli sull'API, consulta ApplyPendingMaintenanceAction AWS CLICommand Reference.

`authorize-db-security-group-ingress`

Il seguente esempio di codice mostra come utilizzare`authorize-db-security-group-ingress`.

AWS CLI

Per associare un ruolo AWS Identity and Access Management (IAM) a un'istanza DB

L'esempio `authorize-db-security-group-ingress` seguente configura il gruppo di sicurezza predefinito con una regola di ingresso per l'intervallo CIDR IP 192.0.2.0/24.

```
aws rds authorize-db-security-group-ingress \
    --db-security-group-name default \
    --cidrip 192.0.2.0/24
```

Output:

```
{
    "DBSecurityGroup": {
        "OwnerId": "123456789012",
        "DBSecurityGroupName": "default",
        "DBSecurityGroupDescription": "default",
        "EC2SecurityGroups": [],
        "IPRanges": [
            {
                "Status": "authorizing",
                "CIDRIP": "192.0.2.0/24"
            }
        ],
        "DBSecurityGroupArn": "arn:aws:rds:us-east-1:111122223333:secgrp:default"
    }
}
```

Per ulteriori informazioni, consulta Autorizzazione dell'accesso a un gruppo di sicurezza database da un intervallo IP nella Guida per l'utente di Amazon RDS.

- Per i dettagli sull'API, consulta AuthorizeDbSecurityGroupIngress AWS CLICommand Reference.

`backtrack-db-cluster`

Il seguente esempio di codice mostra come utilizzare`backtrack-db-cluster`.

AWS CLI

Come eseguire il backtrack di un cluster di database Aurora

L'esempio `backtrack-db-cluster` seguente esegue il backtrack del sample-cluster del cluster di database con il timestamp 19 marzo 2018, ore 10:00.

```
aws rds backtrack-db-cluster --db-cluster-identifier sample-cluster --backtrack-
to 2018-03-19T10:00:00+00:00
```

Questo comando genera un blocco JSON che riconosce la modifica alla risorsa RDS.

- Per i dettagli sull'API, consulta BacktrackDbCluster AWS CLICommand Reference.

`cancel-export-task`

Il seguente esempio di codice mostra come utilizzare`cancel-export-task`.

AWS CLI

Come annullare l'esportazione di uno snapshot in Amazon S3

L'esempio `cancel-export-task` seguente annulla un'attività di esportazione in corso che sta esportando uno snapshot in Amazon S3.

```
aws rds cancel-export-task \
    --export-task-identifier my-s3-export-1
```

Output:

```
{
    "ExportTaskIdentifier": "my-s3-export-1",
    "SourceArn": "arn:aws:rds:us-east-1:123456789012:snapshot:publisher-final-
snapshot",
    "SnapshotTime": "2019-03-24T20:01:09.815Z",
```

```
    "S3Bucket": "amzn-s3-demo-bucket",
    "S3Prefix": "",
    "IamRoleArn": "arn:aws:iam::123456789012:role/service-role/export-snap-S3-role",
    "KmsKeyId": "arn:aws:kms:us-east-1:123456789012:key/abcd0000-7bfd-4594-af38-
aabbccddeeff",
    "Status": "CANCELING",
    "PercentProgress": 0,
    "TotalExtractedDataInGB": 0
}
```

Per ulteriori informazioni, consulta Annullamento di un'attività di esportazione di snapshot nella Guida per l'utente di Amazon RDS o Annullamento di un'attività di esportazione di snapshot nella Guida per l'utente di Amazon Aurora.

- Per i dettagli sull'API, consulta CancelExportTask AWS CLICommand Reference.

`copy-db-cluster-parameter-group`

Il seguente esempio di codice mostra come utilizzare`copy-db-cluster-parameter-group`.

AWS CLI

Come copiare un gruppo di parametri del cluster di database

L'esempio `copy-db-cluster-parameter-group` seguente crea una copia di un gruppo di parametri del cluster di database.

```
aws rds copy-db-cluster-parameter-group \
    --source-db-cluster-parameter-group-identifier mydbclusterpg \
    --target-db-cluster-parameter-group-identifier mydbclusterpgcopy \
    --target-db-cluster-parameter-group-description "Copy of mydbclusterpg parameter
 group"
```

Output:

```
{
    "DBClusterParameterGroup": {
        "DBClusterParameterGroupName": "mydbclusterpgcopy",
        "DBClusterParameterGroupArn": "arn:aws:rds:us-east-1:123456789012:cluster-
pg:mydbclusterpgcopy",
        "DBParameterGroupFamily": "aurora-mysql5.7",
        "Description": "Copy of mydbclusterpg parameter group"
```

```
    }
}
```

Per ulteriori informazioni, consulta Copia di un gruppo di parametri cluster database nella Guida per l'utente di Amazon Aurora.

- Per i dettagli sull'API, consulta CopyDbClusterParameterGroup AWS CLICommand Reference.

`copy-db-cluster-snapshot`

Il seguente esempio di codice mostra come utilizzare`copy-db-cluster-snapshot`.

AWS CLI

Come copiare uno snapshot del cluster di database

L'esempio `copy-db-cluster-snapshot` seguente crea una copia di uno snapshot del cluster di database, inclusi i relativi tag.

```
aws rds copy-db-cluster-snapshot \
    --source-db-cluster-snapshot-identifier arn:aws:rds:us-
east-1:123456789012:cluster-snapshot:rds:myaurora-2019-06-04-09-16
    --target-db-cluster-snapshot-identifier myclustersnapshotcopy \
    --copy-tags
```

Output:

```
{
    "DBClusterSnapshot": {
        "AvailabilityZones": [
            "us-east-1a",
            "us-east-1b",
            "us-east-1e"
        ],
        "DBClusterSnapshotIdentifier": "myclustersnapshotcopy",
        "DBClusterIdentifier": "myaurora",
        "SnapshotCreateTime": "2019-06-04T09:16:42.649Z",
        "Engine": "aurora-mysql",
        "AllocatedStorage": 0,
        "Status": "available",
        "Port": 0,
        "VpcId": "vpc-6594f31c",
        "ClusterCreateTime": "2019-04-15T14:18:42.785Z",
```

```
        "MasterUsername": "myadmin",
        "EngineVersion": "5.7.mysql_aurora.2.04.2",
        "LicenseModel": "aurora-mysql",
        "SnapshotType": "manual",
        "PercentProgress": 100,
        "StorageEncrypted": true,
        "KmsKeyId": "arn:aws:kms:us-east-1:123456789012:key/AKIAIOSFODNN7EXAMPLE",
        "DBClusterSnapshotArn": "arn:aws:rds:us-east-1:123456789012:cluster-
snapshot:myclustersnapshotcopy",
        "IAMDatabaseAuthenticationEnabled": false
    }
}
```

Per ulteriori informazioni, consulta Copia di uno snapshot nella Guida per l'utente di Amazon Aurora.

- Per i dettagli sull'API, consulta CopyDbClusterSnapshot AWS CLICommand Reference.

`copy-db-parameter-group`

Il seguente esempio di codice mostra come utilizzare`copy-db-parameter-group`.

AWS CLI

Come copiare un gruppo di parametri del cluster di database

L'esempio `copy-db-parameter-group` seguente crea una copia di un gruppo di parametri del database.

```
aws rds copy-db-parameter-group \
    --source-db-parameter-group-identifier mydbpg \
    --target-db-parameter-group-identifier mydbpgcopy \
    --target-db-parameter-group-description "Copy of mydbpg parameter group"
```

Output:

```
{
    "DBParameterGroup": {
        "DBParameterGroupName": "mydbpgcopy",
        "DBParameterGroupArn": "arn:aws:rds:us-east-1:814387698303:pg:mydbpgcopy",
        "DBParameterGroupFamily": "mysql5.7",
        "Description": "Copy of mydbpg parameter group"
    }
```

```
}
```

Per ulteriori informazioni, consulta Copia di un gruppo di parametri database nella Guida per l'utente di Amazon RDS.

- Per i dettagli sull'API, consulta CopyDbParameterGroup AWS CLICommand Reference.

`copy-db-snapshot`

Il seguente esempio di codice mostra come utilizzare`copy-db-snapshot`.

AWS CLI

Come copiare uno snapshot database

L'esempio `copy-db-snapshot` seguente crea una copia di uno snapshot database.

```
aws rds copy-db-snapshot \
    --source-db-snapshot-identifier rds:database-mysql-2019-06-06-08-38
    --target-db-snapshot-identifier mydbsnapshotcopy
```

Output:

```
{
    "DBSnapshot": {
        "VpcId": "vpc-6594f31c",
        "Status": "creating",
        "Encrypted": true,
        "SourceDBSnapshotIdentifier": "arn:aws:rds:us-
east-1:123456789012:snapshot:rds:database-mysql-2019-06-06-08-38",
        "MasterUsername": "admin",
        "Iops": 1000,
        "Port": 3306,
        "LicenseModel": "general-public-license",
        "DBSnapshotArn": "arn:aws:rds:us-
east-1:123456789012:snapshot:mydbsnapshotcopy",
        "EngineVersion": "5.6.40",
        "OptionGroupName": "default:mysql-5-6",
        "ProcessorFeatures": [],
        "Engine": "mysql",
        "StorageType": "io1",
        "DbiResourceId": "db-ZI7UJ5BLKMBYFGX7FDENCKADC4",
        "KmsKeyId": "arn:aws:kms:us-east-1:123456789012:key/AKIAIOSFODNN7EXAMPLE",
```

```
        "SnapshotType": "manual",
        "IAMDatabaseAuthenticationEnabled": false,
        "SourceRegion": "us-east-1",
        "DBInstanceIdentifier": "database-mysql",
        "InstanceCreateTime": "2019-04-30T15:45:53.663Z",
        "AvailabilityZone": "us-east-1f",
        "PercentProgress": 0,
        "AllocatedStorage": 100,
        "DBSnapshotIdentifier": "mydbsnapshotcopy"
    }
}
```

Per ulteriori informazioni, consulta Copia di uno snapshot nella Guida per l'utente di Amazon RDS.

- Per i dettagli sull'API, consulta CopyDbSnapshot AWS CLICommand Reference.

`copy-option-group`

Il seguente esempio di codice mostra come utilizzare`copy-option-group`.

AWS CLI

Come copiare un gruppo di opzioni

L'esempio `copy-option-group` seguente crea una copia di un gruppo di opzioni.

```
aws rds copy-option-group \
    --source-option-group-identifier myoptiongroup \
    --target-option-group-identifier new-option-group \
    --target-option-group-description "My option group copy"
```

Output:

```
{
    "OptionGroup": {
        "Options": [],
        "OptionGroupName": "new-option-group",
        "MajorEngineVersion": "11.2",
        "OptionGroupDescription": "My option group copy",
        "AllowsVpcAndNonVpcInstanceMemberships": true,
        "EngineName": "oracle-ee",
        "OptionGroupArn": "arn:aws:rds:us-east-1:123456789012:og:new-option-group"
    }
```

```
}
```

Per ulteriori informazioni, consulta Esecuzione di una copia di un gruppo di opzioni nella Guida per l'utente di Amazon RDS.

- Per i dettagli sull'API, consulta CopyOptionGroup AWS CLICommand Reference.

`create-blue-green-deployment`

Il seguente esempio di codice mostra come utilizzare`create-blue-green-deployment`.

AWS CLI

Esempio 1: creare una blue/green distribuzione per un'istanza DB RDS for MySQL

L'`create-blue-green-deployment`esempio seguente crea una blue/green distribuzione per un'istanza DB MySQL.

```
aws rds create-blue-green-deployment \
    --blue-green-deployment-name bgd-cli-test-instance \
    --source arn:aws:rds:us-east-1:123456789012:db:my-db-instance \
    --target-engine-version 8.0 \
    --target-db-parameter-group-name mysql-80-group
```

Output:

```
{
    "BlueGreenDeployment": {
        "BlueGreenDeploymentIdentifier": "bgd-v53303651eexfake",
        "BlueGreenDeploymentName": "bgd-cli-test-instance",
        "Source": "arn:aws:rds:us-east-1:123456789012:db:my-db-instance",
        "SwitchoverDetails": [
            {
                "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-
instance"
            },
            {
                "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-
instance-replica-1"
            },
            {
                "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-
instance-replica-2"
```

```
            },
            {
                "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-
instance-replica-3"
            }
        ],
        "Tasks": [
            {
                "Name": "CREATING_READ_REPLICA_OF_SOURCE",
                "Status": "PENDING"
            },
            {
                "Name": "DB_ENGINE_VERSION_UPGRADE",
                "Status": "PENDING"
            },
            {
                "Name": "CONFIGURE_BACKUPS",
                "Status": "PENDING"
            },
            {
                "Name": "CREATING_TOPOLOGY_OF_SOURCE",
                "Status": "PENDING"
            }
        ],
        "Status": "PROVISIONING",
        "CreateTime": "2022-02-25T21:18:51.183000+00:00"
    }
}
```

Per ulteriori informazioni, consulta Creare una blue/green distribuzione nella Amazon RDS User Guide.

Esempio 2: creare una blue/green distribuzione per un cluster Aurora MySQL DB

L'`create-blue-green-deployment`esempio seguente crea una blue/green distribuzione per un cluster Aurora MySQL DB.

```
aws rds create-blue-green-deployment \
    --blue-green-deployment-name my-blue-green-deployment \
    --source arn:aws:rds:us-east-1:123456789012:cluster:my-aurora-mysql-cluster \
    --target-engine-version 8.0 \
    --target-db-cluster-parameter-group-name ams-80-binlog-enabled \
    --target-db-parameter-group-name mysql-80-cluster-group
```

Output:


```
{
      "BlueGreenDeployment": {
        "BlueGreenDeploymentIdentifier": "bgd-wi89nwzglccsfake",
        "BlueGreenDeploymentName": "my-blue-green-deployment",
        "Source": "arn:aws:rds:us-east-1:123456789012:cluster:my-aurora-mysql-
cluster",
        "SwitchoverDetails": [
          {
            "SourceMember": "arn:aws:rds:us-east-1:123456789012:cluster:my-aurora-
mysql-cluster",
            "Status": "PROVISIONING"
          },
          {
            "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-aurora-mysql-
cluster-1",
            "Status": "PROVISIONING"
          },
          {
            "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-aurora-mysql-
cluster-2",
            "Status": "PROVISIONING"
          },
          {
            "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-aurora-mysql-
cluster-3",
            "Status": "PROVISIONING"
          },
          {
            "SourceMember": "arn:aws:rds:us-east-1:123456789012:cluster-endpoint:my-
excluded-member-endpoint",
            "Status": "PROVISIONING"
          },
          {
            "SourceMember": "arn:aws:rds:us-east-1:123456789012:cluster-endpoint:my-
reader-endpoint",
            "Status": "PROVISIONING"
          }
        ],
        "Tasks": [
          {
            "Name": "CREATING_READ_REPLICA_OF_SOURCE",
            "Status": "PENDING"
```

```
            },
            {
              "Name": "DB_ENGINE_VERSION_UPGRADE",
              "Status": "PENDING"
            },
            {
              "Name": "CREATE_DB_INSTANCES_FOR_CLUSTER",
              "Status": "PENDING"
            },
            {
              "Name": "CREATE_CUSTOM_ENDPOINTS",
              "Status": "PENDING"
            }
          ],
          "Status": "PROVISIONING",
          "CreateTime": "2022-02-25T21:12:00.288000+00:00"
        }
}
```

Per ulteriori informazioni, consulta Creare una blue/green distribuzione nella Guida per l'utente di Amazon Aurora.

- Per i dettagli sull'API, consulta CreateBlueGreenDeployment AWS CLICommand Reference.

`create-db-cluster-endpoint`

Il seguente esempio di codice mostra come utilizzare`create-db-cluster-endpoint`.

AWS CLI

Come creare un endpoint del cluster di database personalizzato

L'esempio `create-db-cluster-endpoint` seguente crea un endpoint del cluster di database personalizzato e lo associa al cluster Aurora DB specificato.

```
aws rds create-db-cluster-endpoint \
    --db-cluster-endpoint-identifier mycustomendpoint \
    --endpoint-type reader \
    --db-cluster-identifier mydbcluster \
    --static-members dbinstance1 dbinstance2
```

Output:

```
{
    "DBClusterEndpointIdentifier": "mycustomendpoint",
    "DBClusterIdentifier": "mydbcluster",
    "DBClusterEndpointResourceIdentifier": "cluster-endpoint-ANPAJ4AE5446DAEXAMPLE",
    "Endpoint": "mycustomendpoint.cluster-custom-cnpexample.us-
east-1.rds.amazonaws.com",
    "Status": "creating",
    "EndpointType": "CUSTOM",
    "CustomEndpointType": "READER",
    "StaticMembers": [
        "dbinstance1",
        "dbinstance2"
    ],
    "ExcludedMembers": [],
    "DBClusterEndpointArn": "arn:aws:rds:us-east-1:123456789012:cluster-
endpoint:mycustomendpoint"
}
```

Per ulteriori informazioni, consulta Gestione delle connessioni Amazon Aurora nella Guida per l'utente di Amazon Aurora.

- Per i dettagli sull'API, consulta CreateDbClusterEndpoint AWS CLICommand Reference.

`create-db-cluster-parameter-group`

Il seguente esempio di codice mostra come utilizzare`create-db-cluster-parameter-group`.

AWS CLI

Come creare un gruppo di parametri del cluster di database

L'esempio `create-db-cluster-parameter-group` seguente crea un gruppo di parametri del cluster di database.

```
aws rds create-db-cluster-parameter-group \
    --db-cluster-parameter-group-name mydbclusterparametergroup \
    --db-parameter-group-family aurora5.6 \
    --description "My new cluster parameter group"
```

Output:

```
{
```

```
    "DBClusterParameterGroup": {
        "DBClusterParameterGroupName": "mydbclusterparametergroup",
        "DBParameterGroupFamily": "aurora5.6",
        "Description": "My new cluster parameter group",
        "DBClusterParameterGroupArn": "arn:aws:rds:us-east-1:123456789012:cluster-
pg:mydbclusterparametergroup"
    }
}
```

Per ulteriori informazioni, consulta Creazione di un gruppo di parametri cluster database nella Guida per l'utente di Amazon Aurora.

- Per i dettagli sull'API, consulta CreateDbClusterParameterGroup AWS CLICommand Reference.

`create-db-cluster-snapshot`

Il seguente esempio di codice mostra come utilizzare`create-db-cluster-snapshot`.

AWS CLI

Come creare uno snapshot del cluster di database

L'esempio `create-db-cluster-snapshot` seguente crea uno snapshot del cluster di database.

```
aws rds create-db-cluster-snapshot \
    --db-cluster-identifier mydbcluster \
    --db-cluster-snapshot-identifier mydbclustersnapshot
```

Output:

```
{
    "DBClusterSnapshot": {
        "AvailabilityZones": [
            "us-east-1a",
            "us-east-1b",
            "us-east-1e"
        ],
        "DBClusterSnapshotIdentifier": "mydbclustersnapshot",
        "DBClusterIdentifier": "mydbcluster",
        "SnapshotCreateTime": "2019-06-18T21:21:00.469Z",
        "Engine": "aurora-mysql",
```

```
        "AllocatedStorage": 1,
        "Status": "creating",
        "Port": 0,
        "VpcId": "vpc-6594f31c",
        "ClusterCreateTime": "2019-04-15T14:18:42.785Z",
        "MasterUsername": "myadmin",
        "EngineVersion": "5.7.mysql_aurora.2.04.2",
        "LicenseModel": "aurora-mysql",
        "SnapshotType": "manual",
        "PercentProgress": 0,
        "StorageEncrypted": true,
        "KmsKeyId": "arn:aws:kms:us-east-1:123456789012:key/AKIAIOSFODNN7EXAMPLE",
        "DBClusterSnapshotArn": "arn:aws:rds:us-east-1:123456789012:cluster-
snapshot:mydbclustersnapshot",
        "IAMDatabaseAuthenticationEnabled": false
    }
}
```

Per ulteriori informazioni, consulta Creazione di una snapshot di cluster database nella Guida per l'utente di Amazon Aurora.

- Per i dettagli sull'API, consulta CreateDbClusterSnapshot AWS CLICommand Reference.

`create-db-cluster`

Il seguente esempio di codice mostra come utilizzare`create-db-cluster`.

AWS CLI

Esempio 1: come creare un cluster di database compatibile con MySQL

L'esempio `create-db-cluster` seguente crea un cluster di database compatibile con MySQL 5.7 utilizzando la versione del motore predefinita. Sostituisci la password di esempio `secret99` con una password sicura. Se utilizzi la console per creare un cluster di database, Amazon RDS crea automaticamente l'istanza di database core di scrittura per il cluster di database. Tuttavia, quando si utilizza la AWS CLI per creare un cluster DB, è necessario creare in modo esplicito l'istanza Writer DB per il cluster DB utilizzando il comando `create-db-instance` AWS CLI.

```
aws rds create-db-cluster \
    --db-cluster-identifier sample-cluster \
    --engine aurora-mysql \
    --engine-version 5.7 \
```

```
    --master-username admin \
    --master-user-password secret99 \
    --db-subnet-group-name default \
    --vpc-security-group-ids sg-0b9130572daf3dc16
```

Output:


```
{
    "DBCluster": {
        "DBSubnetGroup": "default",
        "VpcSecurityGroups": [
            {
                "VpcSecurityGroupId": "sg-0b9130572daf3dc16",
                "Status": "active"
            }
        ],
        "AllocatedStorage": 1,
        "AssociatedRoles": [],
        "PreferredBackupWindow": "09:12-09:42",
        "ClusterCreateTime": "2023-02-27T23:21:33.048Z",
        "DeletionProtection": false,
        "IAMDatabaseAuthenticationEnabled": false,
        "ReadReplicaIdentifiers": [],
        "EngineMode": "provisioned",
        "Engine": "aurora-mysql",
        "StorageEncrypted": false,
        "MultiAZ": false,
        "PreferredMaintenanceWindow": "mon:04:31-mon:05:01",
        "HttpEndpointEnabled": false,
        "BackupRetentionPeriod": 1,
        "DbClusterResourceId": "cluster-ANPAJ4AE5446DAEXAMPLE",
        "DBClusterIdentifier": "sample-cluster",
        "AvailabilityZones": [
            "us-east-1a",
            "us-east-1b",
            "us-east-1e"
        ],
        "MasterUsername": "master",
        "EngineVersion": "5.7.mysql_aurora.2.11.1",
        "DBClusterArn": "arn:aws:rds:us-east-1:123456789012:cluster:sample-cluster",
        "DBClusterMembers": [],
        "Port": 3306,
        "Status": "creating",
```

```
        "Endpoint": "sample-cluster.cluster-cnpexample.us-east-1.rds.amazonaws.com",
        "DBClusterParameterGroup": "default.aurora-mysql5.7",
        "HostedZoneId": "Z2R2ITUGPM61AM",
        "ReaderEndpoint": "sample-cluster.cluster-ro-cnpexample.us-
east-1.rds.amazonaws.com",
        "CopyTagsToSnapshot": false
    }
}
```

Esempio 2: come creare un cluster di database compatibile con PostgreSQL

L'esempio `create-db-cluster` seguente crea un cluster di database compatibile con PostgreSQL utilizzando la versione del motore predefinita. Sostituisci la password `secret99` di esempio con una password sicura. Se utilizzi la console per creare un cluster di database, Amazon RDS crea automaticamente l'istanza di database core di scrittura per il cluster di database. Tuttavia, quando si utilizza la AWS CLI per creare un cluster DB, è necessario creare in modo esplicito l'istanza Writer DB per il cluster DB utilizzando il comando `create-db-instance` AWS CLI.

```
aws rds create-db-cluster \
    --db-cluster-identifier sample-pg-cluster \
    --engine aurora-postgresql \
    --master-username master \
    --master-user-password secret99 \
    --db-subnet-group-name default \
    --vpc-security-group-ids sg-0b9130572daf3dc16
```

Output:

```
{
    "DBCluster": {
        "Endpoint": "sample-pg-cluster.cluster-cnpexample.us-
east-1.rds.amazonaws.com",
        "HttpEndpointEnabled": false,
        "DBClusterMembers": [],
        "EngineMode": "provisioned",
        "CopyTagsToSnapshot": false,
        "HostedZoneId": "Z2R2ITUGPM61AM",
        "IAMDatabaseAuthenticationEnabled": false,
        "AllocatedStorage": 1,
        "VpcSecurityGroups": [
```

```
            {
                "VpcSecurityGroupId": "sg-0b9130572daf3dc16",
                "Status": "active"
            }
        ],
        "DeletionProtection": false,
        "StorageEncrypted": false,
        "BackupRetentionPeriod": 1,
        "PreferredBackupWindow": "09:56-10:26",
        "ClusterCreateTime": "2023-02-27T23:26:08.371Z",
        "DBClusterParameterGroup": "default.aurora-postgresql13",
        "EngineVersion": "13.7",
        "Engine": "aurora-postgresql",
        "Status": "creating",
        "DBClusterIdentifier": "sample-pg-cluster",
        "MultiAZ": false,
        "Port": 5432,
        "DBClusterArn": "arn:aws:rds:us-east-1:123456789012:cluster:sample-pg-
cluster",
        "AssociatedRoles": [],
        "DbClusterResourceId": "cluster-ANPAJ4AE5446DAEXAMPLE",
        "PreferredMaintenanceWindow": "wed:03:33-wed:04:03",
        "ReaderEndpoint": "sample-pg-cluster.cluster-ro-cnpexample.us-
east-1.rds.amazonaws.com",
        "MasterUsername": "master",
        "AvailabilityZones": [
            "us-east-1a",
            "us-east-1b",
            "us-east-1c"
        ],
        "ReadReplicaIdentifiers": [],
        "DBSubnetGroup": "default"
    }
}
```

Per ulteriori informazioni, consulta Creazione di un cluster database Amazon Aurora nella Guida per l'utente di Amazon Aurora.

- Per i dettagli sull'API, consulta Command CreateDbClusterReference AWS CLI .

`create-db-instance-read-replica`

Il seguente esempio di codice mostra come utilizzare`create-db-instance-read-replica`.

AWS CLI

Come creare una replica di lettura dell'istanza database

Questo esempio crea una replica di lettura di un'istanza database esistente denominata `test-instance`. La replica di lettura è denominata `test-instance-repl`.

```
aws rds create-db-instance-read-replica \
    --db-instance-identifier test-instance-repl \
    --source-db-instance-identifier test-instance
```

Output:

```
{
    "DBInstance": {
        "IAMDatabaseAuthenticationEnabled": false,
        "MonitoringInterval": 0,
        "DBInstanceArn": "arn:aws:rds:us-east-1:123456789012:db:test-instance-repl",
        "ReadReplicaSourceDBInstanceIdentifier": "test-instance",
        "DBInstanceIdentifier": "test-instance-repl",
        ...some output truncated...
    }
}
```

- Per i dettagli sull'API, consulta CreateDbInstanceReadReplica AWS CLICommand Reference.

`create-db-instance`

Il seguente esempio di codice mostra come utilizzare`create-db-instance`.

AWS CLI

Come creare un'istanza database

L'esempio `create-db-instance` seguente utilizza le opzioni richieste per avviare una nuova istanza database.

```
aws rds create-db-instance \
    --db-instance-identifier test-mysql-instance \
    --db-instance-class db.t3.micro \
    --engine mysql \
```

```
    --master-username admin \
    --master-user-password secret99 \
    --allocated-storage 20
```

Output:

```
{
    "DBInstance": {
        "DBInstanceIdentifier": "test-mysql-instance",
        "DBInstanceClass": "db.t3.micro",
        "Engine": "mysql",
        "DBInstanceStatus": "creating",
        "MasterUsername": "admin",
        "AllocatedStorage": 20,
        "PreferredBackupWindow": "12:55-13:25",
        "BackupRetentionPeriod": 1,
        "DBSecurityGroups": [],
        "VpcSecurityGroups": [
            {
                "VpcSecurityGroupId": "sg-12345abc",
                "Status": "active"
            }
        ],
        "DBParameterGroups": [
            {
                "DBParameterGroupName": "default.mysql5.7",
                "ParameterApplyStatus": "in-sync"
            }
        ],
        "DBSubnetGroup": {
            "DBSubnetGroupName": "default",
            "DBSubnetGroupDescription": "default",
            "VpcId": "vpc-2ff2ff2f",
            "SubnetGroupStatus": "Complete",
            "Subnets": [
                {
                    "SubnetIdentifier": "subnet-########",
                    "SubnetAvailabilityZone": {
                        "Name": "us-west-2c"
                    },
                    "SubnetStatus": "Active"
                },
                {
```

```
                "SubnetIdentifier": "subnet-########",
                "SubnetAvailabilityZone": {
                    "Name": "us-west-2d"
                },
                "SubnetStatus": "Active"
            },
            {
                "SubnetIdentifier": "subnet-########",
                "SubnetAvailabilityZone": {
                    "Name": "us-west-2a"
                },
                "SubnetStatus": "Active"
            },
            {
                "SubnetIdentifier": "subnet-########",
                "SubnetAvailabilityZone": {
                    "Name": "us-west-2b"
                },
                "SubnetStatus": "Active"
            }
        ]
    },
    "PreferredMaintenanceWindow": "sun:08:07-sun:08:37",
    "PendingModifiedValues": {
        "MasterUserPassword": "****"
    },
    "MultiAZ": false,
    "EngineVersion": "5.7.22",
    "AutoMinorVersionUpgrade": true,
    "ReadReplicaDBInstanceIdentifiers": [],
    "LicenseModel": "general-public-license",
    "OptionGroupMemberships": [
        {
            "OptionGroupName": "default:mysql-5-7",
            "Status": "in-sync"
        }
    ],
    "PubliclyAccessible": true,
    "StorageType": "gp2",
    "DbInstancePort": 0,
    "StorageEncrypted": false,
    "DbiResourceId": "db-5555EXAMPLE44444444EXAMPLE",
    "CACertificateIdentifier": "rds-ca-2019",
    "DomainMemberships": [],
```

```
        "CopyTagsToSnapshot": false,
        "MonitoringInterval": 0,
        "DBInstanceArn": "arn:aws:rds:us-west-2:123456789012:db:test-mysql-
instance",
        "IAMDatabaseAuthenticationEnabled": false,
        "PerformanceInsightsEnabled": false,
        "DeletionProtection": false,
        "AssociatedRoles": []
    }
}
```

Per ulteriori informazioni, consulta la pagina Creazione di un'istanza database Amazon RDS nella Guida per l'utente di Amazon RDS.

- Per i dettagli sull'API, consulta Create DBInstance in AWS CLI Command Reference.

`create-db-parameter-group`

Il seguente esempio di codice mostra come utilizzare`create-db-parameter-group`.

AWS CLI

Come creare un gruppo di parametri del database

L'esempio `create-db-parameter-group` seguente crea un gruppo di parametri database.

```
aws rds create-db-parameter-group \
    --db-parameter-group-name mydbparametergroup \
    --db-parameter-group-family MySQL5.6 \
    --description "My new parameter group"
```

Output:

```
{
    "DBParameterGroup": {
        "DBParameterGroupName": "mydbparametergroup",
        "DBParameterGroupFamily": "mysql5.6",
        "Description": "My new parameter group",
        "DBParameterGroupArn": "arn:aws:rds:us-
east-1:123456789012:pg:mydbparametergroup"
    }
}
```

Per ulteriori informazioni, consulta Creazione di un gruppo di parametri database nella Guida per l'utente di Amazon RDS.

- Per i dettagli sull'API, consulta Create DBParameter Group in AWS CLI Command Reference.

`create-db-proxy-endpoint`

Il seguente esempio di codice mostra come utilizzare`create-db-proxy-endpoint`.

AWS CLI

Come creare un endpoint proxy database per un database RDS

L'esempio `create-db-proxy-endpoint` seguente crea un endpoint proxy database personalizzato.

```
aws rds create-db-proxy-endpoint \
    --db-proxy-name proxyExample \
    --db-proxy-endpoint-name "proxyep1" \
    --vpc-subnet-ids subnetgroup1 subnetgroup2
```

Output:

```
{
"DBProxyEndpoint": {
        "DBProxyEndpointName": "proxyep1",
        "DBProxyEndpointArn": "arn:aws:rds:us-east-1:123456789012:db-proxy-
endpoint:prx-endpoint-0123a01b12345c0ab",
        "DBProxyName": "proxyExample",
        "Status": "creating",
        "VpcId": "vpc-1234567",
        "VpcSecurityGroupIds": [
            "sg-1234",
            "sg-5678"
        ],
        "VpcSubnetIds": [
            "subnetgroup1",
            "subnetgroup2"
        ],
        "Endpoint": "proxyep1.endpoint.proxy-ab0cd1efghij.us-
east-1.rds.amazonaws.com",
        "CreatedDate": "2023-04-05T16:09:33.452000+00:00",
        "TargetRole": "READ_WRITE",
```

```
        "IsDefault": false
    }
}
```

Per ulteriori informazioni, consulta Creazione di un endpoint proxy nella Guida per l'utente di Amazon RDS e Creazione di un endpoint proxy nella Guida per l'utente di Amazon Aurora.

- Per i dettagli sull'API, consulta CreateDbProxyEndpoint AWS CLICommand Reference.

`create-db-proxy`

Il seguente esempio di codice mostra come utilizzare`create-db-proxy`.

AWS CLI

Come creare un proxy database per un database RDS

L'esempio `create-db-proxy` seguente crea un proxy database.

```
aws rds create-db-proxy \
    --db-proxy-name proxyExample \
    --engine-family MYSQL \
    --auth
 Description="proxydescription1",AuthScheme="SECRETS",SecretArn="arn:aws:secretsmanager:us-
west-2:123456789123:secret:secretName-1234f",IAMAuth="DISABLED",ClientPasswordAuthType="MYSQ
 \
    --role-arn arn:aws:iam::123456789123:role/ProxyRole \
    --vpc-subnet-ids subnetgroup1 subnetgroup2
```

Output:

```
{
"DBProxy": {
        "DBProxyName": "proxyExample",
        "DBProxyArn": "arn:aws:rds:us-east-1:123456789012:db-
proxy:prx-0123a01b12345c0ab",
        "EngineFamily": "MYSQL",
        "VpcId": "vpc-1234567",
        "VpcSecuritytGroupIds": [
            "sg-1234",
            "sg-5678",
            "sg-9101"
```

```
        ],
        "VpcSubnetIds": [
            "subnetgroup1",
            "subnetgroup2"
        ],
        "Auth": "[
            {
                "Description": "proxydescription1",
                "AuthScheme": "SECRETS",
                "SecretArn": "arn:aws:secretsmanager:us-
west-2:123456789123:secret:proxysecret1-Abcd1e",
                "IAMAuth": "DISABLED"
            }
        ]",
        "RoleArn": "arn:aws:iam::12345678912:role/ProxyRole",
        "Endpoint": "proxyExample.proxy-ab0cd1efghij.us-east-1.rds.amazonaws.com",
        "RequireTLS": false,
        "IdleClientTimeout": 1800,
        "DebuggingLogging": false,
        "CreatedDate": "2023-04-05T16:09:33.452000+00:00",
        "UpdatedDate": "2023-04-13T01:49:38.568000+00:00"
    }
}
```

Per ulteriori informazioni, consulta Creazione di un RDS Proxy nella Guida per l'utente di Amazon RDS e Creazione di un RDS Proxy nella Guida per l'utente di Amazon Aurora.

- Per i dettagli sull'API, consulta CreateDbProxy AWS CLICommand Reference.

`create-db-security-group`

Il seguente esempio di codice mostra come utilizzare`create-db-security-group`.

AWS CLI

Come creare un gruppo di sicurezza database Amazon RDS

Il comando `create-db-security-group` seguente crea un nuovo gruppo di sicurezza del database Amazon RDS:

```
aws rds create-db-security-group --db-security-group-name mysecgroup --db-security-
group-description "My Test Security Group"
```

Nell'esempio il nuovo gruppo di sicurezza database è denominato `mysecgroup` e ha una descrizione.

Output:

```
{
    "DBSecurityGroup": {
        "OwnerId": "123456789012",
        "DBSecurityGroupName": "mysecgroup",
        "DBSecurityGroupDescription": "My Test Security Group",
        "VpcId": "vpc-a1b2c3d4",
        "EC2SecurityGroups": [],
        "IPRanges": [],
        "DBSecurityGroupArn": "arn:aws:rds:us-west-2:123456789012:secgrp:mysecgroup"
    }
}
```

- Per i dettagli sull'API, consulta CreateDbSecurityGroup AWS CLICommand Reference.

`create-db-shard-group`

Il seguente esempio di codice mostra come utilizzare`create-db-shard-group`.

AWS CLI

Esempio 1: come creare un cluster di database primario Aurora PostgreSQL

L'esempio `create-db-cluster` seguente crea un cluster di database primario Aurora PostgreSQL SQL compatibile con Aurora Serverless v2 e Aurora Limitless Database.

```
aws rds create-db-cluster \
    --db-cluster-identifier my-sv2-cluster \
    --engine aurora-postgresql \
    --engine-version 15.2-limitless \
    --storage-type aurora-iopt1 \
    --serverless-v2-scaling-configuration MinCapacity=2,MaxCapacity=16 \
    --enable-limitless-database \
    --master-username myuser \
    --master-user-password mypassword \
    --enable-cloudwatch-logs-exports postgresql
```

Output:

```
{
    "DBCluster": {
        "AllocatedStorage": 1,
        "AvailabilityZones": [
            "us-east-2b",
            "us-east-2c",
            "us-east-2a"
        ],
        "BackupRetentionPeriod": 1,
        "DBClusterIdentifier": "my-sv2-cluster",
        "DBClusterParameterGroup": "default.aurora-postgresql15",
        "DBSubnetGroup": "default",
        "Status": "creating",
        "Endpoint": "my-sv2-cluster.cluster-cekycexample.us-
east-2.rds.amazonaws.com",
        "ReaderEndpoint": "my-sv2-cluster.cluster-ro-cekycexample.us-
east-2.rds.amazonaws.com",
        "MultiAZ": false,
        "Engine": "aurora-postgresql",
        "EngineVersion": "15.2-limitless",
        "Port": 5432,
        "MasterUsername": "myuser",
        "PreferredBackupWindow": "06:05-06:35",
        "PreferredMaintenanceWindow": "mon:08:25-mon:08:55",
        "ReadReplicaIdentifiers": [],
        "DBClusterMembers": [],
        "VpcSecurityGroups": [
            {
                "VpcSecurityGroupId": "sg-########",
                "Status": "active"
            }
        ],
        "HostedZoneId": "Z2XHWR1EXAMPLE",
        "StorageEncrypted": false,
        "DbClusterResourceId": "cluster-XYEDT6ML6FHIXH4Q2J1EXAMPLE",
        "DBClusterArn": "arn:aws:rds:us-east-2:123456789012:cluster:my-sv2-cluster",
        "AssociatedRoles": [],
        "IAMDatabaseAuthenticationEnabled": false,
        "ClusterCreateTime": "2024-02-19T16:24:07.771000+00:00",
        "EnabledCloudwatchLogsExports": [
            "postgresql"
        ],
        "EngineMode": "provisioned",
```

```
        "DeletionProtection": false,
        "HttpEndpointEnabled": false,
        "CopyTagsToSnapshot": false,
        "CrossAccountClone": false,
        "DomainMemberships": [],
        "TagList": [],
        "StorageType": "aurora-iopt1",
        "AutoMinorVersionUpgrade": true,
        "ServerlessV2ScalingConfiguration": {
            "MinCapacity": 2.0,
            "MaxCapacity": 16.0
        },
        "NetworkType": "IPV4",
        "IOOptimizedNextAllowedModificationTime":
 "2024-03-21T16:24:07.781000+00:00",
        "LimitlessDatabase": {
            "Status": "not-in-use",
            "MinRequiredACU": 96.0
        }
    }
}
```

Esempio 2: come creare l'istanza database primaria (istanza di scrittura)

L'esempio `create-db-instance` seguente crea un'istanza database primaria (di scrittura) Aurora Serverless v2. Se utilizzi la console per creare un cluster di database, Amazon RDS crea automaticamente l'istanza di database core di scrittura per il cluster di database. Tuttavia, quando si utilizza la AWS CLI per creare un cluster DB, è necessario creare in modo esplicito l'istanza Writer DB per il cluster DB utilizzando il comando `create-db-instance` AWS CLI.

```
aws rds create-db-instance \
    --db-instance-identifier my-sv2-instance \
    --db-cluster-identifier my-sv2-cluster \
    --engine aurora-postgresql \
    --db-instance-class db.serverless
```

Output:

```
{
    "DBInstance": {
        "DBInstanceIdentifier": "my-sv2-instance",
        "DBInstanceClass": "db.serverless",
```

```
        "Engine": "aurora-postgresql",
        "DBInstanceStatus": "creating",
        "MasterUsername": "myuser",
        "AllocatedStorage": 1,
        "PreferredBackupWindow": "06:05-06:35",
        "BackupRetentionPeriod": 1,
        "DBSecurityGroups": [],
        "VpcSecurityGroups": [
            {
                "VpcSecurityGroupId": "sg-########",
                "Status": "active"
            }
        ],
        "DBParameterGroups": [
            {
                "DBParameterGroupName": "default.aurora-postgresql15",
                "ParameterApplyStatus": "in-sync"
            }
        ],
        "DBSubnetGroup": {
            "DBSubnetGroupName": "default",
            "DBSubnetGroupDescription": "default",
            "VpcId": "vpc-########",
            "SubnetGroupStatus": "Complete",
            "Subnets": [
                {
                    "SubnetIdentifier": "subnet-########",
                    "SubnetAvailabilityZone": {
                        "Name": "us-east-2c"
                    },
                    "SubnetOutpost": {},
                    "SubnetStatus": "Active"
                },
                {
                    "SubnetIdentifier": "subnet-########",
                    "SubnetAvailabilityZone": {
                        "Name": "us-east-2a"
                    },
                    "SubnetOutpost": {},
                    "SubnetStatus": "Active"
                },
                {
                    "SubnetIdentifier": "subnet-########",
                    "SubnetAvailabilityZone": {
```

```
                    "Name": "us-east-2b"
                },
                "SubnetOutpost": {},
                "SubnetStatus": "Active"
            }
        ]
    },
    "PreferredMaintenanceWindow": "fri:09:01-fri:09:31",
    "PendingModifiedValues": {
        "PendingCloudwatchLogsExports": {
            "LogTypesToEnable": [
                "postgresql"
            ]
        }
    },
    "MultiAZ": false,
    "EngineVersion": "15.2-limitless",
    "AutoMinorVersionUpgrade": true,
    "ReadReplicaDBInstanceIdentifiers": [],
    "LicenseModel": "postgresql-license",
    "OptionGroupMemberships": [
        {
            "OptionGroupName": "default:aurora-postgresql-15",
            "Status": "in-sync"
        }
    ],
    "PubliclyAccessible": false,
    "StorageType": "aurora-iopt1",
    "DbInstancePort": 0,
    "DBClusterIdentifier": "my-sv2-cluster",
    "StorageEncrypted": false,
    "DbiResourceId": "db-BIQTE3B3K3RM7M74SK5EXAMPLE",
    "CACertificateIdentifier": "rds-ca-rsa2048-g1",
    "DomainMemberships": [],
    "CopyTagsToSnapshot": false,
    "MonitoringInterval": 0,
    "PromotionTier": 1,
    "DBInstanceArn": "arn:aws:rds:us-east-2:123456789012:db:my-sv2-instance",
    "IAMDatabaseAuthenticationEnabled": false,
    "PerformanceInsightsEnabled": false,
    "DeletionProtection": false,
    "AssociatedRoles": [],
    "TagList": [],
    "CustomerOwnedIpEnabled": false,
```

```
        "BackupTarget": "region",
        "NetworkType": "IPV4",
        "StorageThroughput": 0,
        "CertificateDetails": {
            "CAIdentifier": "rds-ca-rsa2048-g1"
        },
        "DedicatedLogVolume": false
    }
}
```

Esempio 3: come creare il gruppo di shard di database

L'esempio `create-db-shard-group` seguente crea un gruppo shard del database nel cluster database primario Aurora PostgreSQL.

```
aws rds create-db-shard-group \
    --db-shard-group-identifier my-db-shard-group \
    --db-cluster-identifier my-sv2-cluster \
    --max-acu 768
```

Output:

```
{
    "DBShardGroupResourceId": "shardgroup-a6e3a0226aa243e2ac6c7a1234567890",
    "DBShardGroupIdentifier": "my-db-shard-group",
    "DBClusterIdentifier": "my-sv2-cluster",
    "MaxACU": 768.0,
    "ComputeRedundancy": 0,
    "Status": "creating",
    "PubliclyAccessible": false,
    "Endpoint": "my-sv2-cluster.limitless-cekycexample.us-east-2.rds.amazonaws.com"
}
```

Per ulteriori informazioni, consulta Utilizzo di Amazon Aurora serverless v2 nella Guida per l'utente di Amazon Aurora.

- Per i dettagli sull'API, consulta Command CreateDbShardGroupReference AWS CLI .

`create-db-snapshot`

Il seguente esempio di codice mostra come utilizzare`create-db-snapshot`.

AWS CLI

Come creare uno snapshot del database

L'esempio `create-db-snapshot` seguente crea uno snapshot del database.

```
aws rds create-db-snapshot \
    --db-instance-identifier database-mysql \
    --db-snapshot-identifier mydbsnapshot
```

Output:

```
{
    "DBSnapshot": {
        "DBSnapshotIdentifier": "mydbsnapshot",
        "DBInstanceIdentifier": "database-mysql",
        "Engine": "mysql",
        "AllocatedStorage": 100,
        "Status": "creating",
        "Port": 3306,
        "AvailabilityZone": "us-east-1b",
        "VpcId": "vpc-6594f31c",
        "InstanceCreateTime": "2019-04-30T15:45:53.663Z",
        "MasterUsername": "admin",
        "EngineVersion": "5.6.40",
        "LicenseModel": "general-public-license",
        "SnapshotType": "manual",
        "Iops": 1000,
        "OptionGroupName": "default:mysql-5-6",
        "PercentProgress": 0,
        "StorageType": "io1",
        "Encrypted": true,
        "KmsKeyId": "arn:aws:kms:us-east-1:123456789012:key/AKIAIOSFODNN7EXAMPLE",
        "DBSnapshotArn": "arn:aws:rds:us-east-1:123456789012:snapshot:mydbsnapshot",
        "IAMDatabaseAuthenticationEnabled": false,
        "ProcessorFeatures": [],
        "DbiResourceId": "db-AKIAIOSFODNN7EXAMPLE"
    }
}
```

Per ulteriori informazioni, consulta Creazione di uno snapshot di database nella Guida per l'utente di Amazon RDS.

- Per i dettagli sull'API, consulta Create DBSnapshot in AWS CLI Command Reference.

`create-db-subnet-group`

Il seguente esempio di codice mostra come utilizzare`create-db-subnet-group`.

AWS CLI

Come creare un gruppo di sottoreti database

L'esempio `create-db-subnet-group` seguente crea un gruppo di sottoreti database denominato `mysubnetgroup` utilizzando sottoreti esistenti.

```
aws rds create-db-subnet-group \
    --db-subnet-group-name mysubnetgroup \
    --db-subnet-group-description "test DB subnet group" \
    --subnet-ids
 '["subnet-0a1dc4e1a6f123456","subnet-070dd7ecb3aaaaaaa","subnet-00f5b198bc0abcdef"]'
```

Output:

```
{
    "DBSubnetGroup": {
        "DBSubnetGroupName": "mysubnetgroup",
        "DBSubnetGroupDescription": "test DB subnet group",
        "VpcId": "vpc-0f08e7610a1b2c3d4",
        "SubnetGroupStatus": "Complete",
        "Subnets": [
            {
                "SubnetIdentifier": "subnet-070dd7ecb3aaaaaaa",
                "SubnetAvailabilityZone": {
                    "Name": "us-west-2b"
                },
                "SubnetStatus": "Active"
            },
            {
                "SubnetIdentifier": "subnet-00f5b198bc0abcdef",
                "SubnetAvailabilityZone": {
                    "Name": "us-west-2d"
                },
                "SubnetStatus": "Active"
            },
```

```
            {
                "SubnetIdentifier": "subnet-0a1dc4e1a6f123456",
                "SubnetAvailabilityZone": {
                    "Name": "us-west-2b"
                },
                "SubnetStatus": "Active"
            }
        ],
        "DBSubnetGroupArn": "arn:aws:rds:us-
west-2:0123456789012:subgrp:mysubnetgroup"
    }
}
```

Per ulteriori informazioni, consulta Creazione di un'istanza database in un VPC nella Guida per l'utente di Amazon RDS.

- Per i dettagli sull'API, consulta CreateDbSubnetGroup AWS CLICommand Reference.

`create-event-subscription`

Il seguente esempio di codice mostra come utilizzare`create-event-subscription`.

AWS CLI

Come creare una sottoscrizione a un evento

L'`create-event-subscription`esempio seguente crea una sottoscrizione per gli eventi di backup e ripristino per le istanze DB nell' AWS account corrente. Le notifiche vengono inviate a un argomento Amazon Simple Notification Service specificato da `--sns-topic-arn`.

```
aws rds create-event-subscription \
    --subscription-name my-instance-events \
    --source-type db-instance \
    --event-categories '["backup","recovery"]' \
    --sns-topic-arn arn:aws:sns:us-east-1:123456789012:interesting-events
```

Output:

```
{
    "EventSubscription": {
        "Status": "creating",
        "CustSubscriptionId": "my-instance-events",
```

```
        "SubscriptionCreationTime": "Tue Jul 31 23:22:01 UTC 2018",
        "EventCategoriesList": [
            "backup",
            "recovery"
        ],
        "SnsTopicArn": "arn:aws:sns:us-east-1:123456789012:interesting-events",
        "CustomerAwsId": "123456789012",
        "EventSubscriptionArn": "arn:aws:rds:us-east-1:123456789012:es:my-instance-
events",
        "SourceType": "db-instance",
        "Enabled": true
    }
}
```

- Per i dettagli sull'API, consulta CreateEventSubscription AWS CLICommand Reference.

`create-global-cluster`

Il seguente esempio di codice mostra come utilizzare`create-global-cluster`.

AWS CLI

Come creare un nuovo cluster di database globale

L'esempio `create-global-cluster` seguente crea un nuovo cluster di database globale compatibile con Aurora MySQL.

```
aws rds create-global-cluster \
    --global-cluster-identifier myglobalcluster \
    --engine aurora-mysql
```

Output:

```
{
    "GlobalCluster": {
        "GlobalClusterIdentifier": "myglobalcluster",
        "GlobalClusterResourceId": "cluster-f0e523bfe07aabb",
        "GlobalClusterArn": "arn:aws:rds::123456789012:global-
cluster:myglobalcluster",
        "Status": "available",
        "Engine": "aurora-mysql",
        "EngineVersion": "5.7.mysql_aurora.2.07.2",
```

```
        "StorageEncrypted": false,
        "DeletionProtection": false,
        "GlobalClusterMembers": []
    }
}
```

Per ulteriori informazioni, consulta Creazione di un database globale Aurora nella Guida per l'utente di Amazon Aurora.

- Per i dettagli sull'API, consulta CreateGlobalCluster AWS CLICommand Reference.

`create-option-group`

Il seguente esempio di codice mostra come utilizzare`create-option-group`.

AWS CLI

Come creare un gruppo di opzioni Amazon RDS

Il comando `create-option-group` seguente crea un nuovo gruppo di opzioni Amazon RDS per la versione `Oracle Enterprise Edition 11.2`, is named ``MyOptionGroup` e include una descrizione.

```
aws rds create-option-group \
    --option-group-name MyOptionGroup \
    --engine-name oracle-ee \
    --major-engine-version 11.2 \
    --option-group-description "Oracle Database Manager Database Control"
```

Output:

```
{
    "OptionGroup": {
        "OptionGroupName": "myoptiongroup",
        "OptionGroupDescription": "Oracle Database Manager Database Control",
        "EngineName": "oracle-ee",
        "MajorEngineVersion": "11.2",
        "Options": [],
        "AllowsVpcAndNonVpcInstanceMemberships": true,
        "OptionGroupArn": "arn:aws:rds:us-west-2:123456789012:og:myoptiongroup"
    }
}
```

- Per i dettagli sull'API, consulta CreateOptionGroup AWS CLICommand Reference.

`delete-blue-green-deployment`

Il seguente esempio di codice mostra come utilizzare`delete-blue-green-deployment`.

AWS CLI

Esempio 1: come eliminare le risorse in ambiente verde per un'istanza database RDS per MySQL

L'esempio `delete-blue-green-deployment` seguente elimina le risorse in un ambiente verde per un'istanza database RDS per MySQL.

```
aws rds delete-blue-green-deployment \
    --blue-green-deployment-identifier bgd-v53303651eexfake \
    --delete-target
```

Output:

```
{
    "BlueGreenDeployment": {
        "BlueGreenDeploymentIdentifier": "bgd-v53303651eexfake",
        "BlueGreenDeploymentName": "bgd-cli-test-instance",
        "Source": "arn:aws:rds:us-east-1:123456789012:db:my-db-instance",
        "Target": "arn:aws:rds:us-east-1:123456789012:db:my-db-instance-green-
rkfbpe",
        "SwitchoverDetails": [
            {
                "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-
instance",
                "TargetMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-
instance-green-rkfbpe",
                "Status": "AVAILABLE"
            },
            {
                "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-
instance-replica-1",
                "TargetMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-
instance-replica-1-green-j382ha",
                "Status": "AVAILABLE"
            },
            {
```

```
                "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-
instance-replica-2",
                "TargetMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-
instance-replica-2-green-ejv4ao",
                "Status": "AVAILABLE"
            },
            {
                "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-
instance-replica-3",
                "TargetMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-
instance-replica-3-green-vlpz3t",
                "Status": "AVAILABLE"
            }
        ],
        "Tasks": [
            {
                "Name": "CREATING_READ_REPLICA_OF_SOURCE",
                "Status": "COMPLETED"
            },
            {
                "Name": "DB_ENGINE_VERSION_UPGRADE",
                "Status": "COMPLETED"
            },
            {
                "Name": "CONFIGURE_BACKUPS",
                "Status": "COMPLETED"
            },
            {
                "Name": "CREATING_TOPOLOGY_OF_SOURCE",
                "Status": "COMPLETED"
            }
        ],
        "Status": "DELETING",
        "CreateTime": "2022-02-25T21:18:51.183000+00:00",
        "DeleteTime": "2022-02-25T22:25:31.331000+00:00"
    }
}
```

Per ulteriori informazioni, consulta Eliminazione di una blue/green distribuzione nella Amazon RDS User Guide.

Esempio 2: come eliminare le risorse in ambiente verde per un cluster di database Aurora MySQL

L'esempio `delete-blue-green-deployment` seguente elimina le risorse in un ambiente verde per un cluster di database Aurora MySQL.

```
aws rds delete-blue-green-deployment \
    --blue-green-deployment-identifier bgd-wi89nwzglccsfake \
    --delete-target
```

Output:

```
{
    "BlueGreenDeployment": {
        "BlueGreenDeploymentIdentifier": "bgd-wi89nwzglccsfake",
        "BlueGreenDeploymentName": "my-blue-green-deployment",
        "Source": "arn:aws:rds:us-east-1:123456789012:cluster:my-aurora-mysql-
cluster",
        "Target": "arn:aws:rds:us-east-1:123456789012:cluster:my-aurora-mysql-
cluster-green-3rnukl",
        "SwitchoverDetails": [
            {
                "SourceMember": "arn:aws:rds:us-east-1:123456789012:cluster:my-
aurora-mysql-cluster",
                "TargetMember": "arn:aws:rds:us-east-1:123456789012:cluster:my-
aurora-mysql-cluster-green-3rnukl",
                "Status": "AVAILABLE"
            },
            {
                "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-aurora-
mysql-cluster-1",
                "TargetMember": "arn:aws:rds:us-east-1:123456789012:db:my-aurora-
mysql-cluster-1-green-gpmaxf",
                "Status": "AVAILABLE"
            },
            {
                "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-aurora-
mysql-cluster-2",
                "TargetMember": "arn:aws:rds:us-east-1:123456789012:db:my-aurora-
mysql-cluster-2-green-j2oajq",
                "Status": "AVAILABLE"
            },
            {
                "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-aurora-
mysql-cluster-3",
```

```
                "TargetMember": "arn:aws:rds:us-east-1:123456789012:db:my-aurora-
mysql-cluster-3-green-mkxies",
                "Status": "AVAILABLE"
            },
            {
                "SourceMember": "arn:aws:rds:us-east-1:123456789012:cluster-
endpoint:my-excluded-member-endpoint",
                "TargetMember": "arn:aws:rds:us-east-1:123456789012:cluster-
endpoint:my-excluded-member-endpoint-green-4sqjrq",
                "Status": "AVAILABLE"
            },
            {
                "SourceMember": "arn:aws:rds:us-east-1:123456789012:cluster-
endpoint:my-reader-endpoint",
                "TargetMember": "arn:aws:rds:us-east-1:123456789012:cluster-
endpoint:my-reader-endpoint-green-gwwzlg",
                "Status": "AVAILABLE"
            }
        ],
        "Tasks": [
            {
                "Name": "CREATING_READ_REPLICA_OF_SOURCE",
                "Status": "COMPLETED"
            },
            {
                "Name": "DB_ENGINE_VERSION_UPGRADE",
                "Status": "COMPLETED"
            },
            {
                "Name": "CREATE_DB_INSTANCES_FOR_CLUSTER",
                "Status": "COMPLETED"
            },
            {
                "Name": "CREATE_CUSTOM_ENDPOINTS",
                "Status": "COMPLETED"
            }
        ],
        "Status": "DELETING",
        "CreateTime": "2022-02-25T21:12:00.288000+00:00",
        "DeleteTime": "2022-02-25T22:29:11.336000+00:00"
    }
}
```

Per ulteriori informazioni, consulta Eliminazione di una blue/green distribuzione nella Guida per l'utente di Amazon Aurora.

- Per i dettagli sull'API, consulta AWS CLI Command DeleteBlueGreenDeploymentReference.

`delete-db-cluster-endpoint`

Il seguente esempio di codice mostra come utilizzare`delete-db-cluster-endpoint`.

AWS CLI

Come eliminare un endpoint del cluster di database personalizzato

L'esempio `delete-db-cluster-endpoint` seguente elimina l'endpoint del cluster di database personalizzato specificato.

```
aws rds delete-db-cluster-endpoint \
    --db-cluster-endpoint-identifier mycustomendpoint
```

Output:

```
{
    "DBClusterEndpointIdentifier": "mycustomendpoint",
    "DBClusterIdentifier": "mydbcluster",
    "DBClusterEndpointResourceIdentifier": "cluster-endpoint-ANPAJ4AE5446DAEXAMPLE",
    "Endpoint": "mycustomendpoint.cluster-custom-cnpexample.us-
east-1.rds.amazonaws.com",
    "Status": "deleting",
    "EndpointType": "CUSTOM",
    "CustomEndpointType": "READER",
    "StaticMembers": [
        "dbinstance1",
        "dbinstance2",
        "dbinstance3"
    ],
    "ExcludedMembers": [],
    "DBClusterEndpointArn": "arn:aws:rds:us-east-1:123456789012:cluster-
endpoint:mycustomendpoint"
}
```

Per ulteriori informazioni, consulta Gestione delle connessioni Amazon Aurora nella Guida per l'utente di Amazon Aurora.

- Per i dettagli sull'API, consulta DeleteDbClusterEndpoint AWS CLICommand Reference.

`delete-db-cluster-parameter-group`

Il seguente esempio di codice mostra come utilizzare`delete-db-cluster-parameter-group`.

AWS CLI

Come eliminare un gruppo di parametri del cluster di database

L'esempio `delete-db-cluster-parameter-group` seguente elimina il gruppo di parametri del cluster di database specificato.

```
aws rds delete-db-cluster-parameter-group \
    --db-cluster-parameter-group-name mydbclusterparametergroup
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Uso di gruppi di parametri database e gruppi di parametri cluster database nella Guida per l'utente di Amazon Aurora.

- Per i dettagli sull'API, consulta DeleteDbClusterParameterGroup AWS CLICommand Reference.

`delete-db-cluster-snapshot`

Il seguente esempio di codice mostra come utilizzare`delete-db-cluster-snapshot`.

AWS CLI

Come eliminare uno snapshot del cluster di database

L'esempio `delete-db-cluster-snapshot` seguente elimina lo snapshot del cluster di database specificato.

```
aws rds delete-db-cluster-snapshot \
    --db-cluster-snapshot-identifier mydbclustersnapshot
```

Output:

```
{
```

```
    "DBClusterSnapshot": {
        "AvailabilityZones": [
            "us-east-1a",
            "us-east-1b",
            "us-east-1e"
        ],
        "DBClusterSnapshotIdentifier": "mydbclustersnapshot",
        "DBClusterIdentifier": "mydbcluster",
        "SnapshotCreateTime": "2019-06-18T21:21:00.469Z",
        "Engine": "aurora-mysql",
        "AllocatedStorage": 0,
        "Status": "available",
        "Port": 0,
        "VpcId": "vpc-6594f31c",
        "ClusterCreateTime": "2019-04-15T14:18:42.785Z",
        "MasterUsername": "myadmin",
        "EngineVersion": "5.7.mysql_aurora.2.04.2",
        "LicenseModel": "aurora-mysql",
        "SnapshotType": "manual",
        "PercentProgress": 100,
        "StorageEncrypted": true,
        "KmsKeyId": "arn:aws:kms:us-east-1:123456789012:key/AKIAIOSFODNN7EXAMPLE",
        "DBClusterSnapshotArn": "arn:aws:rds:us-east-1:123456789012:cluster-
snapshot:mydbclustersnapshot",
        "IAMDatabaseAuthenticationEnabled": false
    }
}
```

Per ulteriori informazioni, consulta Eliminazione di uno snapshot nella Guida per l'utente di Amazon Aurora.

- Per i dettagli sull'API, consulta DeleteDbClusterSnapshot AWS CLICommand Reference.

`delete-db-cluster`

Il seguente esempio di codice mostra come utilizzare`delete-db-cluster`.

AWS CLI

Esempio 1: come eliminare un'istanza database in un cluster di database

L'esempio `delete-db-instance` seguente elimina l'istanza database finale in un cluster di database. Non è possibile eliminare un cluster di database se questo contiene istanze database

che non sono in stato di eliminazione. Non è possibile scattare uno snapshot finale quando si elimina un'istanza database in un cluster di database.

```
aws rds delete-db-instance \
    --db-instance-identifier database-3
```

Output:

```
{
    "DBInstance": {
        "DBInstanceIdentifier": "database-3",
        "DBInstanceClass": "db.r4.large",
        "Engine": "aurora-postgresql",
        "DBInstanceStatus": "deleting",

    ...output omitted...

    }
}
```

Per ulteriori informazioni, consulta Eliminazione di un'istanza database da un cluster database Aurora nella Guida per l'utente di Amazon Aurora.

Esempio 2: come eliminare un cluster di database

L'esempio `delete-db-cluster` seguente elimina il cluster di database denominato `mycluster` e crea uno snapshot finale denominato `mycluster-final-snapshot`. Lo stato del cluster di database è disponibile durante l'acquisizione dello snapshot. Per seguire lo stato di avanzamento dell'eliminazione, utilizza il comando dell'`describe-db-clusters` CLI.

```
aws rds delete-db-cluster \
    --db-cluster-identifier mycluster \
    --no-skip-final-snapshot \
    --final-db-snapshot-identifier mycluster-final-snapshot
```

Output:

```
{
    "DBCluster": {
        "AllocatedStorage": 20,
        "AvailabilityZones": [
```

```
            "eu-central-1b",
            "eu-central-1c",
            "eu-central-1a"
        ],
        "BackupRetentionPeriod": 7,
        "DBClusterIdentifier": "mycluster",
        "DBClusterParameterGroup": "default.aurora-postgresql10",
        "DBSubnetGroup": "default-vpc-aa11bb22",
        "Status": "available",

    ...output omitted...

    }
}
```

Per ulteriori informazioni, consulta Cluster Aurora con un'istanza database singola nella Guida per l'utente di Amazon Aurora.

- Per i dettagli sull'API, consulta DeleteDbCluster AWS CLICommand Reference.

`delete-db-instance-automated-backup`

Il seguente esempio di codice mostra come utilizzare`delete-db-instance-automated-backup`.

AWS CLI

Come eliminare un backup automatico replicato da una Regione

L'esempio `delete-db-instance-automated-backup` seguente elimina il backup automatico con il nome della risorsa Amazon (ARN) specificato.

```
aws rds delete-db-instance-automated-backup \
    --db-instance-automated-backups-arn "arn:aws:rds:us-west-2:123456789012:auto-backup:ab-jkib2gfq5rv7replzadausbrktni2bn4example"
```

Output:

```
{
    "DBInstanceAutomatedBackup": {
        "DBInstanceArn": "arn:aws:rds:us-east-1:123456789012:db:new-orcl-db",
        "DbiResourceId": "db-JKIB2GFQ5RV7REPLZA4EXAMPLE",
        "Region": "us-east-1",
```

```
        "DBInstanceIdentifier": "new-orcl-db",
        "RestoreWindow": {},
        "AllocatedStorage": 20,
        "Status": "deleting",
        "Port": 1521,
        "AvailabilityZone": "us-east-1b",
        "VpcId": "vpc-########",
        "InstanceCreateTime": "2020-12-04T15:28:31Z",
        "MasterUsername": "admin",
        "Engine": "oracle-se2",
        "EngineVersion": "12.1.0.2.v21",
        "LicenseModel": "bring-your-own-license",
        "OptionGroupName": "default:oracle-se2-12-1",
        "Encrypted": false,
        "StorageType": "gp2",
        "IAMDatabaseAuthenticationEnabled": false,
        "BackupRetentionPeriod": 7,
        "DBInstanceAutomatedBackupsArn": "arn:aws:rds:us-west-2:123456789012:auto-
backup:ab-jkib2gfq5rv7replzadausbrktni2bn4example"
    }
}
```

Per ulteriori informazioni, consulta Eliminazione dei backup replicati nella Guida per l'utente di Amazon RDS.

- Per i dettagli sull'API, consulta DeleteDbInstanceAutomatedBackup AWS CLICommand Reference.

`delete-db-instance`

Il seguente esempio di codice mostra come utilizzare`delete-db-instance`.

AWS CLI

Come eliminare un'istanza database

L'esempio `delete-db-instance` seguente elimina l'istanza database specificata dopo aver creato uno snapshot database finale denominato `test-instance-final-snap`.

```
aws rds delete-db-instance \
    --db-instance-identifier test-instance \
    --final-db-snapshot-identifier test-instance-final-snap
```

Output:

```
{
    "DBInstance": {
        "DBInstanceIdentifier": "test-instance",
        "DBInstanceStatus": "deleting",
        ...some output truncated...
    }
}
```

- Per i dettagli sull'API, consulta Delete DBInstance in AWS CLI Command Reference.

`delete-db-parameter-group`

Il seguente esempio di codice mostra come utilizzare`delete-db-parameter-group`.

AWS CLI

Come eliminare un gruppo di parametri del database

L'esempio `command` seguente elimina un gruppo di parametri database.

```
aws rds delete-db-parameter-group \
    --db-parameter-group-name mydbparametergroup
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Utilizzo di gruppi di parametri di database nella Guida per l'utente di Amazon RDS.

- Per i dettagli sull'API, consulta Delete DBParameter Group in AWS CLI Command Reference.

`delete-db-proxy-endpoint`

Il seguente esempio di codice mostra come utilizzare`delete-db-proxy-endpoint`.

AWS CLI

Come eliminare un endpoint proxy database per un database RDS

L'esempio `delete-db-proxy-endpoint` seguente elimina un endpoint proxy database per il database di destinazione.

```
aws rds delete-db-proxy-endpoint \
    --db-proxy-endpoint-name proxyEP1
```

Output:

```
{
"DBProxyEndpoint":
    {
        "DBProxyEndpointName": "proxyEP1",
        "DBProxyEndpointArn": "arn:aws:rds:us-east-1:123456789012:db-proxy-
endpoint:prx-endpoint-0123a01b12345c0ab",
        "DBProxyName": "proxyExample",
        "Status": "deleting",
        "VpcId": "vpc-1234567",
        "VpcSecurityGroupIds": [
            "sg-1234",
            "sg-5678"
        ],
        "VpcSubnetIds": [
            "subnetgroup1",
            "subnetgroup2"
        ],
        "Endpoint": "proxyEP1.endpoint.proxy-ab0cd1efghij.us-
east-1.rds.amazonaws.com",
        "CreatedDate": "2023-04-13T01:49:38.568000+00:00",
        "TargetRole": "READ_ONLY",
        "IsDefault": false
    }
}
```

Per ulteriori informazioni, consulta Eliminazione di un endpoint proxy nella Guida per l'utente di Amazon RDS ed Eliminazione di un endpoint proxy nella Guida per l'utente di Amazon Aurora.

- Per i dettagli sull'API, consulta DeleteDbProxyEndpoint AWS CLICommand Reference.

`delete-db-proxy`

Il seguente esempio di codice mostra come utilizzare`delete-db-proxy`.

AWS CLI

Come eliminare un proxy database per un database RDS

L'esempio `delete-db-proxy` seguente elimina un proxy database.

```
aws rds delete-db-proxy \
    --db-proxy-name proxyExample
```

Output:

```
{
        "DBProxy":
        {
            "DBProxyName": "proxyExample",
            "DBProxyArn": "arn:aws:rds:us-east-1:123456789012:db-
proxy:prx-0123a01b12345c0ab",
            "Status": "deleting",
            "EngineFamily": "PostgreSQL",
            "VpcId": "vpc-1234567",
            "VpcSecurityGroupIds": [
                "sg-1234",
                "sg-5678"
            ],
            "VpcSubnetIds": [
                "subnetgroup1",
                "subnetgroup2"
            ],
            "Auth": "[
                {
                "Description": "proxydescription`"
                "AuthScheme": "SECRETS",
                "SecretArn": "arn:aws:secretsmanager:us-
west-2:123456789123:secret:proxysecret1-Abcd1e",
                "IAMAuth": "DISABLED"
                } ],
            "RoleArn": "arn:aws:iam::12345678912:role/ProxyPostgreSQLRole",
            "Endpoint": "proxyExample.proxy-ab0cd1efghij.us-
east-1.rds.amazonaws.com",
            "RequireTLS": false,
            "IdleClientTimeout": 1800,
            "DebuggingLogging": false,
        "CreatedDate": "2023-04-05T16:09:33.452000+00:00",
        "UpdatedDate": "2023-04-13T01:49:38.568000+00:00"
    }
}
```

Per ulteriori informazioni, consulta Eliminazione di un proxy RDS nella Guida per l'utente di Amazon RDS e Eliminazione di un proxy RDS nella Guida per l'utente di Amazon Aurora.

- Per i dettagli sull'API, consulta DeleteDbProxy AWS CLICommand Reference.

`delete-db-security-group`

Il seguente esempio di codice mostra come utilizzare`delete-db-security-group`.

AWS CLI

Come eliminare un gruppo di sicurezza del database

L'esempio `delete-db-security-group` seguente elimina un gruppo di sicurezza database denominato `mysecuritygroup`.

```
aws rds delete-db-security-group \
    --db-security-group-name mysecuritygroup
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Working with DB security groups (piattaforma EC2 -Classic) nella Amazon RDS User Guide.

- Per i dettagli sull'API, consulta AWS CLI Command DeleteDbSecurityGroupReference.

`delete-db-shard-group`

Il seguente esempio di codice mostra come utilizzare`delete-db-shard-group`.

AWS CLI

Esempio 1: come eliminare in modo non corretto un gruppo di shard del database

L'esempio `delete-db-shard-group` seguente mostra l'errore che si verifica quando si tenta di eliminare un gruppo di shard del database prima di eliminare tutti i database e gli schemi.

```
aws rds delete-db-shard-group \
    --db-shard-group-identifier limitless-test-shard-grp
```

Output:

```
An error occurred (InvalidDBShardGroupState) when calling the DeleteDBShardGroup
 operation: Unable to delete the DB shard group limitless-test-db-shard-group.
Delete all of your Limitless Database databases and schemas, then try again.
```

Esempio 2: come eliminare in modo corretto un gruppo di shard del database

L'esempio `delete-db-shard-group` seguente elimina un gruppo di shard del database dopo aver eliminato tutti i database e gli schemi, incluso lo schema `public`.

```
aws rds delete-db-shard-group \
    --db-shard-group-identifier limitless-test-shard-grp
```

Output:

```
{
    "DBShardGroupResourceId": "shardgroup-7bb446329da94788b3f957746example",
    "DBShardGroupIdentifier": "limitless-test-shard-grp",
    "DBClusterIdentifier": "limitless-test-cluster",
    "MaxACU": 768.0,
    "ComputeRedundancy": 0,
    "Status": "deleting",
    "PubliclyAccessible": true,
    "Endpoint": "limitless-test-cluster.limitless-cekycexample.us-
east-2.rds.amazonaws.com"
}
```

Per ulteriori informazioni, consulta Eliminazione di cluster e istanze database di Aurora nella Guida per l'utente di Amazon Aurora.

- Per i dettagli sull'API, consulta DeleteDbShardGroup AWS CLICommand Reference.

`delete-db-snapshot`

Il seguente esempio di codice mostra come utilizzare`delete-db-snapshot`.

AWS CLI

Come eliminare uno snapshot del database

L'esempio `delete-db-snapshot` seguente elimina lo snapshot database specificato.

```
aws rds delete-db-snapshot \
    --db-snapshot-identifier mydbsnapshot
```

Output:

```
{
    "DBSnapshot": {
        "DBSnapshotIdentifier": "mydbsnapshot",
        "DBInstanceIdentifier": "database-mysql",
        "SnapshotCreateTime": "2019-06-18T22:08:40.702Z",
        "Engine": "mysql",
        "AllocatedStorage": 100,
        "Status": "deleted",
        "Port": 3306,
        "AvailabilityZone": "us-east-1b",
        "VpcId": "vpc-6594f31c",
        "InstanceCreateTime": "2019-04-30T15:45:53.663Z",
        "MasterUsername": "admin",
        "EngineVersion": "5.6.40",
        "LicenseModel": "general-public-license",
        "SnapshotType": "manual",
        "Iops": 1000,
        "OptionGroupName": "default:mysql-5-6",
        "PercentProgress": 100,
        "StorageType": "io1",
        "Encrypted": true,
        "KmsKeyId": "arn:aws:kms:us-east-1:123456789012:key/AKIAIOSFODNN7EXAMPLE",
        "DBSnapshotArn": "arn:aws:rds:us-east-1:123456789012:snapshot:mydbsnapshot",
        "IAMDatabaseAuthenticationEnabled": false,
        "ProcessorFeatures": [],
        "DbiResourceId": "db-AKIAIOSFODNN7EXAMPLE"
    }
}
```

Per ulteriori informazioni, consulta Eliminazione di uno snapshot nella Guida per l'utente di Amazon RDS.

- Per i dettagli sull'API, consulta DeleteDbSnapshot AWS CLICommand Reference.

`delete-db-subnet-group`

Il seguente esempio di codice mostra come utilizzare`delete-db-subnet-group`.

AWS CLI

Come eliminare un gruppo di sottoreti di database

L'esempio `delete-db-subnet-group` seguente elimina il gruppo di sottoreti database denominato `mysubnetgroup`.

```
aws rds delete-db-subnet-group --db-subnet-group-name mysubnetgroup
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Uso di un'istanza database in un VPC nella Guida per l'utente di Amazon RDS.

- Per i dettagli sull'API, consulta DeleteDbSubnetGroup AWS CLICommand Reference.

`delete-event-subscription`

Il seguente esempio di codice mostra come utilizzare`delete-event-subscription`.

AWS CLI

Come eliminare un abbonamento all'evento

L'esempio `delete-event-subscription` seguente elimina l'abbonamento all'evento specificato.

```
aws rds delete-event-subscription --subscription-name my-instance-events
```

Output:

```
{
    "EventSubscription": {
        "EventSubscriptionArn": "arn:aws:rds:us-east-1:123456789012:es:my-instance-
events",
        "CustomerAwsId": "123456789012",
        "Enabled": false,
        "SourceIdsList": [
            "test-instance"
        ],
        "SourceType": "db-instance",
        "EventCategoriesList": [
            "backup",
```

```
            "recovery"
        ],
        "SubscriptionCreationTime": "2018-07-31 23:22:01.893",
        "CustSubscriptionId": "my-instance-events",
        "SnsTopicArn": "arn:aws:sns:us-east-1:123456789012:interesting-events",
        "Status": "deleting"
    }
}
```

- Per i dettagli sull'API, consulta DeleteEventSubscription AWS CLICommand Reference.

`delete-global-cluster`

Il seguente esempio di codice mostra come utilizzare`delete-global-cluster`.

AWS CLI

Come eliminare un cluster di database globale

L'esempio `delete-global-cluster` seguente elimina un cluster di database globale compatibile con Aurora MySQL. L'output mostra il cluster che stai eliminando, ma i comandi `describe-global-clusters` successivi non elencano tale cluster di database.

```
aws rds delete-global-cluster \
    --global-cluster-identifier myglobalcluster
```

Output:

```
{
    "GlobalCluster": {
        "GlobalClusterIdentifier": "myglobalcluster",
        "GlobalClusterResourceId": "cluster-f0e523bfe07aabb",
        "GlobalClusterArn": "arn:aws:rds::123456789012:global-
cluster:myglobalcluster",
        "Status": "available",
        "Engine": "aurora-mysql",
        "EngineVersion": "5.7.mysql_aurora.2.07.2",
        "StorageEncrypted": false,
        "DeletionProtection": false,
        "GlobalClusterMembers": []
    }
}
```

Per ulteriori informazioni, consulta Eliminazione di un database globale Aurora nella Guida per l'utente di Amazon Aurora.

- Per i dettagli sull'API, consulta DeleteGlobalCluster AWS CLICommand Reference.

`delete-option-group`

Il seguente esempio di codice mostra come utilizzare`delete-option-group`.

AWS CLI

Come eliminare un gruppo di opzioni

L'esempio `delete-option-group` seguente elimina il gruppo di opzioni specificato.

```
aws rds delete-option-group \
    --option-group-name myoptiongroup
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Eliminazione di un gruppo di utenti IAM nella Guida per l'utente di Amazon RDS.

- Per i dettagli sull'API, consulta DeleteOptionGroup AWS CLICommand Reference.

`deregister-db-proxy-targets`

Il seguente esempio di codice mostra come utilizzare`deregister-db-proxy-targets`.

AWS CLI

Come annullare la registrazione di una destinazione del proxy database dal gruppo di destinazioni del database

L'esempio `deregister-db-proxy-targets` seguente rimuove l'associazione tra il proxy `proxyExample` e la sua destinazione.

```
aws rds deregister-db-proxy-targets \
    --db-proxy-name proxyExample \
    --db-instance-identifiers database-1
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Eliminazione di un proxy RDS nella Guida per l'utente di Amazon RDS e Eliminazione di un proxy RDS nella Guida per l'utente di Amazon Aurora.

- Per i dettagli sull'API, consulta DeregisterDbProxyTargets AWS CLICommand Reference.

`describe-account-attributes`

Il seguente esempio di codice mostra come utilizzare`describe-account-attributes`.

AWS CLI

Come descrivere gli attributi dell'account

L'`describe-account-attributes`esempio seguente recupera gli attributi per l' AWS account corrente.

```
aws rds describe-account-attributes
```

Output:

```
{
    "AccountQuotas": [
        {
            "Max": 40,
            "Used": 4,
            "AccountQuotaName": "DBInstances"
        },
        {
            "Max": 40,
            "Used": 0,
            "AccountQuotaName": "ReservedDBInstances"
        },
        {
            "Max": 100000,
            "Used": 40,
            "AccountQuotaName": "AllocatedStorage"
        },
        {
            "Max": 25,
            "Used": 0,
            "AccountQuotaName": "DBSecurityGroups"
        },
        {
```

```
            "Max": 20,
            "Used": 0,
            "AccountQuotaName": "AuthorizationsPerDBSecurityGroup"
        },
        {
            "Max": 50,
            "Used": 1,
            "AccountQuotaName": "DBParameterGroups"
        },
        {
            "Max": 100,
            "Used": 3,
            "AccountQuotaName": "ManualSnapshots"
        },
        {
            "Max": 20,
            "Used": 0,
            "AccountQuotaName": "EventSubscriptions"
        },
        {
            "Max": 50,
            "Used": 1,
            "AccountQuotaName": "DBSubnetGroups"
        },
        {
            "Max": 20,
            "Used": 1,
            "AccountQuotaName": "OptionGroups"
        },
        {
            "Max": 20,
            "Used": 6,
            "AccountQuotaName": "SubnetsPerDBSubnetGroup"
        },
        {
            "Max": 5,
            "Used": 0,
            "AccountQuotaName": "ReadReplicasPerMaster"
        },
        {
            "Max": 40,
            "Used": 1,
            "AccountQuotaName": "DBClusters"
        },
```

```
        {
            "Max": 50,
            "Used": 0,
            "AccountQuotaName": "DBClusterParameterGroups"
        },
        {
            "Max": 5,
            "Used": 0,
            "AccountQuotaName": "DBClusterRoles"
        }
    ]
}
```

- Per i dettagli sull'API, vedere DescribeAccountAttributesin AWS CLI Command Reference.

`describe-blue-green-deployments`

Il seguente esempio di codice mostra come utilizzare`describe-blue-green-deployments`.

AWS CLI

Esempio 1: descrivere una blue/green distribuzione di un'istanza DB RDS dopo il completamento della creazione

L'`describe-blue-green-deployment`esempio seguente recupera i dettagli di una blue/green distribuzione dopo il completamento della creazione.

```
aws rds describe-blue-green-deployments \
    --blue-green-deployment-identifier bgd-v53303651eexfake
```

Output:

```
{
    "BlueGreenDeployments": [
        {
            "BlueGreenDeploymentIdentifier": "bgd-v53303651eexfake",
            "BlueGreenDeploymentName": "bgd-cli-test-instance",
            "Source": "arn:aws:rds:us-east-1:123456789012:db:my-db-instance",
            "Target": "arn:aws:rds:us-east-1:123456789012:db:my-db-instance-green-
rkfbpe",
            "SwitchoverDetails": [
                {
```

```
                    "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-
instance",
                    "TargetMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-
instance-green-rkfbpe",
                    "Status": "AVAILABLE"
                },
                {
                    "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-
instance-replica-1",
                    "TargetMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-
instance-replica-1-green-j382ha",
                    "Status": "AVAILABLE"
                },
                {
                    "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-
instance-replica-2",
                    "TargetMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-
instance-replica-2-green-ejv4ao",
                    "Status": "AVAILABLE"
                },
                {
                    "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-
instance-replica-3",
                    "TargetMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-
instance-replica-3-green-vlpz3t",
                    "Status": "AVAILABLE"
                }
            ],
            "Tasks": [
                {
                    "Name": "CREATING_READ_REPLICA_OF_SOURCE",
                    "Status": "COMPLETED"
                },
                {
                    "Name": "DB_ENGINE_VERSION_UPGRADE",
                    "Status": "COMPLETED"
                },
                {
                    "Name": "CONFIGURE_BACKUPS",
                    "Status": "COMPLETED"
                },
                {
                    "Name": "CREATING_TOPOLOGY_OF_SOURCE",
                    "Status": "COMPLETED"
```

```
                }
            ],
            "Status": "AVAILABLE",
            "CreateTime": "2022-02-25T21:18:51.183000+00:00"
        }
    ]
}
```

Per ulteriori informazioni, consulta Visualizzazione di una blue/green distribuzione nella Amazon RDS User Guide.

Esempio 2: Per descrivere una blue/green distribuzione per un cluster Aurora MySQL DB

L'`describe-blue-green-deployment`esempio seguente recupera i dettagli di una distribuzione. blue/green

```
aws rds describe-blue-green-deployments \
    --blue-green-deployment-identifier bgd-wi89nwzglccsfake
```

Output:

```
{
    "BlueGreenDeployments": [
        {
            "BlueGreenDeploymentIdentifier": "bgd-wi89nwzglccsfake",
            "BlueGreenDeploymentName": "my-blue-green-deployment",
            "Source": "arn:aws:rds:us-east-1:123456789012:cluster:my-aurora-mysql-
cluster",
            "Target": "arn:aws:rds:us-east-1:123456789012:cluster:my-aurora-mysql-
cluster-green-3rnukl",
            "SwitchoverDetails": [
                {
                    "SourceMember": "arn:aws:rds:us-east-1:123456789012:cluster:my-
aurora-mysql-cluster",
                    "TargetMember": "arn:aws:rds:us-east-1:123456789012:cluster:my-
aurora-mysql-cluster-green-3rnukl",
                    "Status": "AVAILABLE"
                },
                {
                    "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-
aurora-mysql-cluster-1",
                    "TargetMember": "arn:aws:rds:us-east-1:123456789012:db:my-
aurora-mysql-cluster-1-green-gpmaxf",
```

```
                    "Status": "AVAILABLE"
                },
                {
                    "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-
aurora-mysql-cluster-2",
                    "TargetMember": "arn:aws:rds:us-east-1:123456789012:db:my-
aurora-mysql-cluster-2-green-j2oajq",
                    "Status": "AVAILABLE"
                },
                {
                    "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-
aurora-mysql-cluster-3",
                    "TargetMember": "arn:aws:rds:us-east-1:123456789012:db:my-
aurora-mysql-cluster-3-green-mkxies",
                    "Status": "AVAILABLE"
                },
                {
                    "SourceMember": "arn:aws:rds:us-east-1:123456789012:cluster-
endpoint:my-excluded-member-endpoint",
                    "TargetMember": "arn:aws:rds:us-east-1:123456789012:cluster-
endpoint:my-excluded-member-endpoint-green-4sqjrq",
                    "Status": "AVAILABLE"
                },
                {
                    "SourceMember": "arn:aws:rds:us-east-1:123456789012:cluster-
endpoint:my-reader-endpoint",
                    "TargetMember": "arn:aws:rds:us-east-1:123456789012:cluster-
endpoint:my-reader-endpoint-green-gwwzlg",
                    "Status": "AVAILABLE"
                }
            ],
            "Tasks": [
                {
                    "Name": "CREATING_READ_REPLICA_OF_SOURCE",
                    "Status": "COMPLETED"
                },
                {
                    "Name": "DB_ENGINE_VERSION_UPGRADE",
                    "Status": "COMPLETED"
                },
                {
                    "Name": "CREATE_DB_INSTANCES_FOR_CLUSTER",
                    "Status": "COMPLETED"
                },
```

```
                {
                    "Name": "CREATE_CUSTOM_ENDPOINTS",
                    "Status": "COMPLETED"
                }
            ],
            "Status": "AVAILABLE",
            "CreateTime": "2022-02-25T21:12:00.288000+00:00"
        }
    ]
}
```

Per ulteriori informazioni, consulta Visualizzazione di una blue/green distribuzione nella Guida per l'utente di Amazon Aurora.

Esempio 3: Per descrivere una blue/green distribuzione per un cluster Aurora MySQL dopo lo switchover

L'`describe-blue-green-deployment`esempio seguente recupera i dettagli su una blue/green distribuzione dopo che l'ambiente verde è stato promosso a ambiente di produzione.

```
aws rds describe-blue-green-deployments \
    --blue-green-deployment-identifier bgd-wi89nwzglccsfake
```

Output:

```
{
    "BlueGreenDeployments": [
        {
            "BlueGreenDeploymentIdentifier": "bgd-wi89nwzglccsfake",
            "BlueGreenDeploymentName": "my-blue-green-deployment",
            "Source": "arn:aws:rds:us-east-1:123456789012:cluster:my-aurora-mysql-
cluster-old1",
            "Target": "arn:aws:rds:us-east-1:123456789012:cluster:my-aurora-mysql-
cluster",
            "SwitchoverDetails": [
                {
                    "SourceMember": "arn:aws:rds:us-east-1:123456789012:cluster:my-
aurora-mysql-cluster-old1",
                    "TargetMember": "arn:aws:rds:us-east-1:123456789012:cluster:my-
aurora-mysql-cluster",
                    "Status": "SWITCHOVER_COMPLETED"
                },
                {
```

```
                    "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-
aurora-mysql-cluster-1-old1",
                    "TargetMember": "arn:aws:rds:us-east-1:123456789012:db:my-
aurora-mysql-cluster-1",
                    "Status": "SWITCHOVER_COMPLETED"
                },
                {
                    "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-
aurora-mysql-cluster-2-old1",
                    "TargetMember": "arn:aws:rds:us-east-1:123456789012:db:my-
aurora-mysql-cluster-2",
                    "Status": "SWITCHOVER_COMPLETED"
                },
                {
                    "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-
aurora-mysql-cluster-3-old1",
                    "TargetMember": "arn:aws:rds:us-east-1:123456789012:db:my-
aurora-mysql-cluster-3",
                    "Status": "SWITCHOVER_COMPLETED"
                },
                {
                    "SourceMember": "arn:aws:rds:us-east-1:123456789012:cluster-
endpoint:my-excluded-member-endpoint-old1",
                    "TargetMember": "arn:aws:rds:us-east-1:123456789012:cluster-
endpoint:my-excluded-member-endpoint",
                    "Status": "SWITCHOVER_COMPLETED"
                },
                {
                    "SourceMember": "arn:aws:rds:us-east-1:123456789012:cluster-
endpoint:my-reader-endpoint-old1",
                    "TargetMember": "arn:aws:rds:us-east-1:123456789012:cluster-
endpoint:my-reader-endpoint",
                    "Status": "SWITCHOVER_COMPLETED"
                }
            ],
            "Tasks": [
                {
                    "Name": "CREATING_READ_REPLICA_OF_SOURCE",
                    "Status": "COMPLETED"
                },
                {
                    "Name": "DB_ENGINE_VERSION_UPGRADE",
                    "Status": "COMPLETED"
                },
```

```
                {
                    "Name": "CREATE_DB_INSTANCES_FOR_CLUSTER",
                    "Status": "COMPLETED"
                },
                {
                    "Name": "CREATE_CUSTOM_ENDPOINTS",
                    "Status": "COMPLETED"
                }
            ],
            "Status": "SWITCHOVER_COMPLETED",
            "CreateTime": "2022-02-25T22:38:49.522000+00:00"
        }
    ]
}
```

Per ulteriori informazioni, consulta Visualizzazione di una blue/green distribuzione nella Guida per l'utente di Amazon Aurora.

Esempio 4: Per descrivere una distribuzione combinata blue/green

L'`describe-blue-green-deployment`esempio seguente recupera i dettagli di una blue/green distribuzione combinata.

```
aws rds describe-blue-green-deployments
```

Output:

```
{
    "BlueGreenDeployments": [
        {
            "BlueGreenDeploymentIdentifier": "bgd-wi89nwzgfakelccs",
            "BlueGreenDeploymentName": "my-blue-green-deployment",
            "Source": "arn:aws:rds:us-east-1:123456789012:cluster:my-aurora-mysql-
cluster",
            "Target": "arn:aws:rds:us-east-1:123456789012:cluster:my-aurora-mysql-
cluster-green-3rnukl",
            "SwitchoverDetails": [
                {
                    "SourceMember": "arn:aws:rds:us-east-1:123456789012:cluster:my-
aurora-mysql-cluster",
                    "TargetMember": "arn:aws:rds:us-east-1:123456789012:cluster:my-
aurora-mysql-cluster-green-3rnukl",
                    "Status": "AVAILABLE"
```

```
                },
                {
                    "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-
aurora-mysql-cluster-1",
                    "TargetMember": "arn:aws:rds:us-east-1:123456789012:db:my-
aurora-mysql-cluster-1-green-gpmaxf",
                    "Status": "AVAILABLE"
                },
                {
                    "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-
aurora-mysql-cluster-2",
                    "TargetMember": "arn:aws:rds:us-east-1:123456789012:db:my-
aurora-mysql-cluster-2-green-j2oajq",
                    "Status": "AVAILABLE"
                },
                {
                    "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-
aurora-mysql-cluster-3",
                    "TargetMember": "arn:aws:rds:us-east-1:123456789012:db:my-
aurora-mysql-cluster-3-green-mkxies",
                    "Status": "AVAILABLE"
                },
                {
                    "SourceMember": "arn:aws:rds:us-east-1:123456789012:cluster-
endpoint:my-excluded-member-endpoint",
                    "TargetMember": "arn:aws:rds:us-east-1:123456789012:cluster-
endpoint:my-excluded-member-endpoint-green-4sqjrq",
                    "Status": "AVAILABLE"
                },
                {
                    "SourceMember": "arn:aws:rds:us-east-1:123456789012:cluster-
endpoint:my-reader-endpoint",
                    "TargetMember": "arn:aws:rds:us-east-1:123456789012:cluster-
endpoint:my-reader-endpoint-green-gwwzlg",
                    "Status": "AVAILABLE"
                }
            ],
            "Tasks": [
                {
                    "Name": "CREATING_READ_REPLICA_OF_SOURCE",
                    "Status": "COMPLETED"
                },
                {
                    "Name": "DB_ENGINE_VERSION_UPGRADE",
```

```
                    "Status": "COMPLETED"
                },
                {
                    "Name": "CREATE_DB_INSTANCES_FOR_CLUSTER",
                    "Status": "COMPLETED"
                },
                {
                    "Name": "CREATE_CUSTOM_ENDPOINTS",
                    "Status": "COMPLETED"
                }
            ],
            "Status": "AVAILABLE",
            "CreateTime": "2022-02-25T21:12:00.288000+00:00"
        },
        {
            "BlueGreenDeploymentIdentifier": "bgd-v5330365fake1eex",
            "BlueGreenDeploymentName": "bgd-cli-test-instance",
            "Source": "arn:aws:rds:us-east-1:123456789012:db:my-db-instance-old1",
            "Target": "arn:aws:rds:us-east-1:123456789012:db:my-db-instance",
            "SwitchoverDetails": [
                {
                    "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-
instance-old1",
                    "TargetMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-
instance",
                    "Status": "SWITCHOVER_COMPLETED"
                },
                {
                    "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-
instance-replica-1-old1",
                    "TargetMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-
instance-replica-1",
                    "Status": "SWITCHOVER_COMPLETED"
                },
                {
                    "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-
instance-replica-2-old1",
                    "TargetMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-
instance-replica-2",
                    "Status": "SWITCHOVER_COMPLETED"
                },
                {
                    "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-
instance-replica-3-old1",
```

```
                    "TargetMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-
instance-replica-3",
                    "Status": "SWITCHOVER_COMPLETED"
                }
            ],
            "Tasks": [
                {
                    "Name": "CREATING_READ_REPLICA_OF_SOURCE",
                    "Status": "COMPLETED"
                },
                {
                    "Name": "DB_ENGINE_VERSION_UPGRADE",
                    "Status": "COMPLETED"
                },
                {
                    "Name": "CONFIGURE_BACKUPS",
                    "Status": "COMPLETED"
                },
                {
                    "Name": "CREATING_TOPOLOGY_OF_SOURCE",
                    "Status": "COMPLETED"
                }
            ],
            "Status": "SWITCHOVER_COMPLETED",
            "CreateTime": "2022-02-25T22:33:22.225000+00:00"
        }
    ]
}
```

Per ulteriori informazioni, consulta Visualizzazione di una blue/green distribuzione nella Guida per l'utente di Amazon RDS e Visualizzazione di una blue/green distribuzione nella Guida per l'utente di Amazon Aurora.

- Per i dettagli sull'API, consulta AWS CLI Command DescribeBlueGreenDeploymentsReference.

`describe-certificates`

Il seguente esempio di codice mostra come utilizzare`describe-certificates`.

AWS CLI

Come descrivere i certificati

L'esempio `describe-certificates` seguente recupera i dettagli del certificato associato alla Regione predefinita dell'utente.

```
aws rds describe-certificates
```

Output:

```
{
    "Certificates": [
        {
            "CertificateIdentifier": "rds-ca-ecc384-g1",
            "CertificateType": "CA",
            "Thumbprint": "2ee3dcc06e50192559b13929e73484354f23387d",
            "ValidFrom": "2021-05-24T22:06:59+00:00",
            "ValidTill": "2121-05-24T23:06:59+00:00",
            "CertificateArn": "arn:aws:rds:us-west-2::cert:rds-ca-ecc384-g1",
            "CustomerOverride": false
        },
        {
            "CertificateIdentifier": "rds-ca-rsa4096-g1",
            "CertificateType": "CA",
            "Thumbprint": "19da4f2af579a8ae1f6a0fa77aa5befd874b4cab",
            "ValidFrom": "2021-05-24T22:03:20+00:00",
            "ValidTill": "2121-05-24T23:03:20+00:00",
            "CertificateArn": "arn:aws:rds:us-west-2::cert:rds-ca-rsa4096-g1",
            "CustomerOverride": false
        },
        {
            "CertificateIdentifier": "rds-ca-rsa2048-g1",
            "CertificateType": "CA",
            "Thumbprint": "7c40cb42714b6fdb2b296f9bbd0e8bb364436a76",
            "ValidFrom": "2021-05-24T21:59:00+00:00",
            "ValidTill": "2061-05-24T22:59:00+00:00",
            "CertificateArn": "arn:aws:rds:us-west-2::cert:rds-ca-rsa2048-g1",
            "CustomerOverride": true,
            "CustomerOverrideValidTill": "2061-05-24T22:59:00+00:00"
        },
        {
            "CertificateIdentifier": "rds-ca-2019",
            "CertificateType": "CA",
            "Thumbprint": "d40ddb29e3750dffa671c3140bbf5f478d1c8096",
            "ValidFrom": "2019-08-22T17:08:50+00:00",
            "ValidTill": "2024-08-22T17:08:50+00:00",
```

```
            "CertificateArn": "arn:aws:rds:us-west-2::cert:rds-ca-2019",
            "CustomerOverride": false
        }
    ],
    "DefaultCertificateForNewLaunches": "rds-ca-rsa2048-g1"
}
```

Per ulteriori informazioni, consulta Utilizzo per SSL/TLS crittografare una connessione a un'istanza DB nella Guida per l'utente di Amazon RDS e Utilizzo SSL/TLS per crittografare una connessione a un cluster DB nella Guida per l'utente di Amazon Aurora.

- Per i dettagli sull'API, consulta DescribeCertificatesCommand Reference.AWS CLI

`describe-db-cluster-backtracks`

Il seguente esempio di codice mostra come utilizzare`describe-db-cluster-backtracks`.

AWS CLI

Come descrivere i backtrack per un cluster di database

L'esempio `describe-db-cluster-backtracks` seguente recupera i dettagli del cluster di database specificato.

```
aws rds describe-db-cluster-backtracks \
    --db-cluster-identifier mydbcluster
```

Output:

```
{
    "DBClusterBacktracks": [
        {
            "DBClusterIdentifier": "mydbcluster",
            "BacktrackIdentifier": "2f5f5294-0dd2-44c9-9f50-EXAMPLE",
            "BacktrackTo": "2021-02-12T04:59:22Z",
            "BacktrackedFrom": "2021-02-12T14:37:31.640Z",
            "BacktrackRequestCreationTime": "2021-02-12T14:36:18.819Z",
            "Status": "COMPLETED"
        },
        {
            "DBClusterIdentifier": "mydbcluster",
            "BacktrackIdentifier": "3c7a6421-af2a-4ea3-ae95-EXAMPLE",
```

```
            "BacktrackTo": "2021-02-11T22:53:46Z",
            "BacktrackedFrom": "2021-02-12T00:09:27.006Z",
            "BacktrackRequestCreationTime": "2021-02-12T00:07:53.487Z",
            "Status": "COMPLETED"
        }
    ]
}
```

Per ulteriori informazioni, consulta Backtracking di un cluster di database Aurora nella Guida per l'utente di Amazon Aurora.

- Per i dettagli sull'API, consulta DescribeDbClusterBacktracks AWS CLICommand Reference.

`describe-db-cluster-endpoints`

Il seguente esempio di codice mostra come utilizzare`describe-db-cluster-endpoints`.

AWS CLI

Esempio 1: come descrivere gli endpoint del cluster di database

L'esempio `describe-db-cluster-endpoints` seguente recupera i dettagli degli endpoint del cluster di database. I tipi più comuni di cluster Aurora hanno due endpoint. Un endpoint ha il tipo `WRITER`. Puoi utilizzare questo endpoint per tutte le istruzioni SQL. L'altro endpoint ha il tipo `READER`. Puoi utilizzare questo endpoint solo per SELECT e altre istruzioni SQL di sola lettura.

```
aws rds describe-db-cluster-endpoints
```

Output:

```
{
    "DBClusterEndpoints": [
        {
            "DBClusterIdentifier": "my-database-1",
            "Endpoint": "my-database-1.cluster-cnpexample.us-
east-1.rds.amazonaws.com",
            "Status": "creating",
            "EndpointType": "WRITER"
        },
        {
            "DBClusterIdentifier": "my-database-1",
```

```
            "Endpoint": "my-database-1.cluster-ro-cnpexample.us-
east-1.rds.amazonaws.com",
            "Status": "creating",
            "EndpointType": "READER"
        },
        {
            "DBClusterIdentifier": "mydbcluster",
            "Endpoint": "mydbcluster.cluster-cnpexamle.us-east-1.rds.amazonaws.com",
            "Status": "available",
            "EndpointType": "WRITER"
        },
        {
            "DBClusterIdentifier": "mydbcluster",
            "Endpoint": "mydbcluster.cluster-ro-cnpexample.us-
east-1.rds.amazonaws.com",
            "Status": "available",
            "EndpointType": "READER"
        }
    ]
}
```

Esempio 2: come descrivere gli endpoint del cluster di database di un singolo cluster di database

L'esempio `describe-db-cluster-endpoints` seguente recupera i dettagli degli endpoint del cluster di database di un singolo cluster di database specificato. I cluster Aurora Serverless hanno un solo endpoint con un tipo `WRITER`.

```
aws rds describe-db-cluster-endpoints \
    --db-cluster-identifier serverless-cluster
```

Output:

```
{
    "DBClusterEndpoints": [
        {
            "Status": "available",
            "Endpoint": "serverless-cluster.cluster-cnpexample.us-
east-1.rds.amazonaws.com",
            "DBClusterIdentifier": "serverless-cluster",
            "EndpointType": "WRITER"
        }
    ]
```

```
}
```

Per ulteriori informazioni, consulta Gestione delle connessioni Amazon Aurora nella Guida per l'utente di Amazon Aurora.

- Per i dettagli sull'API, consulta DescribeDbClusterEndpoints AWS CLICommand Reference.

`describe-db-cluster-parameter-groups`

Il seguente esempio di codice mostra come utilizzare`describe-db-cluster-parameter-groups`.

AWS CLI

Come descrivere i gruppi di parametri del cluster di database

L'esempio `describe-db-cluster-parameter-groups` seguente recupera i dettagli per i gruppi di parametri del cluster di database.

```
aws rds describe-db-cluster-parameter-groups
```

Output:

```
{
    "DBClusterParameterGroups": [
        {
            "DBClusterParameterGroupName": "default.aurora-mysql5.7",
            "DBParameterGroupFamily": "aurora-mysql5.7",
            "Description": "Default cluster parameter group for aurora-mysql5.7",
            "DBClusterParameterGroupArn": "arn:aws:rds:us-
east-1:123456789012:cluster-pg:default.aurora-mysql5.7"
        },
        {
            "DBClusterParameterGroupName": "default.aurora-postgresql9.6",
            "DBParameterGroupFamily": "aurora-postgresql9.6",
            "Description": "Default cluster parameter group for aurora-
postgresql9.6",
            "DBClusterParameterGroupArn": "arn:aws:rds:us-
east-1:123456789012:cluster-pg:default.aurora-postgresql9.6"
        },
        {
```

```
            "DBClusterParameterGroupName": "default.aurora5.6",
            "DBParameterGroupFamily": "aurora5.6",
            "Description": "Default cluster parameter group for aurora5.6",
            "DBClusterParameterGroupArn": "arn:aws:rds:us-
east-1:123456789012:cluster-pg:default.aurora5.6"
        },
        {
            "DBClusterParameterGroupName": "mydbclusterpg",
            "DBParameterGroupFamily": "aurora-mysql5.7",
            "Description": "My DB cluster parameter group",
            "DBClusterParameterGroupArn": "arn:aws:rds:us-
east-1:123456789012:cluster-pg:mydbclusterpg"
        },
        {
            "DBClusterParameterGroupName": "mydbclusterpgcopy",
            "DBParameterGroupFamily": "aurora-mysql5.7",
            "Description": "Copy of mydbclusterpg parameter group",
            "DBClusterParameterGroupArn": "arn:aws:rds:us-
east-1:123456789012:cluster-pg:mydbclusterpgcopy"
        }
    ]
}
```

Per ulteriori informazioni, consulta Uso di gruppi di parametri database e gruppi di parametri cluster database nella Guida per l'utente di Amazon Aurora.

- Per i dettagli sull'API, consulta DescribeDbClusterParameterGroups AWS CLICommand Reference.

`describe-db-cluster-parameters`

Il seguente esempio di codice mostra come utilizzare`describe-db-cluster-parameters`.

AWS CLI

Esempio 1: come descrivere i parametri in un gruppo di parametri del cluster di database

L'esempio `describe-db-cluster-parameters` seguente recupera i dettagli dei parametri in un gruppo di parametri del cluster di database.

```
aws rds describe-db-cluster-parameters \
    --db-cluster-parameter-group-name mydbclusterpg
```

Output:

```
{
    "Parameters": [
        {
            "ParameterName": "allow-suspicious-udfs",
            "Description": "Controls whether user-defined functions that have only
 an xxx symbol for the main function can be loaded",
            "Source": "engine-default",
            "ApplyType": "static",
            "DataType": "boolean",
            "AllowedValues": "0,1",
            "IsModifiable": false,
            "ApplyMethod": "pending-reboot",
            "SupportedEngineModes": [
                "provisioned"
            ]
        },
        {
            "ParameterName": "aurora_lab_mode",
            "ParameterValue": "0",
            "Description": "Enables new features in the Aurora engine.",
            "Source": "engine-default",
            "ApplyType": "static",
            "DataType": "boolean",
            "AllowedValues": "0,1",
            "IsModifiable": true,
            "ApplyMethod": "pending-reboot",
            "SupportedEngineModes": [
                "provisioned"
            ]
        },
        ...some output truncated...
    ]
}
```

Esempio 2: come elencare solo i nomi dei parametri in un gruppo di parametri del cluster di database

L'esempio `describe-db-cluster-parameters` seguente recupera solo i nomi dei parametri in un gruppo di parametri del cluster di database.

```
aws rds describe-db-cluster-parameters \
```

```
    --db-cluster-parameter-group-name default.aurora-mysql5.7 \
    --query 'Parameters[].{ParameterName:ParameterName}'
```

Output:

```
[
    {
        "ParameterName": "allow-suspicious-udfs"
    },
    {
        "ParameterName": "aurora_binlog_read_buffer_size"
    },
    {
        "ParameterName": "aurora_binlog_replication_max_yield_seconds"
    },
    {
        "ParameterName": "aurora_binlog_use_large_read_buffer"
    },
    {
        "ParameterName": "aurora_lab_mode"
    },

    ...some output truncated...
    }
]
```

Esempio 3: come descrivere solo i parametri modificabili in un gruppo di parametri del cluster di database

L'esempio `describe-db-cluster-parameters` seguente recupera solo i nomi dei parametri che è possibile modificare in un gruppo di parametri del cluster di database.

```
aws rds describe-db-cluster-parameters \
    --db-cluster-parameter-group-name default.aurora-mysql5.7 \
    --query 'Parameters[].{ParameterName:ParameterName,IsModifiable:IsModifiable} |
 [?IsModifiable == `true`]'
```

Output:

```
[
    {
        "ParameterName": "aurora_binlog_read_buffer_size",
```

```
        "IsModifiable": true
    },
    {
        "ParameterName": "aurora_binlog_replication_max_yield_seconds",
        "IsModifiable": true
    },
    {
        "ParameterName": "aurora_binlog_use_large_read_buffer",
        "IsModifiable": true
    },
    {
        "ParameterName": "aurora_lab_mode",
        "IsModifiable": true
    },

    ...some output truncated...
    }
]
```

Esempio 4: come descrivere solo i parametri booleani modificabili in un gruppo di parametri del cluster di database

L'esempio `describe-db-cluster-parameters` seguente recupera solo i nomi dei parametri che è possibile modificare in un gruppo di parametri del cluster di database e che hanno un tipo di dati booleano.

```
aws rds describe-db-cluster-parameters \
    --db-cluster-parameter-group-name default.aurora-mysql5.7 \
    --query 'Parameters[].
{ParameterName:ParameterName,DataType:DataType,IsModifiable:IsModifiable} | [?
DataType == `boolean`] | [?IsModifiable == `true`]'
```

Output:

```
[
    {
        "DataType": "boolean",
        "ParameterName": "aurora_binlog_use_large_read_buffer",
        "IsModifiable": true
    },
    {
        "DataType": "boolean",
```

```
        "ParameterName": "aurora_lab_mode",
        "IsModifiable": true
    },
    {
        "DataType": "boolean",
        "ParameterName": "autocommit",
        "IsModifiable": true
    },
    {
        "DataType": "boolean",
        "ParameterName": "automatic_sp_privileges",
        "IsModifiable": true
    },
    ...some output truncated...
    }
]
```

Per ulteriori informazioni, consulta Uso di gruppi di parametri database e gruppi di parametri cluster database nella Guida per l'utente di Amazon Aurora.

- Per i dettagli sull'API, consulta DescribeDbClusterParameters AWS CLICommand Reference.

`describe-db-cluster-snapshot-attributes`

Il seguente esempio di codice mostra come utilizzare`describe-db-cluster-snapshot-attributes`.

AWS CLI

Come descrivere i nomi e i valori degli attributi per uno snapshot del cluster di database

L'esempio `describe-db-cluster-snapshot-attributes` seguente recupera i dettagli dei nomi e dei valori degli attributi per lo snapshot del cluster di database specificato.

```
aws rds describe-db-cluster-snapshot-attributes \
    --db-cluster-snapshot-identifier myclustersnapshot
```

Output:

```
{
    "DBClusterSnapshotAttributesResult": {
        "DBClusterSnapshotIdentifier": "myclustersnapshot",
        "DBClusterSnapshotAttributes": [
```

```
            {
                "AttributeName": "restore",
                "AttributeValues": [
                    "123456789012"
                ]
            }
        ]
    }
}
```

Per ulteriori informazioni, consulta Condivisione di uno snapshot cluster database nella Guida per l'utente di Amazon Aurora.

- Per i dettagli sull'API, consulta DescribeDbClusterSnapshotAttributes AWS CLICommand Reference.

`describe-db-cluster-snapshots`

Il seguente esempio di codice mostra come utilizzare`describe-db-cluster-snapshots`.

AWS CLI

Come descrivere uno snapshot del cluster di database per un cluster di database

L'esempio `describe-db-cluster-snapshots` seguente recupera i dettagli per gli snapshot del cluster di database per il cluster di database specificato.

```
aws rds describe-db-cluster-snapshots \
    --db-cluster-identifier mydbcluster
```

Output:

```
{
    "DBClusterSnapshots": [
        {
            "AvailabilityZones": [
                "us-east-1a",
                "us-east-1b",
                "us-east-1e"
            ],
            "DBClusterSnapshotIdentifier": "myclustersnapshotcopy",
            "DBClusterIdentifier": "mydbcluster",
            "SnapshotCreateTime": "2019-06-04T09:16:42.649Z",
```

```
            "Engine": "aurora-mysql",
            "AllocatedStorage": 0,
            "Status": "available",
            "Port": 0,
            "VpcId": "vpc-6594f31c",
            "ClusterCreateTime": "2019-04-15T14:18:42.785Z",
            "MasterUsername": "myadmin",
            "EngineVersion": "5.7.mysql_aurora.2.04.2",
            "LicenseModel": "aurora-mysql",
            "SnapshotType": "manual",
            "PercentProgress": 100,
            "StorageEncrypted": true,
            "KmsKeyId": "arn:aws:kms:us-east-1:123456789012:key/
AKIAIOSFODNN7EXAMPLE",
            "DBClusterSnapshotArn": "arn:aws:rds:us-east-1:814387698303:cluster-
snapshot:myclustersnapshotcopy",
            "IAMDatabaseAuthenticationEnabled": false
        },
        {
            "AvailabilityZones": [
                "us-east-1a",
                "us-east-1b",
                "us-east-1e"
            ],
            "DBClusterSnapshotIdentifier": "rds:mydbcluster-2019-06-20-09-16",
            "DBClusterIdentifier": "mydbcluster",
            "SnapshotCreateTime": "2019-06-20T09:16:26.569Z",
            "Engine": "aurora-mysql",
            "AllocatedStorage": 0,
            "Status": "available",
            "Port": 0,
            "VpcId": "vpc-6594f31c",
            "ClusterCreateTime": "2019-04-15T14:18:42.785Z",
            "MasterUsername": "myadmin",
            "EngineVersion": "5.7.mysql_aurora.2.04.2",
            "LicenseModel": "aurora-mysql",
            "SnapshotType": "automated",
            "PercentProgress": 100,
            "StorageEncrypted": true,
            "KmsKeyId": "arn:aws:kms:us-east-1:814387698303:key/
AKIAIOSFODNN7EXAMPLE",
            "DBClusterSnapshotArn": "arn:aws:rds:us-east-1:123456789012:cluster-
snapshot:rds:mydbcluster-2019-06-20-09-16",
            "IAMDatabaseAuthenticationEnabled": false
```

```
        }
    ]
}
```

Per ulteriori informazioni, consulta Creazione di una snapshot di cluster database nella Guida per l'utente di Amazon Aurora.

- Per i dettagli sull'API, consulta DescribeDbClusterSnapshots AWS CLICommand Reference.

`describe-db-clusters`

Il seguente esempio di codice mostra come utilizzare`describe-db-clusters`.

AWS CLI

Esempio 1: come descrivere un cluster di database

L'esempio `describe-db-clusters` seguente recupera i dettagli del cluster di database specificato.

```
aws rds describe-db-clusters \
    --db-cluster-identifier mydbcluster
```

Output:

```
{
    "DBClusters": [
        {
            "AllocatedStorage": 1,
            "AvailabilityZones": [
                "us-east-1a",
                "us-east-1b",
                "us-east-1e"
            ],
            "BackupRetentionPeriod": 1,
            "DatabaseName": "mydbcluster",
            "DBClusterIdentifier": "mydbcluster",
            "DBClusterParameterGroup": "default.aurora-mysql5.7",
            "DBSubnetGroup": "default",
            "Status": "available",
            "EarliestRestorableTime": "2019-06-19T09:16:28.210Z",
            "Endpoint": "mydbcluster.cluster-cnpexample.us-
east-1.rds.amazonaws.com",
```

```
            "ReaderEndpoint": "mydbcluster.cluster-ro-cnpexample.us-
east-1.rds.amazonaws.com",
            "MultiAZ": true,
            "Engine": "aurora-mysql",
            "EngineVersion": "5.7.mysql_aurora.2.04.2",
            "LatestRestorableTime": "2019-06-20T22:38:14.908Z",
            "Port": 3306,
            "MasterUsername": "myadmin",
            "PreferredBackupWindow": "09:09-09:39",
            "PreferredMaintenanceWindow": "sat:04:09-sat:04:39",
            "ReadReplicaIdentifiers": [],
            "DBClusterMembers": [
                {
                    "DBInstanceIdentifier": "dbinstance3",
                    "IsClusterWriter": false,
                    "DBClusterParameterGroupStatus": "in-sync",
                    "PromotionTier": 1
                },
                {
                    "DBInstanceIdentifier": "dbinstance1",
                    "IsClusterWriter": false,
                    "DBClusterParameterGroupStatus": "in-sync",
                    "PromotionTier": 1
                },
                {
                    "DBInstanceIdentifier": "dbinstance2",
                    "IsClusterWriter": false,
                    "DBClusterParameterGroupStatus": "in-sync",
                    "PromotionTier": 1
                },
                {
                    "DBInstanceIdentifier": "mydbcluster",
                    "IsClusterWriter": false,
                    "DBClusterParameterGroupStatus": "in-sync",
                    "PromotionTier": 1
                },
                {
                    "DBInstanceIdentifier": "mydbcluster-us-east-1b",
                    "IsClusterWriter": false,
                    "DBClusterParameterGroupStatus": "in-sync",
                    "PromotionTier": 1
                },
                {
                    "DBInstanceIdentifier": "mydbcluster",
```

```
                    "IsClusterWriter": true,
                    "DBClusterParameterGroupStatus": "in-sync",
                    "PromotionTier": 1
                }
            ],
            "VpcSecurityGroups": [
                {
                    "VpcSecurityGroupId": "sg-0b9130572daf3dc16",
                    "Status": "active"
                }
            ],
            "HostedZoneId": "Z2R2ITUGPM61AM",
            "StorageEncrypted": true,
            "KmsKeyId": "arn:aws:kms:us-east-1:814387698303:key/
AKIAIOSFODNN7EXAMPLE",
            "DbClusterResourceId": "cluster-AKIAIOSFODNN7EXAMPLE",
            "DBClusterArn": "arn:aws:rds:us-
east-1:123456789012:cluster:mydbcluster",
            "AssociatedRoles": [],
            "IAMDatabaseAuthenticationEnabled": false,
            "ClusterCreateTime": "2019-04-15T14:18:42.785Z",
            "EngineMode": "provisioned",
            "DeletionProtection": false,
            "HttpEndpointEnabled": false
        }
    ]
}
```

Esempio 2: come elencare determinati attributi di tutti i cluster di database

L'`describe-db-clusters`esempio seguente recupera solo gli `ReaderEndpoint` attributi `DBClusterIdentifierEndpoint`, e di tutti i cluster DB nella regione corrente AWS .

```
aws rds describe-db-clusters \
    --query 'DBClusters[].
{DBClusterIdentifier:DBClusterIdentifier,Endpoint:Endpoint,ReaderEndpoint:ReaderEndpoint}'
```

Output:

```
[
    {
        "Endpoint": "cluster-57-2020-05-01-2270.cluster-cnpexample.us-
east-1.rds.amazonaws.com",
```

```
        "ReaderEndpoint": "cluster-57-2020-05-01-2270.cluster-ro-cnpexample.us-
east-1.rds.amazonaws.com",
        "DBClusterIdentifier": "cluster-57-2020-05-01-2270"
    },
    {
        "Endpoint": "cluster-57-2020-05-01-4615.cluster-cnpexample.us-
east-1.rds.amazonaws.com",
        "ReaderEndpoint": "cluster-57-2020-05-01-4615.cluster-ro-cnpexample.us-
east-1.rds.amazonaws.com",
        "DBClusterIdentifier": "cluster-57-2020-05-01-4615"
    },
    {
        "Endpoint": "pg2-cluster.cluster-cnpexample.us-east-1.rds.amazonaws.com",
        "ReaderEndpoint": "pg2-cluster.cluster-ro-cnpexample.us-
east-1.rds.amazonaws.com",
        "DBClusterIdentifier": "pg2-cluster"
    },
    ...output omitted...
    }
]
```

Esempio 3: come elencare i cluster di database con un attributo specifico

L'esempio `describe-db-clusters` seguente recupera solo gli attributi `DBClusterIdentifier` e `Engine` dei cluster di database che utilizzano il motore database `aurora-postgresql`.

```
aws rds describe-db-clusters \
    --query 'DBClusters[].{DBClusterIdentifier:DBClusterIdentifier,Engine:Engine} |
 [?Engine == `aurora-postgresql`]'
```

Output:

```
[
    {
        "Engine": "aurora-postgresql",
        "DBClusterIdentifier": "pg2-cluster"
    }
]
```

Per ulteriori informazioni, consulta Cluster database Amazon Aurora nella Guida per l'utente di Amazon Aurora.

- Per i dettagli sull'API, consulta AWS CLI Command DescribeDbClustersReference.

`describe-db-engine-versions`

Il seguente esempio di codice mostra come utilizzare`describe-db-engine-versions`.

AWS CLI

Come descrivere le versioni del motore di database per il motore MySQL DB

L'esempio `describe-db-engine-versions` seguente mostra i dettagli di ciascuna versione del motore di database specificato.

```
aws rds describe-db-engine-versions \
    --engine mysql
```

Output:

```
{
    "DBEngineVersions": [
        {
            "Engine": "mysql",
            "EngineVersion": "5.5.46",
            "DBParameterGroupFamily": "mysql5.5",
            "DBEngineDescription": "MySQL Community Edition",
            "DBEngineVersionDescription": "MySQL 5.5.46",
            "ValidUpgradeTarget": [
                {
                    "Engine": "mysql",
                    "EngineVersion": "5.5.53",
                    "Description": "MySQL 5.5.53",
                    "AutoUpgrade": false,
                    "IsMajorVersionUpgrade": false
                },
                {
                    "Engine": "mysql",
                    "EngineVersion": "5.5.54",
                    "Description": "MySQL 5.5.54",
                    "AutoUpgrade": false,
                    "IsMajorVersionUpgrade": false
                },
                {
                    "Engine": "mysql",
```

```
                    "EngineVersion": "5.5.57",
                    "Description": "MySQL 5.5.57",
                    "AutoUpgrade": false,
                    "IsMajorVersionUpgrade": false
                },
                ...some output truncated...
            ]
        }
```

Per ulteriori informazioni, consulta Cos'è Amazon Relational Database Service (Amazon RDS)? nella Guida per l'utente di Amazon RDS.

- Per i dettagli sull'API, consulta Descrivi DBEngine le versioni in AWS CLI Command Reference.

`describe-db-instance-automated-backups`

Il seguente esempio di codice mostra come utilizzare`describe-db-instance-automated-backups`.

AWS CLI

Come descrivere i backup automatici per un'istanza database

L'esempio `describe-db-instance-automated-backups` seguente visualizza i dettagli relativi ai backup automatici per l'istanza di database specificata. I dettagli includono backup automatici replicati in altre AWS regioni.

```
aws rds describe-db-instance-automated-backups \
    --db-instance-identifier new-orcl-db
```

Output:

```
{
    "DBInstanceAutomatedBackups": [
        {
            "DBInstanceArn": "arn:aws:rds:us-east-1:123456789012:db:new-orcl-db",
            "DbiResourceId": "db-JKIB2GFQ5RV7REPLZA4EXAMPLE",
            "Region": "us-east-1",
            "DBInstanceIdentifier": "new-orcl-db",
            "RestoreWindow": {
                "EarliestTime": "2020-12-07T21:05:20.939Z",
                "LatestTime": "2020-12-07T21:05:20.939Z"
```

```
            },
            "AllocatedStorage": 20,
            "Status": "replicating",
            "Port": 1521,
            "InstanceCreateTime": "2020-12-04T15:28:31Z",
            "MasterUsername": "admin",
            "Engine": "oracle-se2",
            "EngineVersion": "12.1.0.2.v21",
            "LicenseModel": "bring-your-own-license",
            "OptionGroupName": "default:oracle-se2-12-1",
            "Encrypted": false,
            "StorageType": "gp2",
            "IAMDatabaseAuthenticationEnabled": false,
            "BackupRetentionPeriod": 14,
            "DBInstanceAutomatedBackupsArn": "arn:aws:rds:us-
west-2:123456789012:auto-backup:ab-jkib2gfq5rv7replzadausbrktni2bn4example"
        }
    ]
}
```

Per ulteriori informazioni, consulta Ricerca di informazioni sui backup replicati nella Guida per l'utente di Amazon RDS.

- Per i dettagli sull'API, consulta AWS CLI Command DescribeDbInstanceAutomatedBackupsReference.

`describe-db-instances`

Il seguente esempio di codice mostra come utilizzare`describe-db-instances`.

AWS CLI

Come descrivere un'istanza database

L'esempio `describe-db-instances` seguente recupera i dettagli dell'istanza database specificata.

```
aws rds describe-db-instances \
    --db-instance-identifier mydbinstancecf
```

Output:

```
{
```

```
    "DBInstances": [
        {
            "DBInstanceIdentifier": "mydbinstancecf",
            "DBInstanceClass": "db.t3.small",
            "Engine": "mysql",
            "DBInstanceStatus": "available",
            "MasterUsername": "masterawsuser",
            "Endpoint": {
                "Address": "mydbinstancecf.abcexample.us-east-1.rds.amazonaws.com",
                "Port": 3306,
                "HostedZoneId": "Z2R2ITUGPM61AM"
            },
            ...some output truncated...
        }
    ]
}
```

- Per i dettagli sull'API, consulta Descrivi DBInstances in AWS CLI Command Reference.

`describe-db-log-files`

Il seguente esempio di codice mostra come utilizzare`describe-db-log-files`.

AWS CLI

Come descrivere i file di log per un'istanza database

L'esempio `describe-db-log-files` seguente recupera i dettagli dei file di log per l'istanza database specificata.

```
aws rds describe-db-log-files -\
    -db-instance-identifier test-instance
```

Output:

```
{
    "DescribeDBLogFiles": [
        {
            "Size": 0,
            "LastWritten": 1533060000000,
            "LogFileName": "error/mysql-error-running.log"
        },
```

```
        {
            "Size": 2683,
            "LastWritten": 1532994300000,
            "LogFileName": "error/mysql-error-running.log.0"
        },
        {
            "Size": 107,
            "LastWritten": 1533057300000,
            "LogFileName": "error/mysql-error-running.log.18"
        },
        {
            "Size": 13105,
            "LastWritten": 1532991000000,
            "LogFileName": "error/mysql-error-running.log.23"
        },
        {
            "Size": 0,
            "LastWritten": 1533061200000,
            "LogFileName": "error/mysql-error.log"
        },
        {
            "Size": 3519,
            "LastWritten": 1532989252000,
            "LogFileName": "mysqlUpgrade"
        }
    ]
}
```

- Per i dettagli sull'API, consulta DescribeDbLogFiles AWS CLICommand Reference.

`describe-db-parameter-groups`

Il seguente esempio di codice mostra come utilizzare`describe-db-parameter-groups`.

AWS CLI

Come descrivere il gruppo di parametri del database

L'esempio `describe-db-parameter-groups` seguente recupera i dettagli di tutti i gruppi di parametri del database.

```
aws rds describe-db-parameter-groups
```

Output:

```
{
    "DBParameterGroups": [
        {
            "DBParameterGroupName": "default.aurora-mysql5.7",
            "DBParameterGroupFamily": "aurora-mysql5.7",
            "Description": "Default parameter group for aurora-mysql5.7",
            "DBParameterGroupArn": "arn:aws:rds:us-
east-1:123456789012:pg:default.aurora-mysql5.7"
        },
        {
            "DBParameterGroupName": "default.aurora-postgresql9.6",
            "DBParameterGroupFamily": "aurora-postgresql9.6",
            "Description": "Default parameter group for aurora-postgresql9.6",
            "DBParameterGroupArn": "arn:aws:rds:us-
east-1:123456789012:pg:default.aurora-postgresql9.6"
        },
        {
            "DBParameterGroupName": "default.aurora5.6",
            "DBParameterGroupFamily": "aurora5.6",
            "Description": "Default parameter group for aurora5.6",
            "DBParameterGroupArn": "arn:aws:rds:us-
east-1:123456789012:pg:default.aurora5.6"
        },
        {
            "DBParameterGroupName": "default.mariadb10.1",
            "DBParameterGroupFamily": "mariadb10.1",
            "Description": "Default parameter group for mariadb10.1",
            "DBParameterGroupArn": "arn:aws:rds:us-
east-1:123456789012:pg:default.mariadb10.1"
        },
        ...some output truncated...
    ]
}
```

Per ulteriori informazioni, consulta Utilizzo di gruppi di parametri di database nella Guida per l'utente di Amazon RDS.

- Per i dettagli sull'API, consulta Descrivi DBParameter i gruppi in AWS CLI Command Reference.

`describe-db-parameters`

Il seguente esempio di codice mostra come utilizzare`describe-db-parameters`.

AWS CLI

Come descrivere i parametri in un gruppo di parametri del database

L'esempio `describe-db-parameters` seguente recupera i dettagli del gruppo di parametri database specificato.

```
aws rds describe-db-parameters \
    --db-parameter-group-name mydbpg
```

Output:

```
{
    "Parameters": [
        {
            "ParameterName": "allow-suspicious-udfs",
            "Description": "Controls whether user-defined functions that have only
 an xxx symbol for the main function can be loaded",
            "Source": "engine-default",
            "ApplyType": "static",
            "DataType": "boolean",
            "AllowedValues": "0,1",
            "IsModifiable": false,
            "ApplyMethod": "pending-reboot"
        },
        {
            "ParameterName": "auto_generate_certs",
            "Description": "Controls whether the server autogenerates SSL key and
 certificate files in the data directory, if they do not already exist.",
            "Source": "engine-default",
            "ApplyType": "static",
            "DataType": "boolean",
            "AllowedValues": "0,1",
            "IsModifiable": false,
            "ApplyMethod": "pending-reboot"
        },
        ...some output truncated...
    ]
}
```

Per ulteriori informazioni, consulta Utilizzo di gruppi di parametri di database nella Guida per l'utente di Amazon RDS.

- Per i dettagli sull'API, consulta Descrivi DBParameters in AWS CLI Command Reference.

`describe-db-proxies`

Il seguente esempio di codice mostra come utilizzare`describe-db-proxies`.

AWS CLI

Come descrivere un proxy database per un database RDS

L'esempio `describe-db-proxies` seguente restituisce le informazioni relative ai proxy database.

```
aws rds describe-db-proxies
```

Output:

```
{
    "DBProxies": [
        {
            "DBProxyName": "proxyExample1",
            "DBProxyArn": "arn:aws:rds:us-east-1:123456789012:db-
proxy:prx-0123a01b12345c0ab",
            "Status": "available",
            "EngineFamily": "PostgreSQL",
            "VpcId": "vpc-1234567",
            "VpcSecurityGroupIds": [
                "sg-1234"
            ],
            "VpcSubnetIds": [
                "subnetgroup1",
                "subnetgroup2"
            ],
            "Auth": "[
                {
                    "Description": "proxydescription1"
                    "AuthScheme": "SECRETS",
                    "SecretArn": "arn:aws:secretsmanager:us-
west-2:123456789123:secret:secretName-1234f",
                    "IAMAuth": "DISABLED"
```

```
                }
            ]",
            "RoleArn": "arn:aws:iam::12345678912??:role/ProxyPostgreSQLRole",
            "Endpoint": "proxyExample1.proxy-ab0cd1efghij.us-
east-1.rds.amazonaws.com",
            "RequireTLS": false,
            "IdleClientTimeout": 1800,
            "DebuggingLogging": false,
            "CreatedDate": "2023-04-05T16:09:33.452000+00:00",
            "UpdatedDate": "2023-04-13T01:49:38.568000+00:00"
        },
        {
            "DBProxyName": "proxyExample2",
            "DBProxyArn": "arn:aws:rds:us-east-1:123456789012:db-
proxy:prx-1234a12b23456c1ab",
            "Status": "available",
            "EngineFamily": "PostgreSQL",
            "VpcId": "sg-1234567",
            "VpcSecurityGroupIds": [
                "sg-1234"
            ],
            "VpcSubnetIds": [
                "subnetgroup1",
                "subnetgroup2"
            ],
            "Auth": "[
                {
                    "Description": "proxydescription2"
                    "AuthScheme": "SECRETS",
                    "SecretArn": "aarn:aws:secretsmanager:us-
west-2:123456789123:secret:secretName-1234f",
                    "IAMAuth": "DISABLED"
                }
            ]",
            "RoleArn": "arn:aws:iam::12345678912:role/ProxyPostgreSQLRole",
            "Endpoint": "proxyExample2.proxy-ab0cd1efghij.us-
east-1.rds.amazonaws.com",
            "RequireTLS": false,
            "IdleClientTimeout": 1800,
            "DebuggingLogging": false,
            "CreatedDate": "2022-01-05T16:19:33.452000+00:00",
            "UpdatedDate": "2023-04-13T01:49:38.568000+00:00"
        }
    ]
```

```
}
```

Per ulteriori informazioni, consulta Visualizzazione di un RDS Proxy nella Guida per l'utente di Amazon RDS e Visualizzazione di un RDS Proxy nella Guida per l'utente di Amazon Aurora.

- Per i dettagli sull'API, consulta DescribeDbProxies AWS CLICommand Reference.

`describe-db-proxy-endpoints`

Il seguente esempio di codice mostra come utilizzare`describe-db-proxy-endpoints`.

AWS CLI

Come descrivere gli endpoint di un proxy database

L'esempio `describe-db-proxy-endpoints` seguente restituisce informazioni sugli endpoint di un proxy database.

```
aws rds describe-db-proxy-endpoints
```

Output:

```
{
    "DBProxyEndpoints": [
        {
            "DBProxyEndpointName": "proxyEndpoint1",
            "DBProxyEndpointArn": "arn:aws:rds:us-east-1:123456789012:db-proxy-
endpoint:prx-endpoint-0123a01b12345c0ab",
            "DBProxyName": "proxyExample",
            "Status": "available",
            "VpcId": "vpc-1234567",
            "VpcSecurityGroupIds": [
                "sg-1234"
            ],
            "VpcSubnetIds": [
                "subnetgroup1",
                "subnetgroup2"
            ],
            "Endpoint": "proxyEndpoint1.endpoint.proxy-ab0cd1efghij.us-
east-1.rds.amazonaws.com",
            "CreatedDate": "2023-04-05T16:09:33.452000+00:00",
            "TargetRole": "READ_WRITE",
            "IsDefault": false
```

```
        },
        {
            "DBProxyEndpointName": "proxyEndpoint2",
            "DBProxyEndpointArn": "arn:aws:rds:us-east-1:123456789012:db-proxy-
endpoint:prx-endpoint-4567a01b12345c0ab",
            "DBProxyName": "proxyExample2",
            "Status": "available",
            "VpcId": "vpc1234567",
            "VpcSecurityGroupIds": [
                "sg-5678"
            ],
            "VpcSubnetIds": [
                "subnetgroup1",
                "subnetgroup2"
            ],
            "Endpoint": "proxyEndpoint2.endpoint.proxy-cd1ef2klmnop.us-
east-1.rds.amazonaws.com",
            "CreatedDate": "2023-04-05T16:09:33.452000+00:00",
            "TargetRole": "READ_WRITE",
            "IsDefault": false
        }
    ]
}
```

Per ulteriori informazioni, consulta Visualizzazione di un endpoint proxy nella Guida per l'utente di Amazon RDS e Creazione di un endpoint proxy nella Guida per l'utente di Amazon Aurora.

- Per i dettagli sull'API, consulta DescribeDbProxyEndpoints AWS CLICommand Reference.

`describe-db-proxy-target-groups`

Il seguente esempio di codice mostra come utilizzare`describe-db-proxy-target-groups`.

AWS CLI

Come descrivere gli endpoint di un proxy database

L'esempio `describe-db-proxy-target-groups` seguente restituisce informazioni sui gruppi di destinazioni del proxy database.

```
aws rds describe-db-proxy-target-groups \
    --db-proxy-name proxyExample
```

Output:

```
{
"TargetGroups":
    {
        "DBProxyName": "proxyExample",
        "TargetGroupName": "default",
        "TargetGroupArn": "arn:aws:rds:us-east-1:123456789012:target-group:prx-
tg-0123a01b12345c0ab",
        "IsDefault": true,
        "Status": "available",
        "ConnectionPoolConfig": {
            "MaxConnectionsPercent": 100,
            "MaxIdleConnectionsPercent": 50,
            "ConnectionBorrowTimeout": 120,
            "SessionPinningFilters": []
        },
        "CreatedDate": "2023-05-02T18:41:19.495000+00:00",
        "UpdatedDate": "2023-05-02T18:41:21.762000+00:00"
    }
}
```

Per ulteriori informazioni, consulta Visualizzazione di un RDS Proxy nella Guida per l'utente di Amazon RDS e Visualizzazione di un RDS Proxy nella Guida per l'utente di Amazon Aurora.

- Per i dettagli sull'API, consulta DescribeDbProxyTargetGroups AWS CLICommand Reference.

`describe-db-proxy-targets`

Il seguente esempio di codice mostra come utilizzare`describe-db-proxy-targets`.

AWS CLI

Come descrivere le destinazioni del proxy database

L'esempio `describe-db-proxy-targets` seguente restituisce informazioni sulle destinazioni del proxy database.

```
aws rds describe-db-proxy-targets \
    --db-proxy-name proxyExample
```

Output:

```
{
    "Targets": [
        {
            "Endpoint": "database1.ab0cd1efghij.us-east-1.rds.amazonaws.com",
            "TrackedClusterId": "database1",
            "RdsResourceId": "database1-instance-1",
            "Port": 3306,
            "Type": "RDS_INSTANCE",
            "Role": "READ_WRITE",
            "TargetHealth": {
                "State": "UNAVAILABLE",
                "Reason": "PENDING_PROXY_CAPACITY",
                "Description": "DBProxy Target is waiting for proxy to scale to
 desired capacity"
            }
        }
    ]
}
```

Per ulteriori informazioni, consulta Visualizzazione di un RDS Proxy nella Guida per l'utente di Amazon RDS e Visualizzazione di un RDS Proxy nella Guida per l'utente di Amazon Aurora.

- Per i dettagli sull'API, consulta DescribeDbProxyTargets AWS CLICommand Reference.

`describe-db-recommendations`

Il seguente esempio di codice mostra come utilizzare`describe-db-recommendations`.

AWS CLI

Esempio 1: come elencare tutte le raccomandazioni database

L'`describe-db-recommendations`esempio seguente elenca tutti i consigli DB presenti nel tuo AWS account.

```
aws rds describe-db-recommendations
```

Output:

```
{
    "DBRecommendations": [
        {
```

```
            "RecommendationId": "12ab3cde-f456-7g8h-9012-i3j45678k9lm",
            "TypeId": "config_recommendation::old_minor_version",
            "Severity": "informational",
            "ResourceArn": "arn:aws:rds:us-west-2:111122223333:db:database-1",
            "Status": "active",
            "CreatedTime": "2024-02-21T23:14:19.292000+00:00",
            "UpdatedTime": "2024-02-21T23:14:19+00:00",
            "Detection": "**[resource-name]** is not running the latest minor DB
engine version",
            "Recommendation": "Upgrade to latest engine version",
            "Description": "Your database resources aren't running the latest minor
DB engine version. The latest minor version contains the latest security fixes and
other improvements.",
            "RecommendedActions": [
                {
                    "ActionId": "12ab34c5de6fg7h89i0jk1lm234n5678",
                    "Operation": "modifyDbInstance",
                    "Parameters": [
                        {
                            "Key": "EngineVersion",
                            "Value": "5.7.44"
                        },
                        {
                            "Key": "DBInstanceIdentifier",
                            "Value": "database-1"
                        }
                    ],
                    "ApplyModes": [
                        "immediately",
                        "next-maintenance-window"
                    ],
                    "Status": "ready",
                    "ContextAttributes": [
                        {
                            "Key": "Recommended value",
                            "Value": "5.7.44"
                        },
                        {
                            "Key": "Current engine version",
                            "Value": "5.7.42"
                        }
                    ]
                }
            ],
```

```
            "Category": "security",
            "Source": "RDS",
            "TypeDetection": "**[resource-count] resources** are not running the
 latest minor DB engine version",
            "TypeRecommendation": "Upgrade to latest engine version",
            "Impact": "Reduced database performance and data security at risk",
            "AdditionalInfo": "We recommend that you maintain your database with the
 latest DB engine minor version as this version includes the latest security and
 functionality fixes. The DB engine minor version upgrades contain only the changes
 which are backward-compatible with earlier minor versions of the same major version
 of the DB engine.",
            "Links": [
                {
                    "Text": "Upgrading an RDS DB instance engine version",
                    "Url": "https://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/
USER_UpgradeDBInstance.Upgrading.html"
                },
                {
                    "Text": "Using Amazon RDS Blue/Green Deployments for database
 updates for Amazon Aurora",
                    "Url": "https://docs.aws.amazon.com/AmazonRDS/latest/
AuroraUserGuide/blue-green-deployments.html"
                },
                {
                    "Text": "Using Amazon RDS Blue/Green Deployments for database
 updates for Amazon RDS",
                    "Url": "https://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/
blue-green-deployments.html"
                }
            ]
        }
    ]
}
```

Per ulteriori informazioni, consulta Visualizzazione e risposta di suggerimenti Amazon RDS nella Guida per l'utente di Amazon RDS e Visualizzazione e risposta a suggerimenti Amazon RDS nella Guida per l'utente di Amazon Aurora.

Esempio 2: come elencare le raccomandazioni database di elevata gravità

L'`describe-db-recommendations`esempio seguente elenca i consigli DB ad alta severità nel tuo AWS account.

```
aws rds describe-db-recommendations \
    --filters Name=severity,Values=high
```

Output:

```
{
    "DBRecommendations": [
        {
            "RecommendationId": "12ab3cde-f456-7g8h-9012-i3j45678k9lm",
            "TypeId": "config_recommendation::rds_extended_support",
            "Severity": "high",
            "ResourceArn": "arn:aws:rds:us-west-2:111122223333:db:database-1",
            "Status": "active",
            "CreatedTime": "2024-02-21T23:14:19.392000+00:00",
            "UpdatedTime": "2024-02-21T23:14:19+00:00",
            "Detection": "Your databases will be auto-enrolled to RDS Extended
 Support on February 29",
            "Recommendation": "Upgrade your major version before February 29, 2024
 to avoid additional charges",
            "Description": "Your PostgreSQL 11 and MySQL 5.7 databases will be
 automatically enrolled into RDS Extended Support on February 29, 2024. To avoid
 the increase in charges due to RDS Extended Support, we recommend upgrading your
 databases to a newer major engine version before February 29, 2024.\nTo learn more
 about the RDS Extended Support pricing, refer to the pricing page.",
            "RecommendedActions": [
                {
                    "ActionId": "12ab34c5de6fg7h89i0jk1lm234n5678",
                    "Parameters": [],
                    "ApplyModes": [
                        "manual"
                    ],
                    "Status": "ready",
                    "ContextAttributes": []
                }
            ],
            "Category": "cost optimization",
            "Source": "RDS",
            "TypeDetection": "Your database will be auto-enrolled to RDS Extended
 Support on February 29",
            "TypeRecommendation": "Upgrade your major version before February 29,
 2024 to avoid additional charges",
            "Impact": "Increase in charges due to RDS Extended Support",
```

```
            "AdditionalInfo": "With Amazon RDS Extended Support, you can continue
 running your database on a major engine version past the RDS end of standard
 support date for an additional cost. This paid feature gives you more time to
 upgrade to a supported major engine version.\nDuring Extended Support, Amazon RDS
 will supply critical CVE patches and bug fixes.",
            "Links": [
                {
                    "Text": "Amazon RDS Extended Support pricing for RDS for MySQL",
                    "Url": "https://aws.amazon.com/rds/mysql/pricing/"
                },
                {
                    "Text": "Amazon RDS Extended Support for RDS for MySQL and
 PostgreSQL databases",
                    "Url": "https://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/
extended-support.html"
                },
                {
                    "Text": "Amazon RDS Extended Support pricing for Amazon Aurora
 PostgreSQL",
                    "Url": "https://aws.amazon.com/rds/aurora/pricing/"
                },
                {
                    "Text": "Amazon RDS Extended Support for Aurora PostgreSQL
 databases",
                    "Url": "https://docs.aws.amazon.com/AmazonRDS/latest/
AuroraUserGuide/extended-support.html"
                },
                {
                    "Text": "Amazon RDS Extended Support pricing for RDS for
 PostgreSQL",
                    "Url": "https://aws.amazon.com/rds/postgresql/pricing/"
                }
            ]
        }
    ]
}
```

Per ulteriori informazioni, consulta Visualizzazione e risposta di suggerimenti Amazon RDS nella Guida per l'utente di Amazon RDS e Visualizzazione e risposta a suggerimenti Amazon RDS nella Guida per l'utente di Amazon Aurora.

Esempio 3: come elencare le raccomandazioni database per un'istanza database specificata

L'esempio `describe-db-recommendations` seguente elenca tutti le raccomandazioni database per un'istanza database specificata.

```
aws rds describe-db-recommendations \
    --filters Name=dbi-resource-id,Values=database-1
```

Output:

```
{
    "DBRecommendations": [
        {
            "RecommendationId": "12ab3cde-f456-7g8h-9012-i3j45678k9lm",
            "TypeId": "config_recommendation::old_minor_version",
            "Severity": "informational",
            "ResourceArn": "arn:aws:rds:us-west-2:111122223333:db:database-1",
            "Status": "active",
            "CreatedTime": "2024-02-21T23:14:19.292000+00:00",
            "UpdatedTime": "2024-02-21T23:14:19+00:00",
            "Detection": "**[resource-name]** is not running the latest minor DB
 engine version",
            "Recommendation": "Upgrade to latest engine version",
            "Description": "Your database resources aren't running the latest minor
 DB engine version. The latest minor version contains the latest security fixes and
 other improvements.",
            "RecommendedActions": [
                {
                    "ActionId": "12ab34c5de6fg7h89i0jk1lm234n5678",
                    "Operation": "modifyDbInstance",
                    "Parameters": [
                        {
                            "Key": "EngineVersion",
                            "Value": "5.7.44"
                        },
                        {
                            "Key": "DBInstanceIdentifier",
                            "Value": "database-1"
                        }
                    ],
                    "ApplyModes": [
                        "immediately",
                        "next-maintenance-window"
                    ],
                    "Status": "ready",
```

```
                    "ContextAttributes": [
                        {
                            "Key": "Recommended value",
                            "Value": "5.7.44"
                        },
                        {
                            "Key": "Current engine version",
                            "Value": "5.7.42"
                        }
                    ]
                }
            ],
            "Category": "security",
            "Source": "RDS",
            "TypeDetection": "**[resource-count] resources** are not running the
 latest minor DB engine version",
            "TypeRecommendation": "Upgrade to latest engine version",
            "Impact": "Reduced database performance and data security at risk",
            "AdditionalInfo": "We recommend that you maintain your database with the
 latest DB engine minor version as this version includes the latest security and
 functionality fixes. The DB engine minor version upgrades contain only the changes
 which are backward-compatible with earlier minor versions of the same major version
 of the DB engine.",
            "Links": [
                {
                    "Text": "Upgrading an RDS DB instance engine version",
                    "Url": "https://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/
USER_UpgradeDBInstance.Upgrading.html"
                },
                {
                    "Text": "Using Amazon RDS Blue/Green Deployments for database
 updates for Amazon Aurora",
                    "Url": "https://docs.aws.amazon.com/AmazonRDS/latest/
AuroraUserGuide/blue-green-deployments.html"
                },
                {
                    "Text": "Using Amazon RDS Blue/Green Deployments for database
 updates for Amazon RDS",
                    "Url": "https://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/
blue-green-deployments.html"
                }
            ]
        }
    ]
```

```
}
```

Per ulteriori informazioni, consulta Visualizzazione e risposta di suggerimenti Amazon RDS nella Guida per l'utente di Amazon RDS e Visualizzazione e risposta a suggerimenti Amazon RDS nella Guida per l'utente di Amazon Aurora.

Esempio 4: come elencare tutti le raccomandazioni database attive

L'`describe-db-recommendations`esempio seguente elenca tutti i consigli DB attivi nel tuo AWS account.

```
aws rds describe-db-recommendations \
    --filters Name=status,Values=active
```

Output:

```
{
    "DBRecommendations": [
        {
            "RecommendationId": "12ab3cde-f456-7g8h-9012-i3j45678k9lm",
            "TypeId": "config_recommendation::old_minor_version",
            "Severity": "informational",
            "ResourceArn": "arn:aws:rds:us-west-2:111122223333:db:database-1",
            "Status": "active",
            "CreatedTime": "2024-02-21T23:14:19.292000+00:00",
            "UpdatedTime": "2024-02-21T23:14:19+00:00",
            "Detection": "**[resource-name]** is not running the latest minor DB
 engine version",
            "Recommendation": "Upgrade to latest engine version",
            "Description": "Your database resources aren't running the latest minor
 DB engine version. The latest minor version contains the latest security fixes and
 other improvements.",
            "RecommendedActions": [
                {
                    "ActionId": "12ab34c5de6fg7h89i0jk1lm234n5678",
                    "Operation": "modifyDbInstance",
                    "Parameters": [
                        {
                            "Key": "EngineVersion",
                            "Value": "5.7.44"
                        },
                        {
                            "Key": "DBInstanceIdentifier",
```

```
                            "Value": "database-1"
                        }
                    ],
                    "ApplyModes": [
                        "immediately",
                        "next-maintenance-window"
                    ],
                    "Status": "ready",
                    "ContextAttributes": [
                        {
                            "Key": "Recommended value",
                            "Value": "5.7.44"
                        },
                        {
                            "Key": "Current engine version",
                            "Value": "5.7.42"
                        }
                    ]
                }
            ],
            "Category": "security",
            "Source": "RDS",
            "TypeDetection": "**[resource-count] resources** are not running the
 latest minor DB engine version",
            "TypeRecommendation": "Upgrade to latest engine version",
            "Impact": "Reduced database performance and data security at risk",
            "AdditionalInfo": "We recommend that you maintain your database with the
 latest DB engine minor version as this version includes the latest security and
 functionality fixes. The DB engine minor version upgrades contain only the changes
 which are backward-compatible with earlier minor versions of the same major version
 of the DB engine.",
            "Links": [
                {
                    "Text": "Upgrading an RDS DB instance engine version",
                    "Url": "https://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/
USER_UpgradeDBInstance.Upgrading.html"
                },
                {
                    "Text": "Using Amazon RDS Blue/Green Deployments for database
 updates for Amazon Aurora",
                    "Url": "https://docs.aws.amazon.com/AmazonRDS/latest/
AuroraUserGuide/blue-green-deployments.html"
                },
                {
```

```
                    "Text": "Using Amazon RDS Blue/Green Deployments for database
 updates for Amazon RDS",
                    "Url": "https://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/
blue-green-deployments.html"
                }
            ]
        }
    ]
}
```

Per ulteriori informazioni, consulta Visualizzazione e risposta di suggerimenti Amazon RDS nella Guida per l'utente di Amazon RDS e Visualizzazione e risposta a suggerimenti Amazon RDS nella Guida per l'utente di Amazon Aurora.

- Per i dettagli sull'API, consulta DescribeDbRecommendations AWS CLICommand Reference.

`describe-db-security-groups`

Il seguente esempio di codice mostra come utilizzare`describe-db-security-groups`.

AWS CLI

Come elencare i gruppi di sicurezza del database

L'esempio `describe-db-security-groups` seguente elenca i gruppi di sicurezza database.

```
aws rds describe-db-security-groups
```

Output:

```
{
    "DBSecurityGroups": [
        {
            "OwnerId": "123456789012",
            "DBSecurityGroupName": "default",
            "DBSecurityGroupDescription": "default",
            "EC2SecurityGroups": [],
            "IPRanges": [],
            "DBSecurityGroupArn": "arn:aws:rds:us-
west-1:111122223333:secgrp:default"
        },
        {
            "OwnerId": "123456789012",
```

```
            "DBSecurityGroupName": "mysecgroup",
            "DBSecurityGroupDescription": "My Test Security Group",
            "VpcId": "vpc-1234567f",
            "EC2SecurityGroups": [],
            "IPRanges": [],
            "DBSecurityGroupArn": "arn:aws:rds:us-
west-1:111122223333:secgrp:mysecgroup"
        }
    ]
}
```

Per ulteriori informazioni, consulta Elenco dei gruppi di sicurezza database disponibili nella Guida per l'utente di Amazon RDS.

- Per i dettagli sull'API, consulta DescribeDbSecurityGroups AWS CLICommand Reference.

`describe-db-shard-groups`

Il seguente esempio di codice mostra come utilizzare`describe-db-shard-groups`.

AWS CLI

Esempio 1: come descrivere i gruppi di shard del database

L'esempio `describe-db-shard-groups` seguente recupera i dettagli dei gruppi di shard database.

```
aws rds describe-db-shard-groups
```

Output:

```
{
    "DBShardGroups": [
        {
            "DBShardGroupResourceId": "shardgroup-7bb446329da94788b3f957746example",
            "DBShardGroupIdentifier": "limitless-test-shard-grp",
            "DBClusterIdentifier": "limitless-test-cluster",
            "MaxACU": 768.0,
            "ComputeRedundancy": 0,
            "Status": "available",
            "PubliclyAccessible": true,
            "Endpoint": "limitless-test-cluster.limitless-cekycexample.us-
east-2.rds.amazonaws.com"
```

```
        },
        {
            "DBShardGroupResourceId": "shardgroup-a6e3a0226aa243e2ac6c7a1234567890",
            "DBShardGroupIdentifier": "my-db-shard-group",
            "DBClusterIdentifier": "my-sv2-cluster",
            "MaxACU": 768.0,
            "ComputeRedundancy": 0,
            "Status": "available",
            "PubliclyAccessible": false,
            "Endpoint": "my-sv2-cluster.limitless-cekycexample.us-
east-2.rds.amazonaws.com"
        }
    ]
}
```

Per ulteriori informazioni, consulta Cluster database Amazon Aurora nella Guida per l'utente di Amazon Aurora.

- Per i dettagli sull'API, consulta DescribeDbShardGroups AWS CLICommand Reference.

`describe-db-snapshot-attributes`

Il seguente esempio di codice mostra come utilizzare`describe-db-snapshot-attributes`.

AWS CLI

Come descrivere i nomi e i valori degli attributi per uno snapshot database

L'esempio `describe-db-snapshot-attributes` seguente descrive i nomi e i valori degli attributi per uno snapshot database.

```
aws rds describe-db-snapshot-attributes \
    --db-snapshot-identifier mydbsnapshot
```

Output:

```
{
    "DBSnapshotAttributesResult": {
        "DBSnapshotIdentifier": "mydbsnapshot",
        "DBSnapshotAttributes": [
            {
                "AttributeName": "restore",
                "AttributeValues": [
```

```
                    "123456789012",
                    "210987654321"
                ]
            }
        ]
    }
}
```

Per ulteriori informazioni, consulta Condivisione di uno snapshot di database nella Guida per l'utente di Amazon RDS.

- Per i dettagli sull'API, consulta DescribeDbSnapshotAttributes AWS CLICommand Reference.

`describe-db-snapshots`

Il seguente esempio di codice mostra come utilizzare`describe-db-snapshots`.

AWS CLI

Esempio 1: come descrivere uno snapshot di database per un'istanza database

L'esempio `describe-db-snapshots` seguente recupera i dettagli di uno snapshot database per un'istanza database.

```
aws rds describe-db-snapshots \
    --db-snapshot-identifier mydbsnapshot
```

Output:

```
{
    "DBSnapshots": [
        {
            "DBSnapshotIdentifier": "mydbsnapshot",
            "DBInstanceIdentifier": "mysqldb",
            "SnapshotCreateTime": "2018-02-08T22:28:08.598Z",
            "Engine": "mysql",
            "AllocatedStorage": 20,
            "Status": "available",
            "Port": 3306,
            "AvailabilityZone": "us-east-1f",
            "VpcId": "vpc-6594f31c",
            "InstanceCreateTime": "2018-02-08T22:24:55.973Z",
            "MasterUsername": "mysqladmin",
```

```
            "EngineVersion": "5.6.37",
            "LicenseModel": "general-public-license",
            "SnapshotType": "manual",
            "OptionGroupName": "default:mysql-5-6",
            "PercentProgress": 100,
            "StorageType": "gp2",
            "Encrypted": false,
            "DBSnapshotArn": "arn:aws:rds:us-
east-1:123456789012:snapshot:mydbsnapshot",
            "IAMDatabaseAuthenticationEnabled": false,
            "ProcessorFeatures": [],
            "DbiResourceId": "db-AKIAIOSFODNN7EXAMPLE"
        }
    ]
}
```

Per ulteriori informazioni, consulta Creazione di uno snapshot di database nella Guida per l'utente di Amazon RDS.

Esempio 2: come trovare il numero di snapshot manuali effettuati

L'`describe-db-snapshots`esempio seguente utilizza l'`length`operatore nell'`--query`opzione per restituire il numero di istantanee manuali che sono state scattate in una particolare AWS regione.

```
aws rds describe-db-snapshots \
    --snapshot-type manual \
    --query "length(*[].{DBSnapshots:SnapshotType})" \
    --region eu-central-1
```

Output:

```
35
```

Per ulteriori informazioni, consulta Creazione di uno snapshot di database nella Guida per l'utente di Amazon RDS.

- Per i dettagli sull'API, consulta Descrivi DBSnapshots in AWS CLI Command Reference.

`describe-db-subnet-groups`

Il seguente esempio di codice mostra come utilizzare`describe-db-subnet-groups`.

AWS CLI

Come descrivere un gruppo di sottoreti database

L'esempio `describe-db-subnet-groups` seguente recupera i dettagli del gruppo di sottoreti database specificato.

```
aws rds describe-db-subnet-groups
```

Output:

```
{
    "DBSubnetGroups": [
        {
            "DBSubnetGroupName": "mydbsubnetgroup",
            "DBSubnetGroupDescription": "My DB Subnet Group",
            "VpcId": "vpc-971c12ee",
            "SubnetGroupStatus": "Complete",
            "Subnets": [
                {
                    "SubnetIdentifier": "subnet-d8c8e7f4",
                    "SubnetAvailabilityZone": {
                        "Name": "us-east-1a"
                    },
                    "SubnetStatus": "Active"
                },
                {
                    "SubnetIdentifier": "subnet-718fdc7d",
                    "SubnetAvailabilityZone": {
                        "Name": "us-east-1f"
                    },
                    "SubnetStatus": "Active"
                },
                {
                    "SubnetIdentifier": "subnet-cbc8e7e7",
                    "SubnetAvailabilityZone": {
                        "Name": "us-east-1a"
                    },
                    "SubnetStatus": "Active"
                },
                {
                    "SubnetIdentifier": "subnet-0ccde220",
                    "SubnetAvailabilityZone": {
```

```
                    "Name": "us-east-1a"
                },
                "SubnetStatus": "Active"
            }
        ],
        "DBSubnetGroupArn": "arn:aws:rds:us-
east-1:123456789012:subgrp:mydbsubnetgroup"
    }
  ]
}
```

Per ulteriori informazioni, consulta Amazon Virtual Private Cloud VPCs e Amazon RDS nella Amazon RDS User Guide.

- Per i dettagli sull'API, consulta AWS CLI Command DescribeDbSubnetGroupsReference.

`describe-engine-default-cluster-parameters`

Il seguente esempio di codice mostra come utilizzare`describe-engine-default-cluster-parameters`.

AWS CLI

Come descrivere il motore predefinito e le informazioni del parametro di sistema per il motore di database Aurora

L'esempio `describe-engine-default-cluster-parameters` seguente recupera i dettagli del motore predefinito e delle informazioni sui parametri di sistema per i cluster Aurora DB con compatibilità MySQL 5.7.

```
aws rds describe-engine-default-cluster-parameters \
    --db-parameter-group-family aurora-mysql5.7
```

Output:

```
{
    "EngineDefaults": {
        "Parameters": [
            {
                "ParameterName": "aurora_load_from_s3_role",
                "Description": "IAM role ARN used to load data from AWS S3",
                "Source": "engine-default",
```

```
                "ApplyType": "dynamic",
                "DataType": "string",
                "IsModifiable": true,
                "SupportedEngineModes": [
                    "provisioned"
                ]
            },
            ...some output truncated...
        ]
    }
}
```

Per ulteriori informazioni, consulta Uso di gruppi di parametri database e gruppi di parametri cluster database nella Guida per l'utente di Amazon Aurora.

- Per i dettagli sull'API, consulta DescribeEngineDefaultClusterParameters AWS CLICommand Reference.

`describe-engine-default-parameters`

Il seguente esempio di codice mostra come utilizzare`describe-engine-default-parameters`.

AWS CLI

Come descrivere il motore predefinito e le informazioni sui parametri di sistema per il motore di database

L'esempio `describe-engine-default-parameters` seguente recupera i dettagli del motore predefinito e delle informazioni sui parametri di sistema per le istanze database MySQL 5.7.

```
aws rds describe-engine-default-parameters \
    --db-parameter-group-family mysql5.7
```

Output:

```
{
    "EngineDefaults": {
        "Parameters": [
            {
                "ParameterName": "allow-suspicious-udfs",
                "Description": "Controls whether user-defined functions that have
 only an xxx symbol for the main function can be loaded",
```

```
                "Source": "engine-default",
                "ApplyType": "static",
                "DataType": "boolean",
                "AllowedValues": "0,1",
                "IsModifiable": false
            },
            ...some output truncated...
        ]
    }
}
```

Per ulteriori informazioni, consulta Utilizzo di gruppi di parametri di database nella Guida per l'utente di Amazon RDS.

- Per i dettagli sull'API, consulta DescribeEngineDefaultParameters AWS CLICommand Reference.

`describe-event-categories`

Il seguente esempio di codice mostra come utilizzare`describe-event-categories`.

AWS CLI

Come descrivere le categorie di eventi

L'esempio `describe-event-categories` seguente recupera i dettagli sulle categorie di eventi per tutte le origini di eventi disponibili.

```
aws rds describe-event-categories
```

Output:

```
{
    "EventCategoriesMapList": [
        {
            "SourceType": "db-instance",
            "EventCategories": [
                "deletion",
                "read replica",
                "failover",
                "restoration",
                "maintenance",
                "low storage",
```

```
                "configuration change",
                "backup",
                "creation",
                "availability",
                "recovery",
                "failure",
                "backtrack",
                "notification"
            ]
        },
        {
            "SourceType": "db-security-group",
            "EventCategories": [
                "configuration change",
                "failure"
            ]
        },
        {
            "SourceType": "db-parameter-group",
            "EventCategories": [
                "configuration change"
            ]
        },
        {
            "SourceType": "db-snapshot",
            "EventCategories": [
                "deletion",
                "creation",
                "restoration",
                "notification"
            ]
        },
        {
            "SourceType": "db-cluster",
            "EventCategories": [
                "failover",
                "failure",
                "notification"
            ]
        },
        {
            "SourceType": "db-cluster-snapshot",
            "EventCategories": [
                "backup"
```

```
            ]
        }
    ]
}
```

- Per i dettagli sull'API, consulta DescribeEventCategories AWS CLICommand Reference.

`describe-event-subscriptions`

Il seguente esempio di codice mostra come utilizzare`describe-event-subscriptions`.

AWS CLI

Come descrivere le sottoscrizioni a eventi

Questo esempio descrive tutti gli abbonamenti agli eventi Amazon RDS per l'account corrente AWS .

```
aws rds describe-event-subscriptions
```

Output:

```
{
    "EventSubscriptionsList": [
        {
            "EventCategoriesList": [
                "backup",
                "recovery"
            ],
            "Enabled": true,
            "EventSubscriptionArn": "arn:aws:rds:us-east-1:123456789012:es:my-
instance-events",
            "Status": "creating",
            "SourceType": "db-instance",
            "CustomerAwsId": "123456789012",
            "SubscriptionCreationTime": "2018-07-31 23:22:01.893",
            "CustSubscriptionId": "my-instance-events",
            "SnsTopicArn": "arn:aws:sns:us-east-1:123456789012:interesting-events"
        },
        ...some output truncated...
    ]
```

```
}
```

- Per i dettagli sulle API, consulta AWS CLI Command DescribeEventSubscriptionsReference.

`describe-events`

Il seguente esempio di codice mostra come utilizzare`describe-events`.

AWS CLI

Come descrivere gli eventi

L'esempio `describe-events` seguente recupera i dettagli per gli eventi che si sono verificati per l'istanza database specificata.

```
aws rds describe-events \
    --source-identifier test-instance \
    --source-type db-instance
```

Output:

```
{
    "Events": [
        {
            "SourceType": "db-instance",
            "SourceIdentifier": "test-instance",
            "EventCategories": [
                "backup"
            ],
            "Message": "Backing up DB instance",
            "Date": "2018-07-31T23:09:23.983Z",
            "SourceArn": "arn:aws:rds:us-east-1:123456789012:db:test-instance"
        },
        {
            "SourceType": "db-instance",
            "SourceIdentifier": "test-instance",
            "EventCategories": [
                "backup"
            ],
            "Message": "Finished DB Instance backup",
            "Date": "2018-07-31T23:15:13.049Z",
```

```
            "SourceArn": "arn:aws:rds:us-east-1:123456789012:db:test-instance"
        }
    ]
}
```

- Per i dettagli sull'API, consulta DescribeEvents AWS CLICommand Reference.

`describe-export-tasks`

Il seguente esempio di codice mostra come utilizzare`describe-export-tasks`.

AWS CLI

Come descrivere le attività di esportazione degli snapshot

L'esempio `describe-export-tasks` seguente restituisce informazioni sulle esportazioni di snapshot in Amazon S3.

```
aws rds describe-export-tasks
```

Output:

```
{
    "ExportTasks": [
        {
            "ExportTaskIdentifier": "test-snapshot-export",
            "SourceArn": "arn:aws:rds:us-west-2:123456789012:snapshot:test-
snapshot",
            "SnapshotTime": "2020-03-02T18:26:28.163Z",
            "TaskStartTime": "2020-03-02T18:57:56.896Z",
            "TaskEndTime": "2020-03-02T19:10:31.985Z",
            "S3Bucket": "amzn-s3-demo-bucket",
            "S3Prefix": "",
            "IamRoleArn": "arn:aws:iam::123456789012:role/service-role/ExportRole",
            "KmsKeyId": "arn:aws:kms:us-west-2:123456789012:key/
abcd0000-7fca-4128-82f2-aabbccddeeff",
            "Status": "COMPLETE",
            "PercentProgress": 100,
            "TotalExtractedDataInGB": 0
        },
        {
            "ExportTaskIdentifier": "my-s3-export",
```

```
            "SourceArn": "arn:aws:rds:us-west-2:123456789012:snapshot:db5-snapshot-
test",
            "SnapshotTime": "2020-03-27T20:48:42.023Z",
            "S3Bucket": "amzn-s3-demo-bucket",
            "S3Prefix": "",
            "IamRoleArn": "arn:aws:iam::123456789012:role/service-role/ExportRole",
            "KmsKeyId": "arn:aws:kms:us-west-2:123456789012:key/
abcd0000-7fca-4128-82f2-aabbccddeeff",
            "Status": "STARTING",
            "PercentProgress": 0,
            "TotalExtractedDataInGB": 0
        }
    ]
}
```

Per ulteriori informazioni, consulta Monitoraggio delle esportazioni di snapshot nella Guida per l'utente di Amazon RDS.

- Per i dettagli sull'API, consulta DescribeExportTasks AWS CLICommand Reference.

`describe-global-clusters`

Il seguente esempio di codice mostra come utilizzare`describe-global-clusters`.

AWS CLI

Come descrivere i cluster di database globali

L'`describe-global-clusters`esempio seguente elenca i cluster DB globali Aurora nella regione corrente. AWS

```
aws rds describe-global-clusters
```

Output:

```
{
    "GlobalClusters": [
        {
            "GlobalClusterIdentifier": "myglobalcluster",
            "GlobalClusterResourceId": "cluster-f5982077e3b5aabb",
            "GlobalClusterArn": "arn:aws:rds::123456789012:global-
cluster:myglobalcluster",
```

```
            "Status": "available",
            "Engine": "aurora-mysql",
            "EngineVersion": "5.7.mysql_aurora.2.07.2",
            "StorageEncrypted": false,
            "DeletionProtection": false,
            "GlobalClusterMembers": []
        }
    ]
}
```

Per ulteriori informazioni, consulta Utilizzo dei database globali Amazon Aurora nella Guida per l'utente di Amazon Aurora.

- Per i dettagli sull'API, consulta AWS CLI Command DescribeGlobalClustersReference.

`describe-option-group-options`

Il seguente esempio di codice mostra come utilizzare`describe-option-group-options`.

AWS CLI

Come descrivere tutte le opzioni disponibili

L'esempio `describe-option-group-options` seguente elenca due opzioni per un'istanza di Oracle Database 19c.

```
aws rds describe-option-group-options \
    --engine-name oracle-ee \
    --major-engine-version 19 \
    --max-items 2
```

Output:

```
{
    "OptionGroupOptions": [
        {
            "Name": "APEX",
            "Description": "Oracle Application Express Runtime Environment",
            "EngineName": "oracle-ee",
            "MajorEngineVersion": "19",
            "MinimumRequiredMinorEngineVersion": "0.0.0.ru-2019-07.rur-2019-07.r1",
            "PortRequired": false,
```

```
            "OptionsDependedOn": [],
            "OptionsConflictsWith": [],
            "Persistent": false,
            "Permanent": false,
            "RequiresAutoMinorEngineVersionUpgrade": false,
            "VpcOnly": false,
            "SupportsOptionVersionDowngrade": false,
            "OptionGroupOptionSettings": [],
            "OptionGroupOptionVersions": [
                {
                    "Version": "19.1.v1",
                    "IsDefault": true
                },
                {
                    "Version": "19.2.v1",
                    "IsDefault": false
                }
            ]
        },
        {
            "Name": "APEX-DEV",
            "Description": "Oracle Application Express Development Environment",
            "EngineName": "oracle-ee",
            "MajorEngineVersion": "19",
            "MinimumRequiredMinorEngineVersion": "0.0.0.ru-2019-07.rur-2019-07.r1",
            "PortRequired": false,
            "OptionsDependedOn": [
                "APEX"
            ],
            "OptionsConflictsWith": [],
            "Persistent": false,
            "Permanent": false,
            "RequiresAutoMinorEngineVersionUpgrade": false,
            "VpcOnly": false,
            "OptionGroupOptionSettings": []
        }
    ],
    "NextToken": "eyJNYXJrZXIiOiBudWxsLCAiYm90b190cnVuY2F0ZV9hbW91bnQiOiAyfQ=="
}
```

Per ulteriori informazioni, consulta Generazione di un elenco delle opzioni e delle impostazioni delle opzioni per un gruppo di opzioni nella Guida per l'utente di Amazon RDS.

- Per i dettagli sull'API, consulta DescribeOptionGroupOptions AWS CLICommand Reference.

`describe-option-groups`

Il seguente esempio di codice mostra come utilizzare`describe-option-groups`.

AWS CLI

Come descrivere i gruppi di opzioni disponibili

L'esempio `describe-option-groups` seguente elenca i gruppi di opzioni per un'istanza di Oracle Database 19c.

```
aws rds describe-option-groups \
    --engine-name oracle-ee \
    --major-engine-version 19
```

Output:

```
{
    "OptionGroupsList": [
        {
            "OptionGroupName": "default:oracle-ee-19",
            "OptionGroupDescription": "Default option group for oracle-ee 19",
            "EngineName": "oracle-ee",
            "MajorEngineVersion": "19",
            "Options": [],
            "AllowsVpcAndNonVpcInstanceMemberships": true,
            "OptionGroupArn": "arn:aws:rds:us-west-1:111122223333:og:default:oracle-
ee-19"
        }
    ]
}
```

Per ulteriori informazioni, consulta Generazione di un elenco delle opzioni e delle impostazioni delle opzioni per un gruppo di opzioni nella Guida per l'utente di Amazon RDS.

- Per i dettagli sull'API, consulta DescribeOptionGroups AWS CLICommand Reference.

`describe-orderable-db-instance-options`

Il seguente esempio di codice mostra come utilizzare`describe-orderable-db-instance-options`.

AWS CLI

Come descrivere le opzioni ordinabili delle istanze database

L'esempio `describe-orderable-db-instance-options` seguente recupera i dettagli delle opzioni ordinabili per le istanze database che eseguono il motore di database MySQL.

```
aws rds describe-orderable-db-instance-options \
    --engine mysql
```

Output:

```
{
    "OrderableDBInstanceOptions": [
        {
            "MinStorageSize": 5,
            "ReadReplicaCapable": true,
            "MaxStorageSize": 6144,
            "AvailabilityZones": [
                {
                    "Name": "us-east-1a"
                },
                {
                    "Name": "us-east-1b"
                },
                {
                    "Name": "us-east-1c"
                },
                {
                    "Name": "us-east-1d"
                }
            ],
            "SupportsIops": false,
            "AvailableProcessorFeatures": [],
            "MultiAZCapable": true,
            "DBInstanceClass": "db.m1.large",
            "Vpc": true,
            "StorageType": "gp2",
            "LicenseModel": "general-public-license",
            "EngineVersion": "5.5.46",
            "SupportsStorageEncryption": false,
            "SupportsEnhancedMonitoring": true,
            "Engine": "mysql",
```

```
            "SupportsIAMDatabaseAuthentication": false,
            "SupportsPerformanceInsights": false
        }
    ]
    ...some output truncated...
}
```

- Per i dettagli sull'API, consulta DescribeOrderableDBInstanceOptions in AWS CLI Command Reference.

`describe-pending-maintenance-actions`

Il seguente esempio di codice mostra come utilizzare`describe-pending-maintenance-actions`.

AWS CLI

Come elencare le risorse con almeno un'azione di manutenzione in sospeso

L'esempio `describe-pending-maintenance-actions` seguente mostra l'azione di manutenzione in sospeso per un'istanza database.

```
aws rds describe-pending-maintenance-actions
```

Output:

```
{
    "PendingMaintenanceActions": [
        {
            "ResourceIdentifier": "arn:aws:rds:us-
west-2:123456789012:cluster:global-db1-cl1",
            "PendingMaintenanceActionDetails": [
                {
                    "Action": "system-update",
                    "Description": "Upgrade to Aurora PostgreSQL 2.4.2"
                }
            ]
        }
    ]
}
```

Per ulteriori informazioni, consulta Manutenzione di un'istanza database nella Guida per l'utente di Amazon RDS.

- Per i dettagli sull'API, consulta DescribePendingMaintenanceActions AWS CLICommand Reference.

`describe-reserved-db-instances-offerings`

Il seguente esempio di codice mostra come utilizzare`describe-reserved-db-instances-offerings`.

AWS CLI

Come descrivere le offerte di istanze database riservate

L'esempio `describe-reserved-db-instances-offerings` seguente recupera i dettagli sulle opzioni di istanze database riservate per `oracle`.

```
aws rds describe-reserved-db-instances-offerings \
    --product-description oracle
```

Output:

```
{
    "ReservedDBInstancesOfferings": [
        {
            "CurrencyCode": "USD",
            "UsagePrice": 0.0,
            "ProductDescription": "oracle-se2(li)",
            "ReservedDBInstancesOfferingId": "005bdee3-9ef4-4182-aa0c-58ef7cb6c2f8",
            "MultiAZ": true,
            "DBInstanceClass": "db.m4.xlarge",
            "OfferingType": "Partial Upfront",
            "RecurringCharges": [
                {
                    "RecurringChargeAmount": 0.594,
                    "RecurringChargeFrequency": "Hourly"
                }
            ],
            "FixedPrice": 4089.0,
            "Duration": 31536000
        },
```

```
    ...some output truncated...
}
```

- Per i dettagli sull'API, consulta DescribeReservedDbInstancesOfferings AWS CLICommand Reference.

`describe-reserved-db-instances`

Il seguente esempio di codice mostra come utilizzare`describe-reserved-db-instances`.

AWS CLI

Come descrivere le istanze database riservate

L'`describe-reserved-db-instances`esempio seguente recupera i dettagli su tutte le istanze DB riservate nell'account corrente AWS .

```
aws rds describe-reserved-db-instances
```

Output:

```
{
    "ReservedDBInstances": [
        {
            "ReservedDBInstanceId": "myreservedinstance",
            "ReservedDBInstancesOfferingId": "12ab34cd-59af-4b2c-a660-1abcdef23456",
            "DBInstanceClass": "db.t3.micro",
            "StartTime": "2020-06-01T13:44:21.436Z",
            "Duration": 31536000,
            "FixedPrice": 0.0,
            "UsagePrice": 0.0,
            "CurrencyCode": "USD",
            "DBInstanceCount": 1,
            "ProductDescription": "sqlserver-ex(li)",
            "OfferingType": "No Upfront",
            "MultiAZ": false,
            "State": "payment-pending",
            "RecurringCharges": [
                {
                    "RecurringChargeAmount": 0.014,
                    "RecurringChargeFrequency": "Hourly"
                }
```

```
            ],
            "ReservedDBInstanceArn": "arn:aws:rds:us-
west-2:123456789012:ri:myreservedinstance",
            "LeaseId": "a1b2c3d4-6b69-4a59-be89-5e11aa446666"
        }
    ]
}
```

Per ulteriori informazioni, consulta Istanze database riservate per Amazon RDS nella Guida per l'utente di Amazon RDS.

- Per i dettagli sull'API, consulta AWS CLI Command DescribeReservedDbInstancesReference.

`describe-source-regions`

Il seguente esempio di codice mostra come utilizzare`describe-source-regions`.

AWS CLI

Come descrivere le Regioni di origine

L'`describe-source-regions`esempio seguente recupera i dettagli su tutte le AWS regioni di origine. Mostra inoltre che i backup automatici possono essere replicati solo dagli Stati Uniti occidentali (Oregon) alla AWS regione di destinazione, Stati Uniti orientali (Virginia settentrionale).

```
aws rds describe-source-regions \
    --region us-east-1
```

Output:

```
{
    "SourceRegions": [
        {
            "RegionName": "af-south-1",
            "Endpoint": "https://rds.af-south-1.amazonaws.com",
            "Status": "available",
            "SupportsDBInstanceAutomatedBackupsReplication": false
        },
        {
            "RegionName": "ap-east-1",
            "Endpoint": "https://rds.ap-east-1.amazonaws.com",
            "Status": "available",
            "SupportsDBInstanceAutomatedBackupsReplication": false
```

```
        },
        {
            "RegionName": "ap-northeast-1",
            "Endpoint": "https://rds.ap-northeast-1.amazonaws.com",
            "Status": "available",
            "SupportsDBInstanceAutomatedBackupsReplication": true
        },
        {
            "RegionName": "ap-northeast-2",
            "Endpoint": "https://rds.ap-northeast-2.amazonaws.com",
            "Status": "available",
            "SupportsDBInstanceAutomatedBackupsReplication": true
        },
        {
            "RegionName": "ap-northeast-3",
            "Endpoint": "https://rds.ap-northeast-3.amazonaws.com",
            "Status": "available",
            "SupportsDBInstanceAutomatedBackupsReplication": false
        },
        {
            "RegionName": "ap-south-1",
            "Endpoint": "https://rds.ap-south-1.amazonaws.com",
            "Status": "available",
            "SupportsDBInstanceAutomatedBackupsReplication": true
        },
        {
            "RegionName": "ap-southeast-1",
            "Endpoint": "https://rds.ap-southeast-1.amazonaws.com",
            "Status": "available",
            "SupportsDBInstanceAutomatedBackupsReplication": true
        },
        {
            "RegionName": "ap-southeast-2",
            "Endpoint": "https://rds.ap-southeast-2.amazonaws.com",
            "Status": "available",
            "SupportsDBInstanceAutomatedBackupsReplication": true
        },
        {
            "RegionName": "ap-southeast-3",
            "Endpoint": "https://rds.ap-southeast-3.amazonaws.com",
            "Status": "available",
            "SupportsDBInstanceAutomatedBackupsReplication": false
        },
        {
```

```
            "RegionName": "ca-central-1",
            "Endpoint": "https://rds.ca-central-1.amazonaws.com",
            "Status": "available",
            "SupportsDBInstanceAutomatedBackupsReplication": true
        },
        {
            "RegionName": "eu-north-1",
            "Endpoint": "https://rds.eu-north-1.amazonaws.com",
            "Status": "available",
            "SupportsDBInstanceAutomatedBackupsReplication": true
        },
        {
            "RegionName": "eu-south-1",
            "Endpoint": "https://rds.eu-south-1.amazonaws.com",
            "Status": "available",
            "SupportsDBInstanceAutomatedBackupsReplication": false
        },
        {
            "RegionName": "eu-west-1",
            "Endpoint": "https://rds.eu-west-1.amazonaws.com",
            "Status": "available",
            "SupportsDBInstanceAutomatedBackupsReplication": true
        },
        {
            "RegionName": "eu-west-2",
            "Endpoint": "https://rds.eu-west-2.amazonaws.com",
            "Status": "available",
            "SupportsDBInstanceAutomatedBackupsReplication": true
        },
        {
            "RegionName": "eu-west-3",
            "Endpoint": "https://rds.eu-west-3.amazonaws.com",
            "Status": "available",
            "SupportsDBInstanceAutomatedBackupsReplication": true
        },
        {
            "RegionName": "me-central-1",
             "Endpoint": "https://rds.me-central-1.amazonaws.com",
             "Status": "available",
             "SupportsDBInstanceAutomatedBackupsReplication": false
        },
        {
            "RegionName": "me-south-1",
            "Endpoint": "https://rds.me-south-1.amazonaws.com",
```

```
            "Status": "available",
            "SupportsDBInstanceAutomatedBackupsReplication": false
        },
        {
            "RegionName": "sa-east-1",
            "Endpoint": "https://rds.sa-east-1.amazonaws.com",
            "Status": "available",
            "SupportsDBInstanceAutomatedBackupsReplication": true
        },
        {
            "RegionName": "us-east-2",
            "Endpoint": "https://rds.us-east-2.amazonaws.com",
            "Status": "available",
            "SupportsDBInstanceAutomatedBackupsReplication": true
        },
        {
            "RegionName": "us-west-1",
            "Endpoint": "https://rds.us-west-1.amazonaws.com",
            "Status": "available",
            "SupportsDBInstanceAutomatedBackupsReplication": true
        },
        {
            "RegionName": "us-west-2",
           "Endpoint": "https://rds.us-west-2.amazonaws.com",
            "Status": "available",
            "SupportsDBInstanceAutomatedBackupsReplication": true
        }
    ]
}
```

Per ulteriori informazioni, consulta Ricerca di informazioni sui backup replicati nella Guida per l'utente di Amazon RDS.

- Per i dettagli sull'API, consulta DescribeSourceRegionsCommand Reference.AWS CLI

`describe-valid-db-instance-modifications`

Il seguente esempio di codice mostra come utilizzare`describe-valid-db-instance-modifications`.

AWS CLI

Come descrivere le modifiche valide per un'istanza database

L'esempio `describe-valid-db-instance-modifications` seguente recupera i dettagli delle modifiche valide per l'istanza database specificata.

```
aws rds describe-valid-db-instance-modifications \
    --db-instance-identifier test-instance
```

Output:

```
{
    "ValidDBInstanceModificationsMessage": {
        "ValidProcessorFeatures": [],
        "Storage": [
            {
                "StorageSize": [
                    {
                        "Step": 1,
                        "To": 20,
                        "From": 20
                    },
                    {
                        "Step": 1,
                        "To": 6144,
                        "From": 22
                    }
                ],
                "ProvisionedIops": [
                    {
                        "Step": 1,
                        "To": 0,
                        "From": 0
                    }
                ],
                "IopsToStorageRatio": [
                    {
                        "To": 0.0,
                        "From": 0.0
                    }
                ],
                "StorageType": "gp2"
            },
            {
                "StorageSize": [
                    {
```

```
                        "Step": 1,
                        "To": 6144,
                        "From": 100
                    }
                ],
                "ProvisionedIops": [
                    {
                        "Step": 1,
                        "To": 40000,
                        "From": 1000
                    }
                ],
                "IopsToStorageRatio": [
                    {
                        "To": 50.0,
                        "From": 1.0
                    }
                ],
                "StorageType": "io1"
            },
            {
                "StorageSize": [
                    {
                        "Step": 1,
                        "To": 20,
                        "From": 20
                    },
                    {
                        "Step": 1,
                        "To": 3072,
                        "From": 22
                    }
                ],
                "ProvisionedIops": [
                    {
                        "Step": 1,
                        "To": 0,
                        "From": 0
                    }
                ],
                "IopsToStorageRatio": [
                    {
                        "To": 0.0,
                        "From": 0.0
```

```
                    }
                ],
                "StorageType": "magnetic"
            }
        ]
    }
}
```

- Per i dettagli sull'API, consulta DescribeValidDbInstanceModifications AWS CLICommand Reference.

`download-db-log-file-portion`

Il seguente esempio di codice mostra come utilizzare`download-db-log-file-portion`.

AWS CLI

Esempio 1: come scaricare la parte più recente di un file di log database

L'esempio `download-db-log-file-portion` seguente scarica solo la parte più recente del file di log e la salva in un file locale denominato `tail.txt`.

```
aws rds download-db-log-file-portion \
    --db-instance-identifier test-instance \
    --log-file-name log.txt \
    --output text > tail.txt
```

Il file salvato potrebbe contenere righe vuote. Appaiono alla fine di ogni parte del file di log durante il download.

Esempio 2: come scaricare un intero file di log database

L'esempio `download-db-log-file-portion` seguente scarica l'intero file di log utilizzando il parametro `--starting-token 0` e salva l'output in un file locale denominato `full.txt`.

```
aws rds download-db-log-file-portion \
    --db-instance-identifier test-instance \
    --log-file-name log.txt \
    --starting-token 0 \
    --output text > full.txt
```

Il file salvato potrebbe contenere righe vuote. Appaiono alla fine di ogni parte del file di log durante il download.

- Per i dettagli sull'API, consulta DownloadDbLogFilePortion AWS CLICommand Reference.

`generate-auth-token`

Il seguente esempio di codice mostra come utilizzare`generate-auth-token`.

AWS CLI

Come generare un token di autenticazione

L'esempio `generate-db-auth-token` seguente genera un token di autenticazione da utilizzare con l'autenticazione del database IAM.

```
aws rds generate-db-auth-token \
    --hostname aurmysql-test.cdgmuqiadpid.us-west-2.rds.amazonaws.com \
    --port 3306 \
    --region us-east-1 \
    --username jane_doe
```

Output:

```
aurmysql-test.cdgmuqiadpid.us-west-2.rds.amazonaws.com:3306/?
Action=connect&DBUser=jane_doe&X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-
Credential=AKIAIESZCNJ3OEXAMPLE%2F20180731%2Fus-east-1%2Frds-db%2Faws4_request&X-
Amz-Date=20180731T235209Z&X-Amz-Expires=900&X-Amz-SignedHeaders=host&X-Amz-
Signature=5a8753ebEXAMPLEa2c724e5667797EXAMPLE9d6ec6e3f427191fa41aeEXAMPLE
```

- Per i dettagli sull'API, consulta GenerateAuthToken AWS CLICommand Reference.

`generate-db-auth-token`

Il seguente esempio di codice mostra come utilizzare`generate-db-auth-token`.

AWS CLI

Come generare un token di autenticazione IAM

L'esempio `generate-db-auth-token` seguente genera un token di autenticazione IAM per connettersi a un database.

```
aws rds generate-db-auth-token \
    --hostname mydb.123456789012.us-east-1.rds.amazonaws.com \
    --port 3306 \
    --region us-east-1 \
    --username db_user
```

Output:

```
mydb.123456789012.us-east-1.rds.amazonaws.com:3306/?
Action=connect&DBUser=db_user&X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-
Credential=AKIAIEXAMPLE%2Fus-east-1%2Frds-db%2Faws4_request&X-Amz-
Date=20210123T011543Z&X-Amz-Expires=900&X-Amz-SignedHeaders=host&X-Amz-
Signature=88987EXAMPLE1EXAMPLE2EXAMPLE3EXAMPLE4EXAMPLE5EXAMPLE6
```

Per ulteriori informazioni, consulta Connessione all'istanza database utilizzando l'autenticazione IAM nella Guida per l'utente di Amazon RDS e Connessione al cluster di database utilizzando l'autenticazione IAM nella Guida per l'utente di Amazon Aurora.

- Per i dettagli sull'API, consulta GenerateDbAuthToken AWS CLICommand Reference.

`list-tags-for-resource`

Il seguente esempio di codice mostra come utilizzare`list-tags-for-resource`.

AWS CLI

Come elencare i tag su una risorsa Amazon RDS

L'esempio `list-tags-for-resource` seguente elenca tutti i tag su un'istanza database.

```
aws rds list-tags-for-resource \
    --resource-name arn:aws:rds:us-east-1:123456789012:db:orcl1
```

Output:

```
{
    "TagList": [
        {
            "Key": "Environment",
            "Value": "test"
```

```
        },
        {
            "Key": "Name",
            "Value": "MyDatabase"
        }
    ]
}
```

Per ulteriori informazioni, consulta la pagina Tagging delle risorse Amazon RDS nella Guida per l'utente di Amazon RDS.

- Per i dettagli sull'API, consulta ListTagsForResource AWS CLICommand Reference.

`modify-certificates`

Il seguente esempio di codice mostra come utilizzare`modify-certificates`.

AWS CLI

Per sovrascrivere temporaneamente il SSL/TLS certificato predefinito del sistema per le nuove istanze DB

L'`modify-certificates`esempio seguente sostituisce temporaneamente il certificato predefinito del sistema SSL/TLS per le nuove istanze DB.

```
aws rds modify-certificates \
    --certificate-identifier rds-ca-2019
```

Output:

```
{
    "Certificate": {
        "CertificateIdentifier": "rds-ca-2019",
        "CertificateType": "CA",
        "Thumbprint": "EXAMPLE123456789012",
        "ValidFrom": "2019-09-19T18:16:53Z",
        "ValidTill": "2024-08-22T17:08:50Z",
        "CertificateArn": "arn:aws:rds:us-east-1::cert:rds-ca-2019",
        "CustomerOverride": true,
        "CustomerOverrideValidTill": "2024-08-22T17:08:50Z"
    }
```

```
}
```

Per ulteriori informazioni, consulta Rotazione del SSL/TLS certificato nella Amazon RDS User Guide e Rotating your SSL/TLS certificate nella Amazon Aurora User Guide.

- Per i dettagli sull'API, consulta ModifyCertificatesCommand Reference.AWS CLI

`modify-current-db-cluster-capacity`

Il seguente esempio di codice mostra come utilizzare`modify-current-db-cluster-capacity`.

AWS CLI

Come dimensionare la capacità di un cluster di database Aurora serverless

L'esempio `modify-current-db-cluster-capacity` seguente scala la capacità di un cluster database Aurora serverless a 8.

```
aws rds modify-current-db-cluster-capacity \
    --db-cluster-identifier mydbcluster \
    --capacity 8
```

Output:

```
{
    "DBClusterIdentifier": "mydbcluster",
    "PendingCapacity": 8,
    "CurrentCapacity": 1,
    "SecondsBeforeTimeout": 300,
    "TimeoutAction": "ForceApplyCapacityChange"
}
```

Per ulteriori informazioni, consulta Dimensionamento manuale della capacità del cluster di database di Aurora Serverless v1 nella Guida per l'utente di Amazon Aurora.

- Per i dettagli sull'API, consulta ModifyCurrentDbClusterCapacity AWS CLICommand Reference.

`modify-db-cluster-endpoint`

Il seguente esempio di codice mostra come utilizzare`modify-db-cluster-endpoint`.

AWS CLI

Come modificare un endpoint del cluster di database personalizzato

L'esempio `modify-db-cluster-endpoint` seguente modifica l'endpoint del cluster di database personalizzato specificato.

```
aws rds modify-db-cluster-endpoint \
    --db-cluster-endpoint-identifier mycustomendpoint \
    --static-members dbinstance1 dbinstance2 dbinstance3
```

Output:

```
{
    "DBClusterEndpointIdentifier": "mycustomendpoint",
    "DBClusterIdentifier": "mydbcluster",
    "DBClusterEndpointResourceIdentifier": "cluster-endpoint-ANPAJ4AE5446DAEXAMPLE",
    "Endpoint": "mycustomendpoint.cluster-custom-cnpexample.us-
east-1.rds.amazonaws.com",
    "Status": "modifying",
    "EndpointType": "CUSTOM",
    "CustomEndpointType": "READER",
    "StaticMembers": [
        "dbinstance1",
        "dbinstance2",
        "dbinstance3"
    ],
    "ExcludedMembers": [],
    "DBClusterEndpointArn": "arn:aws:rds:us-east-1:123456789012:cluster-
endpoint:mycustomendpoint"
}
```

Per ulteriori informazioni, consulta Gestione delle connessioni Amazon Aurora nella Guida per l'utente di Amazon Aurora.

- Per i dettagli sull'API, consulta ModifyDbClusterEndpoint AWS CLICommand Reference.

`modify-db-cluster-parameter-group`

Il seguente esempio di codice mostra come utilizzare`modify-db-cluster-parameter-group`.

AWS CLI

Come modificare i parametri in un gruppo di parametri del cluster di database

L'esempio `modify-db-cluster-parameter-group` seguente modifica i valori dei parametri in un gruppo di parametri del cluster di database.

```
aws rds modify-db-cluster-parameter-group \
    --db-cluster-parameter-group-name mydbclusterpg \
    --
parameters "ParameterName=server_audit_logging,ParameterValue=1,ApplyMethod=immediate"
 \

  "ParameterName=server_audit_logs_upload,ParameterValue=1,ApplyMethod=immediate"
```

Output:

```
{
    "DBClusterParameterGroupName": "mydbclusterpg"
}
```

Per ulteriori informazioni, consulta Uso di gruppi di parametri database e gruppi di parametri cluster database nella Guida per l'utente di Amazon Aurora.

- Per i dettagli sull'API, consulta ModifyDbClusterParameterGroup AWS CLICommand Reference.

`modify-db-cluster-snapshot-attribute`

Il seguente esempio di codice mostra come utilizzare`modify-db-cluster-snapshot-attribute`.

AWS CLI

Come modificare l'attributo dello snapshot del cluster di database

L'esempio `modify-db-cluster-snapshot-attribute` seguente apporta modifiche all'attributo snapshot del cluster di database specificato.

```
aws rds modify-db-cluster-snapshot-attribute \
    --db-cluster-snapshot-identifier myclustersnapshot \
    --attribute-name restore \
```

```
    --values-to-add 123456789012
```

Output:

```
{
    "DBClusterSnapshotAttributesResult": {
        "DBClusterSnapshotIdentifier": "myclustersnapshot",
        "DBClusterSnapshotAttributes": [
            {
                "AttributeName": "restore",
                "AttributeValues": [
                    "123456789012"
                ]
            }
        ]
    }
}
```

Per ulteriori informazioni, consulta Ripristino da uno snapshot cluster database nella Guida per l'utente di Amazon Aurora.

- Per i dettagli sull'API, consulta ModifyDbClusterSnapshotAttribute AWS CLICommand Reference.

`modify-db-cluster`

Il seguente esempio di codice mostra come utilizzare`modify-db-cluster`.

AWS CLI

Esempio 1: come modificare un cluster di database

L'esempio `modify-db-cluster` seguente modifica la password dell'utente master per il cluster di database denominato `cluster-2` e imposta il periodo di conservazione del backup su 14 giorni. Il parametro `--apply-immediately` fa in modo che le modifiche vengano apportate immediatamente, invece di attendere la successiva finestra di manutenzione.

```
aws rds modify-db-cluster \
    --db-cluster-identifier cluster-2 \
    --backup-retention-period 14 \
    --master-user-password newpassword99 \
    --apply-immediately
```

Output:

```
{
    "DBCluster": {
        "AllocatedStorage": 1,
        "AvailabilityZones": [
            "eu-central-1b",
            "eu-central-1c",
            "eu-central-1a"
        ],
        "BackupRetentionPeriod": 14,
        "DatabaseName": "",
        "DBClusterIdentifier": "cluster-2",
        "DBClusterParameterGroup": "default.aurora5.6",
        "DBSubnetGroup": "default-vpc-2305ca49",
        "Status": "available",
        "EarliestRestorableTime": "2020-06-03T02:07:29.637Z",
        "Endpoint": "cluster-2.cluster-############.eu-central-1.rds.amazonaws.com",
        "ReaderEndpoint": "cluster-2.cluster-ro-############.eu-
central-1.rds.amazonaws.com",
        "MultiAZ": false,
        "Engine": "aurora",
        "EngineVersion": "5.6.10a",
        "LatestRestorableTime": "2020-06-04T15:11:25.748Z",
        "Port": 3306,
        "MasterUsername": "admin",
        "PreferredBackupWindow": "01:55-02:25",
        "PreferredMaintenanceWindow": "thu:21:14-thu:21:44",
        "ReadReplicaIdentifiers": [],
        "DBClusterMembers": [
            {
                "DBInstanceIdentifier": "cluster-2-instance-1",
                "IsClusterWriter": true,
                "DBClusterParameterGroupStatus": "in-sync",
                "PromotionTier": 1
            }
        ],
        "VpcSecurityGroups": [
            {
                "VpcSecurityGroupId": "sg-20a5c047",
                "Status": "active"
            }
        ],
        "HostedZoneId": "Z1RLNU0EXAMPLE",
```

```
        "StorageEncrypted": true,
        "KmsKeyId": "arn:aws:kms:eu-central-1:123456789012:key/
d1bd7c8f-5cdb-49ca-8a62-a1b2c3d4e5f6",
        "DbClusterResourceId": "cluster-AGJ7XI77XVIS6FUXHU1EXAMPLE",
        "DBClusterArn": "arn:aws:rds:eu-central-1:123456789012:cluster:cluster-2",
        "AssociatedRoles": [],
        "IAMDatabaseAuthenticationEnabled": false,
        "ClusterCreateTime": "2020-04-03T14:44:02.764Z",
        "EngineMode": "provisioned",
        "DeletionProtection": false,
        "HttpEndpointEnabled": false,
        "CopyTagsToSnapshot": true,
        "CrossAccountClone": false,
        "DomainMemberships": []
    }
}
```

Per ulteriori informazioni, consulta Modifica di un cluster database Amazon Aurora nella Guida per l'utente di Amazon Aurora.

Esempio 2: come associare il gruppo di sicurezza VPC a un cluster di database

L'esempio `modify-db-instance` seguente associa un gruppo di sicurezza VPC specifico e rimuove i gruppi di sicurezza database da un cluster di database.

```
aws rds modify-db-cluster \
    --db-cluster-identifier dbName \
    --vpc-security-group-ids sg-ID
```

Output:

```
{
    "DBCluster": {
        "AllocatedStorage": 1,
        "AvailabilityZones": [
            "us-west-2c",
            "us-west-2b",
            "us-west-2a"
        ],
        "BackupRetentionPeriod": 1,
        "DBClusterIdentifier": "dbName",
        "DBClusterParameterGroup": "default.aurora-mysql8.0",
```

```
        "DBSubnetGroup": "default",
        "Status": "available",
        "EarliestRestorableTime": "2024-02-15T01:12:13.966000+00:00",
        "Endpoint": "dbName.cluster-abcdefghji.us-west-2.rds.amazonaws.com",
        "ReaderEndpoint": "dbName.cluster-ro-abcdefghji.us-
west-2.rds.amazonaws.com",
        "MultiAZ": false,
        "Engine": "aurora-mysql",
        "EngineVersion": "8.0.mysql_aurora.3.04.1",
        "LatestRestorableTime": "2024-02-15T02:25:33.696000+00:00",
        "Port": 3306,
        "MasterUsername": "admin",
        "PreferredBackupWindow": "10:59-11:29",
        "PreferredMaintenanceWindow": "thu:08:54-thu:09:24",
        "ReadReplicaIdentifiers": [],
        "DBClusterMembers": [
            {
                "DBInstanceIdentifier": "dbName-instance-1",
                "IsClusterWriter": true,
                "DBClusterParameterGroupStatus": "in-sync",
                "PromotionTier": 1
            }
        ],
        "VpcSecurityGroups": [
            {
                "VpcSecurityGroupId": "sg-ID",
                "Status": "active"
            }
        ],
        ...output omitted...
    }
}
```

Per ulteriori informazioni, consulta Controllo dell'accesso con i gruppi di sicurezza nella Guida per l'utente di Amazon Aurora.

- Per i dettagli sull'API, consulta ModifyDbCluster AWS CLICommand Reference.

`modify-db-instance`

Il seguente esempio di codice mostra come utilizzare`modify-db-instance`.

AWS CLI

Esempio 1: come modificare un'istanza database

L'esempio `modify-db-instance` seguente associa un gruppo di opzioni e un gruppo di parametri a un'istanza database Microsoft SQL Server compatibile. Il parametro `--apply-immediately` fa sì che l'opzione e i gruppi di parametri vengano associati immediatamente, anziché attendere la successiva finestra di manutenzione.

```
aws rds modify-db-instance \
    --db-instance-identifier database-2 \
    --option-group-name test-se-2017 \
    --db-parameter-group-name test-sqlserver-se-2017 \
    --apply-immediately
```

Output:

```
{
    "DBInstance": {
        "DBInstanceIdentifier": "database-2",
        "DBInstanceClass": "db.r4.large",
        "Engine": "sqlserver-se",
        "DBInstanceStatus": "available",

        ...output omitted...

        "DBParameterGroups": [
            {
                "DBParameterGroupName": "test-sqlserver-se-2017",
                "ParameterApplyStatus": "applying"
            }
        ],
        "AvailabilityZone": "us-west-2d",

        ...output omitted...

        "MultiAZ": true,
        "EngineVersion": "14.00.3281.6.v1",
        "AutoMinorVersionUpgrade": false,
        "ReadReplicaDBInstanceIdentifiers": [],
        "LicenseModel": "license-included",
        "OptionGroupMemberships": [
            {
```

```
                "OptionGroupName": "test-se-2017",
                "Status": "pending-apply"
            }
        ],
        "CharacterSetName": "SQL_Latin1_General_CP1_CI_AS",
        "SecondaryAvailabilityZone": "us-west-2c",
        "PubliclyAccessible": true,
        "StorageType": "gp2",

        ...output omitted...

        "DeletionProtection": false,
        "AssociatedRoles": [],
        "MaxAllocatedStorage": 1000
    }
}
```

Per ulteriori informazioni, consulta Modifica di un'istanza database Amazon RDS nella Guida per l'utente di Amazon RDS.

Esempio 2: come associare un gruppo di sicurezza VPC a un'istanza database

L'esempio `modify-db-instance` seguente associa un gruppo di sicurezza VPC specifico e rimuove i gruppi di sicurezza database da un'istanza database:

```
aws rds modify-db-instance \
    --db-instance-identifier dbName \
    --vpc-security-group-ids sg-ID
```

Output:

```
{
"DBInstance": {
    "DBInstanceIdentifier": "dbName",
    "DBInstanceClass": "db.t3.micro",
    "Engine": "mysql",
    "DBInstanceStatus": "available",
    "MasterUsername": "admin",
    "Endpoint": {
        "Address": "dbName.abcdefghijk.us-west-2.rds.amazonaws.com",
        "Port": 3306,
        "HostedZoneId": "ABCDEFGHIJK1234"
```

```
    },
    "AllocatedStorage": 20,
    "InstanceCreateTime": "2024-02-15T00:37:58.793000+00:00",
    "PreferredBackupWindow": "11:57-12:27",
    "BackupRetentionPeriod": 7,
    "DBSecurityGroups": [],
    "VpcSecurityGroups": [
        {
            "VpcSecurityGroupId": "sg-ID",
            "Status": "active"
        }
    ],
    ... output omitted ...
    "MultiAZ": false,
    "EngineVersion": "8.0.35",
    "AutoMinorVersionUpgrade": true,
    "ReadReplicaDBInstanceIdentifiers": [],
    "LicenseModel": "general-public-license",

    ... output ommited ...
    }
}
```

Per ulteriori informazioni, consulta Controllo dell'accesso con i gruppi di sicurezza nella Guida per l'utente di Amazon RDS.

- Per i dettagli sull'API, consulta Modify DBInstance in AWS CLI Command Reference.

`modify-db-parameter-group`

Il seguente esempio di codice mostra come utilizzare`modify-db-parameter-group`.

AWS CLI

Come modificare un gruppo di parametri del database

L'esempio `modify-db-parameter-group` seguente modifica il valore del parametro `clr enabled` in un gruppo di parametri database. Il parametro `--apply-immediately` fa sì che il gruppo di parametri del database venga modificato immediatamente, anziché attendere la successiva finestra di manutenzione.

```
aws rds modify-db-parameter-group \
```

```
    --db-parameter-group-name test-sqlserver-se-2017 \
    --parameters "ParameterName='clr
 enabled',ParameterValue=1,ApplyMethod=immediate"
```

Output:

```
{
    "DBParameterGroupName": "test-sqlserver-se-2017"
}
```

Per ulteriori informazioni, consulta Modifica di parametri in un gruppo di parametri database nella Guida per l'utente di Amazon RDS.

- Per i dettagli sull'API, consulta Modify DBParameter Group in AWS CLI Command Reference.

`modify-db-proxy-endpoint`

Il seguente esempio di codice mostra come utilizzare`modify-db-proxy-endpoint`.

AWS CLI

Come modificare un endpoint proxy database per un database RDS

L'esempio `modify-db-proxy-endpoint` seguente modifica un endpoint proxy database `proxyEndpoint` per impostare il timeout di lettura su 65 secondi.

```
aws rds modify-db-proxy-endpoint \
    --db-proxy-endpoint-name proxyEndpoint \
    --cli-read-timeout 65
```

Output:

```
{
"DBProxyEndpoint":
    {
        "DBProxyEndpointName": "proxyEndpoint",
        "DBProxyEndpointArn": "arn:aws:rds:us-east-1:123456789012:db-proxy-
endpoint:prx-endpoint-0123a01b12345c0ab",
        "DBProxyName": "proxyExample",
        "Status": "available",
        "VpcId": "vpc-1234567",
        "VpcSecurityGroupIds": [
```

```
            "sg-1234"
        ],
        "VpcSubnetIds": [
            "subnetgroup1",
            "subnetgroup2"
        ],
        "Endpoint": "proxyEndpoint.endpoint.proxyExample-ab0cd1efghij.us-
east-1.rds.amazonaws.com",
        "CreatedDate": "2023-04-05T16:09:33.452000+00:00",
        "TargetRole": "READ_WRITE",
        "IsDefault": "false"
    }
}
```

Per ulteriori informazioni, consulta Modifica di un endpoint proxy nella Guida per l'utente di Amazon RDS e Modifica di un endpoint proxy nella Guida per l'utente di Amazon Aurora.

- Per i dettagli sull'API, consulta ModifyDbProxyEndpoint AWS CLICommand Reference.

`modify-db-proxy-target-group`

Il seguente esempio di codice mostra come utilizzare`modify-db-proxy-target-group`.

AWS CLI

Come modificare gli endpoint di un proxy database

L'esempio `modify-db-proxy-target-group` seguente modifica un gruppo di destinazione di proxy database per impostare il numero massimo di connessioni su 80% e il numero massimo di connessioni inattive su 10%.

```
aws rds modify-db-proxy-target-group \
    --target-group-name default \
    --db-proxy-name proxyExample \
    --connection-pool-config MaxConnectionsPercent=80,MaxIdleConnectionsPercent=10
```

Output:

```
{
"DBProxyTargetGroup":
    {
        "DBProxyName": "proxyExample",
```

```
        "TargetGroupName": "default",
        "TargetGroupArn": "arn:aws:rds:us-east-1:123456789012:target-group:prx-
tg-0123a01b12345c0ab",
        "IsDefault": true,
        "Status": "available",
        "ConnectionPoolConfig": {
            "MaxConnectionsPercent": 80,
            "MaxIdleConnectionsPercent": 10,
            "ConnectionBorrowTimeout": 120,
            "SessionPinningFilters": []
        },
        "CreatedDate": "2023-05-02T18:41:19.495000+00:00",
        "UpdatedDate": "2023-05-02T18:41:21.762000+00:00"
    }
}
```

Per ulteriori informazioni, consulta Modifica di un RDS Proxy nella Guida per l'utente di Amazon RDS e Modifica di un RDS Proxy nella Guida per l'utente di Amazon Aurora.

- Per i dettagli sull'API, consulta ModifyDbProxyTargetGroup AWS CLICommand Reference.

`modify-db-proxy`

Il seguente esempio di codice mostra come utilizzare`modify-db-proxy`.

AWS CLI

Come modificare un proxy database per un database RDS

L'esempio `modify-db-proxy` seguente modifica un proxy database denominato `proxyExample` per richiedere SSL per le sue connessioni.

```
aws rds modify-db-proxy \
    --db-proxy-name proxyExample \
    --require-tls
```

Output:

```
{
"DBProxy":
    {
        "DBProxyName": "proxyExample",
```

```
        "DBProxyArn": "arn:aws:rds:us-east-1:123456789012:db-
proxy:prx-0123a01b12345c0ab",
        "Status": "modifying"
        "EngineFamily": "PostgreSQL",
        "VpcId": "sg-1234567",
        "VpcSecurityGroupIds": [
            "sg-1234"
        ],
        "VpcSubnetIds": [
            "subnetgroup1",
            "subnetgroup2"
        ],
        "Auth": "[
            {
                "Description": "proxydescription1",
                "AuthScheme": "SECRETS",
                "SecretArn": "arn:aws:secretsmanager:us-
west-2:123456789123:secret:proxysecret1-Abcd1e",
                "IAMAuth": "DISABLED"
            }
        ]",
        "RoleArn": "arn:aws:iam::12345678912:role/ProxyPostgreSQLRole",
        "Endpoint": "proxyExample.proxy-ab0cd1efghij.us-east-1.rds.amazonaws.com",
        "RequireTLS": true,
        "IdleClientTimeout": 1800,
        "DebuggingLogging": false,
        "CreatedDate": "2023-04-05T16:09:33.452000+00:00",
        "UpdatedDate": "2023-04-13T01:49:38.568000+00:00"
    }
}
```

Per ulteriori informazioni, consulta Modifica di un RDS Proxy nella Guida per l'utente di Amazon RDS e Creazione di un RDS Proxy nella Guida per l'utente di Amazon Aurora.

- Per i dettagli sull'API, consulta ModifyDbProxy AWS CLICommand Reference.

`modify-db-shard-group`

Il seguente esempio di codice mostra come utilizzare`modify-db-shard-group`.

AWS CLI

Esempio 1: come modificare un gruppo di shard del database

L'esempio `modify-db-shard-group` seguente modifica la capacità massima di un gruppo di shard del database.

```
aws rds modify-db-shard-group \
    --db-shard-group-identifier my-db-shard-group \
    --max-acu 1000
```

Output:

```
{
    "DBShardGroups": [
        {
            "DBShardGroupResourceId": "shardgroup-a6e3a0226aa243e2ac6c7a1234567890",
            "DBShardGroupIdentifier": "my-db-shard-group",
            "DBClusterIdentifier": "my-sv2-cluster",
            "MaxACU": 768.0,
            "ComputeRedundancy": 0,
            "Status": "available",
            "PubliclyAccessible": false,
            "Endpoint": "my-sv2-cluster.limitless-cekycexample.us-
east-2.rds.amazonaws.com"
        }
    ]
}
```

Per ulteriori informazioni, consulta Cluster database Amazon Aurora nella Guida per l'utente di Amazon Aurora.

Esempio 2: come descrivere i gruppi di shard del database

L'esempio `describe-db-shard-groups` seguente recupera i dettagli dei gruppi di shard del database dopo l'esecuzione del comando `modify-db-shard-group`. La capacità massima del gruppo di shard DB `my-db-shard-group` è ora di 1000 unità ACUs di capacità Aurora ().

```
aws rds describe-db-shard-groups
```

Output:

```
{
    "DBShardGroups": [
        {
            "DBShardGroupResourceId": "shardgroup-7bb446329da94788b3f957746example",
```

```
            "DBShardGroupIdentifier": "limitless-test-shard-grp",
            "DBClusterIdentifier": "limitless-test-cluster",
            "MaxACU": 768.0,
            "ComputeRedundancy": 0,
            "Status": "available",
            "PubliclyAccessible": true,
            "Endpoint": "limitless-test-cluster.limitless-cekycexample.us-
east-2.rds.amazonaws.com"
        },
        {
            "DBShardGroupResourceId": "shardgroup-a6e3a0226aa243e2ac6c7a1234567890",
            "DBShardGroupIdentifier": "my-db-shard-group",
            "DBClusterIdentifier": "my-sv2-cluster",
            "MaxACU": 1000.0,
            "ComputeRedundancy": 0,
            "Status": "available",
            "PubliclyAccessible": false,
            "Endpoint": "my-sv2-cluster.limitless-cekycexample.us-
east-2.rds.amazonaws.com"
        }
    ]
}
```

Per ulteriori informazioni, consulta Cluster database Amazon Aurora nella Guida per l'utente di Amazon Aurora.

- Per i dettagli sull'API, consulta AWS CLI Command ModifyDbShardGroupReference.

`modify-db-snapshot-attribute`

Il seguente esempio di codice mostra come utilizzare`modify-db-snapshot-attribute`.

AWS CLI

Esempio 1: consentire a due AWS account di ripristinare un'istantanea del DB

L'`modify-db-snapshot-attribute`esempio seguente concede l'autorizzazione a due AWS account, con gli identificatori `111122223333` e`444455556666`, per ripristinare lo snapshot DB denominato. `mydbsnapshot`

```
aws rds modify-db-snapshot-attribute \
    --db-snapshot-identifier mydbsnapshot \
    --attribute-name restore \
```

```
    --values-to-add {"111122223333","444455556666"}
```

Output:

```
{
    "DBSnapshotAttributesResult": {
        "DBSnapshotIdentifier": "mydbsnapshot",
        "DBSnapshotAttributes": [
            {
                "AttributeName": "restore",
                "AttributeValues": [
                    "111122223333",
                    "444455556666"
                ]
            }
        ]
    }
}
```

Per ulteriori informazioni, consulta Condivisione di uno snapshot nella Guida per l'utente di Amazon RDS.

Esempio 2: impedire a un AWS account di ripristinare un'istantanea del DB

L'`modify-db-snapshot-attribute`esempio seguente rimuove l'autorizzazione da un particolare AWS account per ripristinare lo snapshot DB denominato. `mydbsnapshot` Quando si specifica un solo account, l'identificatore dell'account non può essere racchiuso tra virgolette o parentesi.

```
aws rds modify-db-snapshot-attribute \
    --db-snapshot-identifier mydbsnapshot \
    --attribute-name restore \
    --values-to-remove 444455556666
```

Output:

```
{
    "DBSnapshotAttributesResult": {
        "DBSnapshotIdentifier": "mydbsnapshot",
        "DBSnapshotAttributes": [
            {
                "AttributeName": "restore",
```

```
                "AttributeValues": [
                    "111122223333"
                ]
            }
        ]
    }
}
```

Per ulteriori informazioni, consulta Condivisione di uno snapshot nella Guida per l'utente di Amazon RDS.

- Per i dettagli sull'API, vedere ModifyDbSnapshotAttributein AWS CLI Command Reference.

`modify-db-snapshot-attributes`

Il seguente esempio di codice mostra come utilizzare`modify-db-snapshot-attributes`.

AWS CLI

Come modificare un attributo dello snapshot database

L'`modify-db-snapshot-attribute`esempio seguente consente di ripristinare due identificatori di AWS account `111122223333` e `444455556666` lo snapshot del DB denominato. `mydbsnapshot`

```
aws rds modify-db-snapshot-attribute \
    --db-snapshot-identifier mydbsnapshot \
    --attribute-name restore \
    --values-to-add '["111122223333","444455556666"]'
```

Output:

```
{
    "DBSnapshotAttributesResult": {
        "DBSnapshotIdentifier": "mydbsnapshot",
        "DBSnapshotAttributes": [
            {
                "AttributeName": "restore",
                "AttributeValues": [
                    "111122223333",
                    "444455556666"
                ]
```

```
            }
        ]
    }
}
```

Per ulteriori informazioni, consulta Condivisione di uno snapshot nella Guida per l'utente di Amazon RDS.

- Per i dettagli sull'API, vedere ModifyDbSnapshotAttributesin AWS CLI Command Reference.

`modify-db-snapshot`

Il seguente esempio di codice mostra come utilizzare`modify-db-snapshot`.

AWS CLI

Come modificare uno snapshot database

L'esempio `modify-db-snapshot` seguente aggiorna uno snapshot di PostgeSQL 10.6 denominato `db5-snapshot-upg-test` a PostgreSQL 11.7. La nuova versione del motore database viene mostrata dopo il completamento dell'aggiornamento dello snapshot, quando il suo stato è disponibile.

```
aws rds modify-db-snapshot \
    --db-snapshot-identifier db5-snapshot-upg-test \
    --engine-version 11.7
```

Output:

```
{
    "DBSnapshot": {
        "DBSnapshotIdentifier": "db5-snapshot-upg-test",
        "DBInstanceIdentifier": "database-5",
        "SnapshotCreateTime": "2020-03-27T20:49:17.092Z",
        "Engine": "postgres",
        "AllocatedStorage": 20,
        "Status": "upgrading",
        "Port": 5432,
        "AvailabilityZone": "us-west-2a",
        "VpcId": "vpc-2ff27557",
        "InstanceCreateTime": "2020-03-27T19:59:04.735Z",
        "MasterUsername": "postgres",
```

```
        "EngineVersion": "10.6",
        "LicenseModel": "postgresql-license",
        "SnapshotType": "manual",
        "OptionGroupName": "default:postgres-11",
        "PercentProgress": 100,
        "StorageType": "gp2",
        "Encrypted": false,
        "DBSnapshotArn": "arn:aws:rds:us-west-2:123456789012:snapshot:db5-snapshot-
upg-test",
        "IAMDatabaseAuthenticationEnabled": false,
        "ProcessorFeatures": [],
        "DbiResourceId": "db-GJMF75LM42IL6BTFRE4UZJ5YM4"
    }
}
```

Per ulteriori informazioni, consulta Aggiornamento di uno snapshot database PostgreSQL nella Guida per l'utente di Amazon RDS.

- Per i dettagli sull'API, consulta ModifyDbSnapshot AWS CLICommand Reference.

`modify-db-subnet-group`

Il seguente esempio di codice mostra come utilizzare`modify-db-subnet-group`.

AWS CLI

Come modificare un gruppo di sottoreti database

L'esempio `modify-db-subnet-group` seguente aggiunge una sottorete con l'ID `subnet-08e41f9e230222222` al gruppo di sottoreti database denominato `mysubnetgroup`. Per mantenere le sottoreti esistenti nel gruppo di sottoreti, includete i loro valori IDs as nell'opzione. `--subnet-ids` Assicurati di disporre di un gruppo di sottoreti con almeno due zone di disponibilità diverse nel gruppo di sottoreti database.

```
aws rds modify-db-subnet-group \
    --db-subnet-group-name mysubnetgroup \
    --subnet-ids
 '["subnet-0a1dc4e1a6f123456","subnet-070dd7ecb3aaaaaaa","subnet-00f5b198bc0abcdef","subnet-
```

Output:

```
{
```

```
    "DBSubnetGroup": {
        "DBSubnetGroupName": "mysubnetgroup",
        "DBSubnetGroupDescription": "test DB subnet group",
        "VpcId": "vpc-0f08e7610a1b2c3d4",
        "SubnetGroupStatus": "Complete",
        "Subnets": [
            {
                "SubnetIdentifier": "subnet-08e41f9e230222222",
                "SubnetAvailabilityZone": {
                    "Name": "us-west-2a"
                },
                "SubnetStatus": "Active"
            },
            {
                "SubnetIdentifier": "subnet-070dd7ecb3aaaaaaa",
                "SubnetAvailabilityZone": {
                    "Name": "us-west-2b"
                },
                "SubnetStatus": "Active"
            },
            {
                "SubnetIdentifier": "subnet-00f5b198bc0abcdef",
                "SubnetAvailabilityZone": {
                    "Name": "us-west-2d"
                },
                "SubnetStatus": "Active"
            },
            {
                "SubnetIdentifier": "subnet-0a1dc4e1a6f123456",
                "SubnetAvailabilityZone": {
                    "Name": "us-west-2b"
                },
                "SubnetStatus": "Active"
            }
        ],
        "DBSubnetGroupArn": "arn:aws:rds:us-
west-2:534026745191:subgrp:mysubnetgroup"
    }
}
```

Per ulteriori informazioni, consulta Fase 3: crea un gruppo di sottoreti database nella Guida per l'utente di Amazon RDS.

- Per i dettagli sull'API, consulta Command ModifyDbSubnetGroupReference AWS CLI .

`modify-event-subscription`

Il seguente esempio di codice mostra come utilizzare`modify-event-subscription`.

AWS CLI

Come modificare un abbonamento all'evento

L'esempio `modify-event-subscription` seguente disabilita l'abbonamento all'evento specificato, in modo che non pubblichi più notifiche sull'argomento Amazon Simple Notification Service specificato.

```
aws rds modify-event-subscription \
    --subscription-name my-instance-events \
    --no-enabled
```

Output:

```
{
    "EventSubscription": {
        "EventCategoriesList": [
            "backup",
            "recovery"
        ],
        "CustomerAwsId": "123456789012",
        "SourceType": "db-instance",
        "SubscriptionCreationTime": "Tue Jul 31 23:22:01 UTC 2018",
        "EventSubscriptionArn": "arn:aws:rds:us-east-1:123456789012:es:my-instance-
events",
        "SnsTopicArn": "arn:aws:sns:us-east-1:123456789012:interesting-events",
        "CustSubscriptionId": "my-instance-events",
        "Status": "modifying",
        "Enabled": false
    }
}
```

- Per i dettagli sull'API, consulta ModifyEventSubscription AWS CLICommand Reference.

`modify-global-cluster`

Il seguente esempio di codice mostra come utilizzare`modify-global-cluster`.

AWS CLI

Come modificare un cluster di database globale

L'esempio `modify-global-cluster` seguente abilita la protezione da eliminazione per un cluster di database globale compatibile con Aurora MySQL.

```
aws rds modify-global-cluster \
    --global-cluster-identifier myglobalcluster \
    --deletion-protection
```

Output:

```
{
    "GlobalCluster": {
        "GlobalClusterIdentifier": "myglobalcluster",
        "GlobalClusterResourceId": "cluster-f0e523bfe07aabb",
        "GlobalClusterArn": "arn:aws:rds::123456789012:global-
cluster:myglobalcluster",
        "Status": "available",
        "Engine": "aurora-mysql",
        "EngineVersion": "5.7.mysql_aurora.2.07.2",
        "StorageEncrypted": false,
        "DeletionProtection": true,
        "GlobalClusterMembers": []
    }
}
```

Per ulteriori informazioni, consulta Utilizzo dei database globali Amazon Aurora nella Guida per l'utente di Amazon Aurora.

- Per i dettagli sull'API, consulta ModifyGlobalCluster AWS CLICommand Reference.

`promote-read-replica-db-cluster`

Il seguente esempio di codice mostra come utilizzare`promote-read-replica-db-cluster`.

AWS CLI

Come promuovere una replica di lettura al cluster di database

L'esempio `promote-read-replica-db-cluster` seguente promuove la replica di lettura specificata a un cluster di database autonomo.

```
aws rds promote-read-replica-db-cluster \
    --db-cluster-identifier mydbcluster-1
```

Output:

```
{
    "DBCluster": {
        "AllocatedStorage": 1,
        "AvailabilityZones": [
            "us-east-1a",
            "us-east-1b",
            "us-east-1c"
        ],
        "BackupRetentionPeriod": 1,
        "DatabaseName": "",
        "DBClusterIdentifier": "mydbcluster-1",
        ...some output truncated...
    }
}
```

Per ulteriori informazioni, consulta Promozione di una replica di lettura a cluster database nella Guida per l'utente di Amazon Aurora.

- Per i dettagli sull'API, consulta PromoteReadReplicaDbCluster AWS CLICommand Reference.

`promote-read-replica`

Il seguente esempio di codice mostra come utilizzare`promote-read-replica`.

AWS CLI

Come promuovere una replica di lettura

L'esempio `promote-read-replica` seguente promuove la replica di lettura specificata a diventare un'istanza database autonoma.

```
aws rds promote-read-replica \
    --db-instance-identifier test-instance-repl
```

Output:

```
{
    "DBInstance": {
        "DBInstanceArn": "arn:aws:rds:us-east-1:123456789012:db:test-instance-repl",
        "StorageType": "standard",
        "ReadReplicaSourceDBInstanceIdentifier": "test-instance",
        "DBInstanceStatus": "modifying",
        ...some output truncated...
    }
}
```

- Per i dettagli sull'API, consulta PromoteReadReplica AWS CLICommand Reference.

`purchase-reserved-db-instance`

Il seguente esempio di codice mostra come utilizzare`purchase-reserved-db-instance`.

AWS CLI

Come acquistare un'offerta di istanza database riservata

L'esempio `purchase-reserved-db-instances-offering` seguente acquista un'offerta di istanza database riservata. `reserved-db-instances-offering-id` deve essere un ID di offerta valido, come restituito dal comando `describe-reserved-db-instances-offering`.

aws rds purchase-reserved-db-instances -offering -- reserved-db-instances-offering -id 438012d3-4a52-4cc7-b2e3-8dff72e0e706

- Per i dettagli sull'API, PurchaseReservedDbInstanceconsulta AWS CLI Command Reference.

`purchase-reserved-db-instances-offerings`

Il seguente esempio di codice mostra come utilizzare`purchase-reserved-db-instances-offerings`.

AWS CLI

Esempio 1: come trovare un'istanza database riservata da acquistare

L'esempio `describe-reserved-db-instances-offerings` seguente elenca le istanze database MySQL riservate disponibili con la classe di istanze db.t2.micro e una durata di un anno. L'ID dell'offerta è necessario per acquistare un'istanza database riservata.

```
aws rds describe-reserved-db-instances-offerings \
    --product-description mysql \
    --db-instance-class db.t2.micro \
    --duration 1
```

Output:

```
{
    "ReservedDBInstancesOfferings": [
        {
            "ReservedDBInstancesOfferingId": "8ba30be1-b9ec-447f-8f23-6114e3f4c7b4",
            "DBInstanceClass": "db.t2.micro",
            "Duration": 31536000,
            "FixedPrice": 51.0,
            "UsagePrice": 0.0,
            "CurrencyCode": "USD",
            "ProductDescription": "mysql",
            "OfferingType": "Partial Upfront",
            "MultiAZ": false,
            "RecurringCharges": [
                {
                    "RecurringChargeAmount": 0.006,
                    "RecurringChargeFrequency": "Hourly"
                }
            ]
        },
    ... some output truncated ...
    ]
}
```

Per ulteriori informazioni, consulta Istanze database riservate per Amazon RDS nella Guida per l'utente di Amazon RDS.

Esempio 2: come acquistare un'istanza database riservata

L'esempio `purchase-reserved-db-instances-offering` seguente mostra come acquistare l'offerta di istanza database riservata dell'esempio precedente.

aws rds purchase-reserved-db-instances -offering -- reserved-db-instances-offering -id 8ba30be1-b9ec-447f-8f23-6114e3f4c7b4

Output:

```
{
    "ReservedDBInstance": {
        "ReservedDBInstanceId": "ri-2020-06-29-16-54-57-670",
        "ReservedDBInstancesOfferingId": "8ba30be1-b9ec-447f-8f23-6114e3f4c7b4",
        "DBInstanceClass": "db.t2.micro",
        "StartTime": "2020-06-29T16:54:57.670Z",
        "Duration": 31536000,
        "FixedPrice": 51.0,
        "UsagePrice": 0.0,
        "CurrencyCode": "USD",
        "DBInstanceCount": 1,
        "ProductDescription": "mysql",
        "OfferingType": "Partial Upfront",
        "MultiAZ": false,
        "State": "payment-pending",
        "RecurringCharges": [
            {
                "RecurringChargeAmount": 0.006,
                "RecurringChargeFrequency": "Hourly"
            }
        ],
        "ReservedDBInstanceArn": "arn:aws:rds:us-
west-2:123456789012:ri:ri-2020-06-29-16-54-57-670"
    }
}
```

Per ulteriori informazioni, consulta Istanze database riservate per Amazon RDS nella Guida per l'utente di Amazon RDS.

- Per i dettagli sull'AWS CLI API, PurchaseReservedDbInstancesOfferingsconsulta Command Reference.

`reboot-db-instance`

Il seguente esempio di codice mostra come utilizzare`reboot-db-instance`.

AWS CLI

Come riavviare un'istanza database

L'esempio `reboot-db-instance` seguente avvia il riavvio dell'istanza database specificata.

```
aws rds reboot-db-instance \
```

```
    --db-instance-identifier test-mysql-instance
```

Output:

```
{
    "DBInstance": {
        "DBInstanceIdentifier": "test-mysql-instance",
        "DBInstanceClass": "db.t3.micro",
        "Engine": "mysql",
        "DBInstanceStatus": "rebooting",
        "MasterUsername": "admin",
        "Endpoint": {
            "Address": "test-mysql-instance.############.us-
west-2.rds.amazonaws.com",
            "Port": 3306,
            "HostedZoneId": "Z1PVIF0EXAMPLE"
        },

    ... output omitted...

    }
}
```

Per ulteriori informazioni, consulta Riavvio di un'istanza database nella Guida per l'utente di Amazon RDS.

- Per i dettagli sull'API, consulta Reboot DBInstance in AWS CLI Command Reference.

`reboot-db-shard-group`

Il seguente esempio di codice mostra come utilizzare`reboot-db-shard-group`.

AWS CLI

Esempio 1: come riavviare un gruppo di shard del database

L'esempio `reboot-db-shard-group` seguente riavvia un gruppo di shard del database.

```
aws rds reboot-db-shard-group \
    --db-shard-group-identifier my-db-shard-group
```

Output:

```
{
    "DBShardGroups": [
        {
            "DBShardGroupResourceId": "shardgroup-a6e3a0226aa243e2ac6c7a1234567890",
            "DBShardGroupIdentifier": "my-db-shard-group",
            "DBClusterIdentifier": "my-sv2-cluster",
            "MaxACU": 1000.0,
            "ComputeRedundancy": 0,
            "Status": "available",
            "PubliclyAccessible": false,
            "Endpoint": "my-sv2-cluster.limitless-cekycexample.us-
east-2.rds.amazonaws.com"
        }
    ]
}
```

Per ulteriori informazioni, consulta Riavvio di un cluster Amazon Aurora DB o di un'istanza Amazon Aurora DB nella Guida per l'utente di Amazon Aurora.

Esempio 2: come descrivere i gruppi di shard del database

L'esempio `describe-db-shard-groups` seguente recupera i dettagli dei gruppi di shard del database dopo l'esecuzione del comando `reboot-db-shard-group`. Il gruppo di shard del database `my-db-shard-group` ora è in fase di riavvio.

```
aws rds describe-db-shard-groups
```

Output:

```
{
    "DBShardGroups": [
        {
            "DBShardGroupResourceId": "shardgroup-7bb446329da94788b3f957746example",
            "DBShardGroupIdentifier": "limitless-test-shard-grp",
            "DBClusterIdentifier": "limitless-test-cluster",
            "MaxACU": 768.0,
            "ComputeRedundancy": 0,
            "Status": "available",
            "PubliclyAccessible": true,
            "Endpoint": "limitless-test-cluster.limitless-cekycexample.us-
east-2.rds.amazonaws.com"
        },
```

```
        {
            "DBShardGroupResourceId": "shardgroup-a6e3a0226aa243e2ac6c7a1234567890",
            "DBShardGroupIdentifier": "my-db-shard-group",
            "DBClusterIdentifier": "my-sv2-cluster",
            "MaxACU": 1000.0,
            "ComputeRedundancy": 0,
            "Status": "rebooting",
            "PubliclyAccessible": false,
            "Endpoint": "my-sv2-cluster.limitless-cekycexample.us-
east-2.rds.amazonaws.com"
        }
    ]
}
```

Per ulteriori informazioni, consulta Riavvio di un cluster Amazon Aurora DB o di un'istanza Amazon Aurora DB nella Guida per l'utente di Amazon Aurora.

- Per i dettagli sull'API, consulta RebootDbShardGroup AWS CLICommand Reference.

`register-db-proxy-targets`

Il seguente esempio di codice mostra come utilizzare`register-db-proxy-targets`.

AWS CLI

Come registrare un proxy database con un database

L'esempio `register-db-proxy-targets` seguente crea l'associazione tra un database e un proxy.

```
aws rds register-db-proxy-targets \
    --db-proxy-name proxyExample \
    --db-cluster-identifiers database-5
```

Output:

```
{
    "DBProxyTargets": [
        {
            "RdsResourceId": "database-5",
            "Port": 3306,
            "Type": "TRACKED_CLUSTER",
            "TargetHealth": {
```

```
                "State": "REGISTERING"
            }
        },
        {
            "Endpoint": "database-5instance-1.ab0cd1efghij.us-
east-1.rds.amazonaws.com",
            "RdsResourceId": "database-5",
            "Port": 3306,
            "Type": "RDS_INSTANCE",
            "TargetHealth": {
                "State": "REGISTERING"
            }
        }
    ]
}
```

Per ulteriori informazioni, consulta Creazione di un RDS Proxy nella Guida per l'utente di Amazon RDS e Creazione di un RDS Proxy nella Guida per l'utente di Amazon Aurora.

- Per i dettagli sull'API, consulta RegisterDbProxyTargets AWS CLICommand Reference.

`remove-from-global-cluster`

Il seguente esempio di codice mostra come utilizzare`remove-from-global-cluster`.

AWS CLI

Come scollegare un cluster secondario di Aurora da un cluster di database globale di Aurora

L'esempio `remove-from-global-cluster` seguente distacca un cluster secondario di Aurora da un cluster di database globale di Aurora. Il cluster passa da cluster di sola lettura a cluster autonomo con funzionalità di lettura-scrittura.

```
aws rds remove-from-global-cluster \
    --region us-west-2 \
    --global-cluster-identifier myglobalcluster \
    --db-cluster-identifier arn:aws:rds:us-west-2:123456789012:cluster:DB-1
```

Output:

```
{
    "GlobalCluster": {
```

```
        "GlobalClusterIdentifier": "myglobalcluster",
        "GlobalClusterResourceId": "cluster-abc123def456gh",
        "GlobalClusterArn": "arn:aws:rds::123456789012:global-
cluster:myglobalcluster",
        "Status": "available",
        "Engine": "aurora-postgresql",
        "EngineVersion": "10.11",
        "StorageEncrypted": true,
        "DeletionProtection": false,
        "GlobalClusterMembers": [
            {
                "DBClusterArn": "arn:aws:rds:us-east-1:123456789012:cluster:js-
global-cluster",
                "Readers": [
                    "arn:aws:rds:us-west-2:123456789012:cluster:DB-1"
                ],
                "IsWriter": true
            },
            {
                "DBClusterArn": "arn:aws:rds:us-west-2:123456789012:cluster:DB-1",
                "Readers": [],
                "IsWriter": false,
                "GlobalWriteForwardingStatus": "disabled"
            }
        ]
    }
}
```

Per ulteriori informazioni, consulta Rimozione di un cluster da un database globale Amazon Aurora nella Guida per l'utente di Amazon Aurora.

- Per i dettagli sull'API, consulta RemoveFromGlobalCluster AWS CLICommand Reference.

`remove-option-from-option-group`

Il seguente esempio di codice mostra come utilizzare`remove-option-from-option-group`.

AWS CLI

Come eliminare un'opzione da un gruppo di opzioni

L'esempio `remove-option-from-option-group` seguente rimuove l'opzione `OEM` da `myoptiongroup`.

```
aws rds remove-option-from-option-group \
    --option-group-name myoptiongroup \
    --options OEM \
    --apply-immediately
```

Output:

```
{
    "OptionGroup": {
        "OptionGroupName": "myoptiongroup",
        "OptionGroupDescription": "Test",
        "EngineName": "oracle-ee",
        "MajorEngineVersion": "19",
        "Options": [],
        "AllowsVpcAndNonVpcInstanceMemberships": true,
        "OptionGroupArn": "arn:aws:rds:us-east-1:123456789012:og:myoptiongroup"
    }
}
```

Per ulteriori informazioni, consulta Rimozione di un'opzione da un gruppo di opzioni nella Guida per l'utente di Amazon Aurora.

- Per i dettagli sull'API, consulta RemoveOptionFromOptionGroup AWS CLICommand Reference.

`remove-role-from-db-cluster`

Il seguente esempio di codice mostra come utilizzare`remove-role-from-db-cluster`.

AWS CLI

Per dissociare un ruolo AWS Identity and Access Management (IAM) da un cluster DB

L'esempio `remove-role-from-db-cluster` seguente rimuove un ruolo da un cluster di database.

```
aws rds remove-role-from-db-cluster \
    --db-cluster-identifier mydbcluster \
    --role-arn arn:aws:iam::123456789012:role/RDSLoadFromS3
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Associazione di un ruolo IAM a un cluster DB Amazon Aurora MySQL nella Guida per l'utente di Amazon Aurora.

- Per i dettagli sull'API, consulta RemoveRoleFromDbCluster AWS CLICommand Reference.

`remove-role-from-db-instance`

Il seguente esempio di codice mostra come utilizzare`remove-role-from-db-instance`.

AWS CLI

Per dissociare un ruolo AWS Identity and Access Management (IAM) da un'istanza DB

L'esempio `remove-role-from-db-instance` seguente rimuove il ruolo denominato `rds-s3-integration-role` da un'istanza Oracle DB denominata `test-instance`.

```
aws rds remove-role-from-db-instance \
    --db-instance-identifier test-instance \
    --feature-name S3_INTEGRATION \
    --role-arn arn:aws:iam::111122223333:role/rds-s3-integration-role
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Disabilitazione dell'integrazione di RDS SQL Server con S3 nella Guida per l'utente di Amazon RDS.

- Per i dettagli sull'API, consulta RemoveRoleFromDbInstance AWS CLICommand Reference.

`remove-source-identifier-from-subscription`

Il seguente esempio di codice mostra come utilizzare`remove-source-identifier-from-subscription`.

AWS CLI

Come rimuovere un identificatore di origine da un abbonamento

L'esempio `remove-source-identifier` seguente rimuove l'identificatore di origine specificato da un abbonamento esistente.

```
aws rds remove-source-identifier-from-subscription \
    --subscription-name my-instance-events \
```

```
    --source-identifier test-instance-repl
```

Output:

```
{
    "EventSubscription": {
        "EventSubscriptionArn": "arn:aws:rds:us-east-1:123456789012:es:my-instance-
events",
        "SubscriptionCreationTime": "Tue Jul 31 23:22:01 UTC 2018",
        "EventCategoriesList": [
            "backup",
            "recovery"
        ],
        "SnsTopicArn": "arn:aws:sns:us-east-1:123456789012:interesting-events",
        "Status": "modifying",
        "CustSubscriptionId": "my-instance-events",
        "CustomerAwsId": "123456789012",
        "SourceIdsList": [
            "test-instance"
        ],
        "SourceType": "db-instance",
        "Enabled": false
    }
}
```

- Per i dettagli sull'API, consulta RemoveSourceIdentifierFromSubscription AWS CLICommand Reference.

`remove-tags-from-resource`

Il seguente esempio di codice mostra come utilizzare`remove-tags-from-resource`.

AWS CLI

Come rimuovere tag da una risorsa

L'esempio `remove-tags-from-resource` seguente rimuove i tag da una risorsa.

```
aws rds remove-tags-from-resource \
    --resource-name arn:aws:rds:us-east-1:123456789012:db:mydbinstance \
    --tag-keys Name Environment
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Tagging delle risorse Amazon RDS nella Guida per l'utente di Amazon RDS e Tagging delle risorse Amazon RDS nella Guida per l'utente di Amazon Aurora.

- Per i dettagli sull'API, consulta RemoveTagsFromResource AWS CLICommand Reference.

`reset-db-cluster-parameter-group`

Il seguente esempio di codice mostra come utilizzare`reset-db-cluster-parameter-group`.

AWS CLI

Esempio 1: come reimpostare tutti i parametri sui valori predefiniti

L'esempio `reset-db-cluster-parameter-group` seguente reimposta i valori dei parametri in un gruppo di parametri del cluster di database creato dal cliente sui valori predefiniti.

```
aws rds reset-db-cluster-parameter-group \
    --db-cluster-parameter-group-name mydbclpg \
    --reset-all-parameters
```

Output:

```
{
    "DBClusterParameterGroupName": "mydbclpg"
}
```

Per ulteriori informazioni, consulta Uso di gruppi di parametri database e gruppi di parametri cluster database nella Guida per l'utente di Amazon Aurora.

Esempio 2: come reimpostare i parametri specifici ai valori predefiniti

L'esempio `reset-db-cluster-parameter-group` seguente reimposta i valori dei parametri per parametri specifici sui valori predefiniti in un gruppo di parametri del cluster di database creato dal cliente.

```
aws rds reset-db-cluster-parameter-group \
    --db-cluster-parameter-group-name mydbclpgy \
    --parameters "ParameterName=max_connections,ApplyMethod=immediate" \
```

```
    "ParameterName=max_allowed_packet,ApplyMethod=immediate"
```

Output:

```
{
    "DBClusterParameterGroupName": "mydbclpg"
}
```

Per ulteriori informazioni, consulta Uso di gruppi di parametri database e gruppi di parametri cluster database nella Guida per l'utente di Amazon Aurora.

- Per i dettagli sull'API, consulta ResetDbClusterParameterGroup AWS CLICommand Reference.

`reset-db-parameter-group`

Il seguente esempio di codice mostra come utilizzare`reset-db-parameter-group`.

AWS CLI

Esempio 1: come reimpostare tutti i parametri sui valori predefiniti

L'esempio `reset-db-parameter-group` seguente reimposta tutti i valori dei parametri in un gruppo di parametri database creato dal cliente sui valori predefiniti.

```
aws rds reset-db-parameter-group \
    --db-parameter-group-name mypg \
    --reset-all-parameters
```

Output:

```
{
    "DBParameterGroupName": "mypg"
}
```

Per ulteriori informazioni, consulta Utilizzo di gruppi di parametri di database nella Guida per l'utente di Amazon RDS e Uso di gruppi di parametri database e gruppi di parametri cluster database nella Guida per l'utente di Amazon Aurora.

Esempio 2: come reimpostare i parametri specifici ai valori predefiniti

L'esempio `reset-db-parameter-group` seguente reimposta i valori dei parametri per parametri specifici sui valori predefiniti in un gruppo di parametri database creato dal cliente.

```
aws rds reset-db-parameter-group \
    --db-parameter-group-name mypg \
    --parameters "ParameterName=max_connections,ApplyMethod=immediate" \
                 "ParameterName=max_allowed_packet,ApplyMethod=immediate"
```

Output:

```
{
    "DBParameterGroupName": "mypg"
}
```

Per ulteriori informazioni, consulta Utilizzo di gruppi di parametri di database nella Guida per l'utente di Amazon RDS e Uso di gruppi di parametri database e gruppi di parametri cluster database nella Guida per l'utente di Amazon Aurora.

- Per i dettagli sull'API, consulta ResetDbParameterGroup AWS CLICommand Reference.

`restore-db-cluster-from-s3`

Il seguente esempio di codice mostra come utilizzare`restore-db-cluster-from-s3`.

AWS CLI

Come ripristinare un cluster di database Amazon Aurora da Amazon S3

L'esempio `restore-db-cluster-from-s3` seguente ripristina un cluster di database compatibile con Amazon Aurora MySQL versione 5.7 da un file di backup del database MySQL 5.7 in Amazon S3.

```
aws rds restore-db-cluster-from-s3 \
    --db-cluster-identifier cluster-s3-restore \
    --engine aurora-mysql \
    --master-username admin \
    --master-user-password mypassword \
    --s3-bucket-name amzn-s3-demo-bucket \
    --s3-prefix test-backup \
    --s3-ingestion-role-arn arn:aws:iam::123456789012:role/service-role/TestBackup \
    --source-engine mysql \
```

```
    --source-engine-version 5.7.28
```

Output:

```
{
    "DBCluster": {
        "AllocatedStorage": 1,
        "AvailabilityZones": [
            "us-west-2c",
            "us-west-2a",
            "us-west-2b"
        ],
        "BackupRetentionPeriod": 1,
        "DBClusterIdentifier": "cluster-s3-restore",
        "DBClusterParameterGroup": "default.aurora-mysql5.7",
        "DBSubnetGroup": "default",
        "Status": "creating",
        "Endpoint": "cluster-s3-restore.cluster-co3xyzabc123.us-
west-2.rds.amazonaws.com",
        "ReaderEndpoint": "cluster-s3-restore.cluster-ro-co3xyzabc123.us-
west-2.rds.amazonaws.com",
        "MultiAZ": false,
        "Engine": "aurora-mysql",
        "EngineVersion": "5.7.12",
        "Port": 3306,
        "MasterUsername": "admin",
        "PreferredBackupWindow": "11:15-11:45",
        "PreferredMaintenanceWindow": "thu:12:19-thu:12:49",
        "ReadReplicaIdentifiers": [],
        "DBClusterMembers": [],
        "VpcSecurityGroups": [
            {
                "VpcSecurityGroupId": "sg-########",
                "Status": "active"
            }
        ],
        "HostedZoneId": "Z1PVIF0EXAMPLE",
        "StorageEncrypted": false,
        "DbClusterResourceId": "cluster-SU5THYQQHOWCXZZDGXREXAMPLE",
        "DBClusterArn": "arn:aws:rds:us-west-2:123456789012:cluster:cluster-s3-
restore",
        "AssociatedRoles": [],
        "IAMDatabaseAuthenticationEnabled": false,
```

```
        "ClusterCreateTime": "2020-07-27T14:22:08.095Z",
        "EngineMode": "provisioned",
        "DeletionProtection": false,
        "HttpEndpointEnabled": false,
        "CopyTagsToSnapshot": false,
        "CrossAccountClone": false,
        "DomainMemberships": []
    }
}
```

Per ulteriori informazioni, consulta Migrazione dei dati da MySQL utilizzando un bucket Amazon S3 nella Guida per l'utente di Amazon Aurora.

- Per i dettagli sull'API, consulta RestoreDbClusterFromS3 in AWS CLI Command Reference.

`restore-db-cluster-from-snapshot`

Il seguente esempio di codice mostra come usare`restore-db-cluster-from-snapshot`.

AWS CLI

Come ripristinare un cluster di database da uno snapshot

L'esempio `restore-db-cluster-from-snapshot` seguente ripristina un cluster di database Aurora PostgreSQL compatibile con PostgreSQL versione 10.7 da uno snapshot del cluster di database denominato `test-instance-snapshot`.

```
aws rds restore-db-cluster-from-snapshot \
    --db-cluster-identifier newdbcluster \
    --snapshot-identifier test-instance-snapshot \
    --engine aurora-postgresql \
    --engine-version 10.7
```

Output:

```
{
    "DBCluster": {
        "AllocatedStorage": 1,
        "AvailabilityZones": [
            "us-west-2c",
            "us-west-2a",
            "us-west-2b"
        ],
```

```
        "BackupRetentionPeriod": 7,
        "DatabaseName": "",
        "DBClusterIdentifier": "newdbcluster",
        "DBClusterParameterGroup": "default.aurora-postgresql10",
        "DBSubnetGroup": "default",
        "Status": "creating",
        "Endpoint": "newdbcluster.cluster-############.us-west-2.rds.amazonaws.com",
        "ReaderEndpoint": "newdbcluster.cluster-ro-############.us-
west-2.rds.amazonaws.com",
        "MultiAZ": false,
        "Engine": "aurora-postgresql",
        "EngineVersion": "10.7",
        "Port": 5432,
        "MasterUsername": "postgres",
        "PreferredBackupWindow": "09:33-10:03",
        "PreferredMaintenanceWindow": "sun:12:22-sun:12:52",
        "ReadReplicaIdentifiers": [],
        "DBClusterMembers": [],
        "VpcSecurityGroups": [
            {
                "VpcSecurityGroupId": "sg-########",
                "Status": "active"
            }
        ],
        "HostedZoneId": "Z1PVIF0EXAMPLE",
        "StorageEncrypted": true,
        "KmsKeyId": "arn:aws:kms:us-west-2:123456789012:key/287364e4-33e3-4755-a3b0-
a1b2c3d4e5f6",
        "DbClusterResourceId": "cluster-5DSB5IFQDDUVAWOUWM1EXAMPLE",
        "DBClusterArn": "arn:aws:rds:us-west-2:123456789012:cluster:newdbcluster",
        "AssociatedRoles": [],
        "IAMDatabaseAuthenticationEnabled": false,
        "ClusterCreateTime": "2020-06-05T15:06:58.634Z",
        "EngineMode": "provisioned",
        "DeletionProtection": false,
        "HttpEndpointEnabled": false,
        "CopyTagsToSnapshot": false,
        "CrossAccountClone": false,
        "DomainMemberships": []
    }
}
```

Per ulteriori informazioni, consulta Ripristino da uno snapshot cluster database nella Guida per l'utente di Amazon Aurora.

- Per i dettagli sull'API, consulta RestoreDbClusterFromSnapshot AWS CLICommand Reference.

`restore-db-cluster-to-point-in-time`

Il seguente esempio di codice mostra come utilizzare`restore-db-cluster-to-point-in-time`.

AWS CLI

Come ripristinare un cluster di database a un determinato momento

L'esempio `restore-db-cluster-to-point-in-time` seguente ripristina il cluster di database denominato `database-4` all'ora più recente possibile. L'utilizzo di `copy-on-write` come tipo di ripristino consente di ripristinare il nuovo cluster di database come clone del cluster di database di origine.

```
aws rds restore-db-cluster-to-point-in-time \
    --source-db-cluster-identifier database-4 \
    --db-cluster-identifier sample-cluster-clone \
    --restore-type copy-on-write \
    --use-latest-restorable-time
```

Output:

```
{
    "DBCluster": {
        "AllocatedStorage": 1,
        "AvailabilityZones": [
            "us-west-2c",
            "us-west-2a",
            "us-west-2b"
        ],
        "BackupRetentionPeriod": 7,
        "DatabaseName": "",
        "DBClusterIdentifier": "sample-cluster-clone",
        "DBClusterParameterGroup": "default.aurora-postgresql10",
        "DBSubnetGroup": "default",
        "Status": "creating",
        "Endpoint": "sample-cluster-clone.cluster-############.us-
west-2.rds.amazonaws.com",
```

```
        "ReaderEndpoint": "sample-cluster-clone.cluster-ro-############.us-
west-2.rds.amazonaws.com",
        "MultiAZ": false,
        "Engine": "aurora-postgresql",
        "EngineVersion": "10.7",
        "Port": 5432,
        "MasterUsername": "postgres",
        "PreferredBackupWindow": "09:33-10:03",
        "PreferredMaintenanceWindow": "sun:12:22-sun:12:52",
        "ReadReplicaIdentifiers": [],
        "DBClusterMembers": [],
        "VpcSecurityGroups": [
            {
                "VpcSecurityGroupId": "sg-########",
                "Status": "active"
            }
        ],
        "HostedZoneId": "Z1PVIF0EXAMPLE",
        "StorageEncrypted": true,
        "KmsKeyId": "arn:aws:kms:us-west-2:123456789012:key/287364e4-33e3-4755-a3b0-
a1b2c3d4e5f6",
        "DbClusterResourceId": "cluster-BIZ77GDSA2XBSTNPFW1EXAMPLE",
        "DBClusterArn": "arn:aws:rds:us-west-2:123456789012:cluster:sample-cluster-
clone",
        "AssociatedRoles": [],
        "IAMDatabaseAuthenticationEnabled": false,
        "CloneGroupId": "8d19331a-099a-45a4-b4aa-11aa22bb33cc44dd",
        "ClusterCreateTime": "2020-03-10T19:57:38.967Z",
        "EngineMode": "provisioned",
        "DeletionProtection": false,
        "HttpEndpointEnabled": false,
        "CopyTagsToSnapshot": false,
        "CrossAccountClone": false
    }
}
```

Per ulteriori informazioni, consulta Ripristino di un cluster DB a un'ora specificata nella Guida per l'utente di Amazon Aurora.

- Per i dettagli sull'API, consulta RestoreDbClusterToPointInTime AWS CLICommand Reference.

`restore-db-instance-from-db-snapshot`

Il seguente esempio di codice mostra come utilizzare`restore-db-instance-from-db-snapshot`.

AWS CLI

Come ripristinare un'istanza database da uno snapshot di database

L'esempio `restore-db-instance-from-db-snapshot` seguente crea una nuova istanza database denominata `db7-new-instance` con la classe di istanza database `db.t3.small` dallo snapshot database specificato. L'istanza database di origine da cui è stata acquisita lo snapshot utilizza una classe di istanza database obsoleta, quindi non è possibile aggiornarla.

```
aws rds restore-db-instance-from-db-snapshot \
    --db-instance-identifier db7-new-instance \
    --db-snapshot-identifier db7-test-snapshot \
    --db-instance-class db.t3.small
```

Output:

```
{
    "DBInstance": {
        "DBInstanceIdentifier": "db7-new-instance",
        "DBInstanceClass": "db.t3.small",
        "Engine": "mysql",
        "DBInstanceStatus": "creating",

        ...output omitted...

        "PreferredMaintenanceWindow": "mon:07:37-mon:08:07",
        "PendingModifiedValues": {},
        "MultiAZ": false,
        "EngineVersion": "5.7.22",
        "AutoMinorVersionUpgrade": true,
        "ReadReplicaDBInstanceIdentifiers": [],
        "LicenseModel": "general-public-license",

        ...output omitted...

        "DBInstanceArn": "arn:aws:rds:us-west-2:123456789012:db:db7-new-instance",
        "IAMDatabaseAuthenticationEnabled": false,
```

```
        "PerformanceInsightsEnabled": false,
        "DeletionProtection": false,
        "AssociatedRoles": []
    }
}
```

Per ulteriori informazioni, consulta Ripristino da uno snapshot di database nella Guida per l'utente di Amazon RDS.

- Per i dettagli sull'API, consulta RestoreDbInstanceFromDbSnapshot AWS CLICommand Reference.

`restore-db-instance-from-s3`

Il seguente esempio di codice mostra come utilizzare`restore-db-instance-from-s3`.

AWS CLI

Come ripristinare un'istanza database da un backup in Amazon S3

L'esempio `restore-db-instance-from-s3` seguente crea una nuova istanza database denominata `restored-test-instance` da un backup esistente nel bucket S3 `my-backups`.

```
aws rds restore-db-instance-from-s3 \
    --db-instance-identifier restored-test-instance \
    --allocated-storage 250 --db-instance-class db.m4.large --engine mysql \
    --master-username master --master-user-password secret99 \
    --s3-bucket-name my-backups --s3-ingestion-role-
arn arn:aws:iam::123456789012:role/my-role \
    --source-engine mysql --source-engine-version 5.6.27
```

- Per i dettagli sull'API, consulta RestoreDbInstanceFromS3 in AWS CLI Command Reference.

`restore-db-instance-to-point-in-time`

Il seguente esempio di codice mostra come usare`restore-db-instance-to-point-in-time`.

AWS CLI

Esempio 1: come ripristinare un'istanza database a un punto temporale

L'esempio `restore-db-instance-to-point-in-time` seguente reimposta `test-instance` su una nuova istanza database denominata `restored-test-instance`, a partire dal tempo specificato.

```
aws rds restore-db-instance-to-point-in-time \
    --source-db-instance-identifier test-instance \
    --target-db-instance restored-test-instance \
    --restore-time 2018-07-30T23:45:00.000Z
```

Output:

```
{
    "DBInstance": {
        "AllocatedStorage": 20,
        "DBInstanceArn": "arn:aws:rds:us-east-1:123456789012:db:restored-test-
instance",
        "DBInstanceStatus": "creating",
        "DBInstanceIdentifier": "restored-test-instance",
        ...some output omitted...
    }
}
```

Per ulteriori informazioni, consulta Ripristino a un'ora specifica per un'istanza database nella Guida per l'utente di Amazon RDS.

Esempio 2: come ripristinare un'istanza database a un'ora specificata da un backup replicato

L'esempio `restore-db-instance-to-point-in-time` seguente ripristina un'istanza database Oracle all'ora specificata da un backup automatico replicato.

```
aws rds restore-db-instance-to-point-in-time \
    --source-db-instance-automated-backups-arn "arn:aws:rds:us-
west-2:123456789012:auto-backup:ab-jkib2gfq5rv7replzadausbrktni2bn4example" \
    --target-db-instance-identifier myorclinstance-from-replicated-backup \
    --restore-time 2020-12-08T18:45:00.000Z
```

Output:

```
{
    "DBInstance": {
        "DBInstanceIdentifier": "myorclinstance-from-replicated-backup",
        "DBInstanceClass": "db.t3.micro",
```

```
        "Engine": "oracle-se2",
        "DBInstanceStatus": "creating",
        "MasterUsername": "admin",
        "DBName": "ORCL",
        "AllocatedStorage": 20,
        "PreferredBackupWindow": "07:45-08:15",
        "BackupRetentionPeriod": 14,
        ... some output omitted ...
        "DbiResourceId": "db-KGLXG75BGVIWKQT7NQ4EXAMPLE",
        "CACertificateIdentifier": "rds-ca-2019",
        "DomainMemberships": [],
        "CopyTagsToSnapshot": false,
        "MonitoringInterval": 0,
        "DBInstanceArn": "arn:aws:rds:us-west-2:123456789012:db:myorclinstance-from-
replicated-backup",
        "IAMDatabaseAuthenticationEnabled": false,
        "PerformanceInsightsEnabled": false,
        "DeletionProtection": false,
        "AssociatedRoles": [],
        "TagList": []
    }
}
```

Per ulteriori informazioni, consulta Ripristino a un'ora specificata da un backup replicato nella Guida per l'utente di Amazon RDS.

- Per i dettagli sull'API, consulta RestoreDbInstanceToPointInTime AWS CLICommand Reference.

`start-activity-stream`

Il seguente esempio di codice mostra come utilizzare`start-activity-stream`.

AWS CLI

Come avviare un flusso di attività del database

L'`start-activity-stream`esempio seguente avvia un flusso di attività asincrono per monitorare un cluster Aurora denominato. my-pg-cluster

```
aws rds start-activity-stream \
    --region us-east-1 \
    --mode async \
```

```
    --kms-key-id arn:aws:kms:us-east-1:1234567890123:key/a12c345d-6ef7-890g-
h123-456i789jk0l1 \
    --resource-arn arn:aws:rds:us-east-1:1234567890123:cluster:my-pg-cluster \
    --apply-immediately
```

Output:

```
{
    "KmsKeyId": "arn:aws:kms:us-east-1:1234567890123:key/a12c345d-6ef7-890g-
h123-456i789jk0l1",
    "KinesisStreamName": "aws-rds-das-cluster-0ABCDEFGHI1JKLM2NOPQ3R4S",
    "Status": "starting",
    "Mode": "async",
    "ApplyImmediately": true
}
```

Per ulteriori informazioni, consulta Avvio di un flusso di attività di database nella Guida per l'utente di Amazon Aurora.

- Per i dettagli sull'API, consulta StartActivityStreamCommand Reference.AWS CLI

`start-db-cluster`

Il seguente esempio di codice mostra come utilizzare`start-db-cluster`.

AWS CLI

Come avviare un cluster di database

L'esempio `start-db-cluster` seguente avvia un cluster di database e le relative istanze database.

```
aws rds start-db-cluster \
    --db-cluster-identifier mydbcluster
```

Output:

```
{
    "DBCluster": {
        "AllocatedStorage": 1,
        "AvailabilityZones": [
            "us-east-1a",
```

```
            "us-east-1e",
            "us-east-1b"
        ],
        "BackupRetentionPeriod": 1,
        "DatabaseName": "mydb",
        "DBClusterIdentifier": "mydbcluster",
        ...some output truncated...
    }
}
```

Per ulteriori informazioni, consulta Avvio e arresto di un cluster DB Amazon Aurora nella Guida per l'utente di Amazon Aurora.

- Per i dettagli sull'API, consulta StartDbCluster AWS CLICommand Reference.

`start-db-instance-automated-backups-replication`

Il seguente esempio di codice mostra come utilizzare`start-db-instance-automated-backups-replication`.

AWS CLI

Come abilitare i backup automatici tra Regioni

L'esempio `start-db-instance-automated-backups-replication` seguente replica i backup automatici da un'istanza database nella Regione Stati Uniti orientali (Virginia settentrionale) alla Regione Stati Uniti occidentali (Oregon). Il periodo di conservazione dei backup è di 14 giorni.

```
aws rds start-db-instance-automated-backups-replication \
    --region us-west-2 \
    --source-db-instance-arn "arn:aws:rds:us-east-1:123456789012:db:new-orcl-db" \
    --backup-retention-period 14
```

Output:

```
{
    "DBInstanceAutomatedBackup": {
        "DBInstanceArn": "arn:aws:rds:us-east-1:123456789012:db:new-orcl-db",
        "DbiResourceId": "db-JKIB2GFQ5RV7REPLZA4EXAMPLE",
        "Region": "us-east-1",
        "DBInstanceIdentifier": "new-orcl-db",
```

```
        "RestoreWindow": {},
        "AllocatedStorage": 20,
        "Status": "pending",
        "Port": 1521,
        "InstanceCreateTime": "2020-12-04T15:28:31Z",
        "MasterUsername": "admin",
        "Engine": "oracle-se2",
        "EngineVersion": "12.1.0.2.v21",
        "LicenseModel": "bring-your-own-license",
        "OptionGroupName": "default:oracle-se2-12-1",
        "Encrypted": false,
        "StorageType": "gp2",
        "IAMDatabaseAuthenticationEnabled": false,
        "BackupRetentionPeriod": 14,
        "DBInstanceAutomatedBackupsArn": "arn:aws:rds:us-west-2:123456789012:auto-
backup:ab-jkib2gfq5rv7replzadausbrktni2bn4example"
    }
}
```

Per ulteriori informazioni, consulta Abilitazione dei backup automatici tra Regioni nella Guida per l'utente di Amazon RDS.

- Per i dettagli sull'API, consulta StartDbInstanceAutomatedBackupsReplication AWS CLICommand Reference.

`start-db-instance`

Il seguente esempio di codice mostra come utilizzare`start-db-instance`.

AWS CLI

Come avviare un'istanza database

L'esempio `start-db-instance` seguente avvia la registrazione dell'istanza database specificata.

```
aws rds start-db-instance \
    --db-instance-identifier test-instance
```

Output:

```
{
```

```
    "DBInstance": {
        "DBInstanceStatus": "starting",
        ...some output truncated...
    }
}
```

- Per i dettagli sull'API, consulta StartDbInstance AWS CLICommand Reference.

`start-export-task`

Il seguente esempio di codice mostra come utilizzare`start-export-task`.

AWS CLI

Come esportare uno snapshot in Amazon S3

L'esempio `start-export-task` seguente esporta uno snapshot database denominato `db5-snapshot-test` nel bucket Amazon S3 denominato `amzn-s3-demo-bucket`.

```
aws rds start-export-task \
    --export-task-identifier my-s3-export \
    --source-arn arn:aws:rds:us-west-2:123456789012:snapshot:db5-snapshot-test \
    --s3-bucket-name amzn-s3-demo-bucket \
    --iam-role-arn arn:aws:iam::123456789012:role/service-role/ExportRole \
    --kms-key-id arn:aws:kms:us-west-2:123456789012:key/abcd0000-7fca-4128-82f2-aabbccddeeff
```

Output:

```
{
    "ExportTaskIdentifier": "my-s3-export",
    "SourceArn": "arn:aws:rds:us-west-2:123456789012:snapshot:db5-snapshot-test",
    "SnapshotTime": "2020-03-27T20:48:42.023Z",
    "S3Bucket": "amzn-s3-demo-bucket",
    "IamRoleArn": "arn:aws:iam::123456789012:role/service-role/ExportRole",
    "KmsKeyId": "arn:aws:kms:us-west-2:123456789012:key/abcd0000-7fca-4128-82f2-aabbccddeeff",
    "Status": "STARTING",
    "PercentProgress": 0,
    "TotalExtractedDataInGB": 0
}
```

Per ulteriori informazioni, consulta Esportazione di uno snapshot in un bucket Amazon S3 nella Guida per l'utente di Amazon RDS.

- Per i dettagli sull'API, consulta StartExportTask AWS CLICommand Reference.

`stop-activity-stream`

Il seguente esempio di codice mostra come utilizzare`stop-activity-stream`.

AWS CLI

Come arrestare un flusso di attività del database

L'`stop-activity-stream`esempio seguente interrompe un flusso di attività in un cluster Aurora denominato. my-pg-cluster

```
aws rds stop-activity-stream \
    --region us-east-1 \
    --resource-arn arn:aws:rds:us-east-1:1234567890123:cluster:my-pg-cluster \
    --apply-immediately
```

Output:

```
{
    "KmsKeyId": "arn:aws:kms:us-east-1:1234567890123:key/a12c345d-6ef7-890g-
h123-456i789jk0l1",
    "KinesisStreamName": "aws-rds-das-cluster-0ABCDEFGHI1JKLM2NOPQ3R4S",
    "Status": "stopping"
}
```

Per ulteriori informazioni, consulta Arresto di un flusso di attività nella Guida per l'utente di Amazon Aurora.

- Per i dettagli sull'API, consulta StopActivityStream AWS CLICommand Reference.

`stop-db-cluster`

Il seguente esempio di codice mostra come utilizzare`stop-db-cluster`.

AWS CLI

Come arrestare un cluster di database

L'esempio `stop-db-cluster` seguente arresta un cluster di database e le relative istanze database.

```
aws rds stop-db-cluster \
    --db-cluster-identifier mydbcluster
```

Output:

```
{
    "DBCluster": {
        "AllocatedStorage": 1,
        "AvailabilityZones": [
            "us-east-1a",
            "us-east-1e",
            "us-east-1b"
        ],
        "BackupRetentionPeriod": 1,
        "DatabaseName": "mydb",
        "DBClusterIdentifier": "mydbcluster",
        ...some output truncated...
    }
}
```

Per ulteriori informazioni, consulta Avvio e arresto di un cluster DB Amazon Aurora nella Guida per l'utente di Amazon Aurora.

- Per i dettagli sull'API, consulta StopDbCluster AWS CLICommand Reference.

`stop-db-instance-automated-backups-replication`

Il seguente esempio di codice mostra come utilizzare`stop-db-instance-automated-backups-replication`.

AWS CLI

Come arrestare la replica dei backup automatici

L'esempio `stop-db-instance-automated-backups-replication` seguente termina la replica dei backup automatici nella Regione Stati Uniti occidentali (Oregon). I backup replicati vengono mantenuti in base al periodo di conservazione dei backup impostato.

```
aws rds stop-db-instance-automated-backups-replication \
```

```
    --region us-west-2 \
    --source-db-instance-arn "arn:aws:rds:us-east-1:123456789012:db:new-orcl-db"
```

Output:

```
{
    "DBInstanceAutomatedBackup": {
        "DBInstanceArn": "arn:aws:rds:us-east-1:123456789012:db:new-orcl-db",
        "DbiResourceId": "db-JKIB2GFQ5RV7REPLZA4EXAMPLE",
        "Region": "us-east-1",
        "DBInstanceIdentifier": "new-orcl-db",
        "RestoreWindow": {
            "EarliestTime": "2020-12-04T23:13:21.030Z",
            "LatestTime": "2020-12-07T19:59:57Z"
        },
        "AllocatedStorage": 20,
        "Status": "replicating",
        "Port": 1521,
        "InstanceCreateTime": "2020-12-04T15:28:31Z",
        "MasterUsername": "admin",
        "Engine": "oracle-se2",
        "EngineVersion": "12.1.0.2.v21",
        "LicenseModel": "bring-your-own-license",
        "OptionGroupName": "default:oracle-se2-12-1",
        "Encrypted": false,
        "StorageType": "gp2",
        "IAMDatabaseAuthenticationEnabled": false,
        "BackupRetentionPeriod": 7,
        "DBInstanceAutomatedBackupsArn": "arn:aws:rds:us-west-2:123456789012:auto-
backup:ab-jkib2gfq5rv7replzadausbrktni2bn4example"
    }
}
```

Per ulteriori informazioni, consulta Arresto della replica di backup automatici nella Guida per l'utente di Amazon RDS.

- Per i dettagli sull'API, consulta StopDbInstanceAutomatedBackupsReplication AWS CLICommand Reference.

`stop-db-instance`

Il seguente esempio di codice mostra come utilizzare`stop-db-instance`.

AWS CLI

Come arrestare un'istanza database

L'esempio `stop-db-instance` seguente arresta l'istanza database specificata.

```
aws rds stop-db-instance \
    --db-instance-identifier test-instance
```

Output:

```
{
    "DBInstance": {
        "DBInstanceStatus": "stopping",
        ...some output truncated...
    }
}
```

- Per i dettagli sull'API, consulta StopDbInstance AWS CLICommand Reference.

`switchover-blue-green-deployment`

Il seguente esempio di codice mostra come utilizzare`switchover-blue-green-deployment`.

AWS CLI

Esempio 1: cambiare una blue/green distribuzione per un'istanza DB RDS

L'esempio `switchover-blue-green-deployment` seguente promuove l'ambiente verde specificato a nuovo ambiente di produzione.

```
aws rds switchover-blue-green-deployment \
    --blue-green-deployment-identifier bgd-wi89nwzglccsfake \
    --switchover-timeout 300
```

Output:

```
{
    "BlueGreenDeployment": {
        "BlueGreenDeploymentIdentifier": "bgd-v53303651eexfake",
        "BlueGreenDeploymentName": "bgd-cli-test-instance",
        "Source": "arn:aws:rds:us-east-1:123456789012:db:my-db-instance",
```

```
        "Target": "arn:aws:rds:us-east-1:123456789012:db:my-db-instance-green-
blhi1e",
        "SwitchoverDetails": [
            {
                "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-
instance",
                "TargetMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-
instance-green-blhi1e",
                "Status": "AVAILABLE"
            },
            {
                "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-
instance-replica-1",
                "TargetMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-
instance-replica-1-green-k5fv7u",
                "Status": "AVAILABLE"
            },
            {
                "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-
instance-replica-2",
                "TargetMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-
instance-replica-2-green-ggsh8m",
                "Status": "AVAILABLE"
            },
            {
                "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-
instance-replica-3",
                "TargetMember": "arn:aws:rds:us-east-1:123456789012:db:my-db-
instance-replica-3-green-o2vwm0",
                "Status": "AVAILABLE"
            }
        ],
        "Tasks": [
            {
                "Name": "CREATING_READ_REPLICA_OF_SOURCE",
                "Status": "COMPLETED"
            },
            {
                "Name": "DB_ENGINE_VERSION_UPGRADE",
                "Status": "COMPLETED"
            },
            {
                "Name": "CONFIGURE_BACKUPS",
                "Status": "COMPLETED"
```

```
            },
            {
                "Name": "CREATING_TOPOLOGY_OF_SOURCE",
                "Status": "COMPLETED"
            }
        ],
        "Status": "SWITCHOVER_IN_PROGRESS",
        "CreateTime": "2022-02-25T22:33:22.225000+00:00"
    }
}
```

Per ulteriori informazioni, consulta Switching a blue/green deployment nella Amazon RDS User Guide.

Esempio 2: promuovere una blue/green distribuzione per un cluster Aurora MySQL DB

L'esempio `switchover-blue-green-deployment` seguente promuove l'ambiente verde specificato a nuovo ambiente di produzione.

```
aws rds switchover-blue-green-deployment \
    --blue-green-deployment-identifier bgd-wi89nwzglccsfake \
    --switchover-timeout 300
```

Output:

```
{
    "BlueGreenDeployment": {
        "BlueGreenDeploymentIdentifier": "bgd-wi89nwzglccsfake",
        "BlueGreenDeploymentName": "my-blue-green-deployment",
        "Source": "arn:aws:rds:us-east-1:123456789012:cluster:my-aurora-mysql-
cluster",
        "Target": "arn:aws:rds:us-east-1:123456789012:cluster:my-aurora-mysql-
cluster-green-3ud8z6",
        "SwitchoverDetails": [
            {
                "SourceMember": "arn:aws:rds:us-east-1:123456789012:cluster:my-
aurora-mysql-cluster",
                "TargetMember": "arn:aws:rds:us-east-1:123456789012:cluster:my-
aurora-mysql-cluster-green-3ud8z6",
                "Status": "AVAILABLE"
            },
            {
```

```
                "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-aurora-
mysql-cluster-1",
                "TargetMember": "arn:aws:rds:us-east-1:123456789012:db:my-aurora-
mysql-cluster-1-green-bvxc73",
                "Status": "AVAILABLE"
            },
            {
                "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-aurora-
mysql-cluster-2",
                "TargetMember": "arn:aws:rds:us-east-1:123456789012:db:my-aurora-
mysql-cluster-2-green-7wc4ie",
                "Status": "AVAILABLE"
            },
            {
                "SourceMember": "arn:aws:rds:us-east-1:123456789012:db:my-aurora-
mysql-cluster-3",
                "TargetMember": "arn:aws:rds:us-east-1:123456789012:db:my-aurora-
mysql-cluster-3-green-p4xxkz",
                "Status": "AVAILABLE"
            },
            {
                "SourceMember": "arn:aws:rds:us-east-1:123456789012:cluster-
endpoint:my-excluded-member-endpoint",
                "TargetMember": "arn:aws:rds:us-east-1:123456789012:cluster-
endpoint:my-excluded-member-endpoint-green-np1ikl",
                "Status": "AVAILABLE"
            },
            {
                "SourceMember": "arn:aws:rds:us-east-1:123456789012:cluster-
endpoint:my-reader-endpoint",
                "TargetMember": "arn:aws:rds:us-east-1:123456789012:cluster-
endpoint:my-reader-endpoint-green-miszlf",
                "Status": "AVAILABLE"
            }
        ],
        "Tasks": [
            {
                "Name": "CREATING_READ_REPLICA_OF_SOURCE",
                "Status": "COMPLETED"
            },
            {
                "Name": "DB_ENGINE_VERSION_UPGRADE",
                "Status": "COMPLETED"
            },
```

```
            {
                "Name": "CREATE_DB_INSTANCES_FOR_CLUSTER",
                "Status": "COMPLETED"
            },
            {
                "Name": "CREATE_CUSTOM_ENDPOINTS",
                "Status": "COMPLETED"
            }
        ],
        "Status": "SWITCHOVER_IN_PROGRESS",
        "CreateTime": "2022-02-25T22:38:49.522000+00:00"
    }
}
```

Per ulteriori informazioni, consulta Cambiare una blue/green distribuzione nella Guida per l'utente di Amazon Aurora.

- Per i dettagli sull'API, consulta SwitchoverBlueGreenDeployment AWS CLICommand Reference.

Esempi di Amazon RDS Data Service utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface con Amazon RDS Data Service.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`batch-execute-statement`

Il seguente esempio di codice mostra come usare`batch-execute-statement`.

AWS CLI

Come eseguire un'istruzione SQL in batch

L'esempio `batch-execute-statement` seguente esegue un'istruzione SQL batch su un array di dati con un set di parametri.

```
aws rds-data batch-execute-statement \
    --resource-arn "arn:aws:rds:us-west-2:123456789012:cluster:mydbcluster" \
    --database "mydb" \
    --secret-arn "arn:aws:secretsmanager:us-west-2:123456789012:secret:mysecret" \
    --sql "insert into mytable values (:id, :val)" \
    --parameter-sets "[[{\"name\": \"id\", \"value\": {\"longValue\": 1}},{\"name\":
 \"val\", \"value\": {\"stringValue\": \"ValueOne\"}}],
       [{\"name\": \"id\", \"value\": {\"longValue\": 2}},{\"name\": \"val\",
 \"value\": {\"stringValue\": \"ValueTwo\"}}],
       [{\"name\": \"id\", \"value\": {\"longValue\": 3}},{\"name\": \"val\",
 \"value\": {\"stringValue\": \"ValueThree\"}}]]"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Utilizzo dell'API dati per Aurora Serverless nella Guida per l'utente di Amazon RDS.

- Per i dettagli sull'API, consulta BatchExecuteStatement AWS CLICommand Reference.

`begin-transaction`

Il seguente esempio di codice mostra come utilizzare`begin-transaction`.

AWS CLI

Come avviare una transazione SQL

L'esempio `begin-transaction` seguente esegue una transazione SQL.

```
aws rds-data begin-transaction \
    --resource-arn "arn:aws:rds:us-west-2:123456789012:cluster:mydbcluster" \
    --database "mydb" \
    --secret-arn "arn:aws:secretsmanager:us-west-2:123456789012:secret:mysecret"
```

Output:

```
{
    "transactionId": "ABC1234567890xyz"
}
```

Per ulteriori informazioni, consulta Utilizzo dell'API dati per Aurora Serverless nella Guida per l'utente di Amazon RDS.

- Per i dettagli sull'API, consulta BeginTransaction AWS CLICommand Reference.

`commit-transaction`

Il seguente esempio di codice mostra come utilizzare`commit-transaction`.

AWS CLI

Come eseguire il commit di una transazione SQL

L'esempio `commit-transaction` seguente termina la transazione SQL specificata e conferma le modifiche apportate come parte di essa.

```
aws rds-data commit-transaction \
    --resource-arn "arn:aws:rds:us-west-2:123456789012:cluster:mydbcluster" \
    --secret-arn "arn:aws:secretsmanager:us-west-2:123456789012:secret:mysecret" \
    --transaction-id "ABC1234567890xyz"
```

Output:

```
{
    "transactionStatus": "Transaction Committed"
}
```

Per ulteriori informazioni, consulta Utilizzo dell'API dati per Aurora Serverless nella Guida per l'utente di Amazon RDS.

- Per i dettagli sull'API, consulta CommitTransaction AWS CLICommand Reference.

`execute-statement`

Il seguente esempio di codice mostra come utilizzare`execute-statement`.

AWS CLI

Esempio 1: come eseguire un'istruzione SQL inclusa in una transazione

L'esempio `execute-statement` seguente esegue un'istruzione SQL che fa parte di una transazione.

```
aws rds-data execute-statement \
    --resource-arn "arn:aws:rds:us-west-2:123456789012:cluster:mydbcluster" \
    --database "mydb" \
    --secret-arn "arn:aws:secretsmanager:us-west-2:123456789012:secret:mysecret" \
    --sql "update mytable set quantity=5 where id=201" \
    --transaction-id "ABC1234567890xyz"
```

Output:

```
{
    "numberOfRecordsUpdated": 1
}
```

Esempio 2: come eseguire un'istruzione SQL con parametri

L'esempio `execute-statement` seguente esegue un'istruzione SQL con parametri.

```
aws rds-data execute-statement \
    --resource-arn "arn:aws:rds:us-east-1:123456789012:cluster:mydbcluster" \
    --database "mydb" \
    --secret-arn "arn:aws:secretsmanager:us-east-1:123456789012:secret:mysecret" \
    --sql "insert into mytable values (:id, :val)" \
    --parameters "[{\"name\": \"id\", \"value\": {\"longValue\": 1}},{\"name\":
 \"val\", \"value\": {\"stringValue\": \"value1\"}}]"
```

Output:

```
{
    "numberOfRecordsUpdated": 1
}
```

Per ulteriori informazioni, consulta Utilizzo dell'API dati per Aurora Serverless nella Guida per l'utente di Amazon RDS.

- Per i dettagli sull'API, consulta ExecuteStatement AWS CLICommand Reference.

`rollback-transaction`

Il seguente esempio di codice mostra come utilizzare`rollback-transaction`.

AWS CLI

Come eseguire il rollback di una transazione SQL

L'esempio `rollback-transaction` seguente ripristina la transazione SQL specificata.

```
aws rds-data rollback-transaction \
    --resource-arn "arn:aws:rds:us-west-2:123456789012:cluster:mydbcluster" \
    --secret-arn "arn:aws:secretsmanager:us-west-2:123456789012:secret:mysecret" \
    --transaction-id "ABC1234567890xyz"
```

Output:

```
{
    "transactionStatus": "Rollback Complete"
}
```

Per ulteriori informazioni, consulta Utilizzo dell'API dati per Aurora Serverless nella Guida per l'utente di Amazon RDS.

- Per i dettagli sull'API, consulta RollbackTransaction AWS CLICommand Reference.

Esempi di utilizzo di Amazon RDS Performance Insights AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando Amazon RDS Performance Insights. AWS Command Line Interface

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`create-performance-analysis-report`

Il seguente esempio di codice mostra come usare`create-performance-analysis-report`.

AWS CLI

Come creare un report di analisi delle prestazioni

L'esempio `create-performance-analysis-report` seguente crea un report di analisi delle prestazioni con l'ora di inizio `1682969503` e l'ora di fine `1682979503` per il database `db-abcdefg123456789`.

```
aws pi create-performance-analysis-report \
    --service-type RDS \
    --identifier db-abcdefg123456789 \
    --start-time 1682969503 \
    --end-time 1682979503
```

Output:

```
{
    "AnalysisReportId": "report-0234d3ed98e28fb17"
}
```

Per ulteriori informazioni sulla creazione di report di analisi delle prestazioni, consulta Creazione di un report di analisi delle prestazioni in Approfondimenti sulle prestazioni nella Guida per l'utente di Amazon RDS e Creazione di un report di analisi delle prestazioni in Approfondimenti sulle prestazioni nella Guida per l'utente di Amazon Aurora.

- Per i dettagli sull'API, consulta CreatePerformanceAnalysisReport AWS CLICommand Reference.

`delete-performance-analysis-report`

Il seguente esempio di codice mostra come utilizzare`delete-performance-analysis-report`.

AWS CLI

Come eliminare un report di analisi delle prestazioni

L'esempio `delete-performance-analysis-report` seguente elimina il report di analisi delle prestazioni con l'ID report `report-0d99cc91c4422ee61`.

```
aws pi delete-performance-analysis-report \
    --service-type RDS \
    --identifier db-abcdefg123456789 \
    --analysis-report-id report-0d99cc91c4422ee61
```

Questo comando non produce alcun output.

Per ulteriori informazioni sull'eliminazione di report di analisi delle prestazioni, consulta Eliminazione di un report di analisi delle prestazioni in Approfondimenti sulle prestazioni nella Guida per l'utente di Amazon RDS e Eliminazione di un report di analisi delle prestazioni in Approfondimenti sulle prestazioni nella Guida per l'utente di Amazon Aurora.

- Per i dettagli sull'API, consulta DeletePerformanceAnalysisReport AWS CLICommand Reference.

`describe-dimension-keys`

Il seguente esempio di codice mostra come utilizzare`describe-dimension-keys`.

AWS CLI

Esempio 1: come descrivere le chiavi dimensionali

Questo esempio richiede i nomi di tutti gli eventi di attesa. I dati vengono riepilogati in base al nome dell'evento e ai valori aggregati di tali eventi nel periodo di tempo specificato.

Comando:

```
aws pi describe-dimension-keys --service-type RDS --identifier db-
LKCGOBK26374TPTDFXOIWVCPPM --start-time 1527026400 --end-time 1527080400 --
metric db.load.avg --group-by '{"Group":"db.wait_event"}'
```

Output:

```
{
    "AlignedEndTime": 1.5270804E9,
    "AlignedStartTime": 1.5270264E9,
    "Keys": [
        {
```

```
            "Dimensions": {"db.wait_event.name": "wait/synch/mutex/innodb/
aurora_lock_thread_slot_futex"},
            "Total": 0.05906906851195666
        },
        {
            "Dimensions": {"db.wait_event.name": "wait/io/aurora_redo_log_flush"},
            "Total": 0.015824722186149193
        },
        {
            "Dimensions": {"db.wait_event.name": "CPU"},
            "Total": 0.008014396230265477
        },
        {
            "Dimensions": {"db.wait_event.name": "wait/io/
aurora_respond_to_client"},
            "Total": 0.0036361612526204477
        },
        {
            "Dimensions": {"db.wait_event.name": "wait/io/table/sql/handler"},
            "Total": 0.0019108398419382965
        },
        {
            "Dimensions": {"db.wait_event.name": "wait/synch/cond/mysys/
my_thread_var::suspend"},
            "Total": 8.533847837782684E-4
        },
        {
            "Dimensions": {"db.wait_event.name": "wait/io/file/csv/data"},
            "Total": 6.864181956477376E-4
        },
        {
            "Dimensions": {"db.wait_event.name": "Unknown"},
            "Total": 3.895887056379051E-4
        },
        {
            "Dimensions": {"db.wait_event.name": "wait/synch/mutex/sql/
FILE_AS_TABLE::LOCK_shim_lists"},
            "Total": 3.710368625122906E-5
        },
        {
            "Dimensions": {"db.wait_event.name": "wait/lock/table/sql/handler"},
            "Total": 0
        }
    ]
```

```
}
```

Esempio 2: come trovare l'ID SQL per le istruzioni che contribuiscono maggiormente al caricamento del database

L'esempio `describe-dimension-keys` seguente richiede l'istruzione SQL e l'ID SQL per le 10 istruzioni che hanno contribuito maggiormente al caricamento del database.

```
aws pi describe-dimension-keys \
    --service-type RDS \
    --identifier db-abcdefg123456789 \
    --start-time 2023-05-01T00:00:00Z \
    --end-time 2023-05-01T01:00:00Z \
    --metric db.load.avg \
    --group-by '{"Group": "db.sql", "Dimensions": ["db.sql.id",
 "db.sql.statement"],"Limit": 10}'
```

Output:

```
{
    "AlignedEndTime": 1.5270804E9,
    "AlignedStartTime": 1.5270264E9,
    "Identifier": "db-abcdefg123456789",
    "MetricList": [
        {
            "Keys": [
                {
                    "Dimensions": {"db.sql.id": "AKIAIOSFODNN7EXAMPLE",
 "db.sql.statement": "SELECT * FROM customers WHERE customer_id = 123"},
                    "Total": 25.5,"Partitions": [12.3, 13.2]
                }
            ]
        }
    ]
}
```

Per ulteriori informazioni sulle dimensioni in Approfondimenti sulle prestazioni, consulta Caricamento database nella Guida per l'utente di Amazon RDS e Caricamento database nella Guida per l'utente di Amazon Aurora.

- Per i dettagli sull'API, consulta DescribeDimensionKeys AWS CLICommand Reference.

`get-dimension-key-details`

Il seguente esempio di codice mostra come utilizzare`get-dimension-key-details`.

AWS CLI

Come ottenere dettagli su un gruppo di dimensioni specifico per un'istanza database

L'esempio `get-dimension-key-details` seguente recupera il testo completo di un'istruzione SQL per l'istanza database `db-10BCD2EFGHIJ3KL4M5NO6PQRS5`. `--group` è `db.sql`, e `--group-identifier` è `db.sql.id`. In questo esempio, `example-sql-id` rappresenta un ID SQL recuperato utilizzando le operazioni `describe-dimension-keys` o `get-resource-metrics`. In questo esempio, sono disponibili i dettagli delle dimensioni. Pertanto, Performance Insights recupera il testo completo dell'istruzione SQL, senza troncarla.

```
aws pi get-dimension-key-details \
    --service-type RDS \
    --identifier db-10BCD2EFGHIJ3KL4M5NO6PQRS5 \
    --group db.sql \
    --group-identifier example-sql-id \
    --requested-dimensions statement
```

Output:

```
{
    "Dimensions":[
        {
            "Value": "SELECT e.last_name, d.department_name FROM employees e,
 departments d WHERE e.department_id=d.department_id",
            "Dimension": "db.sql.statement",
            "Status": "AVAILABLE"
        },
    ...
    ]
}
```

Per ulteriori informazioni sulle dimensioni in Approfondimenti sulle prestazioni, consulta Caricamento database nella Guida per l'utente di Amazon RDS e Caricamento database nella Guida per l'utente di Amazon Aurora.

- Per i dettagli sull'API, consulta GetDimensionKeyDetails AWS CLICommand Reference.

`get-performance-analysis-report`

Il seguente esempio di codice mostra come utilizzare`get-performance-analysis-report`.

AWS CLI

Come visualizzare un report di analisi delle prestazioni

L'esempio `get-performance-analysis-report` seguente ottiene il report di analisi delle prestazioni per il database `db-abcdefg123456789` con l'ID report `report-0d99cc91c4422ee61`. La risposta fornisce lo stato, l'ID, i dettagli temporali e gli approfondimenti del report.

```
aws pi get-performance-analysis-report \
    --service-type RDS \
    --identifier db-abcdefg123456789 \
    --analysis-report-id report-0d99cc91c4422ee61
```

Output:

```
{
    "AnalysisReport": {
        "Status": "Succeeded",
        "ServiceType": "RDS",
        "Identifier": "db-abcdefg123456789",
        "StartTime": 1680583486.584,
        "AnalysisReportId": "report-0d99cc91c4422ee61",
        "EndTime": 1680587086.584,
        "CreateTime": 1680587087.139,
        "Insights": [
            ... (Condensed for space)
       ]
    }
}
```

Per ulteriori informazioni sui report di analisi delle prestazioni, consulta Analisi delle prestazioni del database per un periodo di tempo nella Guida per l'utente di Amazon RDS e Analisi delle prestazioni del database per un periodo di tempo nella Guida per l'utente di Amazon Aurora.

- Per i dettagli sull'API, consulta GetPerformanceAnalysisReport AWS CLICommand Reference.

`get-resource-metadata`

Il seguente esempio di codice mostra come utilizzare`get-resource-metadata`.

AWS CLI

Come ottenere i metadati delle risorse per un database

L'esempio `get-resource-metadata` seguente ottiene i metadati delle risorse per il database `db-abcdefg123456789`. La risposta mostra che le statistiche digest di SQL sono abilitate.

```
aws pi get-resource-metadata \
    --service-type RDS \
    --identifier db-abcdefg123456789
```

Output:

```
{
    "Identifier": "db-abcdefg123456789",
    "Features":{
        "SQL_DIGEST_STATISTICS":{
            "Status": "ENABLED"
        }
    }
}
```

Per ulteriori informazioni sulle statistiche SQL per Approfondimenti sulle prestazioni, consulta Statistiche SQL per Performance Insights nella Guida per l'utente di Amazon RDS User Guide e Statistiche SQL per Performance Insights nella Guida per l'utente di Amazon Aurora.

- Per i dettagli sull'API, consulta GetResourceMetadata AWS CLICommand Reference.

`get-resource-metrics`

Il seguente esempio di codice mostra come utilizzare`get-resource-metrics`.

AWS CLI

Come ottenere le metriche delle risorse

Questo esempio richiede punti dati per il gruppo di dimensioni db.wait_event e per la dimensione db.wait_event.name all'interno di quel gruppo. Nella risposta, i punti dati pertinenti vengono raggruppati in base alla dimensione richiesta (db.wait_event.name).

Comando:

```
aws pi get-resource-metrics --service-type RDS --identifier db-LKCGOBK26374TPTDFXOIWVCPPM --start-time 1527026400 --end-time 1527080400 --period-in-seconds 300 --metric db.load.avg --metric-queries file://metric-queries.json
```

Gli argomenti per `--metric-queries` sono memorizzati in un file JSON, `metric-queries.json`. Ecco i contenuti del file:

```
[
    {
        "Metric": "db.load.avg",
        "GroupBy": {
            "Group":"db.wait_event"
        }
    }
]
```

Output:

```
{
    "AlignedEndTime": 1.5270804E9,
    "AlignedStartTime": 1.5270264E9,
    "Identifier": "db-LKCGOBK26374TPTDFXOIWVCPPM",
    "MetricList": [
        {
            "Key": {
                "Metric": "db.load.avg"
            },
            "DataPoints": [
                {
                    "Timestamp": 1527026700.0,
                    "Value": 1.3533333333333333
                },
                {
                    "Timestamp": 1527027000.0,
                    "Value": 0.88
                },
```

```
                <...remaining output omitted...>
            ]
        },
        {
            "Key": {
                "Metric": "db.load.avg",
                "Dimensions": {
                    "db.wait_event.name": "wait/synch/mutex/innodb/
aurora_lock_thread_slot_futex"
                }
            },
            "DataPoints": [
                {
                    "Timestamp": 1527026700.0,
                    "Value": 0.8566666666666667
                },
                {
                    "Timestamp": 1527027000.0,
                    "Value": 0.8633333333333333
                },
                <...remaining output omitted...>
            ],
        },
            <...remaining output omitted...>
    ]
}
```

- Per i dettagli sull'API, consulta GetResourceMetrics AWS CLICommand Reference.

`list-available-resource-dimensions`

Il seguente esempio di codice mostra come utilizzare`list-available-resource-dimensions`.

AWS CLI

Come elencare le dimensioni che possono essere interrogate per un tipo di parametro su un'istanza database

L'esempio `list-available-resource-dimensions` seguente elenca le metriche `db.load` su cui è possibile eseguire una query per il database `db-abcdefg123456789`.

```
aws pi list-available-resource-dimensions \
```

```
    --service-type RDS \
    --identifier db-abcdefg123456789 \
    --metrics db.load
```

Output:

```
{
    "MetricDimensions": [
        {
            "Metric": "db.load",
            "Groups": [
                {
                    "Group": "db.user",
                    "Dimensions": [
                        {
                            "Identifier": "db.user.id"
                        },
                        {
                            "Identifier": "db.user.name"
                        }
                    ]
                },
                {
                    "Group": "db.sql_tokenized",
                    "Dimensions": [
                        {
                            "Identifier": "db.sql_tokenized.id"
                        },
                        {
                            "Identifier": "db.sql_tokenized.db_id"
                        },
                        {
                            "Identifier": "db.sql_tokenized.statement"
                        }
                    ]
                },
                ...
            ]
        }
    ]
}
```

Per ulteriori informazioni sulle dimensioni in Approfondimenti sulle prestazioni, consulta Caricamento database nella Guida per l'utente di Amazon RDS e Caricamento database nella Guida per l'utente di Amazon Aurora.

- Per i dettagli sull'API, consulta ListAvailableResourceDimensions AWS CLICommand Reference.

`list-available-resource-metrics`

Il seguente esempio di codice mostra come utilizzare`list-available-resource-metrics`.

AWS CLI

Come elencare le metriche che possono essere interrogate per un tipo di parametro su un'istanza database

L'esempio `list-available-resource-metrics` seguente elenca le metriche `db.load` su cui è possibile eseguire una query per il database `db-abcdefg123456789`.

```
aws pi list-available-resource-metrics \
    --service-type RDS \
    --identifier db-abcdefg123456789 \
    --metric-types "os" "db"
```

Output:

```
{
    "Metrics": [
        {
            "Description": "The number of virtual CPUs for the DB instance",
            "Metric": "os.general.numVCPUs",
            "Unit": "vCPUs"
        },
        ......,
        {
            "Description": "Time spent reading data file blocks by backends in this
 instance",
            "Metric": "db.IO.read_latency",
            "Unit": "Milliseconds per block"
        },
        ......
```

```
    ]
}
```

Per ulteriori informazioni sulle metriche in Approfondimenti sulle prestazioni, consulta Caricamento database nella Guida per l'utente di Amazon RDS e Caricamento database nella Guida per l'utente di Amazon Aurora.

- Per i dettagli sull'API, consulta ListAvailableResourceMetrics AWS CLICommand Reference.

`list-performance-analysis-reports`

Il seguente esempio di codice mostra come utilizzare`list-performance-analysis-reports`.

AWS CLI

Come elencare i report di analisi delle prestazioni per un database

L'esempio `list-performance-analysis-reports` seguente elenca tutti i report di analisi delle prestazioni disponibili per il database `db-abcdefg123456789`. La risposta elenca tutti i report con i dettagli relativi all'ID, allo stato e al periodo di tempo del report.

```
aws pi list-performance-analysis-reports \
    --service-type RDS \
    --identifier db-abcdefg123456789
```

Output:

```
{
    "AnalysisReports": [
        {
            "Status": "Succeeded",
            "EndTime": 1680587086.584,
            "CreateTime": 1680587087.139,
            "StartTime": 1680583486.584,
            "AnalysisReportId": "report-0d99cc91c4422ee61"
        },
        {
            "Status": "Succeeded",
            "EndTime": 1681491137.914,
            "CreateTime": 1681491145.973,
            "StartTime": 1681487537.914,
```

```
            "AnalysisReportId": "report-002633115cc002233"
        },
        {
            "Status": "Succeeded",
            "EndTime": 1681493499.849,
            "CreateTime": 1681493507.762,
            "StartTime": 1681489899.849,
            "AnalysisReportId": "report-043b1e006b47246f9"
        },
        {
            "Status": "InProgress",
            "EndTime": 1682979503.0,
            "CreateTime": 1682979618.994,
            "StartTime": 1682969503.0,
            "AnalysisReportId": "report-01ad15f9b88bcbd56"
        }
    ]
}
```

Per ulteriori informazioni sui report di analisi delle prestazioni, consulta Analisi delle prestazioni del database per un periodo di tempo nella Guida per l'utente di Amazon RDS e Analisi delle prestazioni del database per un periodo di tempo nella Guida per l'utente di Amazon Aurora.

- Per i dettagli sull'API, consulta ListPerformanceAnalysisReports AWS CLICommand Reference.

`list-tags-for-resource`

Il seguente esempio di codice mostra come utilizzare`list-tags-for-resource`.

AWS CLI

Come elencare tutti i tag per un report di analisi delle prestazioni

L'esempio `list-tags-for-resource` seguente elenca i tag per un report di analisi delle prestazioni con l'ID report `report-0d99cc91c4422ee61`.

```
aws pi list-tags-for-resource \
    --service-type RDS \
    --resource-arn arn:aws:pi:us-west-2:123456789012:perf-reports/RDS/db-
abcdefg123456789/report-0d99cc91c4422ee61
```

Output:

```
{
    "Tags": [
        {
            "Value": "test-tag",
            "Key": "name"
        }
    ]
}
```

Per ulteriori informazioni sull'aggiunta di tag ai report di analisi delle prestazioni, consulta Aggiunta di tag a un report di analisi delle prestazioni in Approfondimenti sulle prestazioni nella Guida per l'utente di Amazon RDS e Aggiunta di tag a un report di analisi delle prestazioni in Approfondimenti sulle prestazioni nella Guida per l'utente di Amazon Aurora.

- Per i dettagli sull'API, consulta ListTagsForResource AWS CLICommand Reference.

`tag-resource`

Il seguente esempio di codice mostra come utilizzare`tag-resource`.

AWS CLI

Come aggiungere un tag a un report di analisi delle prestazioni

L'esempio `tag-resource` seguente aggiunge la chiave di tag `name` con il valore di tag `test-tag` a un report di analisi delle prestazioni con l'ID report `report-0d99cc91c4422ee61`.

```
aws pi tag-resource \
    --service-type RDS \
    --resource-arn arn:aws:pi:us-west-2:123456789012:perf-reports/RDS/db-
abcdefg123456789/report-0d99cc91c4422ee61 \
    --tags Key=name,Value=test-tag
```

Questo comando non produce alcun output.

Per ulteriori informazioni sull'aggiunta di tag ai report di analisi delle prestazioni, consulta Aggiunta di tag a un report di analisi delle prestazioni in Approfondimenti sulle prestazioni nella Guida per l'utente di Amazon RDS e Aggiunta di tag a un report di analisi delle prestazioni in Approfondimenti sulle prestazioni nella Guida per l'utente di Amazon Aurora.

- Per i dettagli sull'API, consulta TagResource AWS CLICommand Reference.

`untag-resource`

Il seguente esempio di codice mostra come utilizzare`untag-resource`.

AWS CLI

Come eliminare un tag di un report di analisi delle prestazioni

L'esempio `untag-resource` seguente elimina il tag `name` del report di analisi delle prestazioni con l'ID report `report-0d99cc91c4422ee61`.

```
aws pi untag-resource \
    --service-type RDS \
    --resource-arn arn:aws:pi:us-west-2:123456789012:perf-reports/RDS/db-abcdefg123456789/report-0d99cc91c4422ee61 \
    --tag-keys name
```

Questo comando non produce alcun output.

Per ulteriori informazioni sull'aggiunta di tag ai report di analisi delle prestazioni, consulta Aggiunta di tag a un report di analisi delle prestazioni in Approfondimenti sulle prestazioni nella Guida per l'utente di Amazon RDS e Aggiunta di tag a un report di analisi delle prestazioni in Approfondimenti sulle prestazioni nella Guida per l'utente di Amazon Aurora.

- Per i dettagli sull'API, consulta UntagResource AWS CLICommand Reference.

Esempi di utilizzo di Amazon Redshift AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando Amazon Redshift. AWS Command Line Interface

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`accept-reserved-node-exchange`

Il seguente esempio di codice mostra come usare`accept-reserved-node-exchange`.

AWS CLI

Come accettare lo scambio di nodi riservati

L'`accept-reserved-node-exchange`esempio seguente accetta lo scambio di un nodo DC1 riservato con un nodo DC2 riservato.

```
aws redshift accept-reserved-node-exchange /
    --reserved-node-id 12345678-12ab-12a1-1a2a-12ab-12a12EXAMPLE /
    --target-reserved-node-offering-id 12345678-12ab-12a1-1a2a-12ab-12a12EXAMPLE
```

Output:

```
{
    "ExchangedReservedNode": {
        "ReservedNodeId": "12345678-12ab-12a1-1a2a-12ab-12a12EXAMPLE",
        "ReservedNodeOfferingId": "12345678-12ab-12a1-1a2a-12ab-12a12EXAMPLE",
        "NodeType": "dc2.large",
        "StartTime": "2019-12-06T21:17:26Z",
        "Duration": 31536000,
        "FixedPrice": 0.0,
        "UsagePrice": 0.0,
        "CurrencyCode": "USD",
        "NodeCount": 1,
        "State": "exchanging",
        "OfferingType": "All Upfront",
        "RecurringCharges": [
            {
                "RecurringChargeAmount": 0.0,
                "RecurringChargeFrequency": "Hourly"
            }
        ],
        "ReservedNodeOfferingType": "Regular"
    }
}
```

Per ulteriori informazioni, consulta Upgrade Reserved Nodes With the AWS CLI nella Amazon Redshift Cluster Management Guide.

- Per i dettagli sull'API, consulta AcceptReservedNodeExchangeCommand Reference.AWS CLI

`authorize-cluster-security-group-ingress`

Il seguente esempio di codice mostra come utilizzare`authorize-cluster-security-group-ingress`.

AWS CLI

L' GroupThis esempio di autorizzazione dell'accesso a una EC2 sicurezza autorizza l'accesso a un gruppo di EC2 sicurezza Amazon denominato. Comando:

```
aws redshift authorize-cluster-security-group-ingress --cluster-security-group-name
 mysecuritygroup --ec2-security-group-name myec2securitygroup --ec2-security-group-
owner-id 123445677890
```

Come autorizzare l'accesso a un intervallo CIDR L'esempio seguente autorizza l'accesso a un intervallo CIDR. Comando:

```
aws redshift authorize-cluster-security-group-ingress --cluster-security-group-name
 mysecuritygroup --cidrip 192.168.100.100/32
```

- Per i dettagli sull'API, consulta AuthorizeClusterSecurityGroupIngressCommand Reference.AWS CLI

`authorize-snapshot-access`

Il seguente esempio di codice mostra come utilizzare`authorize-snapshot-access`.

AWS CLI

Autorizza un AWS account al ripristino Un SnapshotThis esempio autorizza l' AWS account `444455556666` a ripristinare l'istantanea. `my-snapshot-id` Per impostazione predefinita, l'output è in formato JSON. Comando:

```
aws redshift authorize-snapshot-access --snapshot-id my-snapshot-id --account-with-
restore-access 444455556666
```

Risultato:

```
{
   "Snapshot": {
      "Status": "available",
      "SnapshotCreateTime": "2013-07-17T22:04:18.947Z",
      "EstimatedSecondsToCompletion": 0,
      "AvailabilityZone": "us-east-1a",
      "ClusterVersion": "1.0",
      "MasterUsername": "adminuser",
      "Encrypted": false,
      "OwnerAccount": "111122223333",
      "BackupProgressInMegabytes": 11.0,
      "ElapsedTimeInSeconds": 0,
      "DBName": "dev",
      "CurrentBackupRateInMegabytesPerSecond: 0.1534,
      "ClusterCreateTime": "2013-01-22T21:59:29.559Z",
      "ActualIncrementalBackupSizeInMegabytes"; 11.0,
      "SnapshotType": "manual",
      "NodeType": "dw.hs1.xlarge",
      "ClusterIdentifier": "mycluster",
      "TotalBackupSizeInMegabytes": 20.0,
      "Port": 5439,
      "NumberOfNodes": 2,
      "SnapshotIdentifier": "my-snapshot-id"
   }
}
```

- Per i dettagli sull'API, consulta Command AuthorizeSnapshotAccessReference AWS CLI .

`batch-delete-cluster-snapshots`

Il seguente esempio di codice mostra come utilizzare`batch-delete-cluster-snapshots`.

AWS CLI

Come eliminare un set di snapshot del cluster

L'esempio `batch-delete-cluster-snapshots` seguente elimina un set di snapshot del cluster manuali.

```
aws redshift batch-delete-cluster-snapshots \
```

```
        --
identifiers SnapshotIdentifier=mycluster-2019-11-06-14-12 SnapshotIdentifier=mycluster-2019-
```

Output:

```
{
    "Resources": [
        "mycluster-2019-11-06-14-12",
        "mycluster-2019-11-06-14-20"
    ]
}
```

Per ulteriori informazioni, consulta Snapshot di Amazon Redshift nella Guida alla gestione dei cluster di Amazon Redshift.

- Per i dettagli sull'API, consulta BatchDeleteClusterSnapshots AWS CLICommand Reference.

`batch-modify-cluster-snapshots`

Il seguente esempio di codice mostra come utilizzare`batch-modify-cluster-snapshots`.

AWS CLI

Come modificare un set di snapshot del cluster

L'esempio `batch-modify-cluster-snapshots` seguente modifica le impostazioni per un set di snapshot del cluster.

```
aws redshift batch-modify-cluster-snapshots \
    --snapshot-identifier-list mycluster-2019-11-06-16-31 mycluster-2019-11-06-16-32
 \
    --manual-snapshot-retention-period 30
```

Output:

```
{
    "Resources": [
        "mycluster-2019-11-06-16-31",
        "mycluster-2019-11-06-16-32"
    ],
    "Errors": [],
```

```
    "ResponseMetadata": {
        "RequestId": "12345678-12ab-12a1-1a2a-12ab-12a12EXAMPLE",
        "HTTPStatusCode": 200,
        "HTTPHeaders": {
                "x-amzn-requestid": "12345678-12ab-12a1-1a2a-12ab-12a12EXAMPLE,
                "content-type": "text/xml",
                "content-length": "480",
                "date": "Sat, 07 Dec 2019 00:36:09 GMT",
                "connection": "keep-alive"
        },
        "RetryAttempts": 0
    }
}
```

Per ulteriori informazioni, consulta Snapshot di Amazon Redshift nella Guida alla gestione dei cluster di Amazon Redshift.

- Per i dettagli sull'API, consulta BatchModifyClusterSnapshots AWS CLICommand Reference.

`cancel-resize`

Il seguente esempio di codice mostra come utilizzare`cancel-resize`.

AWS CLI

Come annullare il ridimensionamento di un cluster

L'esempio `cancel-resize` seguente annulla una classica operazione di ridimensionamento per un cluster.

```
aws redshift cancel-resize \
    --cluster-identifier mycluster
```

Output:

```
{
    "TargetNodeType": "dc2.large",
    "TargetNumberOfNodes": 2,
    "TargetClusterType": "multi-node",
    "Status": "CANCELLING",
    "ResizeType": "ClassicResize",
    "TargetEncryptionType": "NONE"
```

```
}
```

Per ulteriori informazioni, consulta Ridimensionamento di cluster in Amazon Redshift nella Guida alla gestione dei cluster Amazon Redshift.

- Per i dettagli sull'API, consulta CancelResize AWS CLICommand Reference.

`copy-cluster-snapshot`

Il seguente esempio di codice mostra come utilizzare`copy-cluster-snapshot`.

AWS CLI

L' VersionsThis esempio Get a Description of All Cluster restituisce una descrizione di tutte le versioni del cluster. Per impostazione predefinita, l'output è in formato JSON. Comando:

```
aws redshift copy-cluster-snapshot --source-snapshot-identifier
 cm:examplecluster-2013-01-22-19-27-58 --target-snapshot-identifier my-saved-
snapshot-copy
```

Risultato:

```
{
   "Snapshot": {
      "Status": "available",
      "SnapshotCreateTime": "2013-01-22T19:27:58.931Z",
      "AvailabilityZone": "us-east-1c",
      "ClusterVersion": "1.0",
      "MasterUsername": "adminuser",
      "DBName": "dev",
      "ClusterCreateTime": "2013-01-22T19:23:59.368Z",
      "SnapshotType": "manual",
      "NodeType": "dw.hs1.xlarge",
      "ClusterIdentifier": "examplecluster",
      "Port": 5439,
      "NumberOfNodes": "2",
      "SnapshotIdentifier": "my-saved-snapshot-copy"
   },
   "ResponseMetadata": {
      "RequestId": "3b279691-64e3-11e2-bec0-17624ad140dd"
   }
}
```

- Per i dettagli sull'API, consulta CopyClusterSnapshot AWS CLICommand Reference.

`create-cluster-parameter-group`

Il seguente esempio di codice mostra come utilizzare`create-cluster-parameter-group`.

AWS CLI

L' GroupThis esempio di creazione di un parametro cluster crea un nuovo gruppo di parametri del cluster. Comando:

```
aws redshift create-cluster-parameter-group --parameter-group-name
 myclusterparametergroup --parameter-group-family redshift-1.0 --description "My
 first cluster parameter group"
```

Risultato:

```
{
   "ClusterParameterGroup": {
      "ParameterGroupFamily": "redshift-1.0",
      "Description": "My first cluster parameter group",
      "ParameterGroupName": "myclusterparametergroup"
   },
   "ResponseMetadata": {
      "RequestId": "739448f0-64cc-11e2-8f7d-3b939af52818"
   }
}
```

- Per i dettagli sull'API, consulta AWS CLI Command CreateClusterParameterGroupReference.

`create-cluster-security-group`

Il seguente esempio di codice mostra come utilizzare`create-cluster-security-group`.

AWS CLI

La creazione di un GroupThis esempio di Cluster Security crea un nuovo gruppo di sicurezza del cluster. Per impostazione predefinita, l'output è in formato JSON. Comando:

```
aws redshift create-cluster-security-group --cluster-security-group-name
 mysecuritygroup --description "This is my cluster security group"
```

Risultato:

```
{
   "create_cluster_security_group_response": {
      "create_cluster_security_group_result": {
         "cluster_security_group": {
            "description": "This is my cluster security group",
            "owner_id": "300454760768",
            "cluster_security_group_name": "mysecuritygroup",
            "ec2_security_groups": \[],
            "ip_ranges": \[]
         }
      },
      "response_metadata": {
         "request_id": "5df486a0-343a-11e2-b0d8-d15d0ef48549"
      }
   }
}
```

È inoltre possibile ottenere le stesse informazioni in formato testo utilizzando l'opzione `--output text`. Comando:

Opzione `--output text`. Comando:

Opzione. Comando:

```
aws redshift create-cluster-security-group --cluster-security-group-name
 mysecuritygroup --description "This is my cluster security group" --output text
```

Risultato:

```
This is my cluster security group   300454760768    mysecuritygroup
a0c0bfab-343a-11e2-95d2-c3dc9fe8ab57
```

- Per i dettagli sull'API, vedere CreateClusterSecurityGroupin AWS CLI Command Reference.

`create-cluster-snapshot`

Il seguente esempio di codice mostra come utilizzare`create-cluster-snapshot`.

AWS CLI

L' SnapshotThis esempio di creazione di un cluster crea una nuova istantanea del cluster. Per impostazione predefinita, l'output è in formato JSON. Comando:

```
aws redshift create-cluster-snapshot --cluster-identifier mycluster --snapshot-
identifier my-snapshot-id
```

Risultato:

```
{
   "Snapshot": {
      "Status": "creating",
      "SnapshotCreateTime": "2013-01-22T22:20:33.548Z",
      "AvailabilityZone": "us-east-1a",
      "ClusterVersion": "1.0",
      "MasterUsername": "adminuser",
      "DBName": "dev",
      "ClusterCreateTime": "2013-01-22T21:59:29.559Z",
      "SnapshotType": "manual",
      "NodeType": "dw.hs1.xlarge",
      "ClusterIdentifier": "mycluster",
      "Port": 5439,
      "NumberOfNodes": "2",
      "SnapshotIdentifier": "my-snapshot-id"
   },
   "ResponseMetadata": {
      "RequestId": "f024d1a5-64e1-11e2-88c5-53eb05787dfb"
   }
}
```

- Per i dettagli sull'API, consulta CreateClusterSnapshot AWS CLICommand Reference.

`create-cluster-subnet-group`

Il seguente esempio di codice mostra come utilizzare`create-cluster-subnet-group`.

AWS CLI

L' GroupThis esempio di creazione di una sottorete del cluster crea un nuovo gruppo di sottoreti del cluster.Comando:

```
aws redshift create-cluster-subnet-group --cluster-subnet-group-name mysubnetgroup
 --description "My subnet group" --subnet-ids subnet-763fdd1c
```

Risultato:

```
{
   "ClusterSubnetGroup": {
      "Subnets": [
         {
            "SubnetStatus": "Active",
            "SubnetIdentifier": "subnet-763fdd1c",
            "SubnetAvailabilityZone": {
               "Name": "us-east-1a"
            }
         } ],
      "VpcId": "vpc-7e3fdd14",
      "SubnetGroupStatus": "Complete",
      "Description": "My subnet group",
      "ClusterSubnetGroupName": "mysubnetgroup"
   },
   "ResponseMetadata": {
      "RequestId": "500b8ce2-698f-11e2-9790-fd67517fb6fd"
   }
}
```

- Per i dettagli sull'API, vedere CreateClusterSubnetGroupin Command Reference.AWS CLI

`create-cluster`

Il seguente esempio di codice mostra come utilizzare`create-cluster`.

AWS CLI

L' ParametersThis esempio Create a Cluster with Minimal crea un cluster con il set minimo di parametri. Per impostazione predefinita, l'output è in formato JSON. Comando:

```
aws redshift create-cluster --node-type dw.hs1.xlarge --number-of-nodes 2 --master-
username adminuser --master-user-password TopSecret1 --cluster-identifier mycluster
```

Risultato:

```
{
   "Cluster": {
      "NodeType": "dw.hs1.xlarge",
      "ClusterVersion": "1.0",
      "PubliclyAccessible": "true",
      "MasterUsername": "adminuser",
      "ClusterParameterGroups": [
         {
            "ParameterApplyStatus": "in-sync",
            "ParameterGroupName": "default.redshift-1.0"
         } ],
      "ClusterSecurityGroups": [
         {
            "Status": "active",
            "ClusterSecurityGroupName": "default"
         } ],
      "AllowVersionUpgrade": true,
      "VpcSecurityGroups": \[],
      "PreferredMaintenanceWindow": "sat:03:30-sat:04:00",
      "AutomatedSnapshotRetentionPeriod": 1,
      "ClusterStatus": "creating",
      "ClusterIdentifier": "mycluster",
      "DBName": "dev",
      "NumberOfNodes": 2,
      "PendingModifiedValues": {
         "MasterUserPassword": "\****"
      }
   },
   "ResponseMetadata": {
      "RequestId": "7cf4bcfc-64dd-11e2-bea9-49e0ce183f07"
   }
}
```

- Per i dettagli sull'API, consulta CreateCluster AWS CLICommand Reference.

`create-event-subscription`

Il seguente esempio di codice mostra come utilizzare`create-event-subscription`.

AWS CLI

Come creare una sottoscrizione alle notifiche per un evento

L'esempio `create-event-subscription` seguente crea una sottoscrizione alle notifiche di un evento.

```
aws redshift create-event-subscription \
    --subscription-name mysubscription \
    --sns-topic-arn arn:aws:sns:us-west-2:123456789012:MySNStopic \
    --source-type cluster \
    --source-ids mycluster
```

Output:

```
{
        "EventSubscription": {
        "CustomerAwsId": "123456789012",
        "CustSubscriptionId": "mysubscription",
        "SnsTopicArn": "arn:aws:sns:us-west-2:123456789012:MySNStopic",
        "Status": "active",
        "SubscriptionCreationTime": "2019-12-09T20:05:19.365Z",
        "SourceType": "cluster",
        "SourceIdsList": [
            "mycluster"
        ],
        "EventCategoriesList": [],
        "Severity": "INFO",
        "Enabled": true,
        "Tags": []
    }
}
```

Per ulteriori informazioni, consulta Sottoscrizione alle notifiche eventi di Amazon Redshift nella Guida alla gestione dei cluster di Amazon Redshift.

- Per i dettagli sull'API, consulta CreateEventSubscription AWS CLICommand Reference.

`create-hsm-client-certificate`

Il seguente esempio di codice mostra come utilizzare`create-hsm-client-certificate`.

AWS CLI

Come creare un certificato client HSM

L'esempio `create-hsm-client-certificate` seguente genera un certificato client HSM che un cluster può utilizzare per connettersi a un HSM.

```
aws redshift create-hsm-client-certificate \
    --hsm-client-certificate-identifier myhsmclientcert
```

Output:

```
{
    "HsmClientCertificate": {
        "HsmClientCertificateIdentifier": "myhsmclientcert",
        "HsmClientCertificatePublicKey": "-----BEGIN CERTIFICATE-----
        MIICiEXAMPLECQD6m7oRw0uXOjANBgkqhkiG9w0BAQUFADCBiDELMAkGA1UEBhMC
        VVMxCzAJBgNVBAgTEXAMPLEwDgYDVQQHEwdTZWF0dGxlMQ8wDQYDVQQKEwZBbWF6
        b24xFDASBgNVBAsTC0lBTSBDb25EXAMPLEIwEAYDVQQDEwlUZXN0Q2lsYWMxHzAd
        BgkqhkiG9w0BCQEWEG5vb25lQGFtYXpvbi5jb2EXAMPLETEwNDI1MjA0NTIxWhcN
        MTIwNDI0MjA0NTIxWjCBiDELMAkGA1UEBhMCVVMxCzAJBgNVBEXAMPLEMRAwDgYD
        EXAMPLETZWF0dGxlMQ8wDQYDVQQKEwZBbWF6b24xFDASBgNVBAsTC0lBTSBDb25z
        b2xlMRIwEAEXAMPLEwlUZXN0Q2lsYWMxHzAdBgkqhkiG9w0BCQEWEG5vb25lQGFt
        YXpvbi5jb20wgZ8wDQYJKEXAMPLEAQEBBQADgY0AMIGJAoGBAMaK0dn+a4GmWIWJ
        21uUSfwfEvySWtC2XADZ4nB+BLYgVIk6EXAMPLE3G93vUEIO3IyNoH/f0wYK8m9T
        rDHudUZg3qX4waLG5M43q7Wgc/MbQITxOUSQv7c7ugEXAMPLEzZswY6786m86gpE
        Ibb3OhjZnzcvQAaRHhdlQWIMm2nrAgMBAAEwDQYJKoZIhvcNAQEEXAMPLEEAtCu4
        nUhVVxYUEXAMPLEh8Mg9q6q+auNKyExzyLwaxlAoo7TJHidbtS4J5iNmZgXL0Fkb
        FFBjvSfpJIlJ00zbhNYS5f6GEXAMPLEl0ZxBHjJnyp378OD8uTs7fLvjx79LjSTb
        NYiytVbZPQUQ5Yaxu2jXnimvw3rEXAMPLE=-----END CERTIFICATE-----\n",
    "Tags": []
    }
}
```

Per ulteriori informazioni, consulta Riferimento alle autorizzazioni per l'API Amazon Redshift nella Guida alla gestione dei cluster di Amazon Redshift.

- Per i dettagli sull'API, consulta CreateHsmClientCertificate AWS CLICommand Reference.

`create-hsm-configuration`

Il seguente esempio di codice mostra come utilizzare`create-hsm-configuration`.

AWS CLI

Come creare una configurazione HSM

L'esempio `create-hsm-configuration` seguente crea la configurazione HSM specificata che contiene le informazioni richieste da un cluster per archiviare e utilizzare le chiavi di crittografia del database in un modulo di sicurezza hardware (HSM).

```
aws redshift create-hsm-configuration /
    --hsm-configuration-identifier myhsmconnection
    --description "My HSM connection"
    --hsm-ip-address 192.0.2.09
    --hsm-partition-name myhsmpartition /
    --hsm-partition-password A1b2c3d4 /
    --hsm-server-public-certificate myhsmclientcert
```

Output:

```
{
    "HsmConfiguration": {
        "HsmConfigurationIdentifier": "myhsmconnection",
        "Description": "My HSM connection",
        "HsmIpAddress": "192.0.2.09",
        "HsmPartitionName": "myhsmpartition",
        "Tags": []
    }
}
```

- Per i dettagli sull'API, consulta CreateHsmConfiguration AWS CLICommand Reference.

`create-snapshot-copy-grant`

Il seguente esempio di codice mostra come utilizzare`create-snapshot-copy-grant`.

AWS CLI

Come creare una concessione di copia degli snapshot

L'`create-snapshot-copy-grant`esempio seguente crea una concessione di copia delle istantanee e crittografa le istantanee copiate in una regione di destinazione. AWS

```
aws redshift create-snapshot-copy-grant \
    --snapshot-copy-grant-name mysnapshotcopygrantname
```

Output:

```
{
    "SnapshotCopyGrant": {
        "SnapshotCopyGrantName": "mysnapshotcopygrantname",
        "KmsKeyId": "arn:aws:kms:us-west-2:123456789012:key/
bPxRfih3yCo8nvbEXAMPLEKEY",
        "Tags": []
    }
}
```

Per ulteriori informazioni, consulta Crittografia dei database di Amazon Redshift nella Guida alla gestione dei cluster di Amazon Redshift.

- Per i dettagli sull'API, vedere CreateSnapshotCopyGrantin Command Reference.AWS CLI

`create-snapshot-schedule`

Il seguente esempio di codice mostra come utilizzare`create-snapshot-schedule`.

AWS CLI

Come creare una pianificazione di snapshot

L'esempio `create-snapshot-schedule` seguente crea una pianificazione di snapshot con la descrizione specificata e una frequenza di 12 ore.

```
aws redshift create-snapshot-schedule \
    --schedule-definitions "rate(12 hours)" \
    --schedule-identifier mysnapshotschedule \
    --schedule-description "My schedule description"
```

Output:

```
{
    "ScheduleDefinitions": [
        "rate(12 hours)"
    ],
    "ScheduleIdentifier": "mysnapshotschedule",
    "ScheduleDescription": "My schedule description",
    "Tags": []
}
```

Per ulteriori informazioni, consulta Pianificazioni di snapshot automatici nella Guida alla gestione dei cluster di Amazon Redshift.

- Per i dettagli sull'API, consulta CreateSnapshotSchedule AWS CLICommand Reference.

`create-tags`

Il seguente esempio di codice mostra come utilizzare`create-tags`.

AWS CLI

Come creare tag per un cluster

L'`create-tags`esempio seguente aggiunge la key/value coppia di tag specificata al cluster specificato.

```
aws redshift create-tags \
    --resource-name arn:aws:redshift:us-west-2:123456789012:cluster:mycluster \
    --tags "Key"="mytags","Value"="tag1"
```

Il comando non produce output.

Per ulteriori informazioni, consulta Tagging delle risorse in Amazon Redshift nella Guida alla gestione dei cluster di Amazon Redshift.

- Per i dettagli sull'API, vedere CreateTagsin AWS CLI Command Reference.

`delete-cluster-parameter-group`

Il seguente esempio di codice mostra come utilizzare`delete-cluster-parameter-group`.

AWS CLI

L' GroupThis esempio di eliminazione di un parametro del cluster elimina un parametro del cluster Group.Comando:

```
aws redshift delete-cluster-parameter-group --parameter-group-name
 myclusterparametergroup
```

- Per i dettagli sull'API, vedere DeleteClusterParameterGroupin AWS CLI Command Reference.

`delete-cluster-security-group`

Il seguente esempio di codice mostra come utilizzare`delete-cluster-security-group`.

AWS CLI

Eliminare un GroupThis esempio di Cluster Security elimina un gruppo di sicurezza del cluster.Comando:

```
aws redshift delete-cluster-security-group --cluster-security-group-name
 mysecuritygroup
```

- Per i dettagli sull'API, vedere DeleteClusterSecurityGroupin AWS CLI Command Reference.

`delete-cluster-snapshot`

Il seguente esempio di codice mostra come utilizzare`delete-cluster-snapshot`.

AWS CLI

L' SnapshotThis esempio di eliminazione di un cluster elimina uno snapshot.Comando:

```
aws redshift delete-cluster-snapshot --snapshot-identifier my-snapshot-id
```

- Per i dettagli sull'API, vedere in Command Reference. DeleteClusterSnapshotAWS CLI

`delete-cluster-subnet-group`

Il seguente esempio di codice mostra come utilizzare`delete-cluster-subnet-group`.

AWS CLI

Eliminare un GroupThis esempio di sottorete Cluster elimina un gruppo di sottoreti del cluster.Comando:

```
aws redshift delete-cluster-subnet-group --cluster-subnet-group-name mysubnetgroup
```

Risultato:

```
{
```

```
    "ResponseMetadata": {
        "RequestId": "253fbffd-6993-11e2-bc3a-47431073908a"
    }
}
```

- Per i dettagli sull'API, vedere in Command Reference. DeleteClusterSubnetGroupAWS CLI

`delete-cluster`

Il seguente esempio di codice mostra come utilizzare`delete-cluster`.

AWS CLI

L' SnapshotThis esempio Elimina un cluster senza un cluster finale elimina un cluster, forzando l'eliminazione dei dati in modo che non venga creata alcuna istantanea finale del cluster. Comando:

```
aws redshift delete-cluster --cluster-identifier mycluster --skip-final-cluster-
snapshot
```

L' SnapshotThis esempio Elimina un cluster, Allowing a Final Cluster elimina un cluster, ma specifica uno snapshot finale del cluster. Comando:

```
aws redshift delete-cluster --cluster-identifier mycluster --final-cluster-snapshot-
identifier myfinalsnapshot
```

- Per i dettagli sull'API, vedere in Command Reference. DeleteClusterAWS CLI

`delete-event-subscription`

Il seguente esempio di codice mostra come utilizzare`delete-event-subscription`.

AWS CLI

Come eliminare la sottoscrizione a eventi

L'esempio `delete-event-subscription` seguente elimina la sottoscrizione alle notifiche eventi specificata.

```
aws redshift delete-event-subscription \
```

```
    --subscription-name mysubscription
```

Il comando non produce output.

Per ulteriori informazioni, consulta Sottoscrizione alle notifiche eventi di Amazon Redshift nella Guida alla gestione dei cluster di Amazon Redshift.

- Per i dettagli sull'API, consulta DeleteEventSubscription AWS CLICommand Reference.

`delete-hsm-client-certificate`

Il seguente esempio di codice mostra come utilizzare`delete-hsm-client-certificate`.

AWS CLI

Come eliminare un certificato client HSM

L'esempio `delete-hsm-client-certificate` seguente elimina un certificato client HSM.

```
aws redshift delete-hsm-client-certificate \
    --hsm-client-certificate-identifier myhsmclientcert
```

Il comando non produce output.

Per ulteriori informazioni, consulta Riferimento alle autorizzazioni per l'API Amazon Redshift nella Guida alla gestione dei cluster di Amazon Redshift.

- Per i dettagli sull'API, consulta DeleteHsmClientCertificate AWS CLICommand Reference.

`delete-hsm-configuration`

Il seguente esempio di codice mostra come utilizzare`delete-hsm-configuration`.

AWS CLI

Come eliminare una configurazione HSM

L'`delete-hsm-configuration`esempio seguente elimina la configurazione HSM specificata dall'account corrente AWS .

```
aws redshift delete-hsm-configuration /
```

```
    --hsm-configuration-identifier myhsmconnection
```

Il comando non produce output.

- Per i dettagli sull'API, vedere DeleteHsmConfigurationin AWS CLI Command Reference.

`delete-scheduled-action`

Il seguente esempio di codice mostra come utilizzare`delete-scheduled-action`.

AWS CLI

Come eliminare un'azione pianificata

L'esempio `delete-scheduled-action` seguente elimina un'azione pianificata specificata.

```
aws redshift delete-scheduled-action \
    --scheduled-action-name myscheduledaction
```

Il comando non produce output.

- Per i dettagli sull'API, consulta DeleteScheduledAction AWS CLICommand Reference.

`delete-snapshot-copy-grant`

Il seguente esempio di codice mostra come utilizzare`delete-snapshot-copy-grant`.

AWS CLI

Come eliminare una concessione di copia degli snapshot

L'esempio `delete-snapshot-copy-grant` seguente elimina la concessione di copia degli snapshot specificata.

```
aws redshift delete-snapshot-copy-grant \
    --snapshot-copy-grant-name mysnapshotcopygrantname
```

Il comando non produce output.

Per ulteriori informazioni, consulta Crittografia dei database di Amazon Redshift nella Guida alla gestione dei cluster di Amazon Redshift.

- Per i dettagli sull'API, consulta DeleteSnapshotCopyGrant AWS CLICommand Reference.

`delete-snapshot-schedule`

Il seguente esempio di codice mostra come utilizzare`delete-snapshot-schedule`.

AWS CLI

Come eliminare una pianificazione di snapshot

L'esempio `delete-snapshot-schedule` seguente elimina la pianificazione di snapshot specificata. Prima di eliminare la pianificazione è necessario dissociare i cluster.

```
aws redshift delete-snapshot-schedule \
    --schedule-identifier mysnapshotschedule
```

Il comando non produce output.

Per ulteriori informazioni, consulta Pianificazioni di snapshot automatici nella Guida alla gestione dei cluster di Amazon Redshift.

- Per i dettagli sull'API, consulta DeleteSnapshotSchedule AWS CLICommand Reference.

`delete-tags`

Il seguente esempio di codice mostra come utilizzare`delete-tags`.

AWS CLI

Come eliminare tag da un cluster

L'esempio `delete-tags` seguente elimina i tag con i nomi di chiave specificati dal cluster specificato.

```
aws redshift delete-tags \
    --resource-name arn:aws:redshift:us-west-2:123456789012:cluster:mycluster \
    --tag-keys "clustertagkey" "clustertagvalue"
```

Il comando non produce output.

Per ulteriori informazioni, consulta Tagging delle risorse in Amazon Redshift nella Guida alla gestione dei cluster di Amazon Redshift.

- Per i dettagli sull'API, consulta DeleteTags AWS CLICommand Reference.

`describe-account-attributes`

Il seguente esempio di codice mostra come utilizzare`describe-account-attributes`.

AWS CLI

Per descrivere gli attributi di un AWS account

L'`describe-account-attributes`esempio seguente visualizza gli attributi associati all' AWS account chiamante.

```
aws redshift describe-account-attributes
```

Output:

```
{
    "AccountAttributes": [
        {
            "AttributeName": "max-defer-maintenance-duration",
            "AttributeValues": [
                {
                    "AttributeValue": "45"
                }
            ]
        }
    ]
}
```

- Per i dettagli sull'API, vedere DescribeAccountAttributesin AWS CLI Command Reference.

`describe-cluster-db-revisions`

Il seguente esempio di codice mostra come utilizzare`describe-cluster-db-revisions`.

AWS CLI

Come descrivere le revisioni del database per un cluster

L'esempio `describe-cluster-db-revisions` seguente mostra i dettagli relativi a un array di oggetti `ClusterDbRevision` per il cluster specificato.

```
aws redshift describe-cluster-db-revisions \
```

```
    --cluster-identifier mycluster
```

Output:

```
{
    "ClusterDbRevisions": [
        {
            "ClusterIdentifier": "mycluster",
            "CurrentDatabaseRevision": "11420",
            "DatabaseRevisionReleaseDate": "2019-11-22T16:43:49.597Z",
            "RevisionTargets": []
        }
    ]
}
```

- Per i dettagli sull'API, consulta DescribeClusterDbRevisions AWS CLICommand Reference.

`describe-cluster-parameter-groups`

Il seguente esempio di codice mostra come utilizzare`describe-cluster-parameter-groups`.

AWS CLI

L' GroupsThis esempio Get a Description of All Cluster Parameter restituisce una descrizione di tutti i gruppi di parametri del cluster per l'account, con le intestazioni delle colonne. Per impostazione predefinita, l'output è in formato JSON. Comando:

```
aws redshift describe-cluster-parameter-groups
```

Risultato:

```
{
   "ParameterGroups": [
      {
         "ParameterGroupFamily": "redshift-1.0",
         "Description": "My first cluster parameter group",
         "ParameterGroupName": "myclusterparametergroup"
      } ],
   "ResponseMetadata": {
      "RequestId": "8ceb8f6f-64cc-11e2-bea9-49e0ce183f07"
   }
```

```
}
```

È inoltre possibile ottenere le stesse informazioni in formato testo utilizzando l'opzione `--output text`. Comando:

Opzione `--output text`. Comando:

Opzione. Comando:

```
aws redshift describe-cluster-parameter-groups --output text
```

Risultato:

```
redshift-1.0        My first cluster parameter group        myclusterparametergroup
RESPONSEMETADATA    9e665a36-64cc-11e2-8f7d-3b939af52818
```

- Per i dettagli sull'API, consulta DescribeClusterParameterGroups AWS CLICommand Reference.

`describe-cluster-parameters`

Il seguente esempio di codice mostra come utilizzare`describe-cluster-parameters`.

AWS CLI

L' GroupThis esempio Recupera i parametri per un parametro di cluster specificato recupera i parametri per il gruppo di parametri denominato. Per impostazione predefinita, l'output è in formato JSON. Comando:

```
aws redshift describe-cluster-parameters --parameter-group-name
 myclusterparametergroup
```

Risultato:

```
{
   "Parameters": [
      {
         "Description": "Sets the display format for date and time values.",
         "DataType": "string",
```

```
         "IsModifiable": true,
         "Source": "engine-default",
         "ParameterValue": "ISO, MDY",
         "ParameterName": "datestyle"
      },
      {
         "Description": "Sets the number of digits displayed for floating-point
 values",
         "DataType": "integer",
         "IsModifiable": true,
         "AllowedValues": "-15-2",
         "Source": "engine-default",
         "ParameterValue": "0",
         "ParameterName": "extra_float_digits"
      },
      (...remaining output omitted...)
   ]
}
```

È inoltre possibile ottenere le stesse informazioni in formato testo utilizzando l'opzione `--output text`. Comando:

Opzione `--output text`. Comando:

Opzione. Comando:

```
aws redshift describe-cluster-parameters --parameter-group-name
 myclusterparametergroup --output text
```

Risultato:

```
RESPONSEMETADATA    cdac40aa-64cc-11e2-9e70-918437dd236d
Sets the display format for date and time values.   string  True    engine-default
 ISO, MDY        datestyle
Sets the number of digits displayed for floating-point values       integer True
 -15-2   engine-default  0       extra_float_digits
This parameter applies a user-defined label to a group of queries that are run
 during the same session..    string  True    engine-default  default query_group
require ssl for all databaseconnections     boolean True    true,false      engine-
default  false   require_ssl
Sets the schema search order for names that are not schema-qualified.       string
 True    engine-default  $user, public   search_path
```

```
Aborts any statement that takes over the specified number of milliseconds.  integer
 True    engine-default  0       statement_timeout
wlm json configuration      string  True    engine-default
 \[{"query_concurrency":5}]      wlm_json_configuration
```

- Per i dettagli sull'API, consultate AWS CLI Command DescribeClusterParametersReference.

`describe-cluster-security-groups`

Il seguente esempio di codice mostra come utilizzare`describe-cluster-security-groups`.

AWS CLI

L' GroupsThis esempio Get a Description of All Cluster Security restituisce una descrizione di tutti i gruppi di sicurezza del cluster per l'account. Per impostazione predefinita, l'output è in formato JSON. Comando:

```
aws redshift describe-cluster-security-groups
```

Risultato:

```
{
   "ClusterSecurityGroups": [
      {
         "OwnerId": "100447751468",
         "Description": "default",
         "ClusterSecurityGroupName": "default",
         "EC2SecurityGroups": \[],
         "IPRanges": [
            {
               "Status": "authorized",
               "CIDRIP": "0.0.0.0/0"
            }
         ]
      },
      {
         "OwnerId": "100447751468",
         "Description": "This is my cluster security group",
         "ClusterSecurityGroupName": "mysecuritygroup",
         "EC2SecurityGroups": \[],
         "IPRanges": \[]
      },
```

```
      (...remaining output omitted...)
   ]
}
```

- Per i dettagli sull'API, vedere DescribeClusterSecurityGroupsin AWS CLI Command Reference.

`describe-cluster-snapshots`

Il seguente esempio di codice mostra come utilizzare`describe-cluster-snapshots`.

AWS CLI

L' SnapshotsThis esempio Get a Description of All Cluster restituisce una descrizione di tutte le istantanee del cluster per l'account. Per impostazione predefinita, l'output è in formato JSON. Comando:

```
aws redshift describe-cluster-snapshots
```

Risultato:

```
{
   "Snapshots": [
      {
         "Status": "available",
         "SnapshotCreateTime": "2013-07-17T22:02:22.852Z",
         "EstimatedSecondsToCompletion": -1,
         "AvailabilityZone": "us-east-1a",
         "ClusterVersion": "1.0",
         "MasterUsername": "adminuser",
         "Encrypted": false,
         "OwnerAccount": "111122223333",
         "BackupProgressInMegabytes": 20.0,
         "ElapsedTimeInSeconds": 0,
         "DBName": "dev",
         "CurrentBackupRateInMegabytesPerSecond: 0.0,
         "ClusterCreateTime": "2013-01-22T21:59:29.559Z",
         "ActualIncrementalBackupSizeInMegabytes"; 20.0
         "SnapshotType": "automated",
         "NodeType": "dw.hs1.xlarge",
         "ClusterIdentifier": "mycluster",
         "Port": 5439,
         "TotalBackupSizeInMegabytes": 20.0,
```

```
            "NumberOfNodes": "2",
            "SnapshotIdentifier": "cm:mycluster-2013-01-22-22-04-18"
        },
        {
            "EstimatedSecondsToCompletion": 0,
            "OwnerAccount": "111122223333",
            "CurrentBackupRateInMegabytesPerSecond: 0.1534,
            "ActualIncrementalBackupSizeInMegabytes"; 11.0,
            "NumberOfNodes": "2",
            "Status": "available",
            "ClusterVersion": "1.0",
            "MasterUsername": "adminuser",
            "AccountsWithRestoreAccess": [
               {
                  "AccountID": "444455556666"
               } ],
            "TotalBackupSizeInMegabytes": 20.0,
            "DBName": "dev",
            "BackupProgressInMegabytes": 11.0,
            "ClusterCreateTime": "2013-01-22T21:59:29.559Z",
            "ElapsedTimeInSeconds": 0,
            "ClusterIdentifier": "mycluster",
            "SnapshotCreateTime": "2013-07-17T22:04:18.947Z",
            "AvailabilityZone": "us-east-1a",
            "NodeType": "dw.hs1.xlarge",
            "Encrypted": false,
            "SnapshotType": "manual",
            "Port": 5439,
            "SnapshotIdentifier": "my-snapshot-id"
        } ]
    }
    (...remaining output omitted...)
```

- Per i dettagli sull'API, consulta DescribeClusterSnapshots AWS CLICommand Reference.

`describe-cluster-subnet-groups`

Il seguente esempio di codice mostra come utilizzare`describe-cluster-subnet-groups`.

AWS CLI

L' GroupsThis esempio Get a Description of All Cluster Subnet restituisce una descrizione di tutti i gruppi di sottoreti del cluster. Per impostazione predefinita, l'output è in formato JSON. Comando:

```
aws redshift describe-cluster-subnet-groups
```

Risultato:

```
{
   "ClusterSubnetGroups": [
      {
         "Subnets": [
            {
               "SubnetStatus": "Active",
               "SubnetIdentifier": "subnet-763fdd1c",
               "SubnetAvailabilityZone": {
                  "Name": "us-east-1a"
               }
            }
         ],
         "VpcId": "vpc-7e3fdd14",
         "SubnetGroupStatus": "Complete",
         "Description": "My subnet group",
         "ClusterSubnetGroupName": "mysubnetgroup"
      }
   ],
   "ResponseMetadata": {
      "RequestId": "37fa8c89-6990-11e2-8f75-ab4018764c77"
   }
}
```

- Per i dettagli sull'API, vedere DescribeClusterSubnetGroupsin AWS CLI Command Reference.

`describe-cluster-tracks`

Il seguente esempio di codice mostra come utilizzare`describe-cluster-tracks`.

AWS CLI

Come descrivere le tracce del cluster

L'esempio `describe-cluster-tracks` seguente mostra i dettagli relativi alle tracce di manutenzione disponibili.

```
aws redshift describe-cluster-tracks \
```

```
    --maintenance-track-name current
```

Output:

```
{
    "MaintenanceTracks": [
        {
            "MaintenanceTrackName": "current",
            "DatabaseVersion": "1.0.11420",
            "UpdateTargets": [
                {
                    "MaintenanceTrackName": "preview_features",
                    "DatabaseVersion": "1.0.11746",
                    "SupportedOperations": [
                        {
                            "OperationName": "restore-from-cluster-snapshot"
                        }
                    ]
                },
                {
                    "MaintenanceTrackName": "trailing",
                    "DatabaseVersion": "1.0.11116",
                    "SupportedOperations": [
                        {
                            "OperationName": "restore-from-cluster-snapshot"
                        },
                        {
                            "OperationName": "modify-cluster"
                        }
                    ]
                }
            ]
        }
    ]
}
```

Per ulteriori informazioni, consulta Selezione delle tracce di manutenzione del cluster nella Guida alla gestione dei cluster di Amazon Redshift.

- Per i dettagli sull'API, consulta DescribeClusterTracks AWS CLICommand Reference.

`describe-cluster-versions`

Il seguente esempio di codice mostra come utilizzare`describe-cluster-versions`.

AWS CLI

L' VersionsThis esempio Get a Description of All Cluster restituisce una descrizione di tutte le versioni del cluster. Per impostazione predefinita, l'output è in formato JSON. Comando:

```
aws redshift describe-cluster-versions
```

Risultato:

```
{
   "ClusterVersions": [
      {
      "ClusterVersion": "1.0",
      "Description": "Initial release",
      "ClusterParameterGroupFamily": "redshift-1.0"
      } ],
   "ResponseMetadata": {
      "RequestId": "16a53de3-64cc-11e2-bec0-17624ad140dd"
   }
}
```

- Per i dettagli sull'API, consulta DescribeClusterVersions AWS CLICommand Reference.

`describe-clusters`

Il seguente esempio di codice mostra come utilizzare`describe-clusters`.

AWS CLI

L' ClustersThis esempio Get a Description of All restituisce una descrizione di tutti i cluster dell'account. Per impostazione predefinita, l'output è in formato JSON. Comando:

```
aws redshift describe-clusters
```

Risultato:

```
{
```

```
    "Clusters": [
    {
       "NodeType": "dw.hs1.xlarge",
       "Endpoint": {
          "Port": 5439,
          "Address": "mycluster.coqoarplqhsn.us-east-1.redshift.amazonaws.com"
       },
       "ClusterVersion": "1.0",
       "PubliclyAccessible": "true",
       "MasterUsername": "adminuser",
       "ClusterParameterGroups": [
          {
             "ParameterApplyStatus": "in-sync",
             "ParameterGroupName": "default.redshift-1.0"
          } ],
       "ClusterSecurityGroups": [
          {
             "Status": "active",
             "ClusterSecurityGroupName": "default"
          } ],
       "AllowVersionUpgrade": true,
       "VpcSecurityGroups": \[],
       "AvailabilityZone": "us-east-1a",
       "ClusterCreateTime": "2013-01-22T21:59:29.559Z",
       "PreferredMaintenanceWindow": "sat:03:30-sat:04:00",
       "AutomatedSnapshotRetentionPeriod": 1,
       "ClusterStatus": "available",
       "ClusterIdentifier": "mycluster",
       "DBName": "dev",
       "NumberOfNodes": 2,
       "PendingModifiedValues": {}
    } ],
    "ResponseMetadata": {
       "RequestId": "65b71cac-64df-11e2-8f5b-e90bd6c77476"
    }
}
```


È inoltre possibile ottenere le stesse informazioni in formato testo utilizzando l'opzione `--output text`. Comando:

Opzione `--output text`. Comando:

Opzione. Comando:

```
aws redshift describe-clusters --output text
```

Risultato:

```
dw.hs1.xlarge       1.0     true    adminuser       True    us-east-1a
 2013-01-22T21:59:29.559Z        sat:03:30-sat:04:00     1       available
 mycluster       dev     2
ENDPOINT    5439    mycluster.coqoarplqhsn.us-east-1.redshift.amazonaws.com
in-sync     default.redshift-1.0
active      default
PENDINGMODIFIEDVALUES
RESPONSEMETADATA    934281a8-64df-11e2-b07c-f7fbdd006c67
```

- Per i dettagli sull'API, consulta DescribeClusters AWS CLICommand Reference.

`describe-default-cluster-parameters`

Il seguente esempio di codice mostra come utilizzare`describe-default-cluster-parameters`.

AWS CLI

L' ParametersThis esempio Get a Description of Default Cluster restituisce una descrizione dei parametri di cluster predefiniti per la `redshift-1.0` famiglia. Per impostazione predefinita, l'output è in formato JSON. Comando:

```
aws redshift describe-default-cluster-parameters --parameter-group-family
 redshift-1.0
```

Risultato:

```
{
   "DefaultClusterParameters": {
   "ParameterGroupFamily": "redshift-1.0",
   "Parameters": [
      {
         "Description": "Sets the display format for date and time values.",
         "DataType": "string",
         "IsModifiable": true,
         "Source": "engine-default",
         "ParameterValue": "ISO, MDY",
         "ParameterName": "datestyle"
```

```
      },
      {
         "Description": "Sets the number of digits displayed for floating-point
 values",
         "DataType": "integer",
         "IsModifiable": true,
         "AllowedValues": "-15-2",
         "Source": "engine-default",
         "ParameterValue": "0",
         "ParameterName": "extra_float_digits"
      },
      (...remaining output omitted...)
      ]
   }
}
```

Per visualizzare un elenco di famiglie di gruppi di parametri valide, utilizza il comando `describe-cluster-parameter-groups`.

Comando `describe-cluster-parameter-groups`

comando.

- Per i dettagli sull'API, consulta DescribeDefaultClusterParameters AWS CLICommand Reference.

`describe-event-categories`

Il seguente esempio di codice mostra come utilizzare`describe-event-categories`.

AWS CLI

Come descrivere le categorie di eventi per un cluster

L'esempio `describe-event-categories` seguente mostra i dettagli relativi alle categorie di eventi per un cluster.

```
aws redshift describe-event-categories \
    --source-type cluster
```

Output:

```
{
```

```
    "EventCategoriesMapList": [
        {
            "SourceType": "cluster",
            "Events": [
                {
                    "EventId": "REDSHIFT-EVENT-2000",
                    "EventCategories": [
                        "management"
                    ],
                    "EventDescription": "Cluster <cluster name> created at <time in
 UTC>.",
                    "Severity": "INFO"
                },
                {
                    "EventId": "REDSHIFT-EVENT-2001",
                    "EventCategories": [
                        "management"
                    ],
                    "EventDescription": "Cluster <cluster name> deleted at <time in
 UTC>.",
                    "Severity": "INFO"
                },
                {
                    "EventId": "REDSHIFT-EVENT-3625",
                    "EventCategories": [
                        "monitoring"
                    ],
                    "EventDescription": "The cluster <cluster name> can't be resumed
 with its previous elastic network interface <ENI id>. We will allocate a new
 elastic network interface and associate it with the cluster node.",
                    "Severity": "INFO"
                }
            ]
        }
    ]
}
```

- Per i dettagli sull'API, consulta DescribeEventCategories AWS CLICommand Reference.

`describe-event-subscriptions`

Il seguente esempio di codice mostra come utilizzare`describe-event-subscriptions`.

AWS CLI

Come descrivere le sottoscrizioni a eventi

L'esempio `describe-event-subscriptions` seguente mostra le sottoscrizioni alle notifiche eventi per la sottoscrizione specificata.

```
aws redshift describe-event-subscriptions \
    --subscription-name mysubscription
```

Output:

```
{
    "EventSubscriptionsList": [
        {
            "CustomerAwsId": "123456789012",
            "CustSubscriptionId": "mysubscription",
            "SnsTopicArn": "arn:aws:sns:us-west-2:123456789012:MySNStopic",
            "Status": "active",
            "SubscriptionCreationTime": "2019-12-09T21:50:21.332Z",
            "SourceIdsList": [],
            "EventCategoriesList": [
                "management"
            ],
            "Severity": "ERROR",
            "Enabled": true,
            "Tags": []
        }
    ]
}
```

Per ulteriori informazioni, consulta Sottoscrizione alle notifiche eventi di Amazon Redshift nella Guida alla gestione dei cluster di Amazon Redshift.

- Per i dettagli sull'API, consulta DescribeEventSubscriptions AWS CLICommand Reference.

`describe-events`

Il seguente esempio di codice mostra come utilizzare`describe-events`.

AWS CLI

Come descrivere tutti gli eventi L'esempio seguente restituisce tutti gli eventi. Per impostazione predefinita, l'output è in formato JSON. Comando:

```
aws redshift describe-events
```

Risultato:

```
{
   "Events": [
      {
      "Date": "2013-01-22T19:17:03.640Z",
      "SourceIdentifier": "myclusterparametergroup",
      "Message": "Cluster parameter group myclusterparametergroup has been
 created.",
      "SourceType": "cluster-parameter-group"
      } ],
   "ResponseMetadata": {
      "RequestId": "9f056111-64c9-11e2-9390-ff04f2c1e638"
   }
}
```

È inoltre possibile ottenere le stesse informazioni in formato testo utilizzando l'opzione `--output text`. Comando:

Opzione `--output text`. Comando:

Opzione. Comando:

```
aws redshift describe-events --output text
```

Risultato:

```
2013-01-22T19:17:03.640Z    myclusterparametergroup Cluster parameter group
 myclusterparametergroup has been created.       cluster-parameter-group
RESPONSEMETADATA    8e5fe765-64c9-11e2-bce3-e56f52c50e17
```

- Per i dettagli sull'API, consulta DescribeEvents AWS CLICommand Reference.

`describe-hsm-client-certificates`

Il seguente esempio di codice mostra come utilizzare`describe-hsm-client-certificates`.

AWS CLI

Come descrivere i certificati client HSM

L'esempio `describe-hsm-client-certificates` seguente mostra i dettagli relativi al certificato client HSM specificato.

```
aws redshift describe-hsm-client-certificates \
    --hsm-client-certificate-identifier myhsmclientcert
```

Output:

```
{
    "HsmClientCertificates": [
        {
        "HsmClientCertificateIdentifier": "myhsmclientcert",
        "HsmClientCertificatePublicKey": "-----BEGIN CERTIFICATE-----\
        EXAMPLECAfICCQD6m7oRw0uXOjANBgkqhkiG9w0BAQUFADCBiDELMAkGA1UEBhMC
        VVMxCzAJBgNVBAEXAMPLERAwDgYDVQQHEwdTZWF0dGxlMQ8wDQYDVQQKEwZBbWF6
        b24xFDASBgNVBAsTC0lBTSBDb25zEXAMPLEwEAYDVQQDEwlUZXN0Q2lsYWMxHzAd
        BgkqhkiG9w0BCQEWEG5vb25lQGFtYXpvbi5jb20wHhEXAMPLEDI1MjA0EXAMPLEN
        EXAMPLE0MjA0NTIxWjCBiDELMAkGA1UEBhMCVVMxCzAJBgNVBAgTAldBMRAwDgYD
        VQQHEwdTZWF0dGEXAMPLEQYDVQQKEwZBbWF6b24xFDASBgNVBAsTC0lBTSBDb25z
        b2xlMRIwEAYDVQQDEwlUZXN0Q2lsEXAMPLEdBgkqhkiG9w0BCQEWEG5vb25lQGFt
        YXpvbi5jb20wgZ8wDQYJKoZIhvcNAQEBBQADgY0AMIEXAMPLEMaK0dn+a4GmWIWJ
        21uUSfwfEvySWtC2XADZ4nB+BLYgVIk60CpiwsZ3G93vUEIO3IyNoH/f0wYK8m9T
        rDHudUZg3qX4waLG5M43q7Wgc/MbQITxOUSQv7c7ugFFDzQGBzZswY67EXAMPLEE
        EXAMPLEZnzcvQAaRHhdlQWIMm2nrAgMBAAEwDQYJKoZIhvcNAQEFBQADgYEAtCu4
        nUhVVxYUntneD9EXAMPLE6q+auNKyExzyLwaxlAoo7TJHidbtS4J5iNmZgXL0Fkb
        FFBjvSfpJIlJ00zbhNYS5f6GuoEDEXAMPLEBHjJnyp378OD8uTs7fLvjx79LjSTb
        NYiytVbZPQUQ5Yaxu2jXnimvw3rEXAMPLE=-----END CERTIFICATE-----\n",
        "Tags": []
        }
    ]
}
```

Per ulteriori informazioni, consulta Riferimento alle autorizzazioni per l'API Amazon Redshift nella Guida alla gestione dei cluster di Amazon Redshift.

- Per i dettagli sull'API, consulta DescribeHsmClientCertificates AWS CLICommand Reference.

`describe-hsm-configurations`

Il seguente esempio di codice mostra come utilizzare`describe-hsm-configurations`.

AWS CLI

Come descrivere le configurazioni HSM

L'`describe-hsm-configurations`esempio seguente visualizza i dettagli delle configurazioni HSM disponibili per l'account chiamante AWS .

```
aws redshift describe-hsm-configurations /
    --hsm-configuration-identifier myhsmconnection
```

Output:

```
{
    "HsmConfigurations": [
        {
            "HsmConfigurationIdentifier": "myhsmconnection",
            "Description": "My HSM connection",
            "HsmIpAddress": "192.0.2.09",
            "HsmPartitionName": "myhsmpartition",
            "Tags": []
        }
    ]
}
```

- Per i dettagli sull'API, vedere DescribeHsmConfigurationsin AWS CLI Command Reference.

`describe-logging-status`

Il seguente esempio di codice mostra come utilizzare`describe-logging-status`.

AWS CLI

Come descrivere lo stato della registrazione di log per un cluster

L'esempio `describe-logging-status` seguente mostra se le informazioni, ad esempio query e tentativi di connessione, vengono registrate nei log per un cluster.

```
aws redshift describe-logging-status \
    --cluster-identifier mycluster
```

Output:

```
{
    "LoggingEnabled": false
}
```

Per ulteriori informazioni, consulta Logging di controllo dei database nella Guida alla gestione dei cluster di Amazon Redshift.

- Per i dettagli sull'API, consulta DescribeLoggingStatus AWS CLICommand Reference.

`describe-node-configuration-options`

Il seguente esempio di codice mostra come utilizzare`describe-node-configuration-options`.

AWS CLI

Come descrivere le opzioni di configurazione dei nodi

L'esempio `describe-node-configuration-options` seguente mostra le proprietà delle possibili configurazioni dei nodi, ad esempio il tipo di nodo, il numero di nodi e l'utilizzo del disco per lo snapshot del cluster specificato.

```
aws redshift describe-node-configuration-options \
    --action-type restore-cluster \
    --snapshot-identifier rs:mycluster-2019-12-09-16-42-43
```

Output:

```
{
    "NodeConfigurationOptionList": [
        {
            "NodeType": "dc2.large",
            "NumberOfNodes": 2,
            "EstimatedDiskUtilizationPercent": 19.61
        },
        {
            "NodeType": "dc2.large",
            "NumberOfNodes": 4,
```

```
            "EstimatedDiskUtilizationPercent": 9.96
        },
        {
            "NodeType": "ds2.xlarge",
            "NumberOfNodes": 2,
            "EstimatedDiskUtilizationPercent": 1.53
        },
        {
            "NodeType": "ds2.xlarge",
            "NumberOfNodes": 4,
            "EstimatedDiskUtilizationPercent": 0.78
        }
    ]
}
```

Per ulteriori informazioni, consulta Acquisto di nodi riservati di Amazon Redshift nella Guida alla gestione dei cluster di Amazon Redshift.

- Per i dettagli sull'API, consulta DescribeNodeConfigurationOptions AWS CLICommand Reference.

`describe-orderable-cluster-options`

Il seguente esempio di codice mostra come utilizzare`describe-orderable-cluster-options`.

AWS CLI

L' OptionsThis esempio Describing All Orderable Cluster restituisce le descrizioni di tutte le opzioni di cluster ordinabili. Per impostazione predefinita, l'output è in formato JSON. Comando:

```
aws redshift describe-orderable-cluster-options
```

Risultato:

```
{
   "OrderableClusterOptions": [
      {
         "NodeType": "dw.hs1.8xlarge",
         "AvailabilityZones": [
            { "Name": "us-east-1a" },
            { "Name": "us-east-1b" },
            { "Name": "us-east-1c" } ],
```

```
         "ClusterVersion": "1.0",
         "ClusterType": "multi-node"
      },
      {
         "NodeType": "dw.hs1.xlarge",
         "AvailabilityZones": [
            { "Name": "us-east-1a" },
            { "Name": "us-east-1b" },
            { "Name": "us-east-1c" } ],
         "ClusterVersion": "1.0",
         "ClusterType": "multi-node"
      },
      {
      "NodeType": "dw.hs1.xlarge",
      "AvailabilityZones": [
         { "Name": "us-east-1a" },
         { "Name": "us-east-1b" },
         { "Name": "us-east-1c" } ],
      "ClusterVersion": "1.0",
      "ClusterType": "single-node"
      } ],
   "ResponseMetadata": {
      "RequestId": "f6000035-64cb-11e2-9135-ff82df53a51a"
   }
}
```

È inoltre possibile ottenere le stesse informazioni in formato testo utilizzando l'opzione `--output text`. Comando:

Opzione `--output text`. Comando:

Opzione. Comando:

```
aws redshift describe-orderable-cluster-options --output text
```

Risultato:

```
dw.hs1.8xlarge      1.0     multi-node
us-east-1a
us-east-1b
us-east-1c
dw.hs1.xlarge       1.0     multi-node
us-east-1a
```

```
us-east-1b
us-east-1c
dw.hs1.xlarge       1.0     single-node
us-east-1a
us-east-1b
us-east-1c
RESPONSEMETADATA    e648696b-64cb-11e2-bec0-17624ad140dd
```

- Per i dettagli sull'API, vedere DescribeOrderableClusterOptionsin AWS CLI Command Reference.

`describe-reserved-node-offerings`

Il seguente esempio di codice mostra come utilizzare`describe-reserved-node-offerings`.

AWS CLI

L' OfferingsThis esempio Descrivi il nodo riservato mostra tutte le offerte di nodi riservati disponibili per Purchase.Command:

```
aws redshift describe-reserved-node-offerings
```

Risultato:

```
{
   "ReservedNodeOfferings": [
      {
         "OfferingType": "Heavy Utilization",
         "FixedPrice": "",
         "NodeType": "dw.hs1.xlarge",
         "UsagePrice": "",
         "RecurringCharges": [
            {
               "RecurringChargeAmount": "",
               "RecurringChargeFrequency": "Hourly"
            } ],
         "Duration": 31536000,
         "ReservedNodeOfferingId": "ceb6a579-cf4c-4343-be8b-d832c45ab51c"
      },
      {
         "OfferingType": "Heavy Utilization",
         "FixedPrice": "",
```

```
            "NodeType": "dw.hs1.8xlarge",
            "UsagePrice": "",
            "RecurringCharges": [
               {
               "RecurringChargeAmount": "",
               "RecurringChargeFrequency": "Hourly"
               } ],
            "Duration": 31536000,
            "ReservedNodeOfferingId": "e5a2ff3b-352d-4a9c-ad7d-373c4cab5dd2"
         },
         ...remaining output omitted...
      ],
      "ResponseMetadata": {
         "RequestId": "8b1a1a43-75ff-11e2-9666-e142fe91ddd1"
      }
}
```

Se desideri acquistare un'offerta di nodi riservati, puoi chiamare `purchase-reserved-node-offering` utilizzando un nodo valido. ReservedNodeOfferingId

`purchase-reserved-node-offering`utilizzando un valido ReservedNodeOfferingId.

utilizzando un valido ReservedNodeOfferingId.

ReservedNodeOfferingId.

.

- Per i dettagli sull'API, consulta DescribeReservedNodeOfferings AWS CLICommand Reference.

`describe-reserved-nodes`

Il seguente esempio di codice mostra come utilizzare`describe-reserved-nodes`.

AWS CLI

Descrivi l' NodesThis esempio riservato mostra un'offerta di nodi riservati che è stata acquistata.Comando:

```
aws redshift describe-reserved-nodes
```

Risultato:

```
{
   "ResponseMetadata": {
      "RequestId": "bc29ce2e-7600-11e2-9949-4b361e7420b7"
   },
   "ReservedNodes": [
      {
         "OfferingType": "Heavy Utilization",
         "FixedPrice": "",
         "NodeType": "dw.hs1.xlarge",
         "ReservedNodeId": "1ba8e2e3-bc01-4d65-b35d-a4a3e931547e",
         "UsagePrice": "",
         "RecurringCharges": [
            {
               "RecurringChargeAmount": "",
               "RecurringChargeFrequency": "Hourly"
            } ],
         "NodeCount": 1,
         "State": "payment-pending",
         "StartTime": "2013-02-13T17:08:39.051Z",
         "Duration": 31536000,
         "ReservedNodeOfferingId": "ceb6a579-cf4c-4343-be8b-d832c45ab51c"
      }
   ]
}
```

- Per i dettagli sull'API, consulta DescribeReservedNodesCommand Reference.AWS CLI

`describe-resize`

Il seguente esempio di codice mostra come utilizzare`describe-resize`.

AWS CLI

Descrivi l' ResizeThis esempio che descrive l'ultimo ridimensionamento di un cluster. La richiesta riguarda tre nodi di tipo `dw.hs1.8xlarge`. Comando:

```
aws redshift describe-resize --cluster-identifier mycluster
```

Risultato:

```
{
   "Status": "NONE",
```

```
   "TargetClusterType": "multi-node",
   "TargetNodeType": "dw.hs1.8xlarge",
   "ResponseMetadata": {
      "RequestId": "9f52b0b4-7733-11e2-aa9b-318b2909bd27"
   },
   "TargetNumberOfNodes": "3"
}
```

- Per i dettagli sull'API, consulta DescribeResize AWS CLICommand Reference.

`describe-scheduled-actions`

Il seguente esempio di codice mostra come utilizzare`describe-scheduled-actions`.

AWS CLI

Come descrivere le azioni pianificate

L'esempio `describe-scheduled-actions` seguente mostra i dettagli relativi a tutte le azioni attualmente pianificate.

```
aws redshift describe-scheduled-actions
```

Output:

```
{
    "ScheduledActions": [
        {
            "ScheduledActionName": "resizecluster",
            "TargetAction": {
                "ResizeCluster": {
                    "ClusterIdentifier": "mycluster",
                    "NumberOfNodes": 4,
                    "Classic": false
                }
            },
            "Schedule": "at(2019-12-10T00:07:00)",
            "IamRole": "arn:aws:iam::123456789012:role/myRedshiftRole",
            "State": "ACTIVE",
            "NextInvocations": [
                "2019-12-10T00:07:00Z"
            ]
```

```
        }
    ]
}
```

- Per i dettagli sull'API, consulta DescribeScheduledActions AWS CLICommand Reference.

`describe-snapshot-copy-grants`

Il seguente esempio di codice mostra come utilizzare`describe-snapshot-copy-grants`.

AWS CLI

Come descrivere le concessioni di copia degli snapshot

L'esempio `describe-snapshot-copy-grants` seguente mostra i dettagli relativi alla concessione di copia degli snapshot specificata.

```
aws redshift describe-snapshot-copy-grants \
    --snapshot-copy-grant-name mysnapshotcopygrantname
```

Output:

```
{
    "SnapshotCopyGrants": [
        {
            "SnapshotCopyGrantName": "mysnapshotcopygrantname",
            "KmsKeyId": "arn:aws:kms:us-west-2:123456789012:key/
bPxRfih3yCo8nvbEXAMPLEKEY",
            "Tags": []
        }
    ]
}
```

Per ulteriori informazioni, consulta Crittografia dei database di Amazon Redshift nella Guida alla gestione dei cluster di Amazon Redshift.

- Per i dettagli sull'API, consulta DescribeSnapshotCopyGrants AWS CLICommand Reference.

`describe-snapshot-schedules`

Il seguente esempio di codice mostra come utilizzare`describe-snapshot-schedules`.

AWS CLI

Come descrivere le pianificazioni di snapshot

L'esempio `describe-snapshot-schedules` seguente mostra i dettagli relativi alla pianificazione di snapshot del cluster specificata.

```
aws redshift describe-snapshot-schedules \
    --cluster-identifier mycluster \
    --schedule-identifier mysnapshotschedule
```

Output:

```
{
    "SnapshotSchedules": [
        {
            "ScheduleDefinitions": [
                "rate(12 hours)"
            ],
            "ScheduleIdentifier": "mysnapshotschedule",
            "ScheduleDescription": "My schedule description",
            "Tags": [],
            "AssociatedClusterCount": 1,
            "AssociatedClusters": [
                {
                    "ClusterIdentifier": "mycluster",
                    "ScheduleAssociationState": "ACTIVE"
                }
            ]
        }
    ]
}
```

Per ulteriori informazioni, consulta Pianificazioni di snapshot automatici nella Guida alla gestione dei cluster di Amazon Redshift.

- Per i dettagli sull'API, consulta DescribeSnapshotSchedules AWS CLICommand Reference.

`describe-storage`

Il seguente esempio di codice mostra come utilizzare`describe-storage`.

AWS CLI

Come descrivere l'archiviazione

L'esempio `describe-storage` seguente mostra i dettagli relativi all'archiviazione di backup e alle dimensioni di archiviazione provvisorie per l'account.

```
aws redshift describe-storage
```

Output:

```
{
    "TotalBackupSizeInMegaBytes": 193149.0,
    "TotalProvisionedStorageInMegaBytes": 655360.0
}
```

Per ulteriori informazioni, consulta Gestione dell'archiviazione di snapshot nella Guida alla gestione dei cluster di Amazon Redshift.

- Per i dettagli sull'API, consulta DescribeStorage AWS CLICommand Reference.

`describe-table-restore-status`

Il seguente esempio di codice mostra come utilizzare`describe-table-restore-status`.

AWS CLI

Come descrivere lo stato delle richieste di ripristino delle tabelle da uno snapshot del cluster

L'esempio `describe-table-restore-status` seguente mostra i dettagli relativi alle richieste di ripristino delle tabelle effettuate per il cluster specificato.

```
aws redshift describe-table-restore-status /
    --cluster-identifier mycluster
```

Output:

```
{
    "TableRestoreStatusDetails": [
        {
            "TableRestoreRequestId": "z1116630-0e80-46f4-ba86-bd9670411ebd",
            "Status": "IN_PROGRESS",
```

```
            "RequestTime": "2019-12-27T18:22:12.257Z",
            "ClusterIdentifier": "mycluster",
            "SnapshotIdentifier": "mysnapshotid",
            "SourceDatabaseName": "dev",
            "SourceSchemaName": "public",
            "SourceTableName": "mytable",
            "TargetDatabaseName": "dev",
            "TargetSchemaName": "public",
            "NewTableName": "mytable-clone"
        }
    ]
}
```

Per ulteriori informazioni, consulta Ripristino di una tabella da uno snapshot nella Guida alla gestione dei cluster di Amazon Redshift.

- Per i dettagli sull'API, consulta DescribeTableRestoreStatus AWS CLICommand Reference.

`describe-tags`

Il seguente esempio di codice mostra come utilizzare`describe-tags`.

AWS CLI

Come descrivere i tag

L'esempio `describe-tags` seguente mostra le risorse del cluster specificato che sono associate ai nomi e ai valori dei tag specificati.

```
aws redshift describe-tags \
    --resource-name arn:aws:redshift:us-west-2:123456789012:cluster:mycluster \
    --tag-keys clustertagkey \
    --tag-values clustertagvalue
```

Output:

```
{
    "TaggedResources": [
                    {
            "Tag": {
                "Key": "clustertagkey",
                "Value": "clustertagvalue"
            },
```

```
            "ResourceName": "arn:aws:redshift:us-
west-2:123456789012:cluster:mycluster",
            "ResourceType": "cluster"
        }
    ]
}
```

Per ulteriori informazioni, consulta Tagging delle risorse in Amazon Redshift nella Guida alla gestione dei cluster di Amazon Redshift.

- Per i dettagli sull'API, consulta DescribeTags AWS CLICommand Reference.

`disable-snapshot-copy`

Il seguente esempio di codice mostra come utilizzare`disable-snapshot-copy`.

AWS CLI

Come disabilitare la copia di snapshot per un cluster

L'esempio `disable-snapshot-copy` seguente disabilita la copia automatica di uno snapshot per il cluster specificato.

```
aws redshift disable-snapshot-copy \
    --cluster-identifier mycluster
```

Output:

```
{
    "Cluster": {
        "ClusterIdentifier": "mycluster",
        "NodeType": "dc2.large",
        "ClusterStatus": "available",
        "ClusterAvailabilityStatus": "Available",
        "MasterUsername": "adminuser",
        "DBName": "dev",
        "Endpoint": {
            "Address": "mycluster.cmeaswqeuae.us-west-2.redshift.amazonaws.com",
            "Port": 5439
        },
        "ClusterCreateTime": "2019-12-05T18:44:36.991Z",
        "AutomatedSnapshotRetentionPeriod": 3,
        "ManualSnapshotRetentionPeriod": -1,
```

```
        "ClusterSecurityGroups": [],
        "VpcSecurityGroups": [
            {
                "VpcSecurityGroupId": "sh-i9b431cd",
                "Status": "active"
            }
        ],
        "ClusterParameterGroups": [
            {
                "ParameterGroupName": "default.redshift-1.0",
                "ParameterApplyStatus": "in-sync"
            }
        ],
        "ClusterSubnetGroupName": "default",
        "VpcId": "vpc-b1fel7t9",
        "AvailabilityZone": "us-west-2f",
        "PreferredMaintenanceWindow": "sat:16:00-sat:16:30",
        "PendingModifiedValues": {
            "NodeType": "dc2.large",
            "NumberOfNodes": 2,
            "ClusterType": "multi-node"
        },
        "ClusterVersion": "1.0",
        "AllowVersionUpgrade": true,
        "NumberOfNodes": 4,
        "PubliclyAccessible": false,
        "Encrypted": false,
        "Tags": [
            {
                "Key": "mytags",
                "Value": "tag1"
            }
        ],
        "EnhancedVpcRouting": false,
        "IamRoles": [
            {
                "IamRoleArn": "arn:aws:iam::123456789012:role/myRedshiftRole",
                "ApplyStatus": "in-sync"
            }
        ],
        "MaintenanceTrackName": "current",
        "DeferredMaintenanceWindows": [],
        "ExpectedNextSnapshotScheduleTime": "2019-12-10T04:42:43.390Z",
        "ExpectedNextSnapshotScheduleTimeStatus": "OnTrack",
```

```
        "NextMaintenanceWindowStartTime": "2019-12-14T16:00:00Z"
    }
}
```

Per ulteriori informazioni, consulta Copiare gli snapshot in un'altra AWS regione nella Amazon Redshift Cluster Management Guide.

- Per i dettagli sull'API, consulta Command DisableSnapshotCopyReference AWS CLI .

`enable-snapshot-copy`

Il seguente esempio di codice mostra come utilizzare`enable-snapshot-copy`.

AWS CLI

Come abilitare la copia di snapshot per un cluster

L'esempio `enable-snapshot-copy` seguente abilita la copia automatica di uno snapshot per il cluster specificato.

```
aws redshift enable-snapshot-copy \
    --cluster-identifier mycluster \
    --destination-region us-west-1
```

Output:

```
{
    "Cluster": {
        "ClusterIdentifier": "mycluster",
        "NodeType": "dc2.large",
        "ClusterStatus": "available",
        "ClusterAvailabilityStatus": "Available",
        "MasterUsername": "adminuser",
        "DBName": "dev",
        "Endpoint": {
            "Address": "mycluster.cmeaswqeuae.us-west-2.redshift.amazonaws.com",
            "Port": 5439
        },
        "ClusterCreateTime": "2019-12-05T18:44:36.991Z",
        "AutomatedSnapshotRetentionPeriod": 3,
        "ManualSnapshotRetentionPeriod": -1,
        "ClusterSecurityGroups": [],
        "VpcSecurityGroups": [
```

```
            {
                "VpcSecurityGroupId": "sh-f4c731cd",
                "Status": "active"
            }
        ],
        "ClusterParameterGroups": [
            {
                "ParameterGroupName": "default.redshift-1.0",
                "ParameterApplyStatus": "in-sync"
            }
        ],
        "ClusterSubnetGroupName": "default",
        "VpcId": "vpc-b1ael7t9",
        "AvailabilityZone": "us-west-2f",
        "PreferredMaintenanceWindow": "sat:16:00-sat:16:30",
        "PendingModifiedValues": {
            "NodeType": "dc2.large",
            "NumberOfNodes": 2,
            "ClusterType": "multi-node"
        },
        "ClusterVersion": "1.0",
        "AllowVersionUpgrade": true,
        "NumberOfNodes": 4,
        "PubliclyAccessible": false,
        "Encrypted": false,
        "ClusterSnapshotCopyStatus": {
            "DestinationRegion": "us-west-1",
            "RetentionPeriod": 7,
            "ManualSnapshotRetentionPeriod": -1
        },
        "Tags": [
            {
                "Key": "mytags",
                "Value": "tag1"
            }
        ],
        "EnhancedVpcRouting": false,
        "IamRoles": [
            {
                "IamRoleArn": "arn:aws:iam::123456789012:role/myRedshiftRole",
                "ApplyStatus": "in-sync"
            }
        ],
        "MaintenanceTrackName": "current",
```

```
        "DeferredMaintenanceWindows": [],
        "ExpectedNextSnapshotScheduleTime": "2019-12-10T04:42:43.390Z",
        "ExpectedNextSnapshotScheduleTimeStatus": "OnTrack",
        "NextMaintenanceWindowStartTime": "2019-12-14T16:00:00Z"
    }
}
```

Per ulteriori informazioni, consulta Copiare gli snapshot in un'altra AWS regione nella Amazon Redshift Cluster Management Guide.

- Per i dettagli sull'API, consulta Command EnableSnapshotCopyReference AWS CLI .

`get-cluster-credentials`

Il seguente esempio di codice mostra come utilizzare`get-cluster-credentials`.

AWS CLI

Per ottenere le credenziali del cluster per un account AWS

L'esempio `get-cluster-credentials` seguente recupera le credenziali temporanee che consentono l'accesso a un database Amazon Redshift.

```
aws redshift get-cluster-credentials \
    --db-user adminuser --db-name dev \
    --cluster-identifier mycluster
```

Output:

```
{
    "DbUser": "IAM:adminuser",
    "DbPassword": "AMAFUyyuros/QjxPTtgzcsuQsqzIasdzJEN04aCtWDzXx1O9d6UmpkBtvEeqFly/
EXAMPLE==",
    "Expiration": "2019-12-10T17:25:05.770Z"
}
```

Per ulteriori informazioni, consulta Generazione di credenziali di database IAM mediante la CLI o l'API di Amazon Redshift nella Guida alla gestione dei cluster di Amazon Redshift.

- Per i dettagli sull'API, consulta GetClusterCredentials AWS CLICommand Reference.

`get-reserved-node-exchange-offerings`

Il seguente esempio di codice mostra come utilizzare`get-reserved-node-exchange-offerings`.

AWS CLI

Come ottenere offerte di scambio di nodi riservati

L'esempio `get-reserved-node-exchange-offerings` seguente recupera un array di `ReservedNodeOfferings` DC2 che corrispondono al nodo riservato `DC1`.

```
aws redshift get-reserved-node-exchange-offerings \
    --reserved-node-id 12345678-12ab-12a1-1a2a-12ab-12a12EXAMPLE
```

Output:

```
{
    "ReservedNodeOfferings": [
        {
            "ReservedNodeOfferingId": "12345678-12ab-12a1-1a2a-12ab-12a12EXAMPLE",
            "NodeType": "dc2.large",
            "Duration": 31536000,
            "FixedPrice": 0.0,
            "UsagePrice": 0.0,
            "CurrencyCode": "USD",
            "OfferingType": "All Upfront",
            "RecurringCharges": [
                {
                    "RecurringChargeAmount": 0.0,
                    "RecurringChargeFrequency": "Hourly"
                }
            ],
            "ReservedNodeOfferingType": "Regular"
        }
    ]
}
```

Per ulteriori informazioni, consulta Upgrade Reserved Nodes With the AWS CLI nella Amazon Redshift Cluster Management Guide.

- Per i dettagli sull'API, consulta GetReservedNodeExchangeOfferingsCommand Reference.AWS CLI

`modify-cluster-iam-roles`

Il seguente esempio di codice mostra come utilizzare`modify-cluster-iam-roles`.

AWS CLI

Come modificare il ruolo IAM per un cluster

L'`modify-cluster-iam-roles`esempio seguente rimuove il ruolo AWS IAM specificato dal cluster specificato.

```
aws redshift modify-cluster-iam-roles \
    --cluster-identifier mycluster \
    --remove-iam-roles arn:aws:iam::123456789012:role/myRedshiftRole
```

Output:

```
{
    "Cluster": {
        "ClusterIdentifier": "mycluster",
        "NodeType": "dc2.large",
        "ClusterStatus": "available",
        "ClusterAvailabilityStatus": "Available",
        "MasterUsername": "adminuser",
        "DBName": "dev",
        "Endpoint": {
            "Address": "mycluster.cmeaswqeuae.us-west-2.redshift.amazonaws.com",
            "Port": 5439
        },
        "ClusterCreateTime": "2019-12-05T18:44:36.991Z",
        "AutomatedSnapshotRetentionPeriod": 3,
        "ManualSnapshotRetentionPeriod": -1,
        "ClusterSecurityGroups": [],
        "VpcSecurityGroups": [
            {
                "VpcSecurityGroupId": "sh-f9b731sd",
                "Status": "active"
            }
        ],
        "ClusterParameterGroups": [
            {
                "ParameterGroupName": "default.redshift-1.0",
                "ParameterApplyStatus": "in-sync"
```

```
            }
        ],
        "ClusterSubnetGroupName": "default",
        "VpcId": "vpc-b2fal7t9",
        "AvailabilityZone": "us-west-2f",
        "PreferredMaintenanceWindow": "sat:16:00-sat:16:30",
        "PendingModifiedValues": {
            "NodeType": "dc2.large",
            "NumberOfNodes": 2,
            "ClusterType": "multi-node"
        },
        "ClusterVersion": "1.0",
        "AllowVersionUpgrade": true,
        "NumberOfNodes": 4,
        "PubliclyAccessible": false,
        "Encrypted": false,
        "ClusterSnapshotCopyStatus": {
            "DestinationRegion": "us-west-1",
            "RetentionPeriod": 7,
            "ManualSnapshotRetentionPeriod": -1
        },
        "Tags": [
            {
                "Key": "mytags",
                "Value": "tag1"
            }
        ],
        "EnhancedVpcRouting": false,
        "IamRoles": [],
        "MaintenanceTrackName": "current",
        "DeferredMaintenanceWindows": [],
        "ExpectedNextSnapshotScheduleTime": "2019-12-11T04:42:55.631Z",
        "ExpectedNextSnapshotScheduleTimeStatus": "OnTrack",
        "NextMaintenanceWindowStartTime": "2019-12-14T16:00:00Z"
    }
}
```

Per ulteriori informazioni, consulta Utilizzo di policy basate su identità (policy IAM) per Amazon Redshift nella Guida alla gestione dei cluster di Amazon Redshift.

- Per i dettagli sull'API, vedere ModifyClusterIamRolesin AWS CLI Command Reference.

`modify-cluster-maintenance`

Il seguente esempio di codice mostra come utilizzare`modify-cluster-maintenance`.

AWS CLI

Come modificare la manutenzione del cluster

L'esempio `modify-cluster-maintenance` seguente posticipa di 30 giorni la manutenzione del cluster specificato.

```
aws redshift modify-cluster-maintenance \
    --cluster-identifier mycluster \
    --defer-maintenance \
    --defer-maintenance-duration 30
```

Output:

```
{
    "Cluster": {
        "ClusterIdentifier": "mycluster",
        "NodeType": "dc2.large",
        "ClusterStatus": "available",
        "ClusterAvailabilityStatus": "Available",
        "MasterUsername": "adminuser",
        "DBName": "dev",
        "Endpoint": {
            "Address": "mycluster.cmeaswqeuae.us-west-2.redshift.amazonaws.com",
            "Port": 5439
        },
        "ClusterCreateTime": "2019-12-05T18:44:36.991Z",
        "AutomatedSnapshotRetentionPeriod": 3,
        "ManualSnapshotRetentionPeriod": -1,
        "ClusterSecurityGroups": [],
        "VpcSecurityGroups": [
            {
                "VpcSecurityGroupId": "sh-a1a123ab",
                "Status": "active"
            }
        ],
        "ClusterParameterGroups": [
            {
                "ParameterGroupName": "default.redshift-1.0",
```

```
                "ParameterApplyStatus": "in-sync"
            }
        ],
        "ClusterSubnetGroupName": "default",
        "VpcId": "vpc-b1ael7t9",
        "AvailabilityZone": "us-west-2f",
        "PreferredMaintenanceWindow": "sat:16:00-sat:16:30",
        "PendingModifiedValues": {
            "NodeType": "dc2.large",
            "NumberOfNodes": 2,
            "ClusterType": "multi-node"
        },
        "ClusterVersion": "1.0",
        "AllowVersionUpgrade": true,
        "NumberOfNodes": 4,
        "PubliclyAccessible": false,
        "Encrypted": false,
        "ClusterSnapshotCopyStatus": {
            "DestinationRegion": "us-west-1",
            "RetentionPeriod": 7,
            "ManualSnapshotRetentionPeriod": -1
        },
        "Tags": [
            {
                "Key": "mytags",
                "Value": "tag1"
            }
        ],
        "EnhancedVpcRouting": false,
        "IamRoles": [],
        "MaintenanceTrackName": "current",
        "DeferredMaintenanceWindows": [
            {
                "DeferMaintenanceIdentifier": "dfm-mUdVIfFcT1B4SGhw6fyF",
                "DeferMaintenanceStartTime": "2019-12-10T18:18:39.354Z",
                "DeferMaintenanceEndTime": "2020-01-09T18:18:39.354Z"
            }
        ],
        "ExpectedNextSnapshotScheduleTime": "2019-12-11T04:42:55.631Z",
        "ExpectedNextSnapshotScheduleTimeStatus": "OnTrack",
        "NextMaintenanceWindowStartTime": "2020-01-11T16:00:00Z"
    }
}
```

Per ulteriori informazioni, consulta Manutenzione del cluster nella Guida alla gestione dei cluster di Amazon Redshift.

- Per i dettagli sull'API, consulta ModifyClusterMaintenance AWS CLICommand Reference.

`modify-cluster-parameter-group`

Il seguente esempio di codice mostra come utilizzare`modify-cluster-parameter-group`.

AWS CLI

Come modificare i parametri in un gruppo di parametri

L'esempio `modify-cluster-parameter-group` seguente modifica il parametro wlm_json_configuration per la gestione dei carichi di lavoro. Il comando accetta i parametri da un file con il contenuto JSON mostrato di seguito.

```
aws redshift modify-cluster-parameter-group \
    --parameter-group-name myclusterparametergroup \
    --parameters file://modify_pg.json
```

Contenuto di `modify_pg.json`:

```
[
    {
        "ParameterName": "wlm_json_configuration",
        "ParameterValue": "[{\"user_group\":\"example_user_group1\",\"query_group\":
 \"example_query_group1\", \"query_concurrency\":7},{\"query_concurrency\":5}]"
    }
]
```

Output:

```
{
   "ParameterGroupStatus": "Your parameter group has been updated but changes won't
 get applied until you reboot the associated Clusters.",
   "ParameterGroupName": "myclusterparametergroup",
   "ResponseMetadata": {
      "RequestId": "09974cc0-64cd-11e2-bea9-49e0ce183f07"
   }
}
```

- Per i dettagli sull'API, consulta ModifyClusterParameterGroup AWS CLICommand Reference.

`modify-cluster-snapshot-schedule`

Il seguente esempio di codice mostra come utilizzare`modify-cluster-snapshot-schedule`.

AWS CLI

Come modificare la pianificazione di snapshot del cluster

L'esempio `modify-cluster-snapshot-schedule` seguente rimuove la pianificazione di snapshot specificata dal cluster specificato.

```
aws redshift modify-cluster-snapshot-schedule \
    --cluster-identifier mycluster \
    --schedule-identifier mysnapshotschedule \
    --disassociate-schedule
```

Il comando non produce output.

Per ulteriori informazioni, consulta Pianificazioni di snapshot automatici nella Guida alla gestione dei cluster di Amazon Redshift.

- Per i dettagli sull'API, consulta ModifyClusterSnapshotSchedule AWS CLICommand Reference.

`modify-cluster-snapshot`

Il seguente esempio di codice mostra come utilizzare`modify-cluster-snapshot`.

AWS CLI

Come modificare uno snapshot del cluster

L'esempio `modify-cluster-snapshot` seguente imposta il periodo di conservazione manuale per lo snapshot del cluster specificato su un valore di 10 giorni.

```
aws redshift modify-cluster-snapshot \
    --snapshot-identifier mycluster-2019-11-06-16-32 \
    --manual-snapshot-retention-period 10
```

Output:

```
{
    "Snapshot": {
        "SnapshotIdentifier": "mycluster-2019-11-06-16-32",
        "ClusterIdentifier": "mycluster",
        "SnapshotCreateTime": "2019-12-07T00:34:05.633Z",
        "Status": "available",
        "Port": 5439,
        "AvailabilityZone": "us-west-2f",
        "ClusterCreateTime": "2019-12-05T18:44:36.991Z",
        "MasterUsername": "adminuser",
        "ClusterVersion": "1.0",
        "SnapshotType": "manual",
        "NodeType": "dc2.large",
        "NumberOfNodes": 2,
        "DBName": "dev",
        "VpcId": "vpc-b1ce17t9",
        "Encrypted": false,
        "EncryptedWithHSM": false,
        "OwnerAccount": "123456789012",
        "TotalBackupSizeInMegaBytes": 64384.0,
        "ActualIncrementalBackupSizeInMegaBytes": 24.0,
        "BackupProgressInMegaBytes": 24.0,
        "CurrentBackupRateInMegaBytesPerSecond": 13.0011,
        "EstimatedSecondsToCompletion": 0,
        "ElapsedTimeInSeconds": 1,
        "Tags": [
            {
                "Key": "mytagkey",
                "Value": "mytagvalue"
            }
        ],
        "EnhancedVpcRouting": false,
        "MaintenanceTrackName": "current",
        "ManualSnapshotRetentionPeriod": 10,
        "ManualSnapshotRemainingDays": 6,
        "SnapshotRetentionStartTime": "2019-12-07T00:34:07.479Z"
    }
}
```


Per ulteriori informazioni, consulta Snapshot di Amazon Redshift nella Guida alla gestione dei cluster di Amazon Redshift.

- Per i dettagli sull'API, consulta ModifyClusterSnapshot AWS CLICommand Reference.

`modify-cluster-subnet-group`

Il seguente esempio di codice mostra come utilizzare`modify-cluster-subnet-group`.

AWS CLI

L' GroupThis esempio di modifica delle sottoreti in una sottorete del cluster mostra come modificare l'elenco delle sottoreti in un gruppo di sottoreti della cache. Per impostazione predefinita, l'output è in formato JSON. Comando:

```
aws redshift modify-cluster-subnet-group --cluster-subnet-group-name mysubnetgroup
 --subnet-ids subnet-763fdd1 subnet-ac830e9
```

Risultato:

```
{
   "ClusterSubnetGroup":
   {
      "Subnets": [
         {
            "SubnetStatus": "Active",
            "SubnetIdentifier": "subnet-763fdd1c",
            "SubnetAvailabilityZone":
               { "Name": "us-east-1a" }
         },
         {
            "SubnetStatus": "Active",
            "SubnetIdentifier": "subnet-ac830e9",
            "SubnetAvailabilityZone":
               { "Name": "us-east-1b" }
         } ],
      "VpcId": "vpc-7e3fdd14",
      "SubnetGroupStatus": "Complete",
      "Description": "My subnet group",
      "ClusterSubnetGroupName": "mysubnetgroup"
   },
   "ResponseMetadata": {
      "RequestId": "8da93e89-8372-f936-93a8-873918938197a"
   }
}
```

- Per i dettagli sull'API, vedere in Command Reference. ModifyClusterSubnetGroupAWS CLI

`modify-cluster`

Il seguente esempio di codice mostra come utilizzare`modify-cluster`.

AWS CLI

Associare un gruppo di sicurezza a un ClusterThis esempio mostra come associare un gruppo di sicurezza del cluster al Cluster.command specificato:

```
aws redshift modify-cluster --cluster-identifier mycluster --cluster-security-groups
 mysecuritygroup
```

Modifica la finestra di manutenzione per ClusterThis mostrare come modificare la finestra di manutenzione settimanale preferita per un cluster in modo che diventi la finestra di manutenzione minima di quattro ore che inizia la domenica alle 23:15 e termina il lunedì alle 3:15. Comando:

```
aws redshift modify-cluster --cluster-identifier mycluster --preferred-maintenance-
window Sun:23:15-Mon:03:15
```

Modifica la password principale: nell' ClusterThis esempio viene illustrato come modificare la password principale per un Cluster.Comando:

```
aws redshift modify-cluster --cluster-identifier mycluster --master-user-password
 A1b2c3d4
```

- Per i dettagli sull'API, vedere ModifyClusterin AWS CLI Command Reference.

`modify-event-subscription`

Il seguente esempio di codice mostra come utilizzare`modify-event-subscription`.

AWS CLI

Come modificare la sottoscrizione a eventi

L'esempio `modify-event-subscription` seguente disabilita la sottoscrizione alle notifiche eventi specificata.

```
aws redshift modify-event-subscription \
    --subscription-name mysubscription \
    --no-enabled
```

Output:

```
{
    "EventSubscription": {
        "CustomerAwsId": "123456789012",
        "CustSubscriptionId": "mysubscription",
        "SnsTopicArn": "arn:aws:sns:us-west-2:123456789012:MySNStopic",
        "Status": "active",
        "SubscriptionCreationTime": "2019-12-09T21:50:21.332Z",
        "SourceIdsList": [],
        "EventCategoriesList": [
            "management"
        ],
        "Severity": "ERROR",
        "Enabled": false,
        "Tags": []
    }
}
```

Per ulteriori informazioni, consulta Sottoscrizione alle notifiche eventi di Amazon Redshift nella Guida alla gestione dei cluster di Amazon Redshift.

- Per i dettagli sull'API, consulta ModifyEventSubscription AWS CLICommand Reference.

`modify-scheduled-action`

Il seguente esempio di codice mostra come utilizzare`modify-scheduled-action`.

AWS CLI

Come modificare un'azione pianificata

L'esempio `modify-scheduled-action` seguente aggiunge una descrizione all'azione pianificata esistente specificata.

```
aws redshift modify-scheduled-action \
    --scheduled-action-name myscheduledaction \
    --scheduled-action-description "My scheduled action"
```

Output:

```
{
```

```
    "ScheduledActionName": "myscheduledaction",
    "TargetAction": {
        "ResizeCluster": {
            "ClusterIdentifier": "mycluster",
            "NumberOfNodes": 2,
            "Classic": false
        }
    },
    "Schedule": "at(2019-12-25T00:00:00)",
    "IamRole": "arn:aws:iam::123456789012:role/myRedshiftRole",
    "ScheduledActionDescription": "My scheduled action",
    "State": "ACTIVE",
    "NextInvocations": [
        "2019-12-25T00:00:00Z"
    ]
}
```

- Per i dettagli sull'API, consulta ModifyScheduledAction AWS CLICommand Reference.

`modify-snapshot-copy-retention-period`

Il seguente esempio di codice mostra come utilizzare`modify-snapshot-copy-retention-period`.

AWS CLI

Come modificare il periodo di conservazione delle copie di snapshot

L'`modify-snapshot-copy-retention-period`esempio seguente modifica il numero di giorni in cui conservare le istantanee per il cluster specificato nella AWS regione di destinazione dopo che sono state copiate dalla regione di origine. AWS

```
aws redshift modify-snapshot-copy-retention-period \
    --cluster-identifier mycluster \
    --retention-period 15
```

Output:

```
{
    "Cluster": {
        "ClusterIdentifier": "mycluster",
        "NodeType": "dc2.large",
```

```
        "ClusterStatus": "available",
        "ClusterAvailabilityStatus": "Available",
        "MasterUsername": "adminuser",
        "DBName": "dev",
        "Endpoint": {
            "Address": "mycluster.cmeaswqeuae.us-west-2.redshift.amazonaws.com",
            "Port": 5439
        },
        "ClusterCreateTime": "2019-12-05T18:44:36.991Z",
        "AutomatedSnapshotRetentionPeriod": 3,
        "ManualSnapshotRetentionPeriod": -1,
        "ClusterSecurityGroups": [],
        "VpcSecurityGroups": [
            {
                "VpcSecurityGroupId": "sh-a1a123ab",
                "Status": "active"
            }
        ],
        "ClusterParameterGroups": [
            {
                "ParameterGroupName": "default.redshift-1.0",
                "ParameterApplyStatus": "in-sync"
            }
        ],
        "ClusterSubnetGroupName": "default",
        "VpcId": "vpc-b1fet7t9",
        "AvailabilityZone": "us-west-2f",
        "PreferredMaintenanceWindow": "sat:16:00-sat:16:30",
        "PendingModifiedValues": {
            "NodeType": "dc2.large",
            "NumberOfNodes": 2,
            "ClusterType": "multi-node"
        },
        "ClusterVersion": "1.0",
        "AllowVersionUpgrade": true,
        "NumberOfNodes": 4,
        "PubliclyAccessible": false,
        "Encrypted": false,
        "ClusterSnapshotCopyStatus": {
            "DestinationRegion": "us-west-1",
            "RetentionPeriod": 15,
            "ManualSnapshotRetentionPeriod": -1
        },
        "Tags": [
```

```
            {
                "Key": "mytags",
                "Value": "tag1"
            }
        ],
        "EnhancedVpcRouting": false,
        "IamRoles": [],
        "MaintenanceTrackName": "current",
        "DeferredMaintenanceWindows": [
            {
                "DeferMaintenanceIdentifier": "dfm-mUdVSfDcT1F4SGhw6fyF",
                "DeferMaintenanceStartTime": "2019-12-10T18:18:39.354Z",
                "DeferMaintenanceEndTime": "2020-01-09T18:18:39.354Z"
            }
        ],
        "NextMaintenanceWindowStartTime": "2020-01-11T16:00:00Z"
    }
}
```

Per ulteriori informazioni, consulta Formato di una pianificazione di snapshot nella Guida alla gestione dei cluster di Amazon Redshift.

- Per i dettagli sull'API, vedere ModifySnapshotCopyRetentionPeriodin AWS CLI Command Reference.

`modify-snapshot-schedule`

Il seguente esempio di codice mostra come utilizzare`modify-snapshot-schedule`.

AWS CLI

Come modificare la pianificazione di snapshot

L'esempio `modify-snapshot-schedule` seguente modifica la pianificazione di snapshot specificata impostando una frequenza di 10 ore.

```
aws redshift modify-snapshot-schedule \
    --schedule-identifier mysnapshotschedule \
    --schedule-definitions "rate(10 hours)"
```

Output:

```
{
```

```
    "ScheduleDefinitions": [
        "rate(10 hours)"
    ],
    "ScheduleIdentifier": "mysnapshotschedule",
    "ScheduleDescription": "My schedule description",
    "Tags": []
}
```

Per ulteriori informazioni, consulta Formato di una pianificazione di snapshot nella Guida alla gestione dei cluster di Amazon Redshift.

- Per i dettagli sull'API, consulta ModifySnapshotSchedule AWS CLICommand Reference.

`purchase-reserved-node-offering`

Il seguente esempio di codice mostra come utilizzare`purchase-reserved-node-offering`.

AWS CLI

L' NodeThis esempio di acquisto di un nodo riservato mostra come acquistare un'offerta di nodi riservati. Il valore di `reserved-node-offering-id` viene ottenuto chiamando `describe-reserved-node-offerings`. Comando:

```
aws redshift purchase-reserved-node-offering --reserved-node-offering-id ceb6a579-cf4c-4343-be8b-d832c45ab51c
```

Risultato:

```
{
   "ReservedNode": {
      "OfferingType": "Heavy Utilization",
      "FixedPrice": "",
      "NodeType": "dw.hs1.xlarge",
      "ReservedNodeId": "1ba8e2e3-bc01-4d65-b35d-a4a3e931547e",
      "UsagePrice": "",
      "RecurringCharges": [
         {
            "RecurringChargeAmount": "",
            "RecurringChargeFrequency": "Hourly"
         }
      ],
      "NodeCount": 1,
```

```
      "State": "payment-pending",
      "StartTime": "2013-02-13T17:08:39.051Z",
      "Duration": 31536000,
      "ReservedNodeOfferingId": "ceb6a579-cf4c-4343-be8b-d832c45ab51c"
   },
   "ResponseMetadata": {
      "RequestId": "01bda7bf-7600-11e2-b605-2568d7396e7f"
   }
}
```

- Per i dettagli sull'API, consulta PurchaseReservedNodeOffering AWS CLICommand Reference.

`reboot-cluster`

Il seguente esempio di codice mostra come utilizzare`reboot-cluster`.

AWS CLI

Riavvia un ClusterThis esempio riavvia un cluster. Per impostazione predefinita, l'output è in formato JSON. Comando:

```
aws redshift reboot-cluster --cluster-identifier mycluster
```

Risultato:

```
{
   "Cluster": {
      "NodeType": "dw.hs1.xlarge",
      "Endpoint": {
         "Port": 5439,
         "Address": "mycluster.coqoarplqhsn.us-east-1.redshift.amazonaws.com"
      },
      "ClusterVersion": "1.0",
      "PubliclyAccessible": "true",
      "MasterUsername": "adminuser",
      "ClusterParameterGroups": [
         {
         "ParameterApplyStatus": "in-sync",
         "ParameterGroupName": "default.redshift-1.0"
         }
      ],
      "ClusterSecurityGroups": [
```

```
            {
            "Status": "active",
            "ClusterSecurityGroupName": "default"
            }
         ],
         "AllowVersionUpgrade": true,
         "VpcSecurityGroups": \[],
         "AvailabilityZone": "us-east-1a",
         "ClusterCreateTime": "2013-01-22T21:59:29.559Z",
         "PreferredMaintenanceWindow": "sun:23:15-mon:03:15",
         "AutomatedSnapshotRetentionPeriod": 1,
         "ClusterStatus": "rebooting",
         "ClusterIdentifier": "mycluster",
         "DBName": "dev",
         "NumberOfNodes": 2,
         "PendingModifiedValues": {}
      },
      "ResponseMetadata": {
         "RequestId": "61c8b564-64e8-11e2-8f7d-3b939af52818"
      }
}
```

- Per i dettagli sull'API, consulta AWS CLI Command RebootClusterReference.

`reset-cluster-parameter-group`

Il seguente esempio di codice mostra come utilizzare`reset-cluster-parameter-group`.

AWS CLI

L' GroupThis esempio di reset Parameters in a Parameter mostra come reimpostare tutti i parametri in un gruppo di parametri.Comando:

```
aws redshift reset-cluster-parameter-group --parameter-group-name
 myclusterparametergroup --reset-all-parameters
```

- Per i dettagli sull'API, consulta AWS CLI Command ResetClusterParameterGroupReference.

`resize-cluster`

Il seguente esempio di codice mostra come utilizzare`resize-cluster`.

AWS CLI

Come ridimensionare un cluster

L'esempio `resize-cluster` seguente ridimensiona il cluster specificato.

```
aws redshift resize-cluster \
    --cluster-identifier mycluster \
    --cluster-type multi-node \
    --node-type dc2.large \
    --number-of-nodes 6 \
    --classic
```

Output:

```
{
    "Cluster": {
        "ClusterIdentifier": "mycluster",
        "NodeType": "dc2.large",
        "ClusterStatus": "resizing",
        "ClusterAvailabilityStatus": "Modifying",
        "MasterUsername": "adminuser",
        "DBName": "dev",
        "Endpoint": {
            "Address": "mycluster.cmeaswqeuae.us-west-2.redshift.amazonaws.com",
            "Port": 5439
        },
        "ClusterCreateTime": "2019-12-05T18:44:36.991Z",
        "AutomatedSnapshotRetentionPeriod": 3,
        "ManualSnapshotRetentionPeriod": -1,
        "ClusterSecurityGroups": [],
        "VpcSecurityGroups": [
            {
                "VpcSecurityGroupId": "sh-a1a123ab",
                "Status": "active"
            }
        ],
        "ClusterParameterGroups": [
            {
                "ParameterGroupName": "default.redshift-1.0",
                "ParameterApplyStatus": "in-sync"
            }
        ],
```

```
        "ClusterSubnetGroupName": "default",
        "VpcId": "vpc-a1abc1a1",
        "AvailabilityZone": "us-west-2f",
        "PreferredMaintenanceWindow": "sat:16:00-sat:16:30",
        "PendingModifiedValues": {
            "NodeType": "dc2.large",
            "NumberOfNodes": 6,
            "ClusterType": "multi-node"
        },
        "ClusterVersion": "1.0",
        "AllowVersionUpgrade": true,
        "NumberOfNodes": 4,
        "PubliclyAccessible": false,
        "Encrypted": false,
        "ClusterSnapshotCopyStatus": {
            "DestinationRegion": "us-west-1",
            "RetentionPeriod": 15,
            "ManualSnapshotRetentionPeriod": -1
        },
        "Tags": [
            {
                "Key": "mytags",
                "Value": "tag1"
            }
        ],
        "EnhancedVpcRouting": false,
        "IamRoles": [],
        "MaintenanceTrackName": "current",
        "DeferredMaintenanceWindows": [
            {
                "DeferMaintenanceIdentifier": "dfm-mUdVCfDcT1B4SGhw6fyF",
                "DeferMaintenanceStartTime": "2019-12-10T18:18:39.354Z",
                "DeferMaintenanceEndTime": "2020-01-09T18:18:39.354Z"
            }
        ],
        "NextMaintenanceWindowStartTime": "2020-01-11T16:00:00Z",
        "ResizeInfo": {
            "ResizeType": "ClassicResize",
            "AllowCancelResize": true
        }
    }
}
```

Per ulteriori informazioni, consulta Ridimensionamento di un cluster nella Guida alla gestione dei cluster di Amazon Redshift.

- Per i dettagli sull'API, consulta ResizeCluster AWS CLICommand Reference.

`restore-from-cluster-snapshot`

Il seguente esempio di codice mostra come utilizzare`restore-from-cluster-snapshot`.

AWS CLI

Ripristina un cluster Da un SnapshotThis esempio ripristina un cluster da uno snapshot.Comando:

```
aws redshift restore-from-cluster-snapshot --cluster-identifier mycluster-clone --
snapshot-identifier my-snapshot-id
```

Risultato:

```
{
   "Cluster": {
      "NodeType": "dw.hs1.xlarge",
      "ClusterVersion": "1.0",
      "PubliclyAccessible": "true",
      "MasterUsername": "adminuser",
      "ClusterParameterGroups": [
         {
         "ParameterApplyStatus": "in-sync",
         "ParameterGroupName": "default.redshift-1.0"
         }
      ],
      "ClusterSecurityGroups": [
         {
         "Status": "active",
         "ClusterSecurityGroupName": "default"
         }
      ],
      "AllowVersionUpgrade": true,
      "VpcSecurityGroups": \[],
      "PreferredMaintenanceWindow": "sun:23:15-mon:03:15",
      "AutomatedSnapshotRetentionPeriod": 1,
      "ClusterStatus": "creating",
      "ClusterIdentifier": "mycluster-clone",
```

```
        "DBName": "dev",
        "NumberOfNodes": 2,
        "PendingModifiedValues": {}
    },
    "ResponseMetadata": {
        "RequestId": "77fd512b-64e3-11e2-8f5b-e90bd6c77476"
    }
}
```

- Per i dettagli sull'API, vedere in Command Reference. RestoreFromClusterSnapshotAWS CLI

`restore-table-from-cluster-snapshot`

Il seguente esempio di codice mostra come utilizzare`restore-table-from-cluster-snapshot`.

AWS CLI

Come ripristinare una tabella da uno snapshot di cluster

L'esempio `restore-table-from-cluster-snapshot` seguente crea una nuova tabella in base alla tabella specificata nello snapshot del cluster specificato.

```
aws redshift restore-table-from-cluster-snapshot /
    --cluster-identifier mycluster /
    --snapshot-identifier mycluster-2019-11-19-16-17 /
    --source-database-name dev /
    --source-schema-name public /
    --source-table-name mytable /
    --target-database-name dev /
    --target-schema-name public /
    --new-table-name mytable-clone
```

Output:

```
{
    "TableRestoreStatus": {
        "TableRestoreRequestId": "a123a12b-abc1-1a1a-a123-a1234ab12345",
        "Status": "PENDING",
        "RequestTime": "2019-12-20T00:20:16.402Z",
        "ClusterIdentifier": "mycluster",
        "SnapshotIdentifier": "mycluster-2019-11-19-16-17",
        "SourceDatabaseName": "dev",
```

```
        "SourceSchemaName": "public",
        "SourceTableName": "mytable",
        "TargetDatabaseName": "dev",
        "TargetSchemaName": "public",
        "NewTableName": "mytable-clone"
    }
}
```

Per ulteriori informazioni, consulta Ripristino di una tabella da uno snapshot nella Guida alla gestione dei cluster di Amazon Redshift.

- Per i dettagli sull'API, consulta RestoreTableFromClusterSnapshot AWS CLICommand Reference.

`revoke-cluster-security-group-ingress`

Il seguente esempio di codice mostra come utilizzare`revoke-cluster-security-group-ingress`.

AWS CLI

L' GroupThis esempio Revoke Access from an EC2 Security revoca l'accesso a un gruppo di sicurezza Amazon denominato. Comando: EC2

```
aws redshift revoke-cluster-security-group-ingress --cluster-security-group-name
 mysecuritygroup --ec2-security-group-name myec2securitygroup --ec2-security-group-
owner-id 123445677890
```

Come revocare l'accesso a un intervallo CIDR L'esempio seguente revoca l'accesso a un intervallo CIDR. Comando:

```
aws redshift revoke-cluster-security-group-ingress --cluster-security-group-name
 mysecuritygroup --cidrip 192.168.100.100/32
```

- Per i dettagli sull'API, consulta Command Reference. RevokeClusterSecurityGroupIngressAWS CLI

`revoke-snapshot-access`

Il seguente esempio di codice mostra come utilizzare`revoke-snapshot-access`.

AWS CLI

Revoca l'autorizzazione di un AWS account al ripristino Un SnapshotThis esempio revoca l'autorizzazione dell' AWS account `444455556666` a ripristinare l'istantanea. `my-snapshot-id` Per impostazione predefinita, l'output è in formato JSON. Comando:

```
aws redshift revoke-snapshot-access --snapshot-id my-snapshot-id --account-with-
restore-access 444455556666
```

Risultato:

```
{
   "Snapshot": {
      "Status": "available",
      "SnapshotCreateTime": "2013-07-17T22:04:18.947Z",
      "EstimatedSecondsToCompletion": 0,
      "AvailabilityZone": "us-east-1a",
      "ClusterVersion": "1.0",
      "MasterUsername": "adminuser",
      "Encrypted": false,
      "OwnerAccount": "111122223333",
      "BackupProgressInMegabytes": 11.0,
      "ElapsedTimeInSeconds": 0,
      "DBName": "dev",
      "CurrentBackupRateInMegabytesPerSecond: 0.1534,
      "ClusterCreateTime": "2013-01-22T21:59:29.559Z",
      "ActualIncrementalBackupSizeInMegabytes"; 11.0,
      "SnapshotType": "manual",
      "NodeType": "dw.hs1.xlarge",
      "ClusterIdentifier": "mycluster",
      "TotalBackupSizeInMegabytes": 20.0,
      "Port": 5439,
      "NumberOfNodes": 2,
      "SnapshotIdentifier": "my-snapshot-id"
   }
}
```

- Per i dettagli sull'API, consulta Command Reference. RevokeSnapshotAccessAWS CLI

`rotate-encryption-key`

Il seguente esempio di codice mostra come utilizzare`rotate-encryption-key`.

AWS CLI

Come ruotare le chiavi di crittografia per un cluster

L'esempio `rotate-encryption-key` seguente ruota la chiave di crittografia per il cluster specificato.

```
aws redshift rotate-encryption-key \
    --cluster-identifier mycluster
```

Output:

```
{
    "Cluster": {
        "ClusterIdentifier": "mycluster",
        "NodeType": "dc2.large",
        "ClusterStatus": "rotating-keys",
        "ClusterAvailabilityStatus": "Modifying",
        "MasterUsername": "adminuser",
        "DBName": "dev",
        "Endpoint": {
            "Address": "mycluster.cmeaswqeuae.us-west-2.redshift.amazonaws.com",
            "Port": 5439
        },
        "ClusterCreateTime": "2019-12-10T19:25:45.886Z",
        "AutomatedSnapshotRetentionPeriod": 30,
        "ManualSnapshotRetentionPeriod": -1,
        "ClusterSecurityGroups": [],
        "VpcSecurityGroups": [
            {
                "VpcSecurityGroupId": "sh-a1a123ab",
                "Status": "active"
            }
        ],
        "ClusterParameterGroups": [
            {
                "ParameterGroupName": "default.redshift-1.0",
                "ParameterApplyStatus": "in-sync"
            }
        ],
        "ClusterSubnetGroupName": "default",
        "VpcId": "vpc-a1abc1a1",
        "AvailabilityZone": "us-west-2a",
```

```
        "PreferredMaintenanceWindow": "sat:16:00-sat:16:30",
        "PendingModifiedValues": {},
        "ClusterVersion": "1.0",
        "AllowVersionUpgrade": true,
        "NumberOfNodes": 2,
        "PubliclyAccessible": false,
        "Encrypted": true,
        "Tags": [],
        "KmsKeyId": "arn:aws:kms:us-west-2:123456789012:key/
bPxRfih3yCo8nvbEXAMPLEKEY",
        "EnhancedVpcRouting": false,
        "IamRoles": [
            {
                "IamRoleArn": "arn:aws:iam::123456789012:role/myRedshiftRole",
                "ApplyStatus": "in-sync"
            }
        ],
        "MaintenanceTrackName": "current",
        "DeferredMaintenanceWindows": [],
        "NextMaintenanceWindowStartTime": "2019-12-14T16:00:00Z"
    }
}
```

Per ulteriori informazioni, consulta Crittografia dei database di Amazon Redshift nella Guida alla gestione dei cluster di Amazon Redshift.

- Per i dettagli sull'API, consulta RotateEncryptionKey AWS CLICommand Reference.

Esempi di utilizzo di Amazon Rekognition AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando Amazon AWS Command Line Interface Rekognition.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`compare-faces`

Il seguente esempio di codice mostra come usare. `compare-faces`

Per ulteriori informazioni, consulta Confronto dei volti nelle immagini.

AWS CLI

Come confrontare i volti in due immagini

Il comando `compare-faces` seguente confronta i volti in due immagini archiviate in un bucket Amazon S3.

```
aws rekognition compare-faces \
    --source-image '{"S3Object":{"Bucket":"MyImageS3Bucket","Name":"source.jpg"}}' \
    --target-image '{"S3Object":{"Bucket":"MyImageS3Bucket","Name":"target.jpg"}}'
```

Output:

```
{
    "UnmatchedFaces": [],
    "FaceMatches": [
        {
            "Face": {
                "BoundingBox": {
                    "Width": 0.12368916720151901,
                    "Top": 0.16007372736930847,
                    "Left": 0.5901257991790771,
                    "Height": 0.25140416622161865
                },
                "Confidence": 100.0,
                "Pose": {
                    "Yaw": -3.7351467609405518,
                    "Roll": -0.10309021919965744,
                    "Pitch": 0.8637830018997192
                },
                "Quality": {
                    "Sharpness": 95.51618957519531,
                    "Brightness": 65.29893493652344
                },
                "Landmarks": [
```

```
                    {
                        "Y": 0.26721030473709106,
                        "X": 0.6204193830490112,
                        "Type": "eyeLeft"
                    },
                    {
                        "Y": 0.26831310987472534,
                        "X": 0.6776827573776245,
                        "Type": "eyeRight"
                    },
                    {
                        "Y": 0.3514654338359833,
                        "X": 0.6241428852081299,
                        "Type": "mouthLeft"
                    },
                    {
                        "Y": 0.35258132219314575,
                        "X": 0.6713621020317078,
                        "Type": "mouthRight"
                    },
                    {
                        "Y": 0.3140771687030792,
                        "X": 0.6428444981575012,
                        "Type": "nose"
                    }
                ]
            },
            "Similarity": 100.0
        }
    ],
    "SourceImageFace": {
        "BoundingBox": {
            "Width": 0.12368916720151901,
            "Top": 0.16007372736930847,
            "Left": 0.5901257991790771,
            "Height": 0.25140416622161865
        },
        "Confidence": 100.0
    }
}
```

Per ulteriori informazioni, consulta Confronto dei volti nelle immagini nella Guida per gli sviluppatori di Amazon Rekognition.

- Per i dettagli sull'API, consulta CompareFaces AWS CLICommand Reference.

`create-collection`

Il seguente esempio di codice mostra come utilizzare`create-collection`.

Per ulteriori informazioni, consulta Creazione di una raccolta.

AWS CLI

Come creare una raccolta

Il comando `create-collection` seguente crea una raccolta con il nome specificato.

```
aws rekognition create-collection \
    --collection-id "MyCollection"
```

Output:

```
{
    "CollectionArn": "aws:rekognition:us-west-2:123456789012:collection/
MyCollection",
    "FaceModelVersion": "4.0",
    "StatusCode": 200
}
```

Per ulteriori informazioni, consulta Creazione di una raccolta nella Guida per gli sviluppatori di Amazon Rekognition.

- Per i dettagli sull'API, consulta CreateCollection AWS CLICommand Reference.

`create-stream-processor`

Il seguente esempio di codice mostra come utilizzare`create-stream-processor`.

AWS CLI

Come creare un nuovo processore di flussi

L'esempio `create-stream-processor` seguente crea un nuovo processore di flussi con la configurazione specificata.

```
aws rekognition create-stream-processor --name my-stream-processor\
    --input '{"KinesisVideoStream":{"Arn":"arn:aws:kinesisvideo:us-
west-2:123456789012:stream/macwebcam/1530559711205"}}'\
    --stream-processor-output '{"KinesisDataStream":{"Arn":"arn:aws:kinesis:us-
west-2:123456789012:stream/AmazonRekognitionRekStream"}}'\
    --role-arn arn:aws:iam::123456789012:role/AmazonRekognitionDetect\
    --settings '{"FaceSearch":
{"CollectionId":"MyCollection","FaceMatchThreshold":85.5}}'
```

Output:

```
{
    "StreamProcessorArn": "arn:aws:rekognition:us-
west-2:123456789012:streamprocessor/my-stream-processor"
}
```

Per ulteriori informazioni, consulta Utilizzo dei video in streaming nella Guida per gli sviluppatori di Amazon Rekognition.

- Per i dettagli sull'API, consulta CreateStreamProcessor AWS CLICommand Reference.

`delete-collection`

Il seguente esempio di codice mostra come utilizzare`delete-collection`.

Per ulteriori informazioni, consulta Eliminazione di una raccolta.

AWS CLI

Come eliminare una raccolta

Il comando `delete-collection` seguente elimina la raccolta specificata.

```
aws rekognition delete-collection \
    --collection-id MyCollection
```

Output:

```
{
    "StatusCode": 200
}
```

Per ulteriori informazioni, consulta Eliminazione di una raccolta nella Guida per gli sviluppatori di Amazon Rekognition.

- Per i dettagli sull'API, consulta DeleteCollection AWS CLICommand Reference.

`delete-faces`

Il seguente esempio di codice mostra come utilizzare`delete-faces`.

Per ulteriori informazioni, consulta Eliminazione dei volti da una raccolta.

AWS CLI

Come eliminare i volti da una raccolta

Il comando `delete-faces` seguente elimina il volto specificato da una raccolta.

```
aws rekognition delete-faces \
    --collection-id MyCollection
    --face-ids '["0040279c-0178-436e-b70a-e61b074e96b0"]'
```

Output:

```
{
    "DeletedFaces": [
        "0040279c-0178-436e-b70a-e61b074e96b0"
    ]
}
```

Per ulteriori informazioni, consulta Eliminazione dei volti da una raccolta nella Guida per gli sviluppatori di Amazon Rekognition.

- Per i dettagli sull'API, consulta DeleteFaces AWS CLICommand Reference.

`delete-stream-processor`

Il seguente esempio di codice mostra come utilizzare`delete-stream-processor`.

AWS CLI

Come eliminare un processore di flussi

Il comando `delete-stream-processor` seguente elimina il processore di flussi specificato.

```
aws rekognition delete-stream-processor \
    --name my-stream-processor
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Utilizzo dei video in streaming nella Guida per gli sviluppatori di Amazon Rekognition.

- Per i dettagli sull'API, consulta DeleteStreamProcessor AWS CLICommand Reference.

`describe-collection`

Il seguente esempio di codice mostra come utilizzare`describe-collection`.

Per ulteriori informazioni, consulta Descrizione di una raccolta.

AWS CLI

Come descrivere una raccolta

L'esempio `describe-collection` seguente visualizza i dettagli della raccolta specificata.

```
aws rekognition describe-collection \
    --collection-id MyCollection
```

Output:

```
{
    "FaceCount": 200,
    "CreationTimestamp": 1569444828.274,
    "CollectionARN": "arn:aws:rekognition:us-west-2:123456789012:collection/
MyCollection",
    "FaceModelVersion": "4.0"
}
```

Per ulteriori informazioni, consulta Descrizione di una raccolta nella Guida per gli sviluppatori di Amazon Rekognition.

- Per i dettagli sull'API, consulta DescribeCollection AWS CLICommand Reference.

`describe-stream-processor`

Il seguente esempio di codice mostra come utilizzare`describe-stream-processor`.

AWS CLI

Come ottenere informazioni su un flusso

Il comando `describe-stream-processor` seguente visualizza i dettagli relativi al processore di flussi specificato.

```
aws rekognition describe-stream-processor \
    --name my-stream-processor
```

Output:

```
{
    "Status": "STOPPED",
    "Name": "my-stream-processor",
    "LastUpdateTimestamp": 1532449292.712,
    "Settings": {
        "FaceSearch": {
            "FaceMatchThreshold": 80.0,
            "CollectionId": "my-collection"
        }
    },
    "RoleArn": "arn:aws:iam::123456789012:role/AmazonRekognitionDetectStream",
    "StreamProcessorArn": "arn:aws:rekognition:us-
west-2:123456789012:streamprocessor/my-stream-processpr",
    "Output": {
        "KinesisDataStream": {
            "Arn": "arn:aws:kinesis:us-west-2:123456789012:stream/
AmazonRekognitionRekStream"
        }
    },
    "Input": {
        "KinesisVideoStream": {
            "Arn": "arn:aws:kinesisvideo:us-west-2:123456789012:stream/
macwebcam/123456789012"
        }
    },
    "CreationTimestamp": 1532449292.712
}
```

Per ulteriori informazioni, consulta Utilizzo dei video in streaming nella Guida per gli sviluppatori di Amazon Rekognition.

- Per i dettagli sull'API, consulta DescribeStreamProcessor AWS CLICommand Reference.

`detect-faces`

Il seguente esempio di codice mostra come utilizzare`detect-faces`.

Per ulteriori informazioni, consulta Rilevamento dei volti in un'immagine.

AWS CLI

Come rilevare volti in un'immagine

Il comando `detect-faces` seguente rileva i volti nell'immagine specificata archiviata in un bucket Amazon S3.

```
aws rekognition detect-faces \
    --image '{"S3Object":{"Bucket":"MyImageS3Bucket","Name":"MyFriend.jpg"}}' \
    --attributes "ALL"
```

Output:

```
{
    "FaceDetails": [
        {
            "Confidence": 100.0,
            "Eyeglasses": {
                "Confidence": 98.91107940673828,
                "Value": false
            },
            "Sunglasses": {
                "Confidence": 99.7966537475586,
                "Value": false
            },
            "Gender": {
                "Confidence": 99.56611633300781,
                "Value": "Male"
            },
            "Landmarks": [
                {
                    "Y": 0.26721030473709106,
```

```
                    "X": 0.6204193830490112,
                    "Type": "eyeLeft"
                },
                {
                    "Y": 0.26831310987472534,
                    "X": 0.6776827573776245,
                    "Type": "eyeRight"
                },
                {
                    "Y": 0.3514654338359833,
                    "X": 0.6241428852081299,
                    "Type": "mouthLeft"
                },
                {
                    "Y": 0.35258132219314575,
                    "X": 0.6713621020317078,
                    "Type": "mouthRight"
                },
                {
                    "Y": 0.3140771687030792,
                    "X": 0.6428444981575012,
                    "Type": "nose"
                },
                {
                    "Y": 0.24662546813488007,
                    "X": 0.6001564860343933,
                    "Type": "leftEyeBrowLeft"
                },
                {
                    "Y": 0.24326619505882263,
                    "X": 0.6303644776344299,
                    "Type": "leftEyeBrowRight"
                },
                {
                    "Y": 0.23818562924861908,
                    "X": 0.6146903038024902,
                    "Type": "leftEyeBrowUp"
                },
                {
                    "Y": 0.24373626708984375,
                    "X": 0.6640064716339111,
                    "Type": "rightEyeBrowLeft"
                },
                {
```

```
                    "Y": 0.24877218902111053,
                    "X": 0.7025929093360901,
                    "Type": "rightEyeBrowRight"
                },
                {
                    "Y": 0.23938551545143127,
                    "X": 0.6823262572288513,
                    "Type": "rightEyeBrowUp"
                },
                {
                    "Y": 0.265746533870697,
                    "X": 0.6112898588180542,
                    "Type": "leftEyeLeft"
                },
                {
                    "Y": 0.2676128149032593,
                    "X": 0.6317071914672852,
                    "Type": "leftEyeRight"
                },
                {
                    "Y": 0.262735515832901,
                    "X": 0.6201658248901367,
                    "Type": "leftEyeUp"
                },
                {
                    "Y": 0.27025148272514343,
                    "X": 0.6206279993057251,
                    "Type": "leftEyeDown"
                },
                {
                    "Y": 0.268223375082016,
                    "X": 0.6658390760421753,
                    "Type": "rightEyeLeft"
                },
                {
                    "Y": 0.2672517001628876,
                    "X": 0.687832236289978,
                    "Type": "rightEyeRight"
                },
                {
                    "Y": 0.26383838057518005,
                    "X": 0.6769183874130249,
                    "Type": "rightEyeUp"
                },
```

```
                {
                    "Y": 0.27138751745224,
                    "X": 0.676596462726593,
                    "Type": "rightEyeDown"
                },
                {
                    "Y": 0.32283174991607666,
                    "X": 0.6350004076957703,
                    "Type": "noseLeft"
                },
                {
                    "Y": 0.3219289481639862,
                    "X": 0.6567046642303467,
                    "Type": "noseRight"
                },
                {
                    "Y": 0.3420318365097046,
                    "X": 0.6450609564781189,
                    "Type": "mouthUp"
                },
                {
                    "Y": 0.3664324879646301,
                    "X": 0.6455618143081665,
                    "Type": "mouthDown"
                },
                {
                    "Y": 0.26721030473709106,
                    "X": 0.6204193830490112,
                    "Type": "leftPupil"
                },
                {
                    "Y": 0.26831310987472534,
                    "X": 0.6776827573776245,
                    "Type": "rightPupil"
                },
                {
                    "Y": 0.26343393325805664,
                    "X": 0.5946047306060791,
                    "Type": "upperJawlineLeft"
                },
                {
                    "Y": 0.3543180525302887,
                    "X": 0.6044883728027344,
                    "Type": "midJawlineLeft"
```

```
                },
                {
                    "Y": 0.4084877669811249,
                    "X": 0.6477024555206299,
                    "Type": "chinBottom"
                },
                {
                    "Y": 0.3562754988670349,
                    "X": 0.707981526851654,
                    "Type": "midJawlineRight"
                },
                {
                    "Y": 0.26580461859703064,
                    "X": 0.7234612107276917,
                    "Type": "upperJawlineRight"
                }
            ],
            "Pose": {
                "Yaw": -3.7351467609405518,
                "Roll": -0.10309021919965744,
                "Pitch": 0.8637830018997192
            },
            "Emotions": [
                {
                    "Confidence": 8.74203109741211,
                    "Type": "SURPRISED"
                },
                {
                    "Confidence": 2.501944065093994,
                    "Type": "ANGRY"
                },
                {
                    "Confidence": 0.7378743290901184,
                    "Type": "DISGUSTED"
                },
                {
                    "Confidence": 3.5296201705932617,
                    "Type": "HAPPY"
                },
                {
                    "Confidence": 1.7162904739379883,
                    "Type": "SAD"
                },
                {
```

```
                "Confidence": 9.518536567687988,
                "Type": "CONFUSED"
            },
            {
                "Confidence": 0.45474427938461304,
                "Type": "FEAR"
            },
            {
                "Confidence": 72.79895782470703,
                "Type": "CALM"
            }
        ],
        "AgeRange": {
            "High": 48,
            "Low": 32
        },
        "EyesOpen": {
            "Confidence": 98.93987274169922,
            "Value": true
        },
        "BoundingBox": {
            "Width": 0.12368916720151901,
            "Top": 0.16007372736930847,
            "Left": 0.5901257991790771,
            "Height": 0.25140416622161865
        },
        "Smile": {
            "Confidence": 93.4493179321289,
            "Value": false
        },
        "MouthOpen": {
            "Confidence": 90.53053283691406,
            "Value": false
        },
        "Quality": {
            "Sharpness": 95.51618957519531,
            "Brightness": 65.29893493652344
        },
        "Mustache": {
            "Confidence": 89.85221099853516,
            "Value": false
        },
        "Beard": {
            "Confidence": 86.1991195678711,
```

```
                "Value": true
            }
        }
    ]
}
```

Per ulteriori informazioni, consulta Rilevamento di volti in un'immagine nella Guida per gli sviluppatori di Amazon Rekognition.

- Per i dettagli sull'API, consulta DetectFaces AWS CLICommand Reference.

`detect-labels`

Il seguente esempio di codice mostra come utilizzare`detect-labels`.

Per ulteriori informazioni, consulta Rilevamento delle etichette in un'immagine.

AWS CLI

Come rilevare un'etichetta in un'immagine

L'esempio `detect-labels` seguente rileva scene e oggetti in un'immagine archiviata in un bucket Amazon S3.

```
aws rekognition detect-labels \
    --image '{"S3Object":{"Bucket":"bucket","Name":"image"}}'
```

Output:

```
{
    "Labels": [
        {
            "Instances": [],
            "Confidence": 99.15271759033203,
            "Parents": [
                {
                    "Name": "Vehicle"
                },
                {
                    "Name": "Transportation"
                }
            ],
            "Name": "Automobile"
```

```
        },
        {
            "Instances": [],
            "Confidence": 99.15271759033203,
            "Parents": [
                {
                    "Name": "Transportation"
                }
            ],
            "Name": "Vehicle"
        },
        {
            "Instances": [],
            "Confidence": 99.15271759033203,
            "Parents": [],
            "Name": "Transportation"
        },
        {
            "Instances": [
                {
                    "BoundingBox": {
                        "Width": 0.10616336017847061,
                        "Top": 0.5039216876029968,
                        "Left": 0.0037978808395564556,
                        "Height": 0.18528179824352264
                    },
                    "Confidence": 99.15271759033203
                },
                {
                    "BoundingBox": {
                        "Width": 0.2429988533258438,
                        "Top": 0.5251884460449219,
                        "Left": 0.7309805154800415,
                        "Height": 0.21577216684818268
                    },
                    "Confidence": 99.1286392211914
                },
                {
                    "BoundingBox": {
                        "Width": 0.14233611524105072,
                        "Top": 0.5333095788955688,
                        "Left": 0.6494812965393066,
                        "Height": 0.15528248250484467
                    },
```

```
            "Confidence": 98.48368072509766
        },
        {
            "BoundingBox": {
                "Width": 0.11086395382881165,
                "Top": 0.5354844927787781,
                "Left": 0.10355594009160995,
                "Height": 0.10271988064050674
            },
            "Confidence": 96.45606231689453
        },
        {
            "BoundingBox": {
                "Width": 0.06254628300666809,
                "Top": 0.5573825240135193,
                "Left": 0.46083059906959534,
                "Height": 0.053911514580249786
            },
            "Confidence": 93.65448760986328
        },
        {
            "BoundingBox": {
                "Width": 0.10105438530445099,
                "Top": 0.534368634223938,
                "Left": 0.5743985772132874,
                "Height": 0.12226245552301407
            },
            "Confidence": 93.06217193603516
        },
        {
            "BoundingBox": {
                "Width": 0.056389667093753815,
                "Top": 0.5235804319381714,
                "Left": 0.9427769780158997,
                "Height": 0.17163699865341187
            },
            "Confidence": 92.6864013671875
        },
        {
            "BoundingBox": {
                "Width": 0.06003860384225845,
                "Top": 0.5441341400146484,
                "Left": 0.22409997880458832,
                "Height": 0.06737709045410156
```

```
            },
            "Confidence": 90.4227066040039
        },
        {
            "BoundingBox": {
                "Width": 0.02848697081208229,
                "Top": 0.5107086896896362,
                "Left": 0,
                "Height": 0.19150497019290924
            },
            "Confidence": 86.65286254882812
        },
        {
            "BoundingBox": {
                "Width": 0.04067881405353546,
                "Top": 0.5566273927688599,
                "Left": 0.316415935754776,
                "Height": 0.03428703173995018
            },
            "Confidence": 85.36471557617188
        },
        {
            "BoundingBox": {
                "Width": 0.043411049991846085,
                "Top": 0.5394920110702515,
                "Left": 0.18293385207653046,
                "Height": 0.0893595889210701
            },
            "Confidence": 82.21705627441406
        },
        {
            "BoundingBox": {
                "Width": 0.031183116137981415,
                "Top": 0.5579366683959961,
                "Left": 0.2853088080883026,
                "Height": 0.03989990055561066
            },
            "Confidence": 81.0157470703125
        },
        {
            "BoundingBox": {
                "Width": 0.031113790348172188,
                "Top": 0.5504819750785828,
                "Left": 0.2580395042896271,
```

```
                    "Height": 0.056484755128622055
                },
                "Confidence": 56.13441467285156
            },
            {
                "BoundingBox": {
                    "Width": 0.08586374670267105,
                    "Top": 0.5438792705535889,
                    "Left": 0.5128012895584106,
                    "Height": 0.08550430089235306
                },
                "Confidence": 52.37760925292969
            }
        ],
        "Confidence": 99.15271759033203,
        "Parents": [
            {
                "Name": "Vehicle"
            },
            {
                "Name": "Transportation"
            }
        ],
        "Name": "Car"
    },
    {
        "Instances": [],
        "Confidence": 98.9914321899414,
        "Parents": [],
        "Name": "Human"
    },
    {
        "Instances": [
            {
                "BoundingBox": {
                    "Width": 0.19360728561878204,
                    "Top": 0.35072067379951477,
                    "Left": 0.43734854459762573,
                    "Height": 0.2742200493812561
                },
                "Confidence": 98.9914321899414
            },
            {
                "BoundingBox": {
```

```
                    "Width": 0.03801717236638069,
                    "Top": 0.5010883808135986,
                    "Left": 0.9155802130699158,
                    "Height": 0.06597328186035156
                },
                "Confidence": 85.02790832519531
            }
        ],
        "Confidence": 98.9914321899414,
        "Parents": [],
        "Name": "Person"
    },
    {
        "Instances": [],
        "Confidence": 93.24951934814453,
        "Parents": [],
        "Name": "Machine"
    },
    {
        "Instances": [
            {
                "BoundingBox": {
                    "Width": 0.03561960905790329,
                    "Top": 0.6468243598937988,
                    "Left": 0.7850857377052307,
                    "Height": 0.08878646790981293
                },
                "Confidence": 93.24951934814453
            },
            {
                "BoundingBox": {
                    "Width": 0.02217046171426773,
                    "Top": 0.6149078607559204,
                    "Left": 0.04757237061858177,
                    "Height": 0.07136218994855881
                },
                "Confidence": 91.5025863647461
            },
            {
                "BoundingBox": {
                    "Width": 0.016197510063648224,
                    "Top": 0.6274210214614868,
                    "Left": 0.6472989320755005,
                    "Height": 0.04955997318029404
```

```
            },
            "Confidence": 85.14686584472656
        },
        {
            "BoundingBox": {
                "Width": 0.020207518711686134,
                "Top": 0.6348286867141724,
                "Left": 0.7295016646385193,
                "Height": 0.07059963047504425
            },
            "Confidence": 83.34547424316406
        },
        {
            "BoundingBox": {
                "Width": 0.020280985161662102,
                "Top": 0.6171894669532776,
                "Left": 0.08744934946298599,
                "Height": 0.05297485366463661
            },
            "Confidence": 79.9981460571289
        },
        {
            "BoundingBox": {
                "Width": 0.018318990245461464,
                "Top": 0.623889148235321,
                "Left": 0.6836880445480347,
                "Height": 0.06730121374130249
            },
            "Confidence": 78.87144470214844
        },
        {
            "BoundingBox": {
                "Width": 0.021310249343514442,
                "Top": 0.6167286038398743,
                "Left": 0.004064912907779217,
                "Height": 0.08317798376083374
            },
            "Confidence": 75.89361572265625
        },
        {
            "BoundingBox": {
                "Width": 0.03604431077837944,
                "Top": 0.7030032277107239,
                "Left": 0.9254803657531738,
```

```
                    "Height": 0.04569442570209503
                },
                "Confidence": 64.402587890625
            },
            {
                "BoundingBox": {
                    "Width": 0.009834849275648594,
                    "Top": 0.5821820497512817,
                    "Left": 0.28094568848609924,
                    "Height": 0.01964157074689865
                },
                "Confidence": 62.79907989501953
            },
            {
                "BoundingBox": {
                    "Width": 0.01475677452981472,
                    "Top": 0.6137543320655823,
                    "Left": 0.5950819253921509,
                    "Height": 0.039063986390829086
                },
                "Confidence": 59.40483474731445
            }
        ],
        "Confidence": 93.24951934814453,
        "Parents": [
            {
                "Name": "Machine"
            }
        ],
        "Name": "Wheel"
    },
    {
        "Instances": [],
        "Confidence": 92.61514282226562,
        "Parents": [],
        "Name": "Road"
    },
    {
        "Instances": [],
        "Confidence": 92.37877655029297,
        "Parents": [
            {
                "Name": "Person"
            }
```

```
            ],
            "Name": "Sport"
        },
        {
            "Instances": [],
            "Confidence": 92.37877655029297,
            "Parents": [
                {
                    "Name": "Person"
                }
            ],
            "Name": "Sports"
        },
        {
            "Instances": [
                {
                    "BoundingBox": {
                        "Width": 0.12326609343290329,
                        "Top": 0.6332163214683533,
                        "Left": 0.44815489649772644,
                        "Height": 0.058117982000112534
                    },
                    "Confidence": 92.37877655029297
                }
            ],
            "Confidence": 92.37877655029297,
            "Parents": [
                {
                    "Name": "Person"
                },
                {
                    "Name": "Sport"
                }
            ],
            "Name": "Skateboard"
        },
        {
            "Instances": [],
            "Confidence": 90.62931060791016,
            "Parents": [
                {
                    "Name": "Person"
                }
            ],
```

```
            "Name": "Pedestrian"
        },
        {
            "Instances": [],
            "Confidence": 88.81334686279297,
            "Parents": [],
            "Name": "Asphalt"
        },
        {
            "Instances": [],
            "Confidence": 88.81334686279297,
            "Parents": [],
            "Name": "Tarmac"
        },
        {
            "Instances": [],
            "Confidence": 88.23201751708984,
            "Parents": [],
            "Name": "Path"
        },
        {
            "Instances": [],
            "Confidence": 80.26520538330078,
            "Parents": [],
            "Name": "Urban"
        },
        {
            "Instances": [],
            "Confidence": 80.26520538330078,
            "Parents": [
                {
                    "Name": "Building"
                },
                {
                    "Name": "Urban"
                }
            ],
            "Name": "Town"
        },
        {
            "Instances": [],
            "Confidence": 80.26520538330078,
            "Parents": [],
            "Name": "Building"
```

```
        },
        {
            "Instances": [],
            "Confidence": 80.26520538330078,
            "Parents": [
                {
                    "Name": "Building"
                },
                {
                    "Name": "Urban"
                }
            ],
            "Name": "City"
        },
        {
            "Instances": [],
            "Confidence": 78.37934875488281,
            "Parents": [
                {
                    "Name": "Car"
                },
                {
                    "Name": "Vehicle"
                },
                {
                    "Name": "Transportation"
                }
            ],
            "Name": "Parking Lot"
        },
        {
            "Instances": [],
            "Confidence": 78.37934875488281,
            "Parents": [
                {
                    "Name": "Car"
                },
                {
                    "Name": "Vehicle"
                },
                {
                    "Name": "Transportation"
                }
            ],
```

```
            "Name": "Parking"
        },
        {
            "Instances": [],
            "Confidence": 74.37590026855469,
            "Parents": [
                {
                    "Name": "Building"
                },
                {
                    "Name": "Urban"
                },
                {
                    "Name": "City"
                }
            ],
            "Name": "Downtown"
        },
        {
            "Instances": [],
            "Confidence": 69.84622955322266,
            "Parents": [
                {
                    "Name": "Road"
                }
            ],
            "Name": "Intersection"
        },
        {
            "Instances": [],
            "Confidence": 57.68518829345703,
            "Parents": [
                {
                    "Name": "Sports Car"
                },
                {
                    "Name": "Car"
                },
                {
                    "Name": "Vehicle"
                },
                {
                    "Name": "Transportation"
                }
```

```
            ],
            "Name": "Coupe"
        },
        {
            "Instances": [],
            "Confidence": 57.68518829345703,
            "Parents": [
                {
                    "Name": "Car"
                },
                {
                    "Name": "Vehicle"
                },
                {
                    "Name": "Transportation"
                }
            ],
            "Name": "Sports Car"
        },
        {
            "Instances": [],
            "Confidence": 56.59492111206055,
            "Parents": [
                {
                    "Name": "Path"
                }
            ],
            "Name": "Sidewalk"
        },
        {
            "Instances": [],
            "Confidence": 56.59492111206055,
            "Parents": [
                {
                    "Name": "Path"
                }
            ],
            "Name": "Pavement"
        },
        {
            "Instances": [],
            "Confidence": 55.58770751953125,
            "Parents": [
                {
```

```
                    "Name": "Building"
                },
                {
                    "Name": "Urban"
                }
            ],
            "Name": "Neighborhood"
        }
    ],
    "LabelModelVersion": "2.0"
}
```

Per ulteriori informazioni, consulta Rilevamento di etichette in un'immagine nella Guida per gli sviluppatori di Amazon Rekognition.

- Per i dettagli sull'API, consulta DetectLabels AWS CLICommand Reference.

`detect-moderation-labels`

Il seguente esempio di codice mostra come utilizzare`detect-moderation-labels`.

Per ulteriori informazioni, consulta Rilevamento di immagini non appropriate.

AWS CLI

Come rilevare contenuti non sicuri in un'immagine

Il comando `detect-moderation-labels` seguente rileva contenuti non sicuri nell'immagine specificata archiviata in un bucket Amazon S3.

```
aws rekognition detect-moderation-labels \
    --image "S3Object={Bucket=MyImageS3Bucket,Name=gun.jpg}"
```

Output:

```
{
    "ModerationModelVersion": "3.0",
    "ModerationLabels": [
        {
            "Confidence": 97.29618072509766,
            "ParentName": "Violence",
            "Name": "Weapon Violence"
        },
```

```
        {
            "Confidence": 97.29618072509766,
            "ParentName": "",
            "Name": "Violence"
        }
    ]
}
```

Per ulteriori informazioni, consulta Rilevamento di immagini non sicure nella Guida per gli sviluppatori di Amazon Rekognition.

- Per i dettagli sull'API, consulta DetectModerationLabels AWS CLICommand Reference.

`detect-text`

Il seguente esempio di codice mostra come utilizzare`detect-text`.

Per ulteriori informazioni, consulta Rilevamento del testo in un'immagine.

AWS CLI

Come rilevare il testo in un'immagine

Il comando `detect-text` seguente rileva il testo nell'immagine specificata.

```
aws rekognition detect-text \
    --image '{"S3Object":{"Bucket":"MyImageS3Bucket","Name":"ExamplePicture.jpg"}}'
```

Output:

```
{
    "TextDetections": [
        {
            "Geometry": {
                "BoundingBox": {
                    "Width": 0.24624845385551453,
                    "Top": 0.28288066387176514,
                    "Left": 0.391388863325119,
                    "Height": 0.022687450051307678
                },
                "Polygon": [
                    {
                        "Y": 0.28288066387176514,
```

```
                    "X": 0.391388863325119
                },
                {
                    "Y": 0.2826388478279114,
                    "X": 0.6376373171806335
                },
                {
                    "Y": 0.30532628297805786,
                    "X": 0.637677013874054
                },
                {
                    "Y": 0.305568128824234,
                    "X": 0.39142853021621704
                }
            ]
        },
        "Confidence": 94.35709381103516,
        "DetectedText": "ESTD 1882",
        "Type": "LINE",
        "Id": 0
    },
    {
        "Geometry": {
            "BoundingBox": {
                "Width": 0.33933889865875244,
                "Top": 0.32603850960731506,
                "Left": 0.34534579515457153,
                "Height": 0.07126858830451965
            },
            "Polygon": [
                {
                    "Y": 0.32603850960731506,
                    "X": 0.34534579515457153
                },
                {
                    "Y": 0.32633158564567566,
                    "X": 0.684684693813324
                },
                {
                    "Y": 0.3976001739501953,
                    "X": 0.684575080871582
                },
                {
                    "Y": 0.3973070979118347,
```

```
                    "X": 0.345236212015152
                }
            ]
        },
        "Confidence": 99.95779418945312,
        "DetectedText": "BRAINS",
        "Type": "LINE",
        "Id": 1
    },
    {
        "Confidence": 97.22098541259766,
        "Geometry": {
            "BoundingBox": {
                "Width": 0.061079490929841995,
                "Top": 0.2843210697174072,
                "Left": 0.391391396522522,
                "Height": 0.021029088646173477
            },
            "Polygon": [
                {
                    "Y": 0.2843210697174072,
                    "X": 0.391391396522522
                },
                {
                    "Y": 0.2828207015991211,
                    "X": 0.4524524509906769
                },
                {
                    "Y": 0.3038259446620941,
                    "X": 0.4534534513950348
                },
                {
                    "Y": 0.30532634258270264,
                    "X": 0.3923923969268799
                }
            ]
        },
        "DetectedText": "ESTD",
        "ParentId": 0,
        "Type": "WORD",
        "Id": 2
    },
    {
        "Confidence": 91.49320983886719,
```

```
            "Geometry": {
                "BoundingBox": {
                    "Width": 0.07007007300853729,
                    "Top": 0.2828207015991211,
                    "Left": 0.5675675868988037,
                    "Height": 0.02250562608242035
                },
                "Polygon": [
                    {
                        "Y": 0.2828207015991211,
                        "X": 0.5675675868988037
                    },
                    {
                        "Y": 0.2828207015991211,
                        "X": 0.6376376152038574
                    },
                    {
                        "Y": 0.30532634258270264,
                        "X": 0.6376376152038574
                    },
                    {
                        "Y": 0.30532634258270264,
                        "X": 0.5675675868988037
                    }
                ]
            },
            "DetectedText": "1882",
            "ParentId": 0,
            "Type": "WORD",
            "Id": 3
        },
        {
            "Confidence": 99.95779418945312,
            "Geometry": {
                "BoundingBox": {
                    "Width": 0.33933934569358826,
                    "Top": 0.32633158564567566,
                    "Left": 0.3453453481197357,
                    "Height": 0.07127484679222107
                },
                "Polygon": [
                    {
                        "Y": 0.32633158564567566,
                        "X": 0.3453453481197357
```

```
                    },
                    {
                        "Y": 0.32633158564567566,
                        "X": 0.684684693813324
                    },
                    {
                        "Y": 0.39759939908981323,
                        "X": 0.6836836934089661
                    },
                    {
                        "Y": 0.39684921503067017,
                        "X": 0.3453453481197357
                    }
                ]
            },
            "DetectedText": "BRAINS",
            "ParentId": 1,
            "Type": "WORD",
            "Id": 4
        }
    ]
}
```

- Per i dettagli sull'API, consulta DetectText AWS CLICommand Reference.

`get-celebrity-info`

Il seguente esempio di codice mostra come utilizzare`get-celebrity-info`.

AWS CLI

Come ottenere informazioni su una celebrità

Il comando `get-celebrity-info` seguente visualizza informazioni sulla celebrità specificata. Il parametro `id` proviene da una precedente chiamata a `recognize-celebrities`.

```
aws rekognition get-celebrity-info --id nnnnnnn
```

Output:

```
{
    "Name": "Celeb A",
```

```
    "Urls": [
        "www.imdb.com/name/aaaaaaaaa"
    ]
}
```

Per ulteriori informazioni, consulta Come recuperare le informazioni su un volto celebre nella Guida per gli sviluppatori di Amazon Rekognition.

- Per i dettagli sull'API, consulta GetCelebrityInfo AWS CLICommand Reference.

`get-celebrity-recognition`

Il seguente esempio di codice mostra come utilizzare`get-celebrity-recognition`.

AWS CLI

Come ottenere i risultati di un'operazione di riconoscimento di celebrità

Il comando `get-celebrity-recognition` seguente visualizza i risultati di un'operazione di riconoscimento di celebrità avviata in precedenza tramite una chiamata a `start-celebrity-recognition`.

```
aws rekognition get-celebrity-recognition  \
    --job-id 1234567890abcdef1234567890abcdef1234567890abcdef1234567890abcdef
```

Output:

```
{
    "NextToken": "3D01ClxlCiT31VsRDkAO3IybLb/h5AtDWSGuhYi
+N1FIJwwPtAkuKzDhL2rV3GcwmNt77+12",
    "Celebrities": [
        {
            "Timestamp": 0,
            "Celebrity": {
                "Confidence": 96.0,
                "Face": {
                    "BoundingBox": {
                        "Width": 0.70333331823349,
                        "Top": 0.16750000417232513,
                        "Left": 0.19555555284023285,
                        "Height": 0.3956249952316284
                    },
                    "Landmarks": [
```

```
                        {
                            "Y": 0.31031012535095215,
                            "X": 0.441436767578125,
                            "Type": "eyeLeft"
                        },
                        {
                            "Y": 0.3081788718700409,
                            "X": 0.6437258720397949,
                            "Type": "eyeRight"
                        },
                        {
                            "Y": 0.39542075991630554,
                            "X": 0.5572493076324463,
                            "Type": "nose"
                        },
                        {
                            "Y": 0.4597957134246826,
                            "X": 0.4579732120037079,
                            "Type": "mouthLeft"
                        },
                        {
                            "Y": 0.45688048005104065,
                            "X": 0.6349081993103027,
                            "Type": "mouthRight"
                        }
                    ],
                    "Pose": {
                        "Yaw": 8.943398475646973,
                        "Roll": -2.0309247970581055,
                        "Pitch": -0.5674862861633301
                    },
                    "Quality": {
                        "Sharpness": 99.40211486816406,
                        "Brightness": 89.47132110595703
                    },
                    "Confidence": 99.99861145019531
                },
                "Name": "CelebrityA",
                "Urls": [
                    "www.imdb.com/name/111111111"
                ],
                "Id": "nnnnnn"
            }
        },
```

```
        {
            "Timestamp": 467,
            "Celebrity": {
                "Confidence": 99.0,
                "Face": {
                    "BoundingBox": {
                        "Width": 0.6877777576446533,
                        "Top": 0.18437500298023224,
                        "Left": 0.20555555820465088,
                        "Height": 0.3868750035762787
                    },
                    "Landmarks": [
                        {
                            "Y": 0.31895750761032104,
                            "X": 0.4411413371562958,
                            "Type": "eyeLeft"
                        },
                        {
                            "Y": 0.3140959143638611,
                            "X": 0.6523157954216003,
                            "Type": "eyeRight"
                        },
                        {
                            "Y": 0.4016456604003906,
                            "X": 0.5682755708694458,
                            "Type": "nose"
                        },
                        {
                            "Y": 0.46894142031669617,
                            "X": 0.4597797095775604,
                            "Type": "mouthLeft"
                        },
                        {
                            "Y": 0.46971091628074646,
                            "X": 0.6286435127258301,
                            "Type": "mouthRight"
                        }
                    ],
                    "Pose": {
                        "Yaw": 10.433465957641602,
                        "Roll": -3.347442388534546,
                        "Pitch": 1.3709543943405151
                    },
                    "Quality": {
```

```
                        "Sharpness": 99.5531005859375,
                        "Brightness": 88.5764389038086
                    },
                    "Confidence": 99.99148559570312
                },
                "Name": "Jane Celebrity",
                "Urls": [
                    "www.imdb.com/name/111111111"
                ],
                "Id": "nnnnnn"
            }
        }
    ],
    "JobStatus": "SUCCEEDED",
    "VideoMetadata": {
        "Format": "QuickTime / MOV",
        "FrameRate": 29.978118896484375,
        "Codec": "h264",
        "DurationMillis": 4570,
        "FrameHeight": 1920,
        "FrameWidth": 1080
    }
}
```

Per ulteriori informazioni, consulta Riconoscimento di un volto celebre in un video archiviato nella Guida per gli sviluppatori di Amazon Rekognition.

- Per i dettagli sull'API, consulta GetCelebrityRecognition AWS CLICommand Reference.

`get-content-moderation`

Il seguente esempio di codice mostra come utilizzare`get-content-moderation`.

AWS CLI

Come ottenere i risultati di un'operazione su contenuti non sicuri

Il comando `get-content-moderation` seguente visualizza i risultati di un'operazione su contenuti non sicuri avviata in precedenza tramite una chiamata a `start-content-moderation`.

```
aws rekognition get-content-moderation \
```

```
    --job-id 1234567890abcdef1234567890abcdef1234567890abcdef1234567890abcdef
```

Output:

```
{
    "NextToken": "dlhcKMHMzpCBGFukz6IO3JMcWiJAamCVhXHt3r6b4b5Tfbyw3q7o+Jeezt
+ZpgfOnW9FCCgQ",
    "ModerationLabels": [
        {
            "Timestamp": 0,
            "ModerationLabel": {
                "Confidence": 97.39583587646484,
                "ParentName": "",
                "Name": "Violence"
            }
        },
        {
            "Timestamp": 0,
            "ModerationLabel": {
                "Confidence": 97.39583587646484,
                "ParentName": "Violence",
                "Name": "Weapon Violence"
            }
        }
    ],
    "JobStatus": "SUCCEEDED",
    "VideoMetadata": {
        "Format": "QuickTime / MOV",
        "FrameRate": 29.97515869140625,
        "Codec": "h264",
        "DurationMillis": 6039,
        "FrameHeight": 1920,
        "FrameWidth": 1080
    }
}
```

Per ulteriori informazioni, consulta Rilevamento di video archiviati non sicuri nella Guida per gli sviluppatori di Amazon Rekognition.

- Per i dettagli sull'API, consulta GetContentModeration AWS CLICommand Reference.

`get-face-detection`

Il seguente esempio di codice mostra come utilizzare`get-face-detection`.

AWS CLI

Come ottenere i risultati di un'operazione di riconoscimento facciale

Il comando `get-face-detection` seguente visualizza i risultati di un'operazione di rilevamento di volti avviata in precedenza tramite una chiamata a `start-face-detection`.

```
aws rekognition get-face-detection \
    --job-id 1234567890abcdef1234567890abcdef1234567890abcdef1234567890abcdef
```

Output:

```
{
    "Faces": [
        {
            "Timestamp": 467,
            "Face": {
                "BoundingBox": {
                    "Width": 0.1560753583908081,
                    "Top": 0.13555361330509186,
                    "Left": -0.0952017530798912,
                    "Height": 0.6934483051300049
                },
                "Landmarks": [
                    {
                        "Y": 0.4013825058937073,
                        "X": -0.041750285774469376,
                        "Type": "eyeLeft"
                    },
                    {
                        "Y": 0.41695496439933777,
                        "X": 0.027979329228401184,
                        "Type": "eyeRight"
                    },
                    {
                        "Y": 0.6375303268432617,
                        "X": -0.04034662991762161,
                        "Type": "mouthLeft"
                    },
```

```
                    {
                        "Y": 0.6497718691825867,
                        "X": 0.013960429467260838,
                        "Type": "mouthRight"
                    },
                    {
                        "Y": 0.5238034129142761,
                        "X": 0.008022055961191654,
                        "Type": "nose"
                    }
                ],
                "Pose": {
                    "Yaw": -58.07863998413086,
                    "Roll": 1.9384294748306274,
                    "Pitch": -24.66305160522461
                },
                "Quality": {
                    "Sharpness": 83.14741516113281,
                    "Brightness": 25.75942611694336
                },
                "Confidence": 87.7622299194336
            }
        },
        {
            "Timestamp": 967,
            "Face": {
                "BoundingBox": {
                    "Width": 0.28559377789497375,
                    "Top": 0.19436298310756683,
                    "Left": 0.024553587660193443,
                    "Height": 0.7216082215309143
                },
                "Landmarks": [
                    {
                        "Y": 0.4650231599807739,
                        "X": 0.16269078850746155,
                        "Type": "eyeLeft"
                    },
                    {
                        "Y": 0.4843238294124603,
                        "X": 0.2782580852508545,
                        "Type": "eyeRight"
                    },
                    {
```

```
                        "Y": 0.71530681848526,
                        "X": 0.1741468608379364,
                        "Type": "mouthLeft"
                    },
                    {
                        "Y": 0.7310671210289001,
                        "X": 0.26857468485832214,
                        "Type": "mouthRight"
                    },
                    {
                        "Y": 0.582602322101593,
                        "X": 0.2566150426864624,
                        "Type": "nose"
                    }
                ],
                "Pose": {
                    "Yaw": 11.487052917480469,
                    "Roll": 5.074230670928955,
                    "Pitch": 15.396159172058105
                },
                "Quality": {
                    "Sharpness": 73.32209777832031,
                    "Brightness": 54.96497344970703
                },
                "Confidence": 99.99998474121094
            }
        }
    ],
    "NextToken":
 "OzL223pDKy9116O/02KXRqFIEAwxjy4PkgYcm3hSo0rdysbXg5Ex0eFgTGEj0ADEac6S037U",
    "JobStatus": "SUCCEEDED",
    "VideoMetadata": {
        "Format": "QuickTime / MOV",
        "FrameRate": 29.970617294311523,
        "Codec": "h264",
        "DurationMillis": 6806,
        "FrameHeight": 1080,
        "FrameWidth": 1920
    }
}
```

Per ulteriori informazioni, consulta Rilevamento di volti in un video archiviato nella Guida per gli sviluppatori di Amazon Rekognition.

- Per i dettagli sull'API, consulta GetFaceDetection AWS CLICommand Reference.

`get-face-search`

Il seguente esempio di codice mostra come utilizzare`get-face-search`.

AWS CLI

Come ottenere i risultati di un'operazione di ricerca di volti

Il comando `get-face-search` seguente visualizza i risultati di un'operazione di ricerca di volti avviata in precedenza tramite una chiamata a `start-face-search`.

```
aws rekognition get-face-search  \
    --job-id 1234567890abcdef1234567890abcdef1234567890abcdef1234567890abcdef
```

Output:

```
{
    "Persons": [
        {
            "Timestamp": 467,
            "FaceMatches": [],
            "Person": {
                "Index": 0,
                "Face": {
                    "BoundingBox": {
                        "Width": 0.1560753583908081,
                        "Top": 0.13555361330509186,
                        "Left": -0.0952017530798912,
                        "Height": 0.6934483051300049
                    },
                    "Landmarks": [
                        {
                            "Y": 0.4013825058937073,
                            "X": -0.041750285774469376,
                            "Type": "eyeLeft"
                        },
                        {
                            "Y": 0.41695496439933777,
                            "X": 0.027979329228401184,
                            "Type": "eyeRight"
                        },
```

```
                    {
                        "Y": 0.6375303268432617,
                        "X": -0.04034662991762161,
                        "Type": "mouthLeft"
                    },
                    {
                        "Y": 0.6497718691825867,
                        "X": 0.013960429467260838,
                        "Type": "mouthRight"
                    },
                    {
                        "Y": 0.5238034129142761,
                        "X": 0.008022055961191654,
                        "Type": "nose"
                    }
                ],
                "Pose": {
                    "Yaw": -58.07863998413086,
                    "Roll": 1.9384294748306274,
                    "Pitch": -24.66305160522461
                },
                "Quality": {
                    "Sharpness": 83.14741516113281,
                    "Brightness": 25.75942611694336
                },
                "Confidence": 87.7622299194336
            }
        }
    },
    {
        "Timestamp": 967,
        "FaceMatches": [
            {
                "Face": {
                    "BoundingBox": {
                        "Width": 0.12368900328874588,
                        "Top": 0.16007399559020996,
                        "Left": 0.5901259779930115,
                        "Height": 0.2514039874076843
                    },
                    "FaceId": "056a95fa-2060-4159-9cab-7ed4daa030fa",
                    "ExternalImageId": "image3.jpg",
                    "Confidence": 100.0,
                    "ImageId": "08f8a078-8929-37fd-8e8f-aadf690e8232"
```

```
                },
                "Similarity": 98.44476318359375
            }
        ],
        "Person": {
            "Index": 1,
            "Face": {
                "BoundingBox": {
                    "Width": 0.28559377789497375,
                    "Top": 0.19436298310756683,
                    "Left": 0.024553587660193443,
                    "Height": 0.7216082215309143
                },
                "Landmarks": [
                    {
                        "Y": 0.4650231599807739,
                        "X": 0.16269078850746155,
                        "Type": "eyeLeft"
                    },
                    {
                        "Y": 0.4843238294124603,
                        "X": 0.2782580852508545,
                        "Type": "eyeRight"
                    },
                    {
                        "Y": 0.71530681848526,
                        "X": 0.1741468608379364,
                        "Type": "mouthLeft"
                    },
                    {
                        "Y": 0.7310671210289001,
                        "X": 0.26857468485832214,
                        "Type": "mouthRight"
                    },
                    {
                        "Y": 0.582602322101593,
                        "X": 0.2566150426864624,
                        "Type": "nose"
                    }
                ],
                "Pose": {
                    "Yaw": 11.487052917480469,
                    "Roll": 5.074230670928955,
                    "Pitch": 15.396159172058105
```

```
                    },
                    "Quality": {
                        "Sharpness": 73.32209777832031,
                        "Brightness": 54.96497344970703
                    },
                    "Confidence": 99.99998474121094
                }
            }
        }
    ],
    "NextToken": "5bkgcezyuaqhtWk3C8OTW6cjRghrwV9XDMivm5B3MXm+Lv6G+L+GejyFHPhoNa/
ldXIC4c/d",
    "JobStatus": "SUCCEEDED",
    "VideoMetadata": {
        "Format": "QuickTime / MOV",
        "FrameRate": 29.970617294311523,
        "Codec": "h264",
        "DurationMillis": 6806,
        "FrameHeight": 1080,
        "FrameWidth": 1920
    }
}
```

Per ulteriori informazioni, consulta Ricerca di volti nei video archiviati nella Guida per gli sviluppatori di Amazon Rekognition.

- Per i dettagli sull'API, consulta GetFaceSearch AWS CLICommand Reference.

`get-label-detection`

Il seguente esempio di codice mostra come utilizzare`get-label-detection`.

AWS CLI

Come ottenere i risultati di un'operazione di rilevamento di oggetti e scene

Il comando `get-label-detection` seguente visualizza i risultati di un'operazione di rilevamento di oggetti e scene avviata in precedenza tramite una chiamata a `start-label-detection`.

```
aws rekognition get-label-detection  \
    --job-id 1234567890abcdef1234567890abcdef1234567890abcdef1234567890abcdef
```

Output:


```
{
    "Labels": [
        {
            "Timestamp": 0,
            "Label": {
                "Instances": [],
                "Confidence": 50.19071578979492,
                "Parents": [
                    {
                        "Name": "Person"
                    },
                    {
                        "Name": "Crowd"
                    }
                ],
                "Name": "Audience"
            }
        },
        {
            "Timestamp": 0,
            "Label": {
                "Instances": [],
                "Confidence": 55.74115753173828,
                "Parents": [
                    {
                        "Name": "Room"
                    },
                    {
                        "Name": "Indoors"
                    },
                    {
                        "Name": "School"
                    }
                ],
                "Name": "Classroom"
            }
        }
    ],
    "JobStatus": "SUCCEEDED",
    "LabelModelVersion": "2.0",
    "VideoMetadata": {
        "Format": "QuickTime / MOV",
```

```
        "FrameRate": 29.970617294311523,
        "Codec": "h264",
        "DurationMillis": 6806,
        "FrameHeight": 1080,
        "FrameWidth": 1920
    },
    "NextToken": "BMugzAi4L72IERzQdbpyMQuEFBsjlo5W0Yx3mfG+sR9mm98E1/
CpObenspRfs/5FBQFs4X7G"
}
```

Per ulteriori informazioni, consulta Rilevamento di etichette in un video nella Guida per gli sviluppatori di Amazon Rekognition.

- Per i dettagli sull'API, consulta GetLabelDetection AWS CLICommand Reference.

`get-person-tracking`

Il seguente esempio di codice mostra come utilizzare`get-person-tracking`.

AWS CLI

Come ottenere i risultati di un'operazione di rilevamento dei movimenti delle persone

Il comando `get-person-tracking` seguente visualizza i risultati di un'operazione di rilevamento dei movimenti delle persone avviata in precedenza tramite una chiamata a `start-person-tracking`.

```
aws rekognition get-person-tracking  \
    --job-id 1234567890abcdef1234567890abcdef1234567890abcdef1234567890abcdef
```

Output:

```
{
    "Persons": [
        {
            "Timestamp": 500,
            "Person": {
                "BoundingBox": {
                    "Width": 0.4151041805744171,
                    "Top": 0.07870370149612427,
                    "Left": 0.0,
                    "Height": 0.9212962985038757
                },
```

```
                "Index": 0
            }
        },
        {
            "Timestamp": 567,
            "Person": {
                "BoundingBox": {
                    "Width": 0.4755208194255829,
                    "Top": 0.07777778059244156,
                    "Left": 0.0,
                    "Height": 0.9194444417953491
                },
                "Index": 0
            }
        }
    ],
    "NextToken": "D/vRIYNyhG79ugdta3f+8cRg9oSRo
+HigGOuxRiYpTn0ExnqTi1CJektVAc4HrAXDv25eHYk",
    "JobStatus": "SUCCEEDED",
    "VideoMetadata": {
        "Format": "QuickTime / MOV",
        "FrameRate": 29.970617294311523,
        "Codec": "h264",
        "DurationMillis": 6806,
        "FrameHeight": 1080,
        "FrameWidth": 1920
    }
}
```


Per ulteriori informazioni, consulta Rilevamento dei percorsi delle persone nella Guida per gli sviluppatori di Amazon Rekognition.

- Per i dettagli sull'API, consulta GetPersonTracking AWS CLICommand Reference.

`index-faces`

Il seguente esempio di codice mostra come utilizzare`index-faces`.

Per ulteriori informazioni, consulta Indicizzazione dei volti in una raccolta.

AWS CLI

Come aggiungere volti a una raccolta

Il comando `index-faces` seguente aggiunge i volti trovati in un'immagine alla raccolta specificata.

```
aws rekognition index-faces \
    --image '{"S3Object":{"Bucket":"MyVideoS3Bucket","Name":"MyPicture.jpg"}}' \
    --collection-id MyCollection \
    --max-faces 1 \
    --quality-filter "AUTO" \
    --detection-attributes "ALL" \
    --external-image-id "MyPicture.jpg"
```

Output:

```
{
    "FaceRecords": [
        {
            "FaceDetail": {
                "Confidence": 99.993408203125,
                "Eyeglasses": {
                    "Confidence": 99.11750030517578,
                    "Value": false
                },
                "Sunglasses": {
                    "Confidence": 99.98249053955078,
                    "Value": false
                },
                "Gender": {
                    "Confidence": 99.92769622802734,
                    "Value": "Male"
                },
                "Landmarks": [
                    {
                        "Y": 0.26750367879867554,
                        "X": 0.6202793717384338,
                        "Type": "eyeLeft"
                    },
                    {
                        "Y": 0.26642778515815735,
                        "X": 0.6787431836128235,
                        "Type": "eyeRight"
                    },
                    {
                        "Y": 0.31361380219459534,
```

```
                        "X": 0.6421601176261902,
                        "Type": "nose"
                    },
                    {
                        "Y": 0.3495299220085144,
                        "X": 0.6216195225715637,
                        "Type": "mouthLeft"
                    },
                    {
                        "Y": 0.35194727778434753,
                        "X": 0.669899046421051,
                        "Type": "mouthRight"
                    },
                    {
                        "Y": 0.26844894886016846,
                        "X": 0.6210268139839172,
                        "Type": "leftPupil"
                    },
                    {
                        "Y": 0.26707562804222107,
                        "X": 0.6817160844802856,
                        "Type": "rightPupil"
                    },
                    {
                        "Y": 0.24834522604942322,
                        "X": 0.6018546223640442,
                        "Type": "leftEyeBrowLeft"
                    },
                    {
                        "Y": 0.24397172033786774,
                        "X": 0.6172008514404297,
                        "Type": "leftEyeBrowUp"
                    },
                    {
                        "Y": 0.24677404761314392,
                        "X": 0.6339119076728821,
                        "Type": "leftEyeBrowRight"
                    },
                    {
                        "Y": 0.24582654237747192,
                        "X": 0.6619398593902588,
                        "Type": "rightEyeBrowLeft"
                    },
                    {
```

```
                    "Y": 0.23973053693771362,
                    "X": 0.6804757118225098,
                    "Type": "rightEyeBrowUp"
                },
                {
                    "Y": 0.24441994726657867,
                    "X": 0.6978968977928162,
                    "Type": "rightEyeBrowRight"
                },
                {
                    "Y": 0.2695908546447754,
                    "X": 0.6085202693939209,
                    "Type": "leftEyeLeft"
                },
                {
                    "Y": 0.26716896891593933,
                    "X": 0.6315826177597046,
                    "Type": "leftEyeRight"
                },
                {
                    "Y": 0.26289820671081543,
                    "X": 0.6202316880226135,
                    "Type": "leftEyeUp"
                },
                {
                    "Y": 0.27123287320137024,
                    "X": 0.6205548048019409,
                    "Type": "leftEyeDown"
                },
                {
                    "Y": 0.2668408751487732,
                    "X": 0.6663622260093689,
                    "Type": "rightEyeLeft"
                },
                {
                    "Y": 0.26741549372673035,
                    "X": 0.6910083889961243,
                    "Type": "rightEyeRight"
                },
                {
                    "Y": 0.2614026665687561,
                    "X": 0.6785826086997986,
                    "Type": "rightEyeUp"
                },
```

```
                    {
                        "Y": 0.27075251936912537,
                        "X": 0.6789616942405701,
                        "Type": "rightEyeDown"
                    },
                    {
                        "Y": 0.3211299479007721,
                        "X": 0.6324167847633362,
                        "Type": "noseLeft"
                    },
                    {
                        "Y": 0.32276326417922974,
                        "X": 0.6558475494384766,
                        "Type": "noseRight"
                    },
                    {
                        "Y": 0.34385165572166443,
                        "X": 0.6444970965385437,
                        "Type": "mouthUp"
                    },
                    {
                        "Y": 0.3671635091304779,
                        "X": 0.6459195017814636,
                        "Type": "mouthDown"
                    }
                ],
                "Pose": {
                    "Yaw": -9.54541015625,
                    "Roll": -0.5709401965141296,
                    "Pitch": 0.6045494675636292
                },
                "Emotions": [
                    {
                        "Confidence": 39.90074157714844,
                        "Type": "HAPPY"
                    },
                    {
                        "Confidence": 23.38753890991211,
                        "Type": "CALM"
                    },
                    {
                        "Confidence": 5.840933322906494,
                        "Type": "CONFUSED"
                    }
```

```
                ],
                "AgeRange": {
                    "High": 63,
                    "Low": 45
                },
                "EyesOpen": {
                    "Confidence": 99.80887603759766,
                    "Value": true
                },
                "BoundingBox": {
                    "Width": 0.18562500178813934,
                    "Top": 0.1618015021085739,
                    "Left": 0.5575000047683716,
                    "Height": 0.24770642817020416
                },
                "Smile": {
                    "Confidence": 99.69740295410156,
                    "Value": false
                },
                "MouthOpen": {
                    "Confidence": 99.97393798828125,
                    "Value": false
                },
                "Quality": {
                    "Sharpness": 95.54405975341797,
                    "Brightness": 63.867706298828125
                },
                "Mustache": {
                    "Confidence": 97.05007934570312,
                    "Value": false
                },
                "Beard": {
                    "Confidence": 87.34505462646484,
                    "Value": false
                }
            },
            "Face": {
                "BoundingBox": {
                    "Width": 0.18562500178813934,
                    "Top": 0.1618015021085739,
                    "Left": 0.5575000047683716,
                    "Height": 0.24770642817020416
                },
                "FaceId": "ce7ed422-2132-4a11-ab14-06c5c410f29f",
```

```
                "ExternalImageId": "example-image.jpg",
                "Confidence": 99.993408203125,
                "ImageId": "8d67061e-90d2-598f-9fbd-29c8497039c0"
            }
        }
    ],
    "UnindexedFaces": [],
    "FaceModelVersion": "3.0",
    "OrientationCorrection": "ROTATE_0"
}
```

Per ulteriori informazioni, consulta Aggiunta di volti a una raccolta nella Guida per gli sviluppatori di Amazon Rekognition.

- Per i dettagli sull'API, consulta IndexFaces AWS CLICommand Reference.

`list-collections`

Il seguente esempio di codice mostra come utilizzare`list-collections`.

Per ulteriori informazioni, consulta Creazione dell'elenco delle raccolte.

AWS CLI

Come elencare le raccolte disponibili

Il `list-collections` comando seguente elenca le raccolte disponibili nell' AWS account.

```
aws rekognition list-collections
```

Output:

```
{
    "FaceModelVersions": [
        "2.0",
        "3.0",
        "3.0",
        "3.0",
        "4.0",
        "1.0",
        "3.0",
        "4.0",
        "4.0",
```

```
        "4.0"
    ],
    "CollectionIds": [
        "MyCollection1",
        "MyCollection2",
        "MyCollection3",
        "MyCollection4",
        "MyCollection5",
        "MyCollection6",
        "MyCollection7",
        "MyCollection8",
        "MyCollection9",
        "MyCollection10"
    ]
}
```

Per ulteriori informazioni, consulta Elenchi delle raccolte nella Guida per gli sviluppatori di Amazon Rekognition.

- Per i dettagli sull'API, vedere ListCollectionsin AWS CLI Command Reference.

`list-faces`

Il seguente esempio di codice mostra come utilizzare`list-faces`.

Per ulteriori informazioni, consulta Creazione dell'elenco dei volti in una raccolta.

AWS CLI

Come elencare i volti in una raccolta

Il comando `list-faces` seguente elenca i volti in un'immagine nella raccolta specificata.

```
aws rekognition list-faces \
    --collection-id MyCollection
```

Output:

```
{
    "FaceModelVersion": "3.0",
    "Faces": [
        {
            "BoundingBox": {
```

```
                "Width": 0.5216310024261475,
                "Top": 0.3256250023841858,
                "Left": 0.13394300639629364,
                "Height": 0.3918749988079071
            },
            "FaceId": "0040279c-0178-436e-b70a-e61b074e96b0",
            "ExternalImageId": "image1.jpg",
            "Confidence": 100.0,
            "ImageId": "f976e487-3719-5e2d-be8b-ea2724c26991"
        },
        {
            "BoundingBox": {
                "Width": 0.5074880123138428,
                "Top": 0.3774999976158142,
                "Left": 0.18302799761295319,
                "Height": 0.3812499940395355
            },
            "FaceId": "086261e8-6deb-4bc0-ac73-ab22323cc38d",
            "ExternalImageId": "image2.jpg",
            "Confidence": 99.99930572509766,
            "ImageId": "ae1593b0-a8f6-5e24-a306-abf529e276fa"
        },
        {
            "BoundingBox": {
                "Width": 0.5574039816856384,
                "Top": 0.37187498807907104,
                "Left": 0.14559100568294525,
                "Height": 0.4181250035762787
            },
            "FaceId": "11c4bd3c-19c5-4eb8-aecc-24feb93a26e1",
            "ExternalImageId": "image3.jpg",
            "Confidence": 99.99960327148438,
            "ImageId": "80739b4d-883f-5b78-97cf-5124038e26b9"
        },
        {
            "BoundingBox": {
                "Width": 0.18562500178813934,
                "Top": 0.1618019938468933,
                "Left": 0.5575000047683716,
                "Height": 0.24770599603652954
            },
            "FaceId": "13692fe4-990a-4679-b14a-5ac23d135eab",
            "ExternalImageId": "image4.jpg",
            "Confidence": 99.99340057373047,
```

```
            "ImageId": "8df18239-9ad1-5acd-a46a-6581ff98f51b"
        },
        {
            "BoundingBox": {
                "Width": 0.5307819843292236,
                "Top": 0.2862499952316284,
                "Left": 0.1564060002565384,
                "Height": 0.3987500071525574
            },
            "FaceId": "2eb5f3fd-e2a9-4b1c-a89f-afa0a518fe06",
            "ExternalImageId": "image5.jpg",
            "Confidence": 99.99970245361328,
            "ImageId": "3c314792-197d-528d-bbb6-798ed012c150"
        },
        {
            "BoundingBox": {
                "Width": 0.5773710012435913,
                "Top": 0.34437501430511475,
                "Left": 0.12396000325679779,
                "Height": 0.4337500035762787
            },
            "FaceId": "57189455-42b0-4839-a86c-abda48b13174",
            "ExternalImageId": "image6.jpg",
            "Confidence": 100.0,
            "ImageId": "0aff2f37-e7a2-5dbc-a3a3-4ef6ec18eaa0"
        },
        {
            "BoundingBox": {
                "Width": 0.5349419713020325,
                "Top": 0.29124999046325684,
                "Left": 0.16389399766921997,
                "Height": 0.40187498927116394
            },
            "FaceId": "745f7509-b1fa-44e0-8b95-367b1359638a",
            "ExternalImageId": "image7.jpg",
            "Confidence": 99.99979400634766,
            "ImageId": "67a34327-48d1-5179-b042-01e52ccfeada"
        },
        {
            "BoundingBox": {
                "Width": 0.41499999165534973,
                "Top": 0.09187500178813934,
                "Left": 0.28083300590515137,
                "Height": 0.3112500011920929
```

```
            },
            "FaceId": "8d3cfc70-4ba8-4b36-9644-90fba29c2dac",
            "ExternalImageId": "image8.jpg",
            "Confidence": 99.99769592285156,
            "ImageId": "a294da46-2cb1-5cc4-9045-61d7ca567662"
        },
        {
            "BoundingBox": {
                "Width": 0.48166701197624207,
                "Top": 0.20999999344348907,
                "Left": 0.21250000596046448,
                "Height": 0.36125001311302185
            },
            "FaceId": "bd4ceb4d-9acc-4ab7-8ef8-1c2d2ba0a66a",
            "ExternalImageId": "image9.jpg",
            "Confidence": 99.99949645996094,
            "ImageId": "5e1a7588-e5a0-5ee3-bd00-c642518dfe3a"
        },
        {
            "BoundingBox": {
                "Width": 0.18562500178813934,
                "Top": 0.1618019938468933,
                "Left": 0.5575000047683716,
                "Height": 0.24770599603652954
            },
            "FaceId": "ce7ed422-2132-4a11-ab14-06c5c410f29f",
            "ExternalImageId": "image10.jpg",
            "Confidence": 99.99340057373047,
            "ImageId": "8d67061e-90d2-598f-9fbd-29c8497039c0"
        }
    ]
}
```

Per ulteriori informazioni, consulta Elenco di volti in una raccolta nella Guida per gli sviluppatori di Amazon Rekognition.

- Per i dettagli sull'API, consulta ListFaces AWS CLICommand Reference.

`list-stream-processors`

Il seguente esempio di codice mostra come utilizzare`list-stream-processors`.

AWS CLI

Come elencare i processori di flussi nell'account corrente

Il comando `list-stream-processors` seguente elenca i processori dei flussi presenti nell'account corrente e il relativo stato.

```
aws rekognition list-stream-processors
```

Output:

```
{
    "StreamProcessors": [
        {
            "Status": "STOPPED",
            "Name": "my-stream-processor"
        }
    ]
}
```

Per ulteriori informazioni, consulta Utilizzo dei video in streaming nella Guida per gli sviluppatori di Amazon Rekognition.

- Per i dettagli sull'API, consulta ListStreamProcessors AWS CLICommand Reference.

`recognize-celebrities`

Il seguente esempio di codice mostra come utilizzare`recognize-celebrities`.

Per ulteriori informazioni, consulta Riconoscimento delle celebrità in un'immagine.

AWS CLI

Come riconoscere le celebrità in un'immagine

Il comando `recognize-celebrities` seguente riconosce le celebrità nell'immagine specificata archiviata in un bucket Amazon S3:

```
aws rekognition recognize-celebrities \
    --image "S3Object={Bucket=MyImageS3Bucket,Name=moviestars.jpg}"
```

Output:


```
{
    "UnrecognizedFaces": [
        {
            "BoundingBox": {
                "Width": 0.14416666328907013,
                "Top": 0.07777778059244156,
                "Left": 0.625,
                "Height": 0.2746031880378723
            },
            "Confidence": 99.9990234375,
            "Pose": {
                "Yaw": 10.80408763885498,
                "Roll": -12.761146545410156,
                "Pitch": 10.96889877319336
            },
            "Quality": {
                "Sharpness": 94.1185531616211,
                "Brightness": 79.18367004394531
            },
            "Landmarks": [
                {
                    "Y": 0.18220913410186768,
                    "X": 0.6702951788902283,
                    "Type": "eyeLeft"
                },
                {
                    "Y": 0.16337193548679352,
                    "X": 0.7188183665275574,
                    "Type": "eyeRight"
                },
                {
                    "Y": 0.20739148557186127,
                    "X": 0.7055801749229431,
                    "Type": "nose"
                },
                {
                    "Y": 0.2889308035373688,
                    "X": 0.687512218952179,
                    "Type": "mouthLeft"
                },
                {
                    "Y": 0.2706988751888275,
```

```
                    "X": 0.7250053286552429,
                    "Type": "mouthRight"
                }
            ]
        }
    ],
    "CelebrityFaces": [
        {
            "MatchConfidence": 100.0,
            "Face": {
                "BoundingBox": {
                    "Width": 0.14000000059604645,
                    "Top": 0.1190476194024086,
                    "Left": 0.82833331823349,
                    "Height": 0.2666666805744171
                },
                "Confidence": 99.99359130859375,
                "Pose": {
                    "Yaw": -10.509642601013184,
                    "Roll": -14.51749324798584,
                    "Pitch": 13.799399375915527
                },
                "Quality": {
                    "Sharpness": 78.74752044677734,
                    "Brightness": 42.201324462890625
                },
                "Landmarks": [
                    {
                        "Y": 0.2290833294391632,
                        "X": 0.8709492087364197,
                        "Type": "eyeLeft"
                    },
                    {
                        "Y": 0.20639978349208832,
                        "X": 0.9153988361358643,
                        "Type": "eyeRight"
                    },
                    {
                        "Y": 0.25417643785476685,
                        "X": 0.8907724022865295,
                        "Type": "nose"
                    },
                    {
                        "Y": 0.32729196548461914,
```

```
                    "X": 0.8876466155052185,
                    "Type": "mouthLeft"
                },
                {
                    "Y": 0.3115464746952057,
                    "X": 0.9238573312759399,
                    "Type": "mouthRight"
                }
            ]
        },
        "Name": "Celeb A",
        "Urls": [
            "www.imdb.com/name/aaaaaaaaa"
        ],
        "Id": "1111111"
    },
    {
        "MatchConfidence": 97.0,
        "Face": {
            "BoundingBox": {
                "Width": 0.13333334028720856,
                "Top": 0.24920634925365448,
                "Left": 0.4449999928474426,
                "Height": 0.2539682686328888
            },
            "Confidence": 99.99979400634766,
            "Pose": {
                "Yaw": 6.557040691375732,
                "Roll": -7.316643714904785,
                "Pitch": 9.272967338562012
            },
            "Quality": {
                "Sharpness": 83.23492431640625,
                "Brightness": 78.83267974853516
            },
            "Landmarks": [
                {
                    "Y": 0.3625510632991791,
                    "X": 0.48898839950561523,
                    "Type": "eyeLeft"
                },
                {
                    "Y": 0.35366007685661316,
                    "X": 0.5313721299171448,
```

```
                    "Type": "eyeRight"
                },
                {
                    "Y": 0.3894785940647125,
                    "X": 0.5173314809799194,
                    "Type": "nose"
                },
                {
                    "Y": 0.44889405369758606,
                    "X": 0.5020005702972412,
                    "Type": "mouthLeft"
                },
                {
                    "Y": 0.4408611059188843,
                    "X": 0.5351271629333496,
                    "Type": "mouthRight"
                }
            ]
        },
        "Name": "Celeb B",
        "Urls": [
            "www.imdb.com/name/bbbbbbbbb"
        ],
        "Id": "2222222"
    },
    {
        "MatchConfidence": 100.0,
        "Face": {
            "BoundingBox": {
                "Width": 0.12416666746139526,
                "Top": 0.2968254089355469,
                "Left": 0.2150000035762787,
                "Height": 0.23650793731212616
            },
            "Confidence": 99.99958801269531,
            "Pose": {
                "Yaw": 7.801797866821289,
                "Roll": -8.326810836791992,
                "Pitch": 7.844768047332764
            },
            "Quality": {
                "Sharpness": 86.93206024169922,
                "Brightness": 79.81291198730469
            },
```

```
                "Landmarks": [
                    {
                        "Y": 0.4027804136276245,
                        "X": 0.2575301229953766,
                        "Type": "eyeLeft"
                    },
                    {
                        "Y": 0.3934555947780609,
                        "X": 0.2956969439983368,
                        "Type": "eyeRight"
                    },
                    {
                        "Y": 0.4309830069541931,
                        "X": 0.2837020754814148,
                        "Type": "nose"
                    },
                    {
                        "Y": 0.48186683654785156,
                        "X": 0.26812544465065,
                        "Type": "mouthLeft"
                    },
                    {
                        "Y": 0.47338807582855225,
                        "X": 0.29905644059181213,
                        "Type": "mouthRight"
                    }
                ]
            },
            "Name": "Celeb C",
            "Urls": [
                "www.imdb.com/name/ccccccccc"
            ],
            "Id": "3333333"
        },
        {
            "MatchConfidence": 97.0,
            "Face": {
                "BoundingBox": {
                    "Width": 0.11916666477918625,
                    "Top": 0.3698412775993347,
                    "Left": 0.008333333767950535,
                    "Height": 0.22698412835597992
                },
                "Confidence": 99.99999237060547,
```

```
                "Pose": {
                    "Yaw": 16.38478660583496,
                    "Roll": -1.0260354280471802,
                    "Pitch": 5.975185394287109
                },
                "Quality": {
                    "Sharpness": 83.23492431640625,
                    "Brightness": 61.408443450927734
                },
                "Landmarks": [
                    {
                        "Y": 0.4632347822189331,
                        "X": 0.049406956881284714,
                        "Type": "eyeLeft"
                    },
                    {
                        "Y": 0.46388113498687744,
                        "X": 0.08722897619009018,
                        "Type": "eyeRight"
                    },
                    {
                        "Y": 0.5020678639411926,
                        "X": 0.0758260041475296,
                        "Type": "nose"
                    },
                    {
                        "Y": 0.544157862663269,
                        "X": 0.054029736667871475,
                        "Type": "mouthLeft"
                    },
                    {
                        "Y": 0.5463630557060242,
                        "X": 0.08464983850717545,
                        "Type": "mouthRight"
                    }
                ]
            },
            "Name": "Celeb D",
            "Urls": [
                "www.imdb.com/name/ddddddddd"
            ],
            "Id": "4444444"
        }
    ]
```

```
}
```

Per ulteriori informazioni, consulta Riconoscimento di volti celebri in un'immagine nella Guida per gli sviluppatori di Amazon Rekognition.

- Per i dettagli sull'API, consulta RecognizeCelebrities AWS CLICommand Reference.

`search-faces-by-image`

Il seguente esempio di codice mostra come utilizzare`search-faces-by-image`.

Per ulteriori informazioni, consulta Ricerca di un volto (immagine).

AWS CLI

Come cercare in una raccolta i volti corrispondenti al volto più grande in un'immagine.

Il comando `search-faces-by-image` seguente cerca in una raccolta i volti corrispondenti al volto più grande nell'immagine specificata:

```
aws rekognition search-faces-by-image \
    --image '{"S3Object":{"Bucket":"MyImageS3Bucket","Name":"ExamplePerson.jpg"}}' \
    --collection-id MyFaceImageCollection

{
    "SearchedFaceBoundingBox": {
        "Width": 0.18562500178813934,
        "Top": 0.1618015021085739,
        "Left": 0.5575000047683716,
        "Height": 0.24770642817020416
    },
    "SearchedFaceConfidence": 99.993408203125,
    "FaceMatches": [
        {
            "Face": {
                "BoundingBox": {
                    "Width": 0.18562500178813934,
                    "Top": 0.1618019938468933,
                    "Left": 0.5575000047683716,
                    "Height": 0.24770599603652954
                },
                "FaceId": "ce7ed422-2132-4a11-ab14-06c5c410f29f",
                "ExternalImageId": "example-image.jpg",
```

```
            "Confidence": 99.99340057373047,
            "ImageId": "8d67061e-90d2-598f-9fbd-29c8497039c0"
        },
        "Similarity": 99.97913360595703
    },
    {
        "Face": {
            "BoundingBox": {
                "Width": 0.18562500178813934,
                "Top": 0.1618019938468933,
                "Left": 0.5575000047683716,
                "Height": 0.24770599603652954
            },
            "FaceId": "13692fe4-990a-4679-b14a-5ac23d135eab",
            "ExternalImageId": "image3.jpg",
            "Confidence": 99.99340057373047,
            "ImageId": "8df18239-9ad1-5acd-a46a-6581ff98f51b"
        },
        "Similarity": 99.97913360595703
    },
    {
        "Face": {
            "BoundingBox": {
                "Width": 0.41499999165534973,
                "Top": 0.09187500178813934,
                "Left": 0.28083300590515137,
                "Height": 0.3112500011920929
            },
            "FaceId": "8d3cfc70-4ba8-4b36-9644-90fba29c2dac",
            "ExternalImageId": "image2.jpg",
            "Confidence": 99.99769592285156,
            "ImageId": "a294da46-2cb1-5cc4-9045-61d7ca567662"
        },
        "Similarity": 99.18069458007812
    },
    {
        "Face": {
            "BoundingBox": {
                "Width": 0.48166701197624207,
                "Top": 0.20999999344348907,
                "Left": 0.21250000596046448,
                "Height": 0.36125001311302185
            },
            "FaceId": "bd4ceb4d-9acc-4ab7-8ef8-1c2d2ba0a66a",
```

```
            "ExternalImageId": "image1.jpg",
            "Confidence": 99.99949645996094,
            "ImageId": "5e1a7588-e5a0-5ee3-bd00-c642518dfe3a"
        },
        "Similarity": 98.66607666015625
    },
    {
        "Face": {
            "BoundingBox": {
                "Width": 0.5349419713020325,
                "Top": 0.29124999046325684,
                "Left": 0.16389399766921997,
                "Height": 0.40187498927116394
            },
            "FaceId": "745f7509-b1fa-44e0-8b95-367b1359638a",
            "ExternalImageId": "image9.jpg",
            "Confidence": 99.99979400634766,
            "ImageId": "67a34327-48d1-5179-b042-01e52ccfeada"
        },
        "Similarity": 98.24278259277344
    },
    {
        "Face": {
            "BoundingBox": {
                "Width": 0.5307819843292236,
                "Top": 0.2862499952316284,
                "Left": 0.1564060002565384,
                "Height": 0.3987500071525574
            },
            "FaceId": "2eb5f3fd-e2a9-4b1c-a89f-afa0a518fe06",
            "ExternalImageId": "image10.jpg",
            "Confidence": 99.99970245361328,
            "ImageId": "3c314792-197d-528d-bbb6-798ed012c150"
        },
        "Similarity": 98.10665893554688
    },
    {
        "Face": {
            "BoundingBox": {
                "Width": 0.5074880123138428,
                "Top": 0.3774999976158142,
                "Left": 0.18302799761295319,
                "Height": 0.3812499940395355
            },
```

```
                "FaceId": "086261e8-6deb-4bc0-ac73-ab22323cc38d",
                "ExternalImageId": "image6.jpg",
                "Confidence": 99.99930572509766,
                "ImageId": "ae1593b0-a8f6-5e24-a306-abf529e276fa"
            },
            "Similarity": 98.10526275634766
        },
        {
            "Face": {
                "BoundingBox": {
                    "Width": 0.5574039816856384,
                    "Top": 0.37187498807907104,
                    "Left": 0.14559100568294525,
                    "Height": 0.4181250035762787
                },
                "FaceId": "11c4bd3c-19c5-4eb8-aecc-24feb93a26e1",
                "ExternalImageId": "image5.jpg",
                "Confidence": 99.99960327148438,
                "ImageId": "80739b4d-883f-5b78-97cf-5124038e26b9"
            },
            "Similarity": 97.94659423828125
        },
        {
            "Face": {
                "BoundingBox": {
                    "Width": 0.5773710012435913,
                    "Top": 0.34437501430511475,
                    "Left": 0.12396000325679779,
                    "Height": 0.4337500035762787
                },
                "FaceId": "57189455-42b0-4839-a86c-abda48b13174",
                "ExternalImageId": "image8.jpg",
                "Confidence": 100.0,
                "ImageId": "0aff2f37-e7a2-5dbc-a3a3-4ef6ec18eaa0"
            },
            "Similarity": 97.93476867675781
        }
    ],

 "FaceModelVersion": "3.0"
}
```

Per ulteriori informazioni, consulta Ricerca di un volto tramite immagine nella Guida per gli sviluppatori di Amazon Rekognition.

- Per i dettagli sull'API, consulta SearchFacesByImage AWS CLICommand Reference.

`search-faces`

Il seguente esempio di codice mostra come utilizzare`search-faces`.

Per ulteriori informazioni, consulta Ricerca di un volto (ID volto).

AWS CLI

Come cercare in una raccolta i volti corrispondenti a un ID volto.

Il comando `search-faces` seguente cerca in una raccolta i volti corrispondenti all'ID volto specificato.

```
aws rekognition search-faces \
    --face-id 8d3cfc70-4ba8-4b36-9644-90fba29c2dac \
    --collection-id MyCollection
```

Output:

```
{
    "SearchedFaceId": "8d3cfc70-4ba8-4b36-9644-90fba29c2dac",
    "FaceModelVersion": "3.0",
    "FaceMatches": [
        {
            "Face": {
                "BoundingBox": {
                    "Width": 0.48166701197624207,
                    "Top": 0.20999999344348907,
                    "Left": 0.21250000596046448,
                    "Height": 0.36125001311302185
                },
                "FaceId": "bd4ceb4d-9acc-4ab7-8ef8-1c2d2ba0a66a",
                "ExternalImageId": "image1.jpg",
                "Confidence": 99.99949645996094,
                "ImageId": "5e1a7588-e5a0-5ee3-bd00-c642518dfe3a"
            },
            "Similarity": 99.30997467041016
        },
```

```
        {
            "Face": {
                "BoundingBox": {
                    "Width": 0.18562500178813934,
                    "Top": 0.1618019938468933,
                    "Left": 0.5575000047683716,
                    "Height": 0.24770599603652954
                },
                "FaceId": "ce7ed422-2132-4a11-ab14-06c5c410f29f",
                "ExternalImageId": "example-image.jpg",
                "Confidence": 99.99340057373047,
                "ImageId": "8d67061e-90d2-598f-9fbd-29c8497039c0"
            },
            "Similarity": 99.24862670898438
        },
        {
            "Face": {
                "BoundingBox": {
                    "Width": 0.18562500178813934,
                    "Top": 0.1618019938468933,
                    "Left": 0.5575000047683716,
                    "Height": 0.24770599603652954
                },
                "FaceId": "13692fe4-990a-4679-b14a-5ac23d135eab",
                "ExternalImageId": "image3.jpg",
                "Confidence": 99.99340057373047,
                "ImageId": "8df18239-9ad1-5acd-a46a-6581ff98f51b"
            },
            "Similarity": 99.24862670898438
        },
        {
            "Face": {
                "BoundingBox": {
                    "Width": 0.5349419713020325,
                    "Top": 0.29124999046325684,
                    "Left": 0.16389399766921997,
                    "Height": 0.40187498927116394
                },
                "FaceId": "745f7509-b1fa-44e0-8b95-367b1359638a",
                "ExternalImageId": "image9.jpg",
                "Confidence": 99.99979400634766,
                "ImageId": "67a34327-48d1-5179-b042-01e52ccfeada"
            },
            "Similarity": 96.73158264160156
```

```
        },
        {
            "Face": {
                "BoundingBox": {
                    "Width": 0.5307819843292236,
                    "Top": 0.2862499952316284,
                    "Left": 0.1564060002565384,
                    "Height": 0.3987500071525574
                },
                "FaceId": "2eb5f3fd-e2a9-4b1c-a89f-afa0a518fe06",
                "ExternalImageId": "image10.jpg",
                "Confidence": 99.99970245361328,
                "ImageId": "3c314792-197d-528d-bbb6-798ed012c150"
            },
            "Similarity": 96.48291015625
        },
        {
            "Face": {
                "BoundingBox": {
                    "Width": 0.5074880123138428,
                    "Top": 0.3774999976158142,
                    "Left": 0.18302799761295319,
                    "Height": 0.3812499940395355
                },
                "FaceId": "086261e8-6deb-4bc0-ac73-ab22323cc38d",
                "ExternalImageId": "image6.jpg",
                "Confidence": 99.99930572509766,
                "ImageId": "ae1593b0-a8f6-5e24-a306-abf529e276fa"
            },
            "Similarity": 96.43287658691406
        },
        {
            "Face": {
                "BoundingBox": {
                    "Width": 0.5574039816856384,
                    "Top": 0.37187498807907104,
                    "Left": 0.14559100568294525,
                    "Height": 0.4181250035762787
                },
                "FaceId": "11c4bd3c-19c5-4eb8-aecc-24feb93a26e1",
                "ExternalImageId": "image5.jpg",
                "Confidence": 99.99960327148438,
                "ImageId": "80739b4d-883f-5b78-97cf-5124038e26b9"
            },
```

```
            "Similarity": 95.25305938720703
        },
        {
            "Face": {
                "BoundingBox": {
                    "Width": 0.5773710012435913,
                    "Top": 0.34437501430511475,
                    "Left": 0.12396000325679779,
                    "Height": 0.4337500035762787
                },
                "FaceId": "57189455-42b0-4839-a86c-abda48b13174",
                "ExternalImageId": "image8.jpg",
                "Confidence": 100.0,
                "ImageId": "0aff2f37-e7a2-5dbc-a3a3-4ef6ec18eaa0"
            },
            "Similarity": 95.22837829589844
        }
    ]
}
```

Per ulteriori informazioni, consulta Ricerca di un volto tramite il relativo ID nella Guida per gli sviluppatori di Amazon Rekognition.

- Per i dettagli sull'API, consulta SearchFaces AWS CLICommand Reference.

`start-celebrity-recognition`

Il seguente esempio di codice mostra come utilizzare`start-celebrity-recognition`.

AWS CLI

Come avviare il riconoscimento di celebrità in un video archiviato

Il comando `start-celebrity-recognition` seguente avvia un processo per cercare celebrità nel file video specificato archiviato in un bucket Amazon S3.

```
aws rekognition start-celebrity-recognition \
    --video "S3Object={Bucket=MyVideoS3Bucket,Name=MyVideoFile.mpg}"
```

Output:

```
{
```

```
    "JobId": "1234567890abcdef1234567890abcdef1234567890abcdef1234567890abcdef"
}
```

Per ulteriori informazioni, consulta Riconoscimento di un volto celebre in un video archiviato nella Guida per gli sviluppatori di Amazon Rekognition.

- Per i dettagli sull'API, consulta StartCelebrityRecognition AWS CLICommand Reference.

`start-content-moderation`

Il seguente esempio di codice mostra come utilizzare`start-content-moderation`.

AWS CLI

Come avviare il riconoscimento di contenuti non sicuri in un video archiviato

Il comando `start-content-moderation` seguente avvia un processo per rilevare contenuti non sicuri nel file video specificato archiviato in un bucket Amazon S3.

```
aws rekognition start-content-moderation \
    --video "S3Object={Bucket=MyVideoS3Bucket,Name=MyVideoFile.mpg}"
```

Output:

```
{
    "JobId": "1234567890abcdef1234567890abcdef1234567890abcdef1234567890abcdef"
}
```

Per ulteriori informazioni, consulta Rilevamento di video archiviati non sicuri nella Guida per gli sviluppatori di Amazon Rekognition.

- Per i dettagli sull'API, consulta StartContentModeration AWS CLICommand Reference.

`start-face-detection`

Il seguente esempio di codice mostra come utilizzare`start-face-detection`.

AWS CLI

Come rilevare i volti in un video

Il comando `start-face-detection` seguente avvia un processo per rilevare volti nel file video specificato archiviato in un bucket Amazon S3.

```
aws rekognition start-face-detection
    --video "S3Object={Bucket=MyVideoS3Bucket,Name=MyVideoFile.mpg}"
```

Output:

```
{
    "JobId": "1234567890abcdef1234567890abcdef1234567890abcdef1234567890abcdef"
}
```

Per ulteriori informazioni, consulta Rilevamento di volti in un video archiviato nella Guida per gli sviluppatori di Amazon Rekognition.

- Per i dettagli sull'API, consulta StartFaceDetection AWS CLICommand Reference.

`start-face-search`

Il seguente esempio di codice mostra come utilizzare`start-face-search`.

AWS CLI

Come cercare in una raccolta i volti corrispondenti ai volti rilevati in un video

Il comando `start-face-search` seguente avvia un processo per rilevare volti in una raccolta corrispondente ai volti individuati nel file video specificato archiviato in un bucket Amazon S3.

```
aws rekognition start-face-search \
    --video "S3Object={Bucket=MyVideoS3Bucket,Name=MyVideoFile.mpg}" \
    --collection-id collection
```

Output:

```
{
    "JobId": "1234567890abcdef1234567890abcdef1234567890abcdef1234567890abcdef"
}
```

Per ulteriori informazioni, consulta Ricerca di volti nei video archiviati nella Guida per gli sviluppatori di Amazon Rekognition.

- Per i dettagli sull'API, consulta StartFaceSearch AWS CLICommand Reference.

`start-label-detection`

Il seguente esempio di codice mostra come utilizzare`start-label-detection`.

AWS CLI

Come rilevare oggetti e scene in un video

Il comando `start-label-detection` seguente avvia un processo per rilevare oggetti e scene nel file video specificato archiviato in un bucket Amazon S3.

```
aws rekognition start-label-detection \
    --video "S3Object={Bucket=MyVideoS3Bucket,Name=MyVideoFile.mpg}"
```

Output:

```
{
    "JobId": "1234567890abcdef1234567890abcdef1234567890abcdef1234567890abcdef"
}
```

Per ulteriori informazioni, consulta Rilevamento di etichette in un video nella Guida per gli sviluppatori di Amazon Rekognition.

- Per i dettagli sull'API, consulta StartLabelDetection AWS CLICommand Reference.

`start-person-tracking`

Il seguente esempio di codice mostra come utilizzare`start-person-tracking`.

AWS CLI

Come avviare il rilevamento dei movimenti delle persone in un video archiviato

Il comando `start-person-tracking` seguente avvia un processo per tracciare i movimenti delle persone nel file video specificato archiviato in un bucket Amazon S3:

```
aws rekognition start-person-tracking \
    --video "S3Object={Bucket=MyVideoS3Bucket,Name=MyVideoFile.mpg}"
```

Output:

```
{
```

```
    "JobId": "1234567890abcdef1234567890abcdef1234567890abcdef1234567890abcdef"
}
```

Per ulteriori informazioni, consulta Rilevamento dei percorsi delle persone nella Guida per gli sviluppatori di Amazon Rekognition.

- Per i dettagli sull'API, consulta StartPersonTracking AWS CLICommand Reference.

`start-stream-processor`

Il seguente esempio di codice mostra come utilizzare`start-stream-processor`.

AWS CLI

Come avviare un processore di flussi

Il comando `start-stream-processor` seguente avvia il processore di flussi video specificato.

```
aws rekognition start-stream-processor \
    --name my-stream-processor
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Utilizzo dei video in streaming nella Guida per gli sviluppatori di Amazon Rekognition.

- Per i dettagli sull'API, consulta StartStreamProcessor AWS CLICommand Reference.

`stop-stream-processor`

Il seguente esempio di codice mostra come utilizzare`stop-stream-processor`.

AWS CLI

Come arrestare un processore di flussi in esecuzione

Il comando `stop-stream-processor` seguente arresta il processore di flussi video in esecuzione specificato.

```
aws rekognition stop-stream-processor \
    --name my-stream-processor
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Utilizzo dei video in streaming nella Guida per gli sviluppatori di Amazon Rekognition.

- Per i dettagli sull'API, consulta StopStreamProcessor AWS CLICommand Reference.

AWS RAM esempi utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with AWS RAM.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`accept-resource-share-invitation`

Il seguente esempio di codice mostra come utilizzare`accept-resource-share-invitation`.

AWS CLI

Come accettare un invito alla condivisione di risorse

L'esempio `accept-resource-share-invitation` seguente accetta l'invito alla condivisione di risorse specificato. I responsabili dell'account invitato possono iniziare immediatamente a utilizzare le risorse della condivisione.

```
aws ram accept-resource-share-invitation \
    --resource-share-invitation-arn arn:aws:ram:us-west-2:111111111111:resource-share-invitation/1e3477be-4a95-46b4-bbe0-c4001EXAMPLE
```

Output:

```
{
    "resourceShareInvitation": {
        "resourceShareInvitationArn": "arn:aws:ram:us-west-2:111111111111:resource-
share-invitation/1e3477be-4a95-46b4-bbe0-c4001EXAMPLE",
        "resourceShareName": "MyLicenseShare",
        "resourceShareArn": "arn:aws:ram:us-west-2:111111111111:resource-
share/27d09b4b-5e12-41d1-a4f2-19dedEXAMPLE",
        "senderAccountId": "111111111111",
        "receiverAccountId": "222222222222",
        "invitationTimestamp": "2021-09-22T15:07:35.620000-07:00",
        "status": "ACCEPTED"
    }
}
```

- Per i dettagli sull'API, consulta AcceptResourceShareInvitation AWS CLICommand Reference.

`associate-resource-share-permission`

Il seguente esempio di codice mostra come utilizzare`associate-resource-share-permission`.

AWS CLI

Come associare un'autorizzazione gestita RAM a una condivisione di risorse

L'esempio `associate-resource-share-permission` seguente sostituisce l'autorizzazione gestita esistente per il tipo di risorsa pertinente con l'autorizzazione gestita specificata. L'accesso a tutte le risorse del tipo di risorsa pertinente è regolato dalla nuova autorizzazione.

```
aws ram associate-resource-share-permission \
    --permission-arn arn:aws:ram::aws:permission/
AWSRAMPermissionGlueDatabaseReadWrite \
    --replace \
    --resource-share-arn arn:aws:ram:us-west-2:123456789012:resource-
share/27d09b4b-5e12-41d1-a4f2-19dedEXAMPLE
```

Output:

```
{
    "returnValue": true
}
```

- Per i dettagli sull'API, consulta AssociateResourceSharePermission AWS CLICommand Reference.

`associate-resource-share`

Il seguente esempio di codice mostra come utilizzare`associate-resource-share`.

AWS CLI

Esempio 1: come associare una risorsa a una condivisione di risorse

L'esempio `associate-resource-share` seguente aggiunge una configurazione di licenza alla condivisione di risorse specificata.

```
aws ram associate-resource-share \
    --resource-share arn:aws:ram:us-west-2:123456789012:resource-
share/27d09b4b-5e12-41d1-a4f2-19dedEXAMPLE \
    --resource-arns arn:aws:license-manager:us-west-2:123456789012:license-
configuration:lic-36be0485f5ae379cc74cf8e92EXAMPLE
```

Output:

```
{
    "resourceShareAssociations": [
        {
           "resourceShareArn": "arn:aws:ram:us-west-2:123456789012:resource-
share/27d09b4b-5e12-41d1-a4f2-19dedEXAMPLE",
           "associatedEntity": "arn:aws:license-manager:us-
west-2:123456789012:license-configuration:lic-36be0485f5ae379cc74cf8e92EXAMPLE",
           "associationType": "RESOURCE",
           "status": "ASSOCIATING",
           "external": false
        }
    ]
}
```

Esempio 2: come associare un principale a una condivisione di risorse

L'esempio `associate-resource-share` seguente concede l'accesso per la condivisione di risorse specificata a tutti gli account dell'unità organizzativa specificata.

```
aws ram associate-resource-share \
```

```
    --resource-share-arn arn:aws:ram:us-west-2:123456789012:resource-
share/27d09b4b-5e12-41d1-a4f2-19dedEXAMPLE \
    --principals arn:aws:organizations::123456789012:ou/o-63bEXAMPLE/ou-46xi-
rEXAMPLE
```

Output:

```
{
    "resourceShareAssociations": [
        {
            "status": "ASSOCIATING",
            "associationType": "PRINCIPAL",
            "associatedEntity": "arn:aws:organizations::123456789012:ou/
o-63bEXAMPLE/ou-46xi-rEXAMPLE",
            "external": false,
            "resourceShareArn": "arn:aws:ram:us-west-2:123456789012:resource-
share/27d09b4b-5e12-41d1-a4f2-19dedEXAMPLE"
        }
    ]
}
```

- Per i dettagli sull'API, consulta AssociateResourceShare AWS CLICommand Reference.

`create-resource-share`

Il seguente esempio di codice mostra come utilizzare`create-resource-share`.

AWS CLI

Esempio 1: come creare una condivisione di risorse

L'esempio `create-resource-share` seguente crea una condivisione di risorse vuota con il nome specificato. È necessario aggiungere separatamente risorse, principali e autorizzazioni alla condivisione.

```
aws ram create-resource-share \
    --name MyNewResourceShare
```

Output:

```
{
    "resourceShare": {
```

```
        "resourceShareArn": "arn:aws:ram:us-west-2:123456789012:resource-
share/4476c27d-8feb-4b21-afe9-7de23EXAMPLE",
        "name": "MyNewResourceShare",
        "owningAccountId": "123456789012",
        "allowExternalPrincipals": true,
        "status": "ACTIVE",
        "creationTime": 1634586271.302,
        "lastUpdatedTime": 1634586271.302
    }
}
```

Esempio 2: creare una condivisione di risorse con AWS gli account come responsabili

L'`create-resource-share`esempio seguente crea una condivisione di risorse e concede l'accesso all' AWS account specificato (222222222222). Se i responsabili specificati non fanno parte della stessa AWS organizzazione, gli inviti vengono inviati e devono essere accettati prima di concedere l'accesso.

```
aws ram create-resource-share \
    --name MyNewResourceShare \
    --principals 222222222222
```

Esempio 3: creare una condivisione di risorse limitata alla propria organizzazione AWS

L'`create-resource-share`esempio seguente crea una condivisione di risorse limitata agli account dell' AWS organizzazione di cui l'account è membro e aggiunge l'unità organizzativa specificata come principale. Tutti gli account di quell'unità organizzativa possono utilizzare le risorse della condivisione di risorse.

```
aws ram create-resource-share \
    --name MyNewResourceShare \
    --no-allow-external-principals \
    --principals arn:aws:organizations::123456789012:ou/o-63bEXAMPLE/ou-46xi-
rEXAMPLE
```

Output:

```
{
    "resourceShare": {
        "resourceShareArn": "arn:aws:ram:us-west-2:123456789012:resource-
share/7be8694e-095c-41ca-9ce8-7be4aEXAMPLE",
        "name": "MyNewResourceShare",
```

```
        "owningAccountId": "123456789012",
        "allowExternalPrincipals": false,
        "status": "ACTIVE",
        "creationTime": 1634587042.49,
        "lastUpdatedTime": 1634587042.49
    }
}
```

- Per i dettagli sull'API, consulta CreateResourceShare AWS CLICommand Reference.

`delete-resource-share`

Il seguente esempio di codice mostra come utilizzare`delete-resource-share`.

AWS CLI

Come eliminare una condivisione di risorse

L'esempio `delete-resource-share` seguente elimina la condivisione di risorse specificata.

```
aws ram delete-resource-share \
    --resource-share-arn arn:aws:ram:us-west-2:123456789012:resource-share/7ab63972-b505-7e2a-420d-6f5d3EXAMPLE
```

Il seguente output indica che l'operazione è riuscita:

```
{
    "returnValue": true
}
```

- Per i dettagli sull'API, consulta DeleteResourceShare AWS CLICommand Reference.

`disassociate-resource-share-permission`

Il seguente esempio di codice mostra come utilizzare`disassociate-resource-share-permission`.

AWS CLI

Come rimuovere un'autorizzazione gestita dalla RAM per un tipo di risorsa da una condivisione di risorse

L'esempio `disassociate-resource-share-permission` seguente rimuove l'autorizzazione gestita da RAM per i database Glue dalla condivisione di risorse specificata.

```
aws ram disassociate-resource-share-permission \
    --resource-share-arn arn:aws:ram:us-west-2:123456789012:resource-share/27d09b4b-5e12-41d1-a4f2-19dedEXAMPLE \
    --permission-arn arn:aws:ram::aws:permission/AWSRAMPermissionGlueDatabaseReadWrite
```

Output:

```
{
    "returnValue": true
}
```

- Per i dettagli sull'API, consulta DisassociateResourceSharePermission AWS CLICommand Reference.

`disassociate-resource-share`

Il seguente esempio di codice mostra come utilizzare`disassociate-resource-share`.

AWS CLI

Come rimuovere una risorsa da una condivisione di risorse

L'esempio `disassociate-resource-share` seguente rimuove la risorsa specificata, in questo caso una sottorete VPC, dalla condivisione di risorse specificata. I principali con accesso alla condivisione di risorse non possono più eseguire operazioni su quella risorsa.

```
aws ram disassociate-resource-share \
    --resource-arns arn:aws:ec2:us-west-2:123456789012:subnet/subnet-0250c25a1fEXAMPLE \
    --resource-share-arn arn:aws:ram:us-west-2:123456789012:resource-share/7ab63972-b505-7e2a-420d-6f5d3EXAMPLE
```

Output:

```
{
```

```
    "resourceShareAssociations": [
        "resourceShareArn": "arn:aws:ram:us-west-2:123456789012:resource-
share/7ab63972-b505-7e2a-420d-6f5d3EXAMPLE",
        "associatedEntity": "arn:aws:ec2:us-west-2:123456789012:subnet/
subnet-0250c25a1fEXAMPLE",
        "associationType": "RESOURCE",
        "status": "DISASSOCIATING",
        "external": false
    ]
}
```

- Per i dettagli sull'API, consulta DisassociateResourceShare AWS CLICommand Reference.

`enable-sharing-with-aws-organization`

Il seguente esempio di codice mostra come utilizzare`enable-sharing-with-aws-organization`.

AWS CLI

Per abilitare la condivisione delle risorse tra AWS Organizzazioni

L'esempio `enable-sharing-with-aws-organization` seguente abilita la condivisione di risorse tra l'organizzazione e le unità organizzative.

```
aws ram enable-sharing-with-aws-organization
```

Il seguente output indica il successo.

```
{
    "returnValue": true
}
```

- Per i dettagli sull'API, consulta EnableSharingWithAwsOrganization AWS CLICommand Reference.

`get-permission`

Il seguente esempio di codice mostra come utilizzare`get-permission`.

AWS CLI

Come recuperare i dettagli di un'autorizzazione gestita dalla RAM

L'esempio `get-permission` seguente mostra i dettagli della versione predefinita dell'autorizzazione gestita da RAM specificata.

```
aws ram get-permission \
    --permission-arn arn:aws:ram::aws:permission/
AWSRAMPermissionGlueTableReadWriteForDatabase
```

Output:

```
{
    "permission": {
        "arn": "arn:aws:ram::aws:permission/
AWSRAMPermissionGlueTableReadWriteForDatabase",
        "version": "2",
        "defaultVersion": true,
        "name": "AWSRAMPermissionGlueTableReadWriteForDatabase",
        "resourceType": "glue:Database",
        "permission": "{\"Effect\":\"Allow\",\"Action\":[\"glue:GetTable
\", \"glue:UpdateTable\", \"glue:DeleteTable\", \"glue:BatchDeleteTable\",
 \"glue:BatchDeleteTableVersion\", \"glue:GetTableVersion\", \"glue:GetTableVersions
\", \"glue:GetPartition\", \"glue:GetPartitions\", \"glue:BatchGetPartition\",
 \"glue:BatchCreatePartition\", \"glue:CreatePartition\", \"glue:UpdatePartition
\", \"glue:BatchDeletePartition\", \"glue:DeletePartition\", \"glue:GetTables\",
 \"glue:SearchTables\"]}",
        "creationTime": 1624912434.431,
        "lastUpdatedTime": 1624912434.431,
        "isResourceTypeDefault": false
    }
}
```

- Per i dettagli sull'API, consulta GetPermission AWS CLICommand Reference.

`get-resource-policies`

Il seguente esempio di codice mostra come utilizzare`get-resource-policies`.

AWS CLI

Come ottenere le policy per una risorsa

L'esempio `get-resource-policies` seguente visualizza le policy di autorizzazione basate sulle risorse per la risorsa specificata associata a una condivisione di risorse.

```
aws ram get-resource-policies \
    --resource-arns arn:aws:ec2:us-west-2:123456789012:subnet/
subnet-0250c25a1fEXAMPLE
```

Output:

```
{
    "policies": [
        "{\"Version\":\"2008-10-17\",\"Statement\":[{\"Sid\":\"RamStatement1\",
\"Effect\":\"Allow\",\"Principal\":{\"AWS\":[]},\"Action\":[\"ec2:RunInstances
\",\"ec2:CreateNetworkInterface\",\"ec2:DescribeSubnets\"],\"Resource\":
\"arn:aws:ec2:us-west-2:123456789012:subnet/subnet-0250c25a1fEXAMPLE\"}]}"
    ]
}
```

- Per i dettagli sull'API, consulta GetResourcePolicies AWS CLICommand Reference.

`get-resource-share-associations`

Il seguente esempio di codice mostra come utilizzare`get-resource-share-associations`.

AWS CLI

Esempio 1: come elencare tutte le associazioni delle risorse per tutti i tipi di risorse

L'esempio `get-resource-share-associations` seguente elenca le associazioni di risorse per tutti i tipi di risorse in tutte le condivisioni di risorse.

```
aws ram get-resource-share-associations \
    --association-type RESOURCE
```

Output:

```
{
```

```
    "resourceShareAssociations": [
        {
            "resourceShareArn": "arn:aws:ram:us-west-2:123456789012:resource-
share/7ab63972-b505-7e2a-420d-6f5d3EXAMPLE",
            "associatedEntity": "arn:aws:ec2:us-west-2:123456789012:subnet/
subnet-0250c25a1fEXAMPLE",
            "resourceShareName": "MySubnetShare",
            "associationType": "RESOURCE",
            "status": "ASSOCIATED",
            "creationTime": 1565303590.973,
            "lastUpdatedTime": 1565303591.695,
            "external": false
        },
        {
            "resourceShareArn": "arn:aws:ram:us-west-2:123456789012:resource-
share/8167bdfe-4480-4a01-8632-315e0EXAMPLE",
            "associatedEntity": "arn:aws:license-manager:us-
west-2:123456789012:license-configuration:lic-36be0485f5ae379cc74cf8e92EXAMPLE",
            "resourceShareName": "MyLicenseShare",
            "associationType": "RESOURCE",
            "status": "ASSOCIATED",
            "creationTime": 1632342958.457,
            "lastUpdatedTime": 1632342958.907,
            "external": false
        }
    ]
}
```

Esempio 2: come elencare le associazioni di principali per una condivisione di risorse

L'esempio `get-resource-share-associations` seguente elenca solo le associazioni principali solo per la condivisione di risorse specificata.

```
aws ram get-resource-share-associations \
   --resource-share-arns arn:aws:ram:us-west-2:123456789012:resource-
share/7be8694e-095c-41ca-9ce8-7be4aEXAMPLE \
   --association-type PRINCIPAL
```

Output:

```
{
    "resourceShareAssociations": [
        {
```

```
            "resourceShareArn": "arn:aws:ram:us-west-2:123456789012:resource-
share/7be8694e-095c-41ca-9ce8-7be4aEXAMPLE",
            "resourceShareName": "MyNewResourceShare",
            "associatedEntity": "arn:aws:organizations::123456789012:ou/
o-63bEXAMPLE/ou-46xi-rEXAMPLE",
            "associationType": "PRINCIPAL",
            "status": "ASSOCIATED",
            "creationTime": 1634587042.49,
            "lastUpdatedTime": 1634587044.291,
            "external": false
        }
    ]
}
```

- Per i dettagli sull'API, consulta GetResourceShareAssociations AWS CLICommand Reference.

`get-resource-share-invitations`

Il seguente esempio di codice mostra come utilizzare`get-resource-share-invitations`.

AWS CLI

Come elencare gli inviti alla condivisione delle risorse

L'esempio `get-resource-share-invitations` seguente elenca gli inviti correnti alla condivisione di risorse.

```
aws ram get-resource-share-invitations
```

Output:

```
{
    "resourceShareInvitations": [
        {
            "resourceShareInvitationArn": "arn:aws:ram:us-
west2-1:111111111111:resource-share-invitation/32b639f0-14b8-7e8f-55ea-
e6117EXAMPLE",
            "resourceShareName": "project-resource-share",
            "resourceShareArn": "arn:aws:ram:us-west-2:111111111111:resource-share/
fcb639f0-1449-4744-35bc-a983fEXAMPLE",
            "senderAccountId": "111111111111",
            "receiverAccountId": "222222222222",
```

```
            "invitationTimestamp": 1565312166.258,
            "status": "PENDING"
        }
    ]
}
```

- Per i dettagli sull'API, consulta GetResourceShareInvitations AWS CLICommand Reference.

`get-resource-shares`

Il seguente esempio di codice mostra come utilizzare`get-resource-shares`.

AWS CLI

Esempio 1: come elencare le risorse condivise che possiedi e che condividi con altri

L'esempio `get-resource-shares` seguente elenca le condivisioni di risorse create e condivise con altri.

```
aws ram get-resource-shares \
    --resource-owner SELF
```

Output:

```
{
    "resourceShares": [
        {
            "resourceShareArn": "arn:aws:ram:us-west-2:123456789012:resource-
share/3ab63985-99d9-1cd2-7d24-75e93EXAMPLE",
            "name": "my-resource-share",
            "owningAccountId": "123456789012",
            "allowExternalPrincipals": false,
            "status": "ACTIVE",
            "tags": [
                {
                    "key": "project",
                    "value": "lima"
                }
            ]
            "creationTime": 1565295733.282,
            "lastUpdatedTime": 1565295733.282
        },
```

```
        {
            "resourceShareArn": "arn:aws:ram:us-west-2:123456789012:resource-
share/7ab63972-b505-7e2a-420d-6f5d3EXAMPLE",
            "name": "my-resource-share",
            "owningAccountId": "123456789012",
            "allowExternalPrincipals": true,
            "status": "ACTIVE",
            "creationTime": 1565295733.282,
            "lastUpdatedTime": 1565295733.282
        }
    ]
}
```

Esempio 2: come elencare le condivisioni di risorse possedute da altri e condivise con te

L'esempio `get-resource-shares` seguente elenca le condivisioni di risorse create e condivise da altri. In questo esempio, non ce ne sono.

```
aws ram get-resource-shares \
    --resource-owner OTHER-ACCOUNTS
```

Output:

```
{
    "resourceShares": []
}
```

- Per i dettagli sull'API, consulta GetResourceShares AWS CLICommand Reference.

`list-pending-invitation-resources`

Il seguente esempio di codice mostra come utilizzare`list-pending-invitation-resources`.

AWS CLI

Come elencare le risorse disponibili in una condivisione di risorse in sospeso

L'esempio `list-pending-invitation-resources` seguente elenca tutte le risorse presenti nella condivisione di risorse associata all'invito specificato.

```
aws ram list-pending-invitation-resources \
```

```
    --resource-share-invitation-arn arn:aws:ram:us-west-2:123456789012:resource-
share-invitation/1e3477be-4a95-46b4-bbe0-c4001EXAMPLE
```

Output:

```
{
   "resources": [
        {
            "arn": "arn:aws:ec2:us-west-2:123456789012:subnet/
subnet-04a555b0e6EXAMPLE",
            "resourceShareArn": "arn:aws:ram:us-west-2:123456789012:resource-
share/7be8694e-095c-41ca-9ce8-7be4aEXAMPLE",
            "creationTime": 1634676051.269,
            "lastUpdatedTime": 1634676052.07,
            "status": "AVAILABLE",
            "type": "ec2:Subnet"
        },
        {
            "arn": "arn:aws:license-manager:us-west-2:123456789012:license-
configuration:lic-36be0485f5ae379cc74cf8e92EXAMPLE",
            "resourceShareArn": "arn:aws:ram:us-west-2:123456789012:resource-
share/7ab63972-b505-7e2a-420d-6f5d3EXAMPLE",
            "creationTime": 1624912434.431,
            "lastUpdatedTime": 1624912434.431,
            "status": "AVAILABLE",
            "type": "license-manager:LicenseConfiguration"
      }
   ]
}
```

- Per i dettagli sull'API, consulta ListPendingInvitationResources AWS CLICommand Reference.

`list-permissions`

Il seguente esempio di codice mostra come utilizzare`list-permissions`.

AWS CLI

Come elencare le autorizzazioni gestite da RAM disponibili

L'`list-permissions`esempio seguente elenca tutte le autorizzazioni gestite dalla RAM disponibili solo per il tipo di risorsa del database AWS Glue.

```
aws ram list-permissions \
    --resource-type glue:Database
```

Output:

```
{
    "permissions": [
        {
            "arn": "arn:aws:ram::aws:permission/
AWSRAMDefaultPermissionGlueDatabase",
            "version": "1",
            "defaultVersion": true,
            "name": "AWSRAMDefaultPermissionGlueDatabase",
            "resourceType": "glue:Database",
            "creationTime": 1592007820.935,
            "lastUpdatedTime": 1592007820.935,
            "isResourceTypeDefault": true
        },
        {
            "arn": "arn:aws:ram::aws:permission/
AWSRAMPermissionGlueAllTablesReadWriteForDatabase",
            "version": "2",
            "defaultVersion": true,
            "name": "AWSRAMPermissionGlueAllTablesReadWriteForDatabase",
            "resourceType": "glue:Database",
            "creationTime": 1624912413.323,
            "lastUpdatedTime": 1624912413.323,
            "isResourceTypeDefault": false
        },
        {
            "arn": "arn:aws:ram::aws:permission/
AWSRAMPermissionGlueDatabaseReadWrite",
            "version": "2",
            "defaultVersion": true,
            "name": "AWSRAMPermissionGlueDatabaseReadWrite",
            "resourceType": "glue:Database",
            "creationTime": 1624912417.4,
            "lastUpdatedTime": 1624912417.4,
            "isResourceTypeDefault": false
        },
        {
            "arn": "arn:aws:ram::aws:permission/
AWSRAMPermissionGlueTableReadWriteForDatabase",
```

```
            "version": "2",
            "defaultVersion": true,
            "name": "AWSRAMPermissionGlueTableReadWriteForDatabase",
            "resourceType": "glue:Database",
            "creationTime": 1624912434.431,
            "lastUpdatedTime": 1624912434.431,
            "isResourceTypeDefault": false
        }
    ]
}
```

L'esempio `list-permissions` seguente visualizza le autorizzazioni gestite da RAM disponibili per tutti i tipi di risorsa.

```
aws ram list-permissions
```

Output:

```
{
    "permissions": [
        {
            "arn": "arn:aws:ram::aws:permission/
AWSRAMBlankEndEntityCertificateAPICSRPassthroughIssuanceCertificateAuthority",
            "version": "1",
            "defaultVersion": true,
            "name":
 "AWSRAMBlankEndEntityCertificateAPICSRPassthroughIssuanceCertificateAuthority",
            "resourceType": "acm-pca:CertificateAuthority",
            "creationTime": 1623264861.085,
            "lastUpdatedTime": 1623264861.085,
            "isResourceTypeDefault": false
        },
        {
            "arn": "arn:aws:ram::aws:permission/AWSRAMDefaultPermissionAppMesh",
            "version": "1",
            "defaultVersion": true,
            "name": "AWSRAMDefaultPermissionAppMesh",
            "resourceType": "appmesh:Mesh",
            "creationTime": 1589307188.584,
            "lastUpdatedTime": 1589307188.584,
            "isResourceTypeDefault": true
        },
        ...TRUNCATED FOR BREVITY...
```

```
        {
            "arn": "arn:aws:ram::aws:permission/
AWSRAMSubordinateCACertificatePathLen0IssuanceCertificateAuthority",
            "version": "1",
            "defaultVersion": true,
            "name":
 "AWSRAMSubordinateCACertificatePathLen0IssuanceCertificateAuthority",
            "resourceType": "acm-pca:CertificateAuthority",
            "creationTime": 1623264876.75,
            "lastUpdatedTime": 1623264876.75,
            "isResourceTypeDefault": false
        }
    ]
}
```

- Per i dettagli sull'API, consulta ListPermissions AWS CLICommand Reference.

`list-principals`

Il seguente esempio di codice mostra come utilizzare`list-principals`.

AWS CLI

Come elencare i principali con accesso a una risorsa

L'esempio `list-principals` seguente visualizza un elenco di principali che possono accedere alle risorse del tipo specificato mediante qualsiasi condivisione di risorse.

```
aws ram list-principals \
    --resource-type ec2:Subnet
```

Output:

```
{
    "principals": [
        {
            "id": "arn:aws:organizations::123456789012:ou/o-gx7EXAMPLE/ou-29c5-
zEXAMPLE",
            "resourceShareArn": "arn:aws:ram:us-west-2:123456789012:resource-
share/7ab63972-b505-7e2a-420d-6f5d3EXAMPLE",
            "creationTime": 1565298209.737,
            "lastUpdatedTime": 1565298211.019,
```

```
            "external": false
        }
    ]
}
```

- Per i dettagli sull'API, consulta ListPrincipals AWS CLICommand Reference.

`list-resource-share-permissions`

Il seguente esempio di codice mostra come utilizzare`list-resource-share-permissions`.

AWS CLI

Come elencare tutte le autorizzazioni gestite da RAM attualmente associate a una condivisione di risorse

L'esempio `list-resource-share-permissions` seguente elenca tutte le autorizzazioni gestite da RAM associate alla condivisione di risorse specificata.

```
aws ram list-resource-share-permissions \
    --resource-share-arn arn:aws:ram:us-west-2:123456789012:resource-
share/27d09b4b-5e12-41d1-a4f2-19dedEXAMPLE
```

Output:

```
{
    "permissions": [
        {
            "arn": "arn:aws:ram::aws:permission/
AWSRAMDefaultPermissionLicenseConfiguration",
            "version": "1",
            "resourceType": "license-manager:LicenseConfiguration",
            "status": "ASSOCIATED",
            "lastUpdatedTime": 1632342984.234
        },
        {
            "arn": "arn:aws:ram::aws:permission/
AWSRAMPermissionGlueDatabaseReadWrite",
            "version": "2",
            "resourceType": "glue:Database",
            "status": "ASSOCIATED",
            "lastUpdatedTime": 1632512462.297
```

```
        }
    ]
}
```

- Per i dettagli sull'API, consulta ListResourceSharePermissions AWS CLICommand Reference.

`list-resource-types`

Il seguente esempio di codice mostra come utilizzare`list-resource-types`.

AWS CLI

Per elencare i tipi di risorse supportati dalla AWS RAM

L'`list-resource-types`esempio seguente elenca tutti i tipi di risorse attualmente supportati dalla AWS RAM.

```
aws ram list-resource-types
```

Output:

```
{
    "resourceTypes": [
        {
            "resourceType": "route53resolver:FirewallRuleGroup",
            "serviceName": "route53resolver"
        },
        {
            "resourceType": "ec2:LocalGatewayRouteTable",
            "serviceName": "ec2"
        },
        ...OUTPUT TRUNCATED FOR BREVITY...
        {
            "resourceType": "ec2:Subnet",
            "serviceName": "ec2"
        },
        {
            "resourceType": "ec2:TransitGatewayMulticastDomain",
            "serviceName": "ec2"
        }
    ]
}
```

- Per i dettagli sull'API, consulta ListResourceTypes AWS CLICommand Reference.

`list-resources`

Il seguente esempio di codice mostra come utilizzare`list-resources`.

AWS CLI

Come elencare le risorse associate a una condivisione di risorse

L'esempio `list-resources` seguente elenca tutte le risorse della condivisione di risorse specificata che sono del tipo specificato.

```
aws ram list-resources \
    --resource-type ec2:Subnet \
    --resource-owner SELF \
    --resource-share-arn arn:aws:ram:us-west-2:123456789012:resource-share/7ab63972-b505-7e2a-420d-6f5d3EXAMPLE
```

Output:

```
{
    "resources": [
        {
            "arn": "aarn:aws:ec2:us-west-2:123456789012:subnet/subnet-0250c25a1f4e15235",
            "type": "ec2:Subnet",
            "resourceShareArn": "arn:aws:ram:us-west-2:123456789012:resource-share/7ab63972-b505-7e2a-420d-6f5d3EXAMPLE",
            "creationTime": 1565301545.023,
            "lastUpdatedTime": 1565301545.947
        }
    ]
}
```

- Per i dettagli sull'API, consulta ListResources AWS CLICommand Reference.

`promote-resource-share-created-from-policy`

Il seguente esempio di codice mostra come utilizzare`promote-resource-share-created-from-policy`.

AWS CLI

Promuovere una condivisione delle risorse basata sulla politica delle risorse alla piena funzionalità della RAM AWS

L'`promote-resource-share-created-from-policy`esempio seguente prende una condivisione di risorse creata implicitamente allegando una policy basata sulle risorse e la converte per renderla completamente funzionale con la AWS console RAM e le relative operazioni CLI e API.

```
aws ram promote-resource-share-created-from-policy \
    --resource-share-arn arn:aws:ram:us-east-1:123456789012:resource-share/91fa8429-2d06-4032-909a-90909EXAMPLE
```

Output:

```
{
    "returnValue": true
}
```

- Per i dettagli sull'API, consulta Command Reference. PromoteResourceShareCreatedFromPolicyAWS CLI

`reject-resource-share-invitation`

Il seguente esempio di codice mostra come utilizzare`reject-resource-share-invitation`.

AWS CLI

Come rifiutare un invito alla condivisione di risorse

L'esempio `reject-resource-share-invitation` seguente rifiuta l'invito alla condivisione di risorse specificato.

```
aws ram reject-resource-share-invitation \
    --resource-share-invitation-arn arn:aws:ram:us-west-2:111111111111:resource-share-invitation/32b639f0-14b8-7e8f-55ea-e6117EXAMPLE
```

Output:

```
"resourceShareInvitations": [
```

```
    {
        "resourceShareInvitationArn": "arn:aws:ram:us-west2-1:111111111111:resource-
share-invitation/32b639f0-14b8-7e8f-55ea-e6117EXAMPLE",
        "resourceShareName": "project-resource-share",
        "resourceShareArn": "arn:aws:ram:us-west-2:111111111111:resource-share/
fcb639f0-1449-4744-35bc-a983fEXAMPLE",
        "senderAccountId": "111111111111",
        "receiverAccountId": "222222222222",
        "invitationTimestamp": 1565319592.463,
        "status": "REJECTED"
    }
]
```

- Per i dettagli sull'API, consulta RejectResourceShareInvitation AWS CLICommand Reference.

`tag-resource`

Il seguente esempio di codice mostra come utilizzare`tag-resource`.

AWS CLI

Come aggiungere tag a una condivisione di risorse

L'esempio `tag-resource` seguente aggiunge la chiave di tag `project` e il valore associato `lima` alla condivisione di risorse specificata.

```
aws ram tag-resource \
    --tags key=project,value=lima \
    --resource-share-arn arn:aws:ram:us-west-2:123456789012:resource-share/7ab63972-
b505-7e2a-420d-6f5d3EXAMPLE
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta TagResource AWS CLICommand Reference.

`untag-resource`

Il seguente esempio di codice mostra come utilizzare`untag-resource`.

AWS CLI

Come rimuovere i tag da una condivisione di risorsa

L'esempio `untag-resource` seguente rimuove la chiave del tag `project` e il valore associato dalla condivisione di risorse specificata.

```
aws ram untag-resource \
    --tag-keys project \
    --resource-share-arn arn:aws:ram:us-west-2:123456789012:resource-share/7ab63972-b505-7e2a-420d-6f5d3EXAMPLE
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta UntagResource AWS CLICommand Reference.

`update-resource-share`

Il seguente esempio di codice mostra come utilizzare`update-resource-share`.

AWS CLI

Come aggiornare una condivisione di risorse

L'`update-resource-share`esempio seguente modifica la condivisione di risorse specificata per consentire i principali esterni che non fanno parte di un' AWS organizzazione.

```
aws ram update-resource-share \
    --allow-external-principals \
    --resource-share-arn arn:aws:ram:us-west-2:123456789012:resource-share/7ab63972-b505-7e2a-420d-6f5d3EXAMPLE
```

Output:

```
{
    "resourceShare": {
        "resourceShareArn": "arn:aws:ram:us-west-2:123456789012:resource-share/7ab63972-b505-7e2a-420d-6f5d3EXAMPLE",
        "name": "my-resource-share",
        "owningAccountId": "123456789012",
        "allowExternalPrincipals": true,
        "status": "ACTIVE",
        "creationTime": 1565295733.282,
        "lastUpdatedTime": 1565303080.023
    }
}
```

- Per i dettagli sull'API, vedere UpdateResourceSharein AWS CLI Command Reference.

Esempi di Resource Explorer utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with Resource Explorer.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`associate-default-view`

Il seguente esempio di codice mostra come utilizzare`associate-default-view`.

AWS CLI

Per impostare una visualizzazione di Resource Explorer come predefinita per la relativa AWS regione

L'`associate-default-view`esempio seguente imposta una vista, come specificato dal relativo ARN, come vista predefinita per la AWS regione in cui si chiama l'operazione.

```
aws resource-explorer-2 associate-default-view \
    --view-arn arn:aws:resource-explorer-2:us-east-1:123456789012:view/My-Main-View/
EXAMPLE8-90ab-cdef-fedc-EXAMPLE11111
```

Output:

```
{
    "ViewArn": "arn:aws:resource-explorer-2:us-east-1:123456789012:view/My-Main-
View/EXAMPLE8-90ab-cdef-fedc-EXAMPLE11111"
```

```
}
```

Per ulteriori informazioni, vedere Impostazione di una visualizzazione predefinita in una AWS regione nella Guida per l'utente di AWS Resource Explorer.

- Per i dettagli sull'API, consulta AssociateDefaultView AWS CLICommand Reference.

`batch-get-view`

Il seguente esempio di codice mostra come utilizzare`batch-get-view`.

AWS CLI

Come recuperare i dettagli su più viste di Esploratore di risorse

L'`batch-get-view`esempio seguente mostra i dettagli relativi a due viste specificate da loro ARNs. Utilizzate gli spazi per separare i multipli ARNs nel parametro --view-arn.

```
aws resource-explorer-2 batch-get-view \
    --view-arns arn:aws:resource-explorer-2:us-east-1:123456789012:view/My-EC2-Only-View/EXAMPLE8-90ab-cdef-fedc-EXAMPLE22222, \
                arn:aws:resource-explorer-2:us-east-1:123456789012:view/My-Main-View/EXAMPLE8-90ab-cdef-fedc-EXAMPLE11111
```

Output:

```
{
    "Views": [
        {
            "Filters": {
                "FilterString": "service:ec2"
            },
            "IncludedProperties": [
                {
                    "Name": "tags"
                }
            ],
            "LastUpdatedAt": "2022-07-13T21:33:45.249000+00:00",
            "Owner": "123456789012",
            "Scope": "arn:aws:iam::123456789012:root",
            "ViewArn": "arn:aws:resource-explorer-2:us-east-1:123456789012:view/My-EC2-Only-View/EXAMPLE8-90ab-cdef-fedc-EXAMPLE22222"
        },
```

```
        {
            "Filters": {
                "FilterString": ""
            },
            "IncludedProperties": [
                {
                    "Name": "tags"
                }
            ],
            "LastUpdatedAt": "2022-07-13T20:34:11.314000+00:00",
            "Owner": "123456789012",
            "Scope": "arn:aws:iam::123456789012:root",
            "ViewArn": "arn:aws:resource-explorer-2:us-east-1:123456789012:view/My-
Main-View/EXAMPLE8-90ab-cdef-fedc-EXAMPLE11111"
        }
    ]
    "Errors": []
}
```

Per ulteriori informazioni sulle visualizzazioni, consulta Informazioni sulle viste di Esploratore di risorse nella Guida per l'utente di Esploratore di risorse AWS Resource Explorer.

- Per i dettagli sull'API, consulta BatchGetViewCommand Reference.AWS CLI

`create-index`

Il seguente esempio di codice mostra come utilizzare`create-index`.

AWS CLI

Per attivare Resource Explorer in una AWS regione creando un indice

L'`create-index`esempio seguente crea un indice locale nella AWS regione in cui viene chiamata l'operazione. La AWS CLI genera automaticamente un valore di `client-token` parametro casuale e lo include nella chiamata a AWS se non si specifica un valore.

```
aws resource-explorer-2 create-index \
    --region us-east-1
```

Output:

```
{
```

```
    "Arn": "arn:aws:resource-explorer-2:us-east-1:123456789012:index/EXAMPLE8-90ab-
cdef-fedc-EXAMPLE22222c",
    "CreatedAt": "2022-11-01T20:00:59.149Z",
    "State": "CREATING"
}
```

Dopo aver creato un indice locale, puoi convertirlo nell'indice di aggregazione per l'account eseguendo il update-index-typecomando.

Per ulteriori informazioni, consulta Attivazione di Resource Explorer in una AWS regione per indicizzare le risorse nella Guida per l'utente di AWS Resource Explorer.

- Per i dettagli sull'API, consulta CreateIndex AWS CLICommand Reference.

`create-view`

Il seguente esempio di codice mostra come utilizzare`create-view`.

AWS CLI

Esempio 1: creare una vista non filtrata per l'indice in una regione AWS

L'`create-view`esempio seguente crea una vista nella AWS regione specificata che restituisce tutti i risultati nella regione senza alcun filtro. La visualizzazione include il campo opzionale Tags sui risultati restituiti. Poiché questa visualizzazione viene creata nella Regione che contiene l'indice aggregatore, può includere i risultati di tutte le Regioni dell'account che contengono un indice dell'Esploratore di risorse.

```
aws resource-explorer-2 create-view \
    --view-name My-Main-View \
    --included-properties Name=tags \
    --region us-east-1
```

Output:

```
{
    "View": {
        "Filters": {
            "FilterString": ""
        },
        "IncludedProperties": [
            {
```

```
                "Name": "tags"
            }
        ],
        "LastUpdatedAt": "2022-07-13T20:34:11.314000+00:00",
        "Owner": "123456789012",
        "Scope": "arn:aws:iam::123456789012:root",
        "ViewArn": "arn:aws:resource-explorer-2:us-east-1:123456789012:view/My-Main-
View/EXAMPLE8-90ab-cdef-fedc-EXAMPLE11111"
    }
}
```

Esempio 2: creare una visualizzazione che restituisca solo le risorse associate ad Amazon EC2

Quanto segue `create-view` crea una visualizzazione nella AWS regione `us-east-1` che restituisce solo le risorse della regione associate al EC2 servizio Amazon. La visualizzazione include il campo opzionale `Tags` sui risultati restituiti. Poiché questa visualizzazione viene creata nella Regione che contiene l'indice aggregatore, può includere i risultati di tutte le Regioni dell'account che contengono un indice dell'Esploratore di risorse.

```
aws resource-explorer-2 create-view \
    --view-name My-EC2-Only-View \
    --included-properties Name=tags \
    --filters FilterString="service:ec2" \
    --region us-east-1
```

Output:

```
{
    "View": {
        "Filters": {
            "FilterString": "service:ec2"
        },
        "IncludedProperties": [
            {
                "Name":"tags"
            }
        ],
        "LastUpdatedAt": "2022-07-13T21:35:09.059Z",
        "Owner": "123456789012",
        "Scope": "arn:aws:iam::123456789012:root",
        "ViewArn": "arn:aws:resource-explorer-2:us-east-1:123456789012:view/My-EC2-
Only-View/EXAMPLE8-90ab-cdef-fedc-EXAMPLE22222"
```

```
    }
}
```

Per ulteriori informazioni, consulta Creazione di viste per la ricerca nella Guida per l'utente di Esploratore di risorse AWS .

- Per i dettagli sull'API, consulta CreateView AWS CLICommand Reference.

`delete-index`

Il seguente esempio di codice mostra come utilizzare`delete-index`.

AWS CLI

Per disattivare Resource Explorer in una AWS regione eliminandone l'indice

L'`delete-index`esempio seguente elimina l'indice Resource Explorer specificato nella AWS regione in cui viene effettuata la richiesta.

```
aws resource-explorer-2 delete-index \
    --arn arn:aws:resource-explorer-2:us-west-2:123456789012:index/EXAMPLE8-90ab-cdef-fedc-EXAMPLE22222 \
    --region us-west-2
```

Output:

```
{
    "Arn": "arn:aws:resource-explorer-2:us-west-2:123456789012:index/EXAMPLE8-90ab-cdef-fedc-EXAMPLE22222",
    "State": "DELETING"
}
```

Per ulteriori informazioni sull'eliminazione di un indice, vedere Disattivazione di AWS Resource Explorer in una AWS regione nella Guida per l'utente di AWS Resource Explorer.

- Per i dettagli sull'API, consulta DeleteIndex AWS CLICommand Reference.

`delete-view`

Il seguente esempio di codice mostra come utilizzare`delete-view`.

AWS CLI

Come eliminare una visualizzazione di Esploratore di risorse

L'esempio `delete-view` seguente elimina una visualizzazione specificata dal relativo ARN.

```
aws resource-explorer-2 delete-view \
    --view-arn arn:aws:resource-explorer-2:us-east-1:123456789012:view/EC2-Only-View/EXAMPLE8-90ab-cdef-fedc-EXAMPLE11111
```

Output:

```
{
    "ViewArn": "arn:aws:resource-explorer-2:us-east-1:123456789012:view/EC2-Only-View/EXAMPLE8-90ab-cdef-fedc-EXAMPLE11111"
}
```

Per ulteriori informazioni, consulta Eliminazione delle viste nella Guida per l'utente di Esploratore di risorse AWS .

- Per i dettagli sull'API, consulta DeleteView AWS CLICommand Reference.

`disassociate-default-view`

Il seguente esempio di codice mostra come utilizzare`disassociate-default-view`.

AWS CLI

Per rimuovere la visualizzazione predefinita di Resource Explorer per una AWS regione

Di seguito viene `disassociate-default-view` rimossa la visualizzazione predefinita di Resource Explorer per la AWS regione in cui viene richiamata l'operazione. Dopo aver eseguito questa operazione, tutte le operazioni di ricerca nella Regione devono specificare in modo esplicito una visualizzazione o l'operazione avrà esito negativo.

```
aws resource-explorer-2 disassociate-default-view
```

Questo comando non produce alcun output.

Per ulteriori informazioni, vedere Impostazione di una visualizzazione predefinita in una AWS regione nella Guida per l'utente di AWS Resource Explorer.

- Per i dettagli sull'API, consulta DisassociateDefaultView AWS CLICommand Reference.

`get-default-view`

Il seguente esempio di codice mostra come utilizzare`get-default-view`.

AWS CLI

Per recuperare la vista Resource Explorer, che è la vista predefinita per la relativa regione AWS

L'`get-default-view`esempio seguente recupera l'ARN della vista che è l'impostazione predefinita per AWS la regione in cui si chiama l'operazione.

```
aws resource-explorer-2 get-default-view
```

Output:

```
{
    "ViewArn": "arn:aws:resource-explorer-2:us-east-1:123456789012:view/default-
view/EXAMPLE8-90ab-cdef-fedc-EXAMPLE11111"
}
```

Per ulteriori informazioni, vedere Impostazione di una visualizzazione predefinita in una AWS regione nella Guida per l'utente di AWS Resource Explorer.

- Per i dettagli sull'API, consulta GetDefaultView AWS CLICommand Reference.

`get-index`

Il seguente esempio di codice mostra come utilizzare`get-index`.

AWS CLI

Esempio 1: come recuperare i dettagli per un indice aggregatore di Esploratore di risorse

L'`get-index`esempio seguente visualizza i dettagli dell'indice Resource Explorer AWS nella regione specificata. Poiché la Regione specificata contiene l'indice aggregatore per l'account, l'output elenca le Regioni che replicano i dati nell'indice di questa Regione.

```
aws resource-explorer-2 get-index \
    --region us-east-1
```

Output:

```
{
    "Arn": "arn:aws:resource-explorer-2:us-east-1:123456789012:index/EXAMPLE8-90ab-
cdef-fedc-EXAMPLE11111",
    "CreatedAt": "2022-07-12T18:59:10.503000+00:00",
    "LastUpdatedAt": "2022-07-13T18:41:58.799000+00:00",
    "ReplicatingFrom": [
        "ap-south-1",
        "us-west-2"
    ],
    "State": "ACTIVE",
    "Tags": {},
    "Type": "AGGREGATOR"
}
```

Esempio 2: come recuperare i dettagli per un indice locale di Esploratore di risorse

L'`get-index`esempio seguente visualizza i dettagli dell'indice Resource Explorer AWS nella regione specificata. Poiché la Regione specificata contiene un indice locale, l'output elenca la Regione in cui replica i dati dall'indice di questa Regione.

```
aws resource-explorer-2 get-index \
    --region us-west-2
```

Output:

```
{
    "Arn": "arn:aws:resource-explorer-2:us-west-2:123456789012:index/EXAMPLE8-90ab-
cdef-fedc-EXAMPLE22222",
    "CreatedAt": "2022-07-12T18:59:10.503000+00:00",
    "LastUpdatedAt": "2022-07-13T18:41:58.799000+00:00",
    "ReplicatingTo": [
        "us-west-2"
    ],
    "State": "ACTIVE",
    "Tags": {},
    "Type": "LOCAL"
}
```

Per ulteriori informazioni sugli indici, vedere Verifica delle AWS aree in cui è attivo Resource Explorer nella Guida per l'utente di AWS Resource Explorer.

- Per i dettagli sull'API, consulta AWS CLI Command GetIndexReference.

`get-view`

Il seguente esempio di codice mostra come utilizzare`get-view`.

AWS CLI

Come recuperare i dettagli su una visualizzazione di Esploratore di risorse

L'esempio `get-view` seguente mostra i dettagli di una visualizzazione specificata dal relativo ARN.

```
aws resource-explorer-2 get-view \
    --view-arn arn:aws:resource-explorer-2:us-east-1:123456789012:view/EC2-Only-View/EXAMPLE8-90ab-cdef-fedc-EXAMPLE11111
```

Output:

```
{
  "Tags" : {},
  "View" : {
        "Filters" : {
            "FilterString" : "service:ec2"
        },
        "IncludedProperties" : [
            {
                "Name" : "tags"
            }
        ],
        "LastUpdatedAt" : "2022-07-13T21:33:45.249Z",
        "Owner" : "123456789012",
        "Scope" : "arn:aws:iam::123456789012:root",
        "ViewArn" : "arn:aws:resource-explorer-2:us-east-1:123456789012:view/EC2-Only-View/EXAMPLE8-90ab-cdef-fedc-EXAMPLE11111"
  }
}
```

Per ulteriori informazioni sulle visualizzazioni, consulta Informazioni sulle viste di Esploratore di risorse nella Guida per l'utente di Esploratore di risorse AWS Resource Explorer.

- Per i dettagli sull'API, consulta GetView AWS CLICommand Reference.

`list-indexes`

Il seguente esempio di codice mostra come utilizzare`list-indexes`.

AWS CLI

Per elencare le AWS regioni in cui Resource Explorer ha indici

L'esempio `list-indexes` seguente elenca gli indici di tutte le Regioni in cui Esploratore di risorse dispone di un indice. La risposta specifica il tipo di ciascun indice, la relativa AWS regione e il relativo ARN.

```
aws resource-explorer-2 list-indexes
```

Output:

```
{
    "Indexes": [
        {
            "Arn": "arn:aws:resource-explorer-2:us-west-2:123456789012:index/
EXAMPLE8-90ab-cdef-fedc-EXAMPLE11111",
            "Region": "us-west-2",
            "Type": "AGGREGATOR"
        },
        {
            "Arn": "arn:aws:resource-explorer-2:us-east-1:123456789012:index/
EXAMPLE8-90ab-cdef-fedc-EXAMPLE22222",
            "Region": "us-east-1",
            "Type": "LOCAL"
        },
        {
            "Arn": "arn:aws:resource-explorer-2:us-east-2:123456789012:index/
EXAMPLE8-90ab-cdef-fedc-EXAMPLE33333",
            "Region": "us-east-2",
            "Type": "LOCAL"
        },
        {
            "Arn": "arn:aws:resource-explorer-2:us-west-1:123456789012:index/
EXAMPLE8-90ab-cdef-fedc-EXAMPLE44444",
            "Region": "us-west-1",
            "Type": "LOCAL"
        }
    ]
```

```
}
```

Per ulteriori informazioni sugli indici, consulta Verifica delle AWS aree in cui è attivo Resource Explorer nella Guida per l'utente di AWS Resource Explorer.

- Per i dettagli sull'API, consulta AWS CLI Command ListIndexesReference.

`list-supported-resource-types`

Il seguente esempio di codice mostra come utilizzare`list-supported-resource-types`.

AWS CLI

Per elencare le AWS regioni in cui Resource Explorer ha indici

L'`list-supported-resource-types`esempio seguente elenca tutti i tipi di risorse attualmente supportati da &AREXlong;. Nella risposta di esempio è incluso un valore `NextToken` che indica la presenza di ulteriore output disponibile da recuperare con chiamate aggiuntive.

```
aws resource-explorer-2 list-supported-resource-types \
    --max-items 10
```

Output:

```
{
    "ResourceTypes": [
        {
            "ResourceType": "cloudfront:cache-policy",
            "Service": "cloudfront"
        },
        {
            "ResourceType": "cloudfront:distribution",
            "Service": "cloudfront"
        },
        {
            "ResourceType": "cloudfront:function",
            "Service": "cloudfront"
        },
        {
            "ResourceType": "cloudfront:origin-access-identity",
            "Service": "cloudfront"
        },
        {
```

```
            "ResourceType": "cloudfront:origin-request-policy",
            "Service": "cloudfront"
        },
        {
            "ResourceType": "cloudfront:realtime-log-config",
            "Service": "cloudfront"
        },
        {
            "ResourceType": "cloudfront:response-headers-policy",
            "Service": "cloudfront"
        },
        {
            "ResourceType": "cloudwatch:alarm",
            "Service": "cloudwatch"
        },
        {
            "ResourceType": "cloudwatch:dashboard",
            "Service": "cloudwatch"
        },
        {
            "ResourceType": "cloudwatch:insight-rule",
            "Service": "cloudwatch"
        }
    ],
    "NextToken": "eyJOZXh0VG9rZW4iOiBudWxsLCAiYm90b190cnVuY2F0ZV9hbW91bnQiOiAxMH0="
}
```

Per ottenere la parte successiva dell'output, chiama nuovamente l'operazione e passa il valore della risposta `NextToken` della chiamata precedente come valore per `--starting-token`. Ripeti l'operazione finché `NextToken` risulta assente dalla risposta.

```
aws resource-explorer-2 list-supported-resource-types \
    --max-items 10  \
    --starting-
token eyJOZXh0VG9rZW4iOiBudWxsLCAiYm90b190cnVuY2F0ZV9hbW91bnQiOiAxMH0=
```

Output:

```
{
    "ResourceTypes": [
        {
            "ResourceType": "cloudwatch:metric-stream",
```

```
            "Service": "cloudwatch"
        },
        {
            "ResourceType": "dynamodb:table",
            "Service": "dynamodb"
        },
        {
            "ResourceType": "ec2:capacity-reservation",
            "Service": "ec2"
        },
        {
            "ResourceType": "ec2:capacity-reservation-fleet",
            "Service": "ec2"
        },
        {
            "ResourceType": "ec2:client-vpn-endpoint",
            "Service": "ec2"
        },
        {
            "ResourceType": "ec2:customer-gateway",
            "Service": "ec2"
        },
        {
            "ResourceType": "ec2:dedicated-host",
            "Service": "ec2"
        },
        {
            "ResourceType": "ec2:dhcp-options",
            "Service": "ec2"
        },
        {
            "ResourceType": "ec2:egress-only-internet-gateway",
            "Service": "ec2"
        },
        {
            "ResourceType": "ec2:elastic-gpu",
            "Service": "ec2"
        }
    ],
    "NextToken": "eyJOZXh0VG9rZW4iOiBudWxsLCAiYm90b190cnVuY2F0ZV9hbW91bnQiOiAyMH0="
}
```

Per ulteriori informazioni sugli indici, consulta Verifica delle AWS regioni in cui è attivo Resource Explorer nella Guida per gli utenti di AWS Resource Explorer.

- Per i dettagli sull'API, consulta AWS CLI Command ListSupportedResourceTypesReference.

`list-tags-for-resource`

Il seguente esempio di codice mostra come utilizzare`list-tags-for-resource`.

AWS CLI

Come elencare i tag allegati a una visualizzazione o a un indice di Esploratore di risorse

L'esempio `list-tags-for-resource` seguente elenca le coppie di chiave di tag e valore collegate alla visualizzazione con l'ARN specificato. È necessario chiamare l'operazione dalla AWS regione che contiene la risorsa.

```
aws resource-explorer-2 list-tags-for-resource \
    --resource-arn arn:aws:resource-explorer-2:us-east-1:123456789012:view/My-View/
EXAMPLE8-90ab-cdef-fedc-EXAMPLE11111
```

Output:

```
{
    "Tags": {
        "application": "MainCorpApp",
        "department": "1234"
    }
}
```

Per ulteriori informazioni sull'assegnazione di tag alle viste, consulta Tagging delle viste per il controllo dell'accesso nella Guida per l'utente di Esploratore di risorse AWS .

- Per i dettagli sull'API, vedere ListTagsForResourcein AWS CLI Command Reference.

`list-views`

Il seguente esempio di codice mostra come utilizzare`list-views`.

AWS CLI

Per elencare le viste di Resource Explorer disponibili in una AWS regione

L'esempio `list-views` seguente elenca tutte le visualizzazioni disponibili nella Regione in cui viene invocata l'operazione.

```
aws resource-explorer-2 list-views
```

Output:

```
{
    "Views": [
        "arn:aws:resource-explorer-2:us-east-1:123456789012:view/EC2-Only-View/
EXAMPLE8-90ab-cdef-fedc-EXAMPLE11111",
        "arn:aws:resource-explorer-2:us-east-1:123456789012:view/Default-All-
Resources-View/EXAMPLE8-90ab-cdef-fedc-EXAMPLE22222",
        "arn:aws:resource-explorer-2:us-east-1:123456789012:view/Production-Only-
View/EXAMPLE8-90ab-cdef-fedc-EXAMPLE33333"
    ]
}
```

Per ulteriori informazioni sulle visualizzazioni, consulta Informazioni sulle viste di Esploratore di risorse nella Guida per l'utente di Esploratore di risorse AWS Resource Explorer.

- Per i dettagli sull'API, consulta ListViews AWS CLICommand Reference.

`search`

Il seguente esempio di codice mostra come utilizzare`search`.

AWS CLI

Esempio 1: come effettuare una ricerca utilizzando la visualizzazione predefinita

L'esempio `search` seguente visualizza tutte le risorse dell'area specificata associate al servizio. La ricerca utilizza la visualizzazione predefinita per la Regione. Nella risposta di esempio è incluso un valore `NextToken` che indica la presenza di ulteriore output disponibile da recuperare con chiamate aggiuntive.

```
aws resource-explorer-2 search \
    --query-string "service:iam"
```

Output:

```
{
    "Count": {
        "Complete": true,
        "TotalResources": 55
    },
    "NextToken":
 "AG9VOEF1KLEXAMPLEOhJHVwo5chEXAMPLER5XiEpNrgsEXAMPLE...b0CmOFOryHEXAMPLE",
    "Resources": [{
        "Arn": "arn:aws:iam::123456789012:policy/service-role/Some-Policy-For-A-
Service-Role",
        "LastReportedAt": "2022-07-21T12:34:42Z",
        "OwningAccountId": "123456789012",
        "Properties": [],
        "Region": "global",
        "ResourceType": "iam:policy",
        "Service": "iam"
    }, {
        "Arn": "arn:aws:iam::123456789012:policy/service-role/Another-Policy-For-A-
Service-Role",
        "LastReportedAt": "2022-07-21T12:34:42Z",
        "OwningAccountId": "123456789012",
        "Properties": [],
        "Region": "global",
        "ResourceType": "iam:policy",
        "Service": "iam"
    }, {
       ... TRUNCATED FOR BREVITY ...
    }],
    "ViewArn": "arn:aws:resource-explorer-2:us-east-1:123456789012:view/my-default-
view/EXAMPLE8-90ab-cdef-fedc-EXAMPLE11111"
}
```

Esempio 2: come effettuare una ricerca utilizzando una visualizzazione specificata

La ricerca di `search` esempio seguente mostra tutte le risorse («*») nella AWS regione specificata che sono visibili attraverso la vista specificata. I risultati includono solo le risorse associate ad Amazon a EC2 causa dei filtri allegati alla vista.

```
aws resource-explorer-2 search \
    -- query-string "*" \
    -- view-arn arn:aws:resource-explorer-2:us-east-1:123456789012:view/My-EC2-view/
EXAMPLE8-90ab-cdef-fedc-EXAMPLE22222
```

Output:

```
HTTP/1.1 200 OK
Date: Tue, 01 Nov 2022 20:00:59 GMT
Content-Type: application/json
Content-Length: <PayloadSizeBytes>

    {
        "Count": {
            "Complete": true,
            "TotalResources": 67
        },
        "Resources": [{
            "Arn": "arn:aws:ec2:us-east-1:123456789012:network-acl/acl-1a2b3c4d",
            "LastReportedAt": "2022-07-21T18:52:02Z",
            "OwningAccountId": "123456789012",
            "Properties": [{
                "Data": [{
                    "Key": "Department",
                    "Value": "AppDevelopment"
                }, {
                    "Key": "Environment",
                    "Value": "Production"
                }],
                "LastReportedAt": "2021-11-15T14:48:29Z",
                "Name": "tags"
            }],
            "Region": "us-east-1",
            "ResourceType": "ec2:network-acl",
            "Service": "ec2"
        }, {
            "Arn": "arn:aws:ec2:us-east-1:123456789012:subnet/subnet-1a2b3c4d",
            "LastReportedAt": "2022-07-21T21:22:23Z",
            "OwningAccountId": "123456789012",
            "Properties": [{
                "Data": [{
                    "Key": "Department",
                    "Value": "AppDevelopment"
                }, {
                    "Key": "Environment",
                    "Value": "Production"
                }],
                "LastReportedAt": "2021-07-29T19:02:39Z",
                "Name": "tags"
```

```
            }],
            "Region": "us-east-1",
            "ResourceType": "ec2:subnet",
            "Service": "ec2"
        }, {
            "Arn": "arn:aws:ec2:us-east-1:123456789012:dhcp-options/dopt-1a2b3c4d",
            "LastReportedAt": "2022-07-21T06:08:53Z",
            "OwningAccountId": "123456789012",
            "Properties": [{
                "Data": [{
                    "Key": "Department",
                    "Value": "AppDevelopment"
                }, {
                    "Key": "Environment",
                    "Value": "Production"
                }],
                "LastReportedAt": "2021-11-15T15:11:05Z",
                "Name": "tags"
            }],
            "Region": "us-east-1",
            "ResourceType": "ec2:dhcpoptions",
            "Service": "ec2"
        }, {
            ... TRUNCATED FOR BREVITY ...
        }],
        "ViewArn": "arn:aws:resource-explorer-2:us-east-1:123456789012:view/My-EC2-
view/EXAMPLE8-90ab-cdef-fedc-EXAMPLE22222"
    }
```


Per ulteriori informazioni, consulta Using AWS Resource Explorer per la ricerca di AWS risorse nella Resource Explorer Users Guide.

- Per informazioni dettagliate sull'API, consulta Search in AWS CLI Command Reference.

`tag-resource`

Il seguente esempio di codice mostra come utilizzare`tag-resource`.

AWS CLI

Come taggare una visualizzazione di Esploratore di risorse

L'esempio `tag-resource` seguente aggiunge la chiave di tag "environment" con il valore "production" alla visualizzazione con l'ARN specificato.

```
aws resource-explorer-2 tag-resource \
    --resource-arn arn:aws:resource-explorer-2:us-east-1:123456789012:view/My-View//
EXAMPLE8-90ab-cdef-fedc-EXAMPLE11111 \
    --tags environment=production
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Tagging delle viste per il controllo dell'accesso nella Guida per l'utente di Esploratore di risorse AWS .

- Per i dettagli sull'API, consulta TagResource AWS CLICommand Reference.

`untag-resource`

Il seguente esempio di codice mostra come utilizzare`untag-resource`.

AWS CLI

Come rimuovere un tag da una visualizzazione di Esploratore di risorse

L'esempio `untag-resource` seguente rimuove i tag con il nome di chiave "environment" dalla visualizzazione con l'ARN specificato.

```
aws resource-explorer-2 untag-resource \
    --resource-arn arn:aws:resource-explorer-2:us-east-1:123456789012:view/My-View//
EXAMPLE8-90ab-cdef-fedc-EXAMPLE11111 \
    --tag-keys environment
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Tagging delle viste per il controllo dell'accesso nella Guida per l'utente di Esploratore di risorse AWS .

- Per i dettagli sull'API, consulta UntagResource AWS CLICommand Reference.

`update-index-type`

Il seguente esempio di codice mostra come utilizzare`update-index-type`.

AWS CLI

Come modificare il tipo di un indice di Esploratore di risorse

L'`update-index-type`esempio seguente converte l'indice specificato da tipo `local aggregator` a tipo per attivare la possibilità di cercare risorse in tutte le AWS regioni dell'account. È necessario inviare la richiesta alla AWS regione che contiene l'indice che si desidera aggiornare.

```
aws resource-explorer-2 update-index-type \
    --arn arn:aws:resource-explorer-2:us-east-1:123456789012:index/EXAMPLE8-90ab-cdef-fedc-EXAMPLE11111 \
    --type aggregator \
    --region us-east-1
```

Output:

```
{
    "Arn":"arn:aws:resource-explorer-2:us-east-1:123456789012:index/EXAMPLE8-90ab-cdef-fedc-EXAMPLE11111",
    "LastUpdatedAt":"2022-07-13T18:41:58.799Z",
    "State":"updating",
    "Type":"aggregator"
}
```

Per ulteriori informazioni sulla modifica di un tipo di indice, consulta Attivazione della ricerca tra Regioni mediante la creazione di un indice di aggregazione nella Guida per l'utente di Esploratore di risorse AWS .

- Per i dettagli sull'API, consulta UpdateIndexType AWS CLICommand Reference.

`update-view`

Il seguente esempio di codice mostra come utilizzare`update-view`.

AWS CLI

Esempio 1: per aggiornare il IncludedProperties campo per una visualizzazione di Resource Explorer

L'esempio `update-view` seguente aggiorna la visualizzazione specificata aggiungendo i `` `tags` `` alle `` `IncludedProperties` `` opzionali. Dopo aver eseguito questa operazione, le operazioni di

ricerca che utilizzano questa visualizzazione includono informazioni sui tag collegati alle risorse che appaiono nei risultati.

```
aws resource-explorer-2 update-view \
    --included-properties Name=tags \
    --view-arn arn:aws:resource-explorer-2:us-east-1:123456789012:view/My-View/
EXAMPLE8-90ab-cdef-fedc-EXAMPLE22222
```

Output:

```
{
    "View": {
        "Filters": {
            "FilterString": ""
        },
        "IncludedProperties": [
            {
                "Name": "tags"
            }
        ],
        "LastUpdatedAt": "2022-07-19T17:41:21.710000+00:00",
        "Owner": "123456789012",
        "Scope": "arn:aws:iam::123456789012:root",
        "ViewArn": "arn:aws:resource-explorer-2:us-east-1:123456789012:view/My-EC2-
Only-View/EXAMPLE8-90ab-cdef-fedc-EXAMPLE11111"
    }
}
```

Esempio 2: come aggiornare i filtri associati a una visualizzazione

L'`update-view`esempio seguente aggiorna la vista specificata per utilizzare un filtro che limita i risultati solo ai tipi di risorse associati al EC2 servizio Amazon.

```
aws resource-explorer-2 update-view \
    --filters FilterString="service:ec2" \
    --view-arn arn:aws:resource-explorer-2:us-east-1:123456789012:view/My-View/
EXAMPLE8-90ab-cdef-fedc-EXAMPLE22222
```

Output:

```
{
    "View": {
```

```
    "Filters": {
        "FilterString": "service:ec2"
    },
    "IncludedProperties": [],
    "LastUpdatedAt": "2022-07-19T17:41:21.710000+00:00",
        "Owner": "123456789012",
        "Scope": "arn:aws:iam::123456789012:root",
        "ViewArn": "arn:aws:resource-explorer-2:us-east-1:123456789012:view/My-View/
EXAMPLE8-90ab-cdef-fedc-EXAMPLE22222"
    }
}
```

Per ulteriori informazioni sulle visualizzazioni, consulta Informazioni sulle viste di Esploratore di risorse nella Guida per l'utente di Esploratore di risorse AWS Resource Explorer.

- Per i dettagli sull'API, consulta UpdateView AWS CLICommand Reference.

Esempi di Resource Groups utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando il AWS Command Line Interface con Resource Groups.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`create-group`

Il seguente esempio di codice mostra come utilizzare`create-group`.

AWS CLI

Esempio 1: come creare un gruppo di risorse basato su tag

L'`create-group`esempio seguente crea un gruppo di risorse basato su tag di EC2 istanze Amazon nella regione corrente. Si basa su una query relativa alle risorse taggate con la chiave `Name` e il valore `WebServers`. Il nome del gruppo è `tbq-WebServer`. La query si trova in un file JSON separato che viene passato al comando.

```
aws resource-groups create-group \
    --name tbq-WebServer \
    --resource-query file://query.json
```

Contenuto di `query.json`:

```
{
    "Type": "TAG_FILTERS_1_0",
    "Query": "{\"ResourceTypeFilters\":[\"AWS::EC2::Instance\"],\"TagFilters\":
[{\"Key\":\"Name\", \"Values\":[\"WebServers\"]}]}"
}
```

Output:

```
{
    "Group": {
        "GroupArn": "arn:aws:resource-groups:us-west-2:123456789012:group/tbq-
WebServer",
        "Name": "tbq-WebServer"
    },
    "ResourceQuery": {
        "Type": "TAG_FILTERS_1_0",
        "Query": "{\"ResourceTypeFilters\":[\"AWS::EC2::Instance\"],\"TagFilters\":
[{\"Key\":\"Name\", \"Values\":[\"WebServers\"]}]}"
    }
}
```

Esempio 2: creare un gruppo di risorse basato sullo CloudFormation stack

L'`create-group`esempio seguente crea un gruppo di risorse AWS CloudFormation basato sullo stack denominato. `sampleCFNstackgroup` La query include tutte le risorse nello CloudFormation stack specificato supportate da AWS Resource Groups.

```
aws resource-groups create-group \
    --name cbq-CFNstackgroup \
```

```
    --resource-query file://query.json
```

Contenuto di `query.json`:

```
{
    "Type": "CLOUDFORMATION_STACK_1_0",
    "Query": "{\"ResourceTypeFilters\":[\"AWS::AllSupported\"],\"StackIdentifier
\":\"arn:aws:cloudformation:us-west-2:123456789012:stack/MyCFNStack/1415z9z0-
z39z-11z8-97z5-500z212zz6fz\"}"
}
```

Output:

```
{
    "Group": {
        "GroupArn": "arn:aws:resource-groups:us-west-2:123456789012:group/cbq-
CFNstackgroup",
        "Name": "cbq-CFNstackgroup"
    },
    "ResourceQuery": {
        "Type": "CLOUDFORMATION_STACK_1_0",
        "Query": "{\"ResourceTypeFilters\":[\"AWS::AllSupported\"],\"StackIdentifier
\":\"arn:aws:cloudformation:us-east-2:123456789012:stack/MyCFNStack/1415z9z0-
z39z-11z8-97z5-500z212zz6fz\"}"}'
    }
}
```

Per ulteriori informazioni, consulta Creazione di gruppi nella Guida per l'utente di Gruppi di risorse AWS .

- Per i dettagli sull'API, vedere CreateGroupin AWS CLI Command Reference.

`delete-group`

Il seguente esempio di codice mostra come utilizzare`delete-group`.

AWS CLI

Come aggiornare la descrizione di un gruppo di risorse

L'esempio `delete-group` seguente aggiorna il gruppo di risorse specificato.

```
aws resource-groups delete-group \
    --group-name tbq-WebServer
```

Output:

```
{
    "Group": {
        "GroupArn": "arn:aws:resource-groups:us-west-2:1234567890:group/tbq-
WebServer",
        "Name": "tbq-WebServer"
    }
}
```

Per ulteriori informazioni, consulta Eliminazione di gruppi nella Guida per l'utente di Gruppi di risorse AWS .

- Per i dettagli sull'API, consulta DeleteGroup AWS CLICommand Reference.

`get-group-query`

Il seguente esempio di codice mostra come utilizzare`get-group-query`.

AWS CLI

Come ottenere la query associata a un gruppo di risorse

L'esempio `get-group-query` seguente visualizza la query associata al gruppo di risorse specificato.

```
aws resource-groups get-group-query \
    --group-name tbq-WebServer
```

Output:

```
{
    "GroupQuery": {
        "GroupName": "tbq-WebServer",
        "ResourceQuery": {
            "Type": "TAG_FILTERS_1_0",
            "Query": "{\"ResourceTypeFilters\":[\"AWS::EC2::Instance\"],\"TagFilters
\":[{\"Key\":\"Name\", \"Values\":[\"WebServers\"]}]}"
```

```
        }
    }
}
```

- Per i dettagli sull'API, consulta GetGroupQuery AWS CLICommand Reference.

`get-group`

Il seguente esempio di codice mostra come utilizzare`get-group`.

AWS CLI

Come ottenere informazioni su un gruppo di risorse

L'esempio `get-group` seguente visualizza i dettagli relativi al gruppo di risorse specificato. Per ottenere la query associata al gruppo, utilizza `get-group-query`.

```
aws resource-groups get-group \
    --group-name tbq-WebServer
```

Output:

```
{
    "Group": {
        "GroupArn": "arn:aws:resource-groups:us-west-2:123456789012:group/tbq-
WebServer",
        "Name": "tbq-WebServer",
        "Description": "A tag-based query resource group of WebServers."
    }
}
```

- Per i dettagli sull'API, consulta GetGroup AWS CLICommand Reference.

`get-tags`

Il seguente esempio di codice mostra come utilizzare`get-tags`.

AWS CLI

Come recuperare i tag associati a un gruppo di risorse

L'esempio `get-tags` seguente visualizza le coppie di chiavi di tag e valori associate al gruppo di risorse specificato (il gruppo stesso e non i relativi membri).

```
aws resource-groups get-tags \
    --arn arn:aws:resource-groups:us-west-2:123456789012:group/tbq-WebServer
```

Output:

```
{
    "Arn": "arn:aws:resource-groups:us-west-2:123456789012:group/tbq-WebServer",
    "Tags": {
        "QueryType": "tags",
        "QueryResources": "ec2-instances"
    }
}
```

- Per i dettagli sull'API, consulta GetTags AWS CLICommand Reference.

`list-group-resources`

Il seguente esempio di codice mostra come utilizzare`list-group-resources`.

AWS CLI

Come elencare tutte le risorse in un gruppo di risorse

Esempio 1: l'esempio `list-resource-groups` seguente elenca tutte le risorse appartenenti gruppo di risorse specificato.

```
aws resource-groups list-group-resources \
    --group-name tbq-WebServer
```

Output:

```
{
    "ResourceIdentifiers": [
        {
            "ResourceArn": "arn:aws:ec2:us-west-2:123456789012:instance/
i-09f77fa38c12345ab",
            "ResourceType": "AWS::EC2::Instance"
```

```
        }
    ]
}
```

Esempio 2: L'esempio seguente elenca tutte le risorse del gruppo che hanno anche un 'tipo di risorsa'::: :Instance'AWS EC2 . :

aws resource-groups list-group-resources --group-name tbq- WebServer --filters Nome=Tipo di risorsa, Valori=::: :Instance AWS EC2

- Per i dettagli sull'API, consulta ListGroupResources AWS CLICommand Reference.

`list-groups`

Il seguente esempio di codice mostra come utilizzare`list-groups`.

AWS CLI

Come elencare i gruppi di risorse disponibili

L'esempio `list-groups` seguente visualizza un elenco di tutti i gruppi di risorse.

```
aws resource-groups list-groups
```

Output:

```
{
    "GroupIdentifiers": [
        {
            "GroupName": "tbq-WebServer",
            "GroupArn": "arn:aws:resource-groups:us-west-2:123456789012:group/tbq-
WebServer3"
        },
        {
            "GroupName": "cbq-CFNStackQuery",
            "GroupArn": "arn:aws:resource-groups:us-west-2:123456789012:group/cbq-
CFNStackQuery"
        }
    ],
    "Groups": [
        {
```

```
            "GroupArn": "arn:aws:resource-groups:us-west-2:123456789012:group/tbq-
WebServer",
            "Name": "tbq-WebServer"
        },
        {
            "GroupArn": "arn:aws:resource-groups:us-west-2:123456789012:group/cbq-
CFNStackQuery",
            "Name": "cbq-CFNStackQuery"
        }
    ]
}
```

- Per i dettagli sull'API, consulta ListGroups AWS CLICommand Reference.

`list-resource-groups`

Il seguente esempio di codice mostra come utilizzare`list-resource-groups`.

AWS CLI

Come elencare tutte le risorse in un gruppo di risorse

L'esempio `list-resource-groups` seguente elenca tutte le risorse appartenenti gruppo di risorse specificato.

```
aws resource-groups list-group-resources \
    --group-name tbq-WebServer
```

Output:

```
{
    "ResourceIdentifiers": [
        {
            "ResourceArn": "arn:aws:ec2:us-west-2:123456789012:instance/
i-09f77fa38c12345ab",
            "ResourceType": "AWS::EC2::Instance"
        }
    ]
}
```

- Per i dettagli sull'API, consulta ListResourceGroups AWS CLICommand Reference.

`put-group-configuration`

Il seguente esempio di codice mostra come utilizzare`put-group-configuration`.

AWS CLI

Come associare una configurazione di servizio a un gruppo di risorse

Esempio 1: l'`put-group-configuration`esempio seguente specifica che il gruppo di risorse deve contenere solo le prenotazioni di EC2 capacità Amazon per le istanze delle famiglie `C5` or`M5`.

```
aws resource-groups put-group-configuration \
    --group MyTestGroup \
    --configuration file://config.json
```

Contenuto di `config.json`:

```
[
    {
        "Type": "AWS::EC2::HostManagement",
        "Parameters": [
            {
                "Name": "allowed-host-families",
                "Values": [ "c5", "m5" ]
            },
                {
                    "Name": "any-host-based-license-configuration",
                    "Values": [ "true" ]
                }
        ]
    },
    {
        "Type": "AWS::ResourceGroups::Generic",
        "Parameters": [
            {
                "Name": "allowed-resource-types",
                "Values": [ "AWS::EC2::Host" ]
            },
            {
                "Name": "deletion-protection",
                "Values": [ "UNLESS_EMPTY" ]
            }
        ]
```

```
    }
]
```

Questo comando non produce alcun output se ha esito positivo.

Per ulteriori informazioni, consulta Configurazioni dei servizi per i gruppi di risorse nella Guida di riferimento dell'API per gruppi di risorse.

- Per i dettagli sull'API, consulta AWS CLI Command PutGroupConfigurationReference.

`search-resources`

Il seguente esempio di codice mostra come utilizzare`search-resources`.

AWS CLI

Come trovare le risorse che corrispondono a una query

L'`search-resources`esempio seguente recupera un elenco di tutte le AWS risorse che corrispondono alla query specificata.

```
aws resource-groups search-resources \
    --resource-query file://query.json
```

Contenuto di `query.json`:

```
{
    "Type": "TAG_FILTERS_1_0",
    "Query": "{\"ResourceTypeFilters\":[\"AWS::EC2::Instance\"],\"TagFilters\":
[{\"Key\":\"Patch Group\", \"Values\":[\"Dev\"]}]}"
}
```

Output:

```
{
    "ResourceIdentifiers": [
        {
            "ResourceArn": "arn:aws:ec2:us-west-2:123456789012:instance/
i-01a23bc45d67890ef",
            "ResourceType": "AWS::EC2::Instance"
        }
    ]
```

```
}
```

- Per i dettagli sull'API, vedere SearchResourcesin AWS CLI Command Reference.

`tag`

Il seguente esempio di codice mostra come utilizzare`tag`.

AWS CLI

Come associare un tag a un gruppo di risorse

L'esempio `tag` seguente collega le coppie di chiavi di tag e valori specificate al gruppo di risorse specificato (il gruppo stesso e non i relativi membri).

```
aws resource-groups tag \
    --tags QueryType=tags,QueryResources=ec2-instances \
    --arn arn:aws:resource-groups:us-west-2:128716708097:group/tbq-WebServer
```

Output:

```
{
    "Arn": "arn:aws:resource-groups:us-west-2:128716708097:group/tbq-WebServer",
    "Tags": {
        "QueryType": "tags",
        "QueryResources": "ec2-instances"
    }
}
```

Per ulteriori informazioni, consulta Gestione dei tag nella Guida per l'utente di Gruppi di risorse AWS .

- Per informazioni dettagliate sull'API, consulta Tag nella documentazione di riferimento dei comandi della AWS CLI .

`untag`

Il seguente esempio di codice mostra come utilizzare`untag`.

AWS CLI

Come rimuovere i tag da un gruppo di risorse

L'esempio `untags` seguente rimuove qualsiasi tag con la chiave specificata dal gruppo di risorse stesso e non dai relativi membri.

```
aws resource-groups untag \
    --arn arn:aws:resource-groups:us-west-2:123456789012:group/tbq-WebServer \
    --keys QueryType
```

Output:

```
{
    "Arn": "arn:aws:resource-groups:us-west-2:123456789012:group/tbq-WebServer",
    "Keys": [
        "QueryType"
    ]
}
```

Per ulteriori informazioni, consulta Gestione dei tag nella Guida per l'utente di Gruppi di risorse AWS .

- Per informazioni dettagliate sull'API, consulta Untag nella documentazione di riferimento dei comandi della AWS CLI .

update-group-query

Il seguente esempio di codice mostra come utilizzare`update-group-query`.

AWS CLI

Esempio 1: come aggiornare la query per un gruppo di risorse basato su tag

L'esempio `update-group-query` seguente aggiorna la query associata al gruppo di risorse basato su tag specificato.

```
aws resource-groups update-group-query \
    --group-name tbq-WebServer \
    --resource-query '{"Type":"TAG_FILTERS_1_0", "Query":"{\"ResourceTypeFilters\":
[\"AWS::EC2::Instance\"],\"TagFilters\":[{\"Key\":\"Name\", \"Values\":[\"WebServers
\"]}]}"}'
```

Output:

```
{
    "Group": {
        "GroupArn": "arn:aws:resource-groups:us-east-2:123456789012:group/tbq-
WebServer",
        "Name": "tbq-WebServer"
    },
    "ResourceQuery": {
        "Type": "TAG_FILTERS_1_0",
        "Query": "{\"ResourceTypeFilters\":[\"AWS::EC2::Instance\"],\"TagFilters\":
[{\"Key\":\"Name\", \"Values\":[\"WebServers\"]}]}"
    }
}
```

Per ulteriori informazioni, consulta Aggiornamento dei gruppi nella Guida per l'utente di Gruppi di risorse AWS .

Esempio 2: Per aggiornare la query per un gruppo di risorse CloudFormation basato sullo stack

L'`update-group-query`esempio seguente aggiorna la query allegata al gruppo di risorse basato sullo AWS CloudFormation stack specificato.

```
aws resource-groups update-group-query \
    --group-name cbq-CFNstackgroup \
    --resource-query '{"Type": "CLOUDFORMATION_STACK_1_0", "Query":
 "{\"ResourceTypeFilters\":[\"AWS::AllSupported\"],\"StackIdentifier\":
\"arn:aws:cloudformation:us-west-2:123456789012:stack/MyCFNStack/1415z9z0-
z39z-11z8-97z5-500z212zz6fz\"}"}'
```

Output:

```
{
    "Group": {
        "GroupArn": "arn:aws:resource-groups:us-west-2:123456789012:group/cbq-
CFNstackgroup",
        "Name": "cbq-CFNstackgroup"
    },
    "ResourceQuery": {
        "Type": "CLOUDFORMATION_STACK_1_0",
        "Query": "{\"ResourceTypeFilters\":[\"AWS::AllSupported\"],\"StackIdentifier
\":\"arn:aws:cloudformation:us-west-2:123456789012:stack/MyCFNStack/1415z9z0-
z39z-11z8-97z5-500z212zz6fz\"}"
    }
```

```
}
```

Per ulteriori informazioni, consulta Aggiornamento dei gruppi nella Guida per l'utente di Gruppi di risorse AWS .

- Per i dettagli sull'API, vedere UpdateGroupQueryin AWS CLI Command Reference.

`update-group`

Il seguente esempio di codice mostra come utilizzareu`pdate-group`.

AWS CLI

Come aggiornare la descrizione di un gruppo di risorse

L'esempio `update-group` seguente aggiorna la descrizione del gruppo di risorse specificato.

```
aws resource-groups update-group \
    --group-name tbq-WebServer \
    --description "Resource group for all web server resources."
```

Output:

```
{
    "Group": {
        "GroupArn": "arn:aws:resource-groups:us-west-2:123456789012:group/tbq-
WebServer",
        "Name": "tbq-WebServer"
        "Description": "Resource group for all web server resources."
    }
}
```

Per ulteriori informazioni, consulta Aggiornamento dei gruppi nella Guida per l'utente di Gruppi di risorse AWS .

- Per i dettagli sull'API, consulta UpdateGroup AWS CLICommand Reference.

Esempi di API di etichettatura dei Resource Groups utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando l'API AWS Command Line Interface with Resource Groups Tagging.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`get-resources`

Il seguente esempio di codice mostra come utilizzare`get-resources`.

AWS CLI

Come ottenere un elenco di risorse taggate

L'esempio `get-resources` seguente visualizza un elenco di risorse dell'account contrassegnate con il nome e il valore della chiave specificati.

```
aws resourcegroupstaggingapi get-resources \
    --tag-filters Key=Environment,Values=Production \
    --tags-per-page 100
```

Output:

```
{
    "ResourceTagMappingList": [
        {
            "ResourceARN": " arn:aws:inspector:us-west-2:123456789012:target/0-
nvgVhaxX/template/0-7sbz2Kz0",
            "Tags": [
                {
                    "Key": "Environment",
                    "Value": "Production"
                }
            ]
        }
```

```
    ]
}
```

Per ulteriori informazioni, consulta GetResourcesil Resource Groups Tagging API Reference.

- Per i dettagli sull'API, consulta GetResources AWS CLICommand Reference.

`get-tag-keys`

Il seguente esempio di codice mostra come utilizzare`get-tag-keys`.

AWS CLI

Come ottenere un elenco di tutte le chiavi di tag

L'esempio `get-tag-keys` seguente recupera l'elenco di tutti i nomi delle chiavi di tag utilizzati dalle risorse nell'account.

```
aws resourcegroupstaggingapi get-tag-keys
```

Output:

```
{
    "TagKeys": [
        "Environment",
        "CostCenter",
        "Department"
    ]
}
```

Per ulteriori informazioni, consulta GetTagKeysil Resource Groups Tagging API Reference.

- Per i dettagli sull'API, consulta GetTagKeys AWS CLICommand Reference.

`get-tag-values`

Il seguente esempio di codice mostra come utilizzare`get-tag-values`.

AWS CLI

Come ottenere un elenco di tutti i valori dei tag

L'esempio `get-tag-values` seguente visualizza tutti i valori utilizzati per la chiave specificata per tutte le risorse nell'account.

```
aws resourcegroupstaggingapi get-tag-values \
    --key=Environment
```

Output:

```
{
    "TagValues": [
        "Alpha",
        "Gamma",
        "Production"
    ]
}
```

Per ulteriori informazioni, consulta GetTagValuesil Resource Groups Tagging API Reference.

- Per i dettagli sull'API, consulta GetTagValues AWS CLICommand Reference.

`tag-resources`

Il seguente esempio di codice mostra come utilizzare`tag-resources`.

AWS CLI

Come associare un tag a una risorsa

L'esempio `tag-resources` seguente contrassegna la risorsa specificata con un nome e un valore di chiave.

```
aws resourcegroupstaggingapi tag-resources \
    --resource-arn-list arn:aws:s3:::MyProductionBucket \
    --tags Environment=Production,CostCenter=1234
```

Output:

```
{
    "FailedResourcesMap": {}
}
```

Per ulteriori informazioni, consulta TagResourcesil Resource Groups Tagging API Reference.

- Per i dettagli sull'API, consulta TagResources AWS CLICommand Reference.

`untag-resources`

Il seguente esempio di codice mostra come utilizzare`untag-resources`.

AWS CLI

Come rimuovere un tag da una risorsa

L'esempio `untag-resources` seguente rimuove le chiavi di tag specificate e i relativi valori associati dalla risorsa specificata.

```
aws resourcegroupstaggingapi untag-resources \
    --resource-arn-list arn:aws:s3:::amzn-s3-demo-bucket \
    --tag-keys Environment CostCenter
```

Output:

```
{
    "FailedResourcesMap": {}
}
```

Per ulteriori informazioni, consulta UntagResourcesil Resource Groups Tagging API Reference.

- Per i dettagli sull'API, consulta UntagResources AWS CLICommand Reference.

Esempi di Route 53 utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with Route 53.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`change-resource-record-sets`

Il seguente esempio di codice mostra come utilizzare`change-resource-record-sets`.

AWS CLI

Come creare, aggiornare o eliminare un set di record delle risorse

Il comando `change-resource-record-sets` seguente crea un set di record delle risorse utilizzando `hosted-zone-id` `Z1R8UBAEXAMPLE` e la configurazione in formato JSON nel file `C:\awscli\route53\change-resource-record-sets.json`:

```
aws route53 change-resource-record-sets --hosted-zone-id Z1R8UBAEXAMPLE --change-batch file://C:\awscli\route53\change-resource-record-sets.json
```

Per ulteriori informazioni, consulta POST ChangeResourceRecordSets nel riferimento dell'API Amazon Route 53.

La configurazione nel file JSON dipende dal tipo di set di record delle risorse che si desidera creare:

BasicWeightedAliasWeighted AliasLatencyLatency AliasFailoverFailover Alias

Sintassi di base:

```
{
  "Comment": "optional comment about the changes in this change batch request",
  "Changes": [
    {
      "Action": "CREATE"|"DELETE"|"UPSERT",
      "ResourceRecordSet": {
        "Name": "DNS domain name",
        "Type": "SOA"|"A"|"TXT"|"NS"|"CNAME"|"MX"|"PTR"|"SRV"|"SPF"|"AAAA",
        "TTL": time to live in seconds,
        "ResourceRecords": [
          {
            "Value": "applicable value for the record type"
          },
          {...}
```

```
        ]
      }
    },
    {...}
  ]
}
```

Sintassi ponderata:

```
{
  "Comment": "optional comment about the changes in this change batch request",
  "Changes": [
    {
      "Action": "CREATE"|"DELETE"|"UPSERT",
      "ResourceRecordSet": {
        "Name": "DNS domain name",
        "Type": "SOA"|"A"|"TXT"|"NS"|"CNAME"|"MX"|"PTR"|"SRV"|"SPF"|"AAAA",
        "SetIdentifier": "unique description for this resource record set",
        "Weight": value between 0 and 255,
        "TTL": time to live in seconds,
        "ResourceRecords": [
          {
            "Value": "applicable value for the record type"
          },
          {...}
        ],
        "HealthCheckId": "optional ID of an Amazon Route 53 health check"
      }
    },
    {...}
  ]
}
```

Sintassi degli alias:

```
{
  "Comment": "optional comment about the changes in this change batch request",
  "Changes": [
    {
      "Action": "CREATE"|"DELETE"|"UPSERT",
      "ResourceRecordSet": {
        "Name": "DNS domain name",
        "Type": "SOA"|"A"|"TXT"|"NS"|"CNAME"|"MX"|"PTR"|"SRV"|"SPF"|"AAAA",
```

```
        "AliasTarget": {
          "HostedZoneId": "hosted zone ID for your CloudFront distribution, Amazon
 S3 bucket, Elastic Load Balancing load balancer, or Amazon Route 53 hosted zone",
          "DNSName": "DNS domain name for your CloudFront distribution, Amazon S3
 bucket, Elastic Load Balancing load balancer, or another resource record set in
 this hosted zone",
          "EvaluateTargetHealth": true|false
        },
        "HealthCheckId": "optional ID of an Amazon Route 53 health check"
      }
    },
    {...}
  ]
}
```

Sintassi ponderata degli alias:

```
{
  "Comment": "optional comment about the changes in this change batch request",
  "Changes": [
    {
      "Action": "CREATE"|"DELETE"|"UPSERT",
      "ResourceRecordSet": {
        "Name": "DNS domain name",
        "Type": "SOA"|"A"|"TXT"|"NS"|"CNAME"|"MX"|"PTR"|"SRV"|"SPF"|"AAAA",
        "SetIdentifier": "unique description for this resource record set",
        "Weight": value between 0 and 255,
        "AliasTarget": {
          "HostedZoneId": "hosted zone ID for your CloudFront distribution, Amazon
 S3 bucket, Elastic Load Balancing load balancer, or Amazon Route 53 hosted zone",
          "DNSName": "DNS domain name for your CloudFront distribution, Amazon S3
 bucket, Elastic Load Balancing load balancer, or another resource record set in
 this hosted zone",
          "EvaluateTargetHealth": true|false
        },
        "HealthCheckId": "optional ID of an Amazon Route 53 health check"
      }
    },
    {...}
  ]
}
```

Sintassi della latenza:

```
{
  "Comment": "optional comment about the changes in this change batch request",
  "Changes": [
    {
      "Action": "CREATE"|"DELETE"|"UPSERT",
      "ResourceRecordSet": {
        "Name": "DNS domain name",
        "Type": "SOA"|"A"|"TXT"|"NS"|"CNAME"|"MX"|"PTR"|"SRV"|"SPF"|"AAAA",
        "SetIdentifier": "unique description for this resource record set",
        "Region": "Amazon EC2 region name",
        "TTL": time to live in seconds,
        "ResourceRecords": [
          {
            "Value": "applicable value for the record type"
          },
          {...}
        ],
        "HealthCheckId": "optional ID of an Amazon Route 53 health check"
      }
    },
    {...}
  ]
}
```

Sintassi degli alias con latenza:

```
{
  "Comment": "optional comment about the changes in this change batch request",
  "Changes": [
    {
      "Action": "CREATE"|"DELETE"|"UPSERT",
      "ResourceRecordSet": {
        "Name": "DNS domain name",
        "Type": "SOA"|"A"|"TXT"|"NS"|"CNAME"|"MX"|"PTR"|"SRV"|"SPF"|"AAAA",
        "SetIdentifier": "unique description for this resource record set",
        "Region": "Amazon EC2 region name",
        "AliasTarget": {
          "HostedZoneId": "hosted zone ID for your CloudFront distribution, Amazon
 S3 bucket, Elastic Load Balancing load balancer, or Amazon Route 53 hosted zone",
          "DNSName": "DNS domain name for your CloudFront distribution, Amazon S3
 bucket, Elastic Load Balancing load balancer, or another resource record set in
 this hosted zone",
          "EvaluateTargetHealth": true|false
```

```
        },
        "HealthCheckId": "optional ID of an Amazon Route 53 health check"
      }
    },
    {...}
  ]
}
```

Sintassi del failover:

```
{
  "Comment": "optional comment about the changes in this change batch request",
  "Changes": [
    {
      "Action": "CREATE"|"DELETE"|"UPSERT",
      "ResourceRecordSet": {
        "Name": "DNS domain name",
        "Type": "SOA"|"A"|"TXT"|"NS"|"CNAME"|"MX"|"PTR"|"SRV"|"SPF"|"AAAA",
        "SetIdentifier": "unique description for this resource record set",
        "Failover": "PRIMARY" | "SECONDARY",
        "TTL": time to live in seconds,
        "ResourceRecords": [
          {
            "Value": "applicable value for the record type"
          },
          {...}
        ],
        "HealthCheckId": "ID of an Amazon Route 53 health check"
      }
    },
    {...}
  ]
}
```

Sintassi degli alias con failover:

```
{
  "Comment": "optional comment about the changes in this change batch request",
  "Changes": [
    {
      "Action": "CREATE"|"DELETE"|"UPSERT",
      "ResourceRecordSet": {
        "Name": "DNS domain name",
```

```
        "Type": "SOA"|"A"|"TXT"|"NS"|"CNAME"|"MX"|"PTR"|"SRV"|"SPF"|"AAAA",
        "SetIdentifier": "unique description for this resource record set",
        "Failover": "PRIMARY" | "SECONDARY",
        "AliasTarget": {
          "HostedZoneId": "hosted zone ID for your CloudFront distribution, Amazon
 S3 bucket, Elastic Load Balancing load balancer, or Amazon Route 53 hosted zone",
          "DNSName": "DNS domain name for your CloudFront distribution, Amazon S3
 bucket, Elastic Load Balancing load balancer, or another resource record set in
 this hosted zone",
          "EvaluateTargetHealth": true|false
        },
        "HealthCheckId": "optional ID of an Amazon Route 53 health check"
      }
    },
    {...}
  ]
}
```

- Per i dettagli sull'API, consulta ChangeResourceRecordSets AWS CLICommand Reference.

`change-tags-for-resource`

Il seguente esempio di codice mostra come utilizzare`change-tags-for-resource`.

AWS CLI

Il comando seguente aggiunge un tag denominato `owner` a una risorsa healthcheck specificata da ID:

```
aws route53 change-tags-for-resource --resource-type healthcheck --resource-
id 6233434j-18c1-34433-ba8e-3443434 --add-tags Key=owner,Value=myboss
```

Il comando seguente rimuove un tag denominato `owner` da una risorsa della zona ospitata specificata da ID:

```
aws route53 change-tags-for-resource --resource-type hostedzone --resource-
id Z1523434445 --remove-tag-keys owner
```

- Per i dettagli sull'API, consulta ChangeTagsForResource AWS CLICommand Reference.

`create-health-check`

Il seguente esempio di codice mostra come utilizzare`create-health-check`.

AWS CLI

Come creare un controllo dell'integrità

Il comando `create-health-check` seguente crea controllo dell'integrità delle risorse utilizzando il riferimento del chiamante `2014-04-01-18:47` e la configurazione in formato JSON nel file `C:\awscli\route53\create-health-check.json`:

```
aws route53 create-health-check --caller-reference 2014-04-01-18:47 --health-check-
config file://C:\awscli\route53\create-health-check.json
```

Sintassi JSON:

```
{
  "IPAddress": "IP address of the endpoint to check",
  "Port": port on the endpoint to check--required when Type is "TCP",
  "Type": "HTTP"|"HTTPS"|"HTTP_STR_MATCH"|"HTTPS_STR_MATCH"|"TCP",
  "ResourcePath": "path of the file that you want Amazon Route 53 to request--all
 Types except TCP",
  "FullyQualifiedDomainName": "domain name of the endpoint to check--all Types
 except TCP",
  "SearchString": "if Type is HTTP_STR_MATCH or HTTPS_STR_MATCH, the string to
 search for in the response body from the specified resource",
  "RequestInterval": 10 | 30,
  "FailureThreshold": integer between 1 and 10
}
```

Per aggiungere il controllo dell'integrità a un set di record di risorse Route 53, utilizza il comando `change-resource-record-sets`.

Per ulteriori informazioni, consulta Controlli dell'integrità e failover DNS di Amazon Route 53 nella Guida per gli sviluppatori di Amazon Route 53.

- Per i dettagli sull'API, consulta CreateHealthCheck AWS CLICommand Reference.

`create-hosted-zone`

Il seguente esempio di codice mostra come utilizzare`create-hosted-zone`.

AWS CLI

Come creare una zona ospitata

Il comando `create-hosted-zone` seguente aggiunge una zona ospitata denominata `example.com` utilizzando il riferimento del chiamante `2014-04-01-18:47`. Il commento facoltativo include uno spazio e pertanto deve essere racchiuso tra virgolette:

```
aws route53 create-hosted-zone --name example.com --caller-
reference 2014-04-01-18:47 --hosted-zone-config Comment="command-line version"
```

Per ulteriori informazioni, consulta Utilizzo delle zone ospitate nella Guida per gli sviluppatori di Amazon Route 53.

- Per i dettagli sull'API, consulta CreateHostedZone AWS CLICommand Reference.

`delete-health-check`

Il seguente esempio di codice mostra come utilizzare`delete-health-check`.

AWS CLI

Come eliminare un controllo dell'integrità

Il comando `delete-health-check` seguente elimina il controllo dell'integrità con `health-check-id` impostato su `e75b48d9-547a-4c3d-88a5-ae4002397608`:

```
aws route53 delete-health-check --health-check-id e75b48d9-547a-4c3d-88a5-
ae4002397608
```

- Per i dettagli sull'API, consulta DeleteHealthCheck AWS CLICommand Reference.

`delete-hosted-zone`

Il seguente esempio di codice mostra come utilizzare`delete-hosted-zone`.

AWS CLI

Come eliminare una zona ospitata

Il comando `delete-hosted-zone` seguente elimina la zona ospitata con `id` `Z36KTIQEXAMPLE`.

```
aws route53 delete-hosted-zone --id Z36KTIQEXAMPLE
```

- Per i dettagli sull'API, consulta DeleteHostedZone AWS CLICommand Reference.

`get-change`

Il seguente esempio di codice mostra come utilizzare`get-change`.

AWS CLI

Come ottenere lo stato di una modifica ai set di record di risorse

Il comando `get-change` seguente ottiene lo stato e altre informazioni sulla richiesta `change-resource-record-sets` il cui `Id` è `/change/CWPIK4URU2I5S`:

```
aws route53 get-change --id /change/CWPIK4URU2I5S
```

- Per i dettagli sull'API, consulta GetChange AWS CLICommand Reference.

`get-health-check`

Il seguente esempio di codice mostra come utilizzare`get-health-check`.

AWS CLI

Come ottenere informazioni su un controllo dell'integrità

Il comando `get-health-check` seguente ottiene informazioni sul controllo dell'integrità il cui `health-check-id` è `02ec8401-9879-4259-91fa-04e66d094674`:

```
aws route53 get-health-check --health-check-id 02ec8401-9879-4259-91fa-04e66d094674
```

- Per i dettagli sull'API, consulta GetHealthCheck AWS CLICommand Reference.

`get-hosted-zone`

Il seguente esempio di codice mostra come utilizzare`get-hosted-zone`.

AWS CLI

Come ottenere informazioni su una zona ospitata

Il comando `get-hosted-zone` seguente ottiene informazioni sulla zona ospitata con `id` `Z1R8UBAEXAMPLE`:

```
aws route53 get-hosted-zone --id Z1R8UBAEXAMPLE
```

- Per i dettagli sull'API, consulta GetHostedZone AWS CLICommand Reference.

`list-health-checks`

Il seguente esempio di codice mostra come utilizzare`list-health-checks`.

AWS CLI

Per elencare i controlli sanitari associati all' AWS account corrente

Il `list-health-checks` comando seguente elenca informazioni dettagliate sui primi 100 controlli sanitari associati all' AWS account corrente. :

```
aws route53 list-health-checks
```

Se disponi di oltre 100 controlli dell'integrità o se desideri elencarli in gruppi più piccoli di 100, includi il parametro `--maxitems`. Ad esempio, per elencare i controlli dell'integrità una alla volta, utilizza il comando seguente:

```
aws route53 list-health-checks --max-items 1
```

Per visualizzare le informazioni sui controlli dell'integrità, prendi il valore di `NextToken` dalla risposta al comando precedente e includilo nel parametro `--starting-token`, ad esempio:

```
aws route53 list-health-checks --max-items 1 --starting-token Z3M3LMPEXAMPLE
```

- Per i dettagli sull'API, consulta ListHealthChecks AWS CLICommand Reference.

`list-hosted-zones-by-name`

Il seguente esempio di codice mostra come utilizzare`list-hosted-zones-by-name`.

AWS CLI

Il comando seguente elenca fino a 100 zone ospitate ordinate per nome di dominio.

```
aws route53 list-hosted-zones-by-name
```

Output:

```
{
  "HostedZones": [
      {
          "ResourceRecordSetCount": 2,
          "CallerReference": "test20150527-2",
          "Config": {
              "Comment": "test2",
              "PrivateZone": false
          },
          "Id": "/hostedzone/Z119WBBTVP5WFX",
          "Name": "2.example.com."
      },
      {
          "ResourceRecordSetCount": 2,
          "CallerReference": "test20150527-1",
          "Config": {
              "Comment": "test",
              "PrivateZone": false
          },
          "Id": "/hostedzone/Z3P5QSUBK4POTI",
          "Name": "www.example.com."
      }
  ],
  "IsTruncated": false,
  "MaxItems": "100"
}
```

Il comando seguente elenca fino le zone ospitate ordinate per nome che inizia con `www.example.com`.

```
aws route53 list-hosted-zones-by-name --dns-name www.example.com
```

Output:

```
{
  "HostedZones": [
      {
          "ResourceRecordSetCount": 2,
```

```
            "CallerReference": "mwunderl20150527-1",
            "Config": {
                "Comment": "test",
                "PrivateZone": false
            },
            "Id": "/hostedzone/Z3P5QSUBK4POTI",
            "Name": "www.example.com."
        }
    ],
    "DNSName": "www.example.com",
    "IsTruncated": false,
    "MaxItems": "100"
}
```

- Per i dettagli sull'API, consulta ListHostedZonesByName AWS CLICommand Reference.

`list-hosted-zones`

Il seguente esempio di codice mostra come utilizzare`list-hosted-zones`.

AWS CLI

Per elencare le zone ospitate associate all' AWS account corrente

Il `list-hosted-zones` comando seguente elenca le informazioni di riepilogo sulle prime 100 zone ospitate associate all' AWS account corrente. :

```
aws route53 list-hosted-zones
```

Se disponi di oltre 100 zone ospitate o se desideri elencarle in gruppi più piccoli di 100, includi il parametro `--max-items`. Ad esempio, per elencare le zone ospitate una alla volta, utilizza il comando seguente:

```
aws route53 list-hosted-zones --max-items 1
```

Per visualizzare le informazioni sulla zona ospitata successiva, prendi il valore di `NextToken` dalla risposta al comando precedente e includilo nel parametro `--starting-token`, ad esempio:

```
aws route53 list-hosted-zones --max-items 1 --starting-token Z3M3LMPEXAMPLE
```

- Per i dettagli sull'API, consulta ListHostedZones AWS CLICommand Reference.

`list-query-logging-configs`

Il seguente esempio di codice mostra come utilizzare`list-query-logging-configs`.

AWS CLI

Come elencare le configurazioni di registrazione dei log delle query

L'`list-query-logging-configs`esempio seguente elenca le informazioni sulle prime 100 configurazioni di registrazione delle query nell' AWS account, per la zona ospitata. `Z1OX3WQEXAMPLE`

```
aws route53 list-query-logging-configs \
    --hosted-zone-id Z1OX3WQEXAMPLE
```

Output:

```
{
    "QueryLoggingConfigs": [
        {
            "Id": "964ff34e-ae03-4f06-80a2-9683cexample",
            "HostedZoneId": "Z1OX3WQEXAMPLE",
            "CloudWatchLogsLogGroupArn": "arn:aws:logs:us-east-1:111122223333:log-
group:/aws/route53/example.com:*"
        }
    ]
}
```

Per ulteriori informazioni, consulta Registrazione delle query DNS nella Guida per gli sviluppatori di Amazon Route 53.

- Per i dettagli sull'API, consulta AWS CLI Command ListQueryLoggingConfigsReference.

`list-resource-record-sets`

Il seguente esempio di codice mostra come utilizzare`list-resource-record-sets`.

AWS CLI

Come aggiornare i set di record della risorsa di una zona ospitata

Il comando `list-resource-record-sets` seguente elenca informazioni di riepilogo sui primi 100 set di record di risorse in una zona ospitata specificata:

```
aws route53 list-resource-record-sets --hosted-zone-id Z2LD58HEXAMPLE
```

Se la zona ospitata contiene più di 100 set di record della risorsa o se desideri elencarle in gruppi più piccoli di 100, includi il parametro `--maxitems`. Ad esempio, per elencare i set di record della risorsa uno alla volta, utilizza il comando seguente:

```
aws route53 list-resource-record-sets --hosted-zone-id Z2LD58HEXAMPLE --max-items 1
```

Per visualizzare le informazioni sul set di record della risorsa successivo nella zona ospitata, prendi il valore di `NextToken` dalla risposta al comando precedente e includilo nel parametro `--starting-token`, ad esempio:

```
aws route53 list-resource-record-sets --hosted-zone-id Z2LD58HEXAMPLE --max-items 1
 --starting-token Z3M3LMPEXAMPLE
```

Per visualizzare tutti i set di record della risorsa con un nome particolare, utilizza il parametro `--query` per filtrarli. Esempio:

```
aws route53 list-resource-record-sets --hosted-zone-id Z2LD58HEXAMPLE --
query "ResourceRecordSets[?Name == 'example.domain.']"
```

- Per i dettagli sull'API, consulta ListResourceRecordSets AWS CLICommand Reference.

Esempi di registrazione del dominio Route 53 utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando la registrazione del dominio AWS Command Line Interface with Route 53.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`check-domain-availability`

Il seguente esempio di codice mostra come utilizzare`check-domain-availability`.

AWS CLI

Come determinare se è possibile registrare un nome di dominio con Route 53

Il comando `check-domain-availability` seguente restituisce informazioni sulla disponibilità del nome di dominio `example.com` per la registrazione tramite Route 53.

Questo comando viene eseguito solo nella Regione `us-east-1`. Se la Regione predefinita è impostata su `us-east-1`, è possibile omettere il parametro `region`.

```
aws route53domains check-domain-availability \
    --region us-east-1 \
    --domain-name example.com
```

Output:

```
{
    "Availability": "UNAVAILABLE"
}
```

Route 53 supporta un gran numero di domini di primo livello (TLDs), come `.com` e `.jp`, ma non supportiamo tutti i domini disponibili. TLDs Se verifichi la disponibilità di un dominio e Route 53 non supporta il dominio di primo livello, `check-domain-availability` restituisce il seguente messaggio.

```
An error occurred (UnsupportedTLD) when calling the CheckDomainAvailability
 operation: <top-level domain> tld is not supported.
```

Per un elenco dei domini TLDs che puoi utilizzare per registrare un dominio con Route 53, consulta Domini che puoi registrare con Amazon Route 53 nella Amazon Route 53 Developer Guide. Per ulteriori informazioni sulla registrazione dei domini con Amazon Route 53, consulta Registrazione di un nuovo dominio nella Guida per gli sviluppatori di Amazon Route 53.

- Per i dettagli sull'API, consulta AWS CLI Command CheckDomainAvailabilityReference.

`check-domain-transferability`

Il seguente esempio di codice mostra come utilizzare`check-domain-transferability`.

AWS CLI

Come determinare se un dominio può essere trasferito a Route 53

Il comando `check-domain-transferability` seguente restituisce informazioni sulla possibilità di trasferire il nome di dominio `example.com` a Route 53.

Questo comando viene eseguito solo nella Regione `us-east-1`. Se la Regione predefinita è impostata su `us-east-1`, è possibile omettere il parametro `region`.

```
aws route53domains check-domain-transferability \
    --region us-east-1 \
    --domain-name example.com
```

Output:

```
{
    "Transferability": {
        "Transferable": "UNTRANSFERABLE"
    }
}
```

Per ulteriori informazioni, consulta Trasferimento della registrazione per un dominio ad Amazon Route 53 nella Guida per gli sviluppatori di Amazon Route 53.

- Per i dettagli sull'API, consulta CheckDomainTransferability AWS CLICommand Reference.

`delete-tags-for-domain`

Il seguente esempio di codice mostra come utilizzare`delete-tags-for-domain`.

AWS CLI

Come eliminare i tag per un dominio

Il comando `delete-tags-for-domain` seguente elimina tre tag dal dominio specificato. Tieni presente che è necessario specificare solo la chiave del tag, e non il suo valore.

Questo comando viene eseguito solo nella Regione `us-east-1`. Se la Regione predefinita è impostata su `us-east-1`, è possibile omettere il parametro `region`.

```
aws route53domains delete-tags-for-domain \
    --region us-east-1 \
    --domain-name example.com \
    --tags-to-delete accounting-key hr-key engineering-key
```

Questo comando non produce alcun output.

Per confermare che i tag sono stati eliminati, puoi eseguire list-tags-for-domain. Per ulteriori informazioni, consulta Tagging delle risorse Amazon Route 53 nella Guida per gli sviluppatori di Amazon Route 53.

- Per i dettagli sull'API, consulta DeleteTagsForDomain AWS CLICommand Reference.

`disable-domain-auto-renew`

Il seguente esempio di codice mostra come utilizzare`disable-domain-auto-renew`.

AWS CLI

Come disabilitare il rinnovo automatico di un dominio

Il comando `disable-domain-auto-renew` seguente configura Route 53 in modo da non rinnovare automaticamente il dominio `example.com` prima della scadenza della sua registrazione.

Questo comando viene eseguito solo nella Regione `us-east-1`. Se la Regione predefinita è impostata su `us-east-1`, è possibile omettere il parametro `region`.

```
aws route53domains disable-domain-auto-renew \
    --region us-east-1 \
    --domain-name example.com
```

Questo comando non produce alcun output.

Per confermare che l'impostazione è stata modificata, puoi eseguire get-domain-detail. Se il rinnovo automatico è disabilitato, il valore di `AutoRenew` è `False`. Per ulteriori

informazioni sul rinnovo automatico, consulta Renewing Registration for a Domain < https://docs.aws.amazon.com/Route53/ latest/DeveloperGuide/domain -renew.html nella Amazon Route 53 Developer Guide.

- Per i dettagli sull'API, consulta DisableDomainAutoRenew AWS CLICommand Reference.

`disable-domain-transfer-lock`

Il seguente esempio di codice mostra come utilizzare`disable-domain-transfer-lock`.

AWS CLI

Come disabilitare il blocco del trasferimento su un dominio

Il comando `disable-domain-transfer-lock` seguente rimuove il blocco del trasferimento sul dominio `example.com` in modo che il dominio possa essere trasferito a un altro registrar. Questo comando modifica lo stato `clientTransferProhibited`.

Questo comando viene eseguito solo nella Regione `us-east-1`. Se la Regione predefinita è impostata su `us-east-1`, è possibile omettere il parametro `region`.

```
aws route53domains disable-domain-transfer-lock \
    --region us-east-1 \
    --domain-name example.com
```

Output:

```
{
    "OperationId": "3f28e0ac-126a-4113-9048-cc930example"
}
```

Per confermare che il blocco del trasferimento è stato modificato, puoi eseguire get-domain-detail. Quando il blocco del trasferimento è disabilitato, il valore di `StatusList` non include `clientTransferProhibited`.

Per ulteriori informazioni sul processo di trasferimento, consulta Trasferimento di un dominio da Amazon Route 53 a un altro registrar nella Guida per gli sviluppatori di Amazon Route 53.

- Per i dettagli sull'API, consulta DisableDomainTransferLock AWS CLICommand Reference.

`enable-domain-auto-renew`

Il seguente esempio di codice mostra come utilizzare`enable-domain-auto-renew`.

AWS CLI

Come abilitare il rinnovo automatico di un dominio

Il comando `enable-domain-auto-renew` seguente configura Route 53 in modo da rinnovare automaticamente il dominio `example.com` prima della scadenza della sua registrazione.

Questo comando viene eseguito solo nella Regione `us-east-1`. Se la Regione predefinita è impostata su `us-east-1`, è possibile omettere il parametro `region`.

```
aws route53domains enable-domain-auto-renew \
    --region us-east-1 \
    --domain-name example.com
```

Questo comando non produce alcun output. Per confermare che l'impostazione è stata modificata, puoi eseguire get-domain-detail. Se il rinnovo automatico è abilitato, il valore di `AutoRenew` è `True`.

Per ulteriori informazioni sul rinnovo automatico, consulta Renewing Registration for a Domain < https://docs.aws.amazon.com/Route53/ latest/DeveloperGuide/domain -renew.html nella Amazon Route 53 Developer Guide.

- Per i dettagli sull'API, consulta EnableDomainAutoRenew AWS CLICommand Reference.

`enable-domain-transfer-lock`

Il seguente esempio di codice mostra come utilizzare`enable-domain-transfer-lock`.

AWS CLI

Come abilitare il blocco del trasferimento su un dominio

Il comando `enable-domain-transfer-lock` seguente blocca il dominio specificato in modo che non possa essere trasferito a un altro registrar. Questo comando modifica lo stato `clientTransferProhibited`.

Questo comando viene eseguito solo nella Regione `us-east-1`. Se la Regione predefinita è impostata su `us-east-1`, è possibile omettere il parametro `region`.

```
aws route53domains enable-domain-transfer-lock \
    --region us-east-1 \
    --domain-name example.com
```

Output:

```
{
    "OperationId": "3f28e0ac-126a-4113-9048-cc930example"
}
```

Per confermare che il blocco del trasferimento è stato modificato, puoi eseguire get-domain-detail. Quando il blocco del trasferimento è abilitato, il valore di `StatusList` include `clientTransferProhibited`.

Per ulteriori informazioni sul processo di trasferimento, consulta Trasferimento di un dominio da Amazon Route 53 a un altro registrar nella Guida per gli sviluppatori di Amazon Route 53.

- Per i dettagli sull'API, consulta EnableDomainTransferLock AWS CLICommand Reference.

`get-contact-reachability-status`

Il seguente esempio di codice mostra come utilizzare`get-contact-reachability-status`.

AWS CLI

Come determinare se il contatto del registrante ha risposto all'email di conferma

Il comando `get-contact-reachability-status` seguente restituisce informazioni sul fatto che il contatto del registrante per il dominio specificato abbia risposto o meno a un'email di conferma.

Questo comando viene eseguito solo nella Regione `us-east-1`. Se la Regione predefinita è impostata su `us-east-1`, è possibile omettere il parametro `region`.

```
aws route53domains get-contact-reachability-status \
    --region us-east-1 \
    --domain-name example.com
```

Output:

```
{
```

```
    "domainName": "example.com",
    "status": "DONE"
}
```

Per ulteriori informazioni, consulta Reinvio di e-mail di autorizzazione e conferma nella Guida per gli sviluppatori di Amazon Route 53.

- Per i dettagli sull'API, consulta GetContactReachabilityStatus AWS CLICommand Reference.

`get-domain-detail`

Il seguente esempio di codice mostra come utilizzare`get-domain-detail`.

AWS CLI

Come ottenere informazioni dettagliate su un dominio specificato

Il comando `get-domain-detail` seguente visualizza informazioni dettagliate sul dominio specificato.

Questo comando viene eseguito solo nella Regione `us-east-1`. Se la Regione predefinita è impostata su `us-east-1`, è possibile omettere il parametro `region`.

```
aws route53domains get-domain-detail \
    --region us-east-1 \
    --domain-name example.com
```

Output:

```
{
    "DomainName": "example.com",
    "Nameservers": [
        {
            "Name": "ns-2048.awsdns-64.com",
            "GlueIps": []
        },
        {
            "Name": "ns-2049.awsdns-65.net",
            "GlueIps": []
        },
        {
            "Name": "ns-2050.awsdns-66.org",
            "GlueIps": []
```

```
        },
        {
            "Name": "ns-2051.awsdns-67.co.uk",
            "GlueIps": []
        }
    ],
    "AutoRenew": true,
    "AdminContact": {
        "FirstName": "Saanvi",
        "LastName": "Sarkar",
        "ContactType": "COMPANY",
        "OrganizationName": "Example",
        "AddressLine1": "123 Main Street",
        "City": "Anytown",
        "State": "WA",
        "CountryCode": "US",
        "ZipCode": "98101",
        "PhoneNumber": "+1.8005551212",
        "Email": "ssarkar@example.com",
        "ExtraParams": []
    },
    "RegistrantContact": {
        "FirstName": "Alejandro",
        "LastName": "Rosalez",
        "ContactType": "COMPANY",
        "OrganizationName": "Example",
        "AddressLine1": "123 Main Street",
        "City": "Anytown",
        "State": "WA",
        "CountryCode": "US",
        "ZipCode": "98101",
        "PhoneNumber": "+1.8005551212",
        "Email": "arosalez@example.com",
        "ExtraParams": []
    },
    "TechContact": {
        "FirstName": "Wang",
        "LastName": "Xiulan",
        "ContactType": "COMPANY",
        "OrganizationName": "Example",
        "AddressLine1": "123 Main Street",
        "City": "Anytown",
        "State": "WA",
        "CountryCode": "US",
```

```
        "ZipCode": "98101",
        "PhoneNumber": "+1.8005551212",
        "Email": "wxiulan@example.com",
        "ExtraParams": []
    },
    "AdminPrivacy": true,
    "RegistrantPrivacy": true,
    "TechPrivacy": true,
    "RegistrarName": "Amazon Registrar, Inc.",
    "WhoIsServer": "whois.registrar.amazon",
    "RegistrarUrl": "http://registrar.amazon.com",
    "AbuseContactEmail": "abuse@registrar.amazon.com",
    "AbuseContactPhone": "+1.2062661000",
    "CreationDate": 1444934889.601,
    "ExpirationDate": 1602787689.0,
    "StatusList": [
        "clientTransferProhibited"
    ]
}
```

- Per i dettagli sull'API, consulta GetDomainDetail AWS CLICommand Reference.

`get-domain-suggestions`

Il seguente esempio di codice mostra come utilizzare`get-domain-suggestions`.

AWS CLI

Come ottenere un elenco dei nomi di dominio suggeriti

Il comando `get-domain-suggestions` seguente visualizza un elenco di nomi di dominio suggeriti in base al nome di dominio `example.com`. La risposta include solo i nomi di dominio disponibili. Questo comando viene eseguito solo nella Regione `us-east-1`. Se la Regione predefinita è impostata su `us-east-1`, è possibile omettere il parametro `region`.

```
aws route53domains get-domain-suggestions \
    --region us-east-1 \
    --domain-name example.com \
    --suggestion-count 10 \
    --only-available
```

Output:

```
{
    "SuggestionsList": [
        {
            "DomainName": "egzaampal.com",
            "Availability": "AVAILABLE"
        },
        {
            "DomainName": "examplelaw.com",
            "Availability": "AVAILABLE"
        },
        {
            "DomainName": "examplehouse.net",
            "Availability": "AVAILABLE"
        },
        {
            "DomainName": "homeexample.net",
            "Availability": "AVAILABLE"
        },
        {
            "DomainName": "examplelist.com",
            "Availability": "AVAILABLE"
       },
        {
            "DomainName": "examplenews.net",
            "Availability": "AVAILABLE"
        },
        {
            "DomainName": "officeexample.com",
            "Availability": "AVAILABLE"
        },
        {
            "DomainName": "exampleworld.com",
            "Availability": "AVAILABLE"
        },
        {
            "DomainName": "exampleart.com",
            "Availability": "AVAILABLE"
        }
    ]
}
```

- Per i dettagli sull'API, consulta GetDomainSuggestions AWS CLICommand Reference.

`get-operation-detail`

Il seguente esempio di codice mostra come utilizzare`get-operation-detail`.

AWS CLI

Come ottenere lo stato corrente di un'operazione

Alcune operazioni di registrazione del dominio funzionano in modalità asincrona e restituiscono una risposta prima che vengano completate. Queste operazioni restituiscono un ID operazione utilizzabile per ottenere lo stato corrente. Il comando `get-operation-detail` seguente restituisce lo stato dell'operazione specificata.

Questo comando viene eseguito solo nella Regione `us-east-1`. Se la Regione predefinita è impostata su `us-east-1`, è possibile omettere il parametro `region`.

```
aws route53domains get-operation-detail \
    --region us-east-1 \
    --operation-id edbd8d63-7fe7-4343-9bc5-54033example
```

Output:

```
{
    "OperationId": "edbd8d63-7fe7-4343-9bc5-54033example",
    "Status": "SUCCESSFUL",
    "DomainName": "example.com",
    "Type": "DOMAIN_LOCK",
    "SubmittedDate": 1573749367.864
}
```

- Per i dettagli sull'API, consulta GetOperationDetail AWS CLICommand Reference.

`list-domains`

Il seguente esempio di codice mostra come utilizzare`list-domains`.

AWS CLI

Per elencare i domini registrati con l'account corrente AWS

Il `list-domains` comando seguente elenca informazioni di riepilogo sui domini registrati con l'account corrente AWS .

Questo comando viene eseguito solo nella Regione `us-east-1`. Se la Regione predefinita è impostata su `us-east-1`, è possibile omettere il parametro `region`.

```
aws route53domains list-domains
    --region us-east-1
```

Output:

```
{
    "Domains": [
        {
            "DomainName": "example.com",
            "AutoRenew": true,
            "TransferLock": true,
            "Expiry": 1602712345.0
        },
        {
            "DomainName": "example.net",
            "AutoRenew": true,
            "TransferLock": true,
            "Expiry": 1602723456.0
        },
        {
            "DomainName": "example.org",
            "AutoRenew": true,
            "TransferLock": true,
            "Expiry": 1602734567.0
        }
    ]
}
```

- Per i dettagli sull'API, vedere ListDomainsin AWS CLI Command Reference.

`list-operations`

Il seguente esempio di codice mostra come utilizzare`list-operations`.

AWS CLI

Come elencare lo stato delle operazioni che restituiscono un ID operazione

Alcune operazioni di registrazione del dominio vengono eseguite in modalità asincrona e restituiscono una risposta prima che vengano completate. Queste operazioni restituiscono un ID operazione utilizzabile per ottenere lo stato corrente. Il comando `list-operations` seguente elenca le informazioni di riepilogo, incluso lo stato, sulle operazioni correnti di registrazione dei domini.

Questo comando viene eseguito solo nella Regione `us-east-1`. Se la Regione predefinita è impostata su `us-east-1`, è possibile omettere il parametro `region`.

```
aws route53domains list-operations
    --region us-east-1
```

Output:

```
{
    "Operations": [
        {
            "OperationId": "aab9822f-1da0-4bf3-8a15-fd4e0example",
            "Status": "SUCCESSFUL",
            "Type": "DOMAIN_LOCK",
            "SubmittedDate": 1455321739.986
        },
        {
            "OperationId": "c24379ed-76be-42f8-bdad-9379bexample",
            "Status": "SUCCESSFUL",
            "Type": "UPDATE_NAMESERVER",
            "SubmittedDate": 1468960475.109
        },
        {
            "OperationId": "f47e1297-ef9e-4c2b-ae1e-a5fcbexample",
            "Status": "SUCCESSFUL",
            "Type": "RENEW_DOMAIN",
            "SubmittedDate": 1473561835.943
        },
        {
            "OperationId": "75584f23-b15f-459e-aed7-dc6f5example",
            "Status": "SUCCESSFUL",
            "Type": "UPDATE_DOMAIN_CONTACT",
            "SubmittedDate": 1547501003.41
        }
    ]
}
```

L'output include tutte le operazioni che restituiscono un ID di operazione e che hai eseguito su tutti i domini che hai mai registrato utilizzando l' AWS account corrente. Per ottenere solo le operazioni inviate dopo una data specifica, è possibile includere il parametro `submitted-since` e indicare una data nel formato Unix con ora UTC (Coordinated Universal Time). Il comando seguente ottiene lo stato di tutte le operazioni inviate dopo le ore 00:00 UTC in data 1° gennaio 2020.

```
aws route53domains list-operations \
    --submitted-since 1577836800
```

- Per i dettagli sull'API, consulta ListOperations AWS CLICommand Reference.

`list-tags-for-domain`

Il seguente esempio di codice mostra come utilizzare`list-tags-for-domain`.

AWS CLI

Come elencare i tag di un dominio

Il comando `list-tags-for-domain` seguente elenca i tag attualmente associati al dominio specificato.

Questo comando viene eseguito solo nella Regione `us-east-1`. Se la Regione predefinita è impostata su `us-east-1`, è possibile omettere il parametro `region`.

```
aws route53domains list-tags-for-domain \
    --region us-east-1 \
    --domain-name example.com
```

Output:

```
{
    "TagList": [
        {
            "Key": "key1",
            "Value": "value1"
        },
        {
            "Key": "key2",
            "Value": "value2"
```

```
        }
    ]
}
```

Per ulteriori informazioni, consulta Tagging delle risorse Amazon Route 53 nella Guida per gli sviluppatori di Amazon Route 53.

- Per i dettagli sull'API, consulta ListTagsForDomain AWS CLICommand Reference.

`register-domain`

Il seguente esempio di codice mostra come utilizzare`register-domain`.

AWS CLI

Come registrare un dominio

Il comando `register-domain` seguente registra un dominio, recuperando tutti i valori dei parametri da un file in formato JSON.

Questo comando viene eseguito solo nella Regione `us-east-1`. Se la Regione predefinita è impostata su `us-east-1`, è possibile omettere il parametro `region`.

```
aws route53domains register-domain \
    --region us-east-1 \
    --cli-input-json file://register-domain.json
```

Contenuto di `register-domain.json`:

```
{
    "DomainName": "example.com",
    "DurationInYears": 1,
    "AutoRenew": true,
    "AdminContact": {
        "FirstName": "Martha",
        "LastName": "Rivera",
        "ContactType": "PERSON",
        "OrganizationName": "Example",
        "AddressLine1": "1 Main Street",
        "City": "Anytown",
        "State": "WA",
        "CountryCode": "US",
```

```
        "ZipCode": "98101",
        "PhoneNumber": "+1.8005551212",
        "Email": "mrivera@example.com"
    },
    "RegistrantContact": {
        "FirstName": "Li",
        "LastName": "Juan",
        "ContactType": "PERSON",
        "OrganizationName": "Example",
        "AddressLine1": "1 Main Street",
        "City": "Anytown",
        "State": "WA",
        "CountryCode": "US",
        "ZipCode": "98101",
        "PhoneNumber": "+1.8005551212",
        "Email": "ljuan@example.com"
    },
    "TechContact": {
        "FirstName": "Mateo",
        "LastName": "Jackson",
        "ContactType": "PERSON",
        "OrganizationName": "Example",
        "AddressLine1": "1 Main Street",
        "City": "Anytown",
        "State": "WA",
        "CountryCode": "US",
        "ZipCode": "98101",
        "PhoneNumber": "+1.8005551212",
        "Email": "mjackson@example.com"
    },
    "PrivacyProtectAdminContact": true,
    "PrivacyProtectRegistrantContact": true,
    "PrivacyProtectTechContact": true
}
```

Output:

```
{
    "OperationId": "b114c44a-9330-47d1-a6e8-a0b11example"
}
```

Per confermare che l'operazione è riuscita, puoi eseguire il comando `get-operation-detail`. Per ulteriori informazioni, consulta get-operation-detail.

Per ulteriori informazioni, consulta Registrazione di un nuovo dominio nella Guida per gli sviluppatori Amazon Route 53.

Per informazioni su quali domini di primo livello (TLDs) richiedono valori `ExtraParams` e quali sono i valori validi, consulta ExtraParamil riferimento alle API di Amazon Route 53.

- Per i dettagli sull'API, consulta AWS CLI Command RegisterDomainReference.

`renew-domain`

Il seguente esempio di codice mostra come utilizzare`renew-domain`.

AWS CLI

Come rinnovare un dominio

Il comando `renew-domain` seguente rinnova il dominio specificato per cinque anni. Per ottenere il valore per `current-expiry-year`, utilizza il comando `get-domain-detail` e converti il valore `ExpirationDate` dal formato Unix.

Questo comando viene eseguito solo nella Regione `us-east-1`. Se la Regione predefinita è impostata su `us-east-1`, è possibile omettere il parametro `region`.

```
aws route53domains renew-domain \
    --region us-east-1 \
    --domain-name example.com \
    --duration-in-years 5 \
    --current-expiry-year 2020
```

Output:

```
{
    "OperationId": "3f28e0ac-126a-4113-9048-cc930example"
}
```

Per confermare che l'operazione è riuscita, puoi eseguire il comando `get-operation-detail`. Per ulteriori informazioni, consulta get-operation-detail.

Il registro per ciascun dominio di primo livello (TLD), come .com o .org, controlla il numero massimo di anni per cui è possibile rinnovare un dominio. Per ottenere il periodo massimo di

rinnovo del dominio, consulta la sezione "Registrazione e periodo di rinnovo" per il dominio di primo livello in Domini che puoi registrare con Amazon Route 53 nella Guida per gli sviluppatori di Amazon Route 53.

Per ulteriori informazioni, consulta Rinnovo della registrazione per un dominio nella Guida per gli sviluppatori di Amazon Route 53.

- Per i dettagli sull'API, consulta RenewDomain AWS CLICommand Reference.

`resend-contact-reachability-email`

Il seguente esempio di codice mostra come utilizzare`resend-contact-reachability-email`.

AWS CLI

Come inviare nuovamente l'email di conferma all'indirizzo email corrente del contatto del registrant

Il comando `resend-contact-reachability-email` seguente invia nuovamente l'email di conferma all'indirizzo email corrente del contatto del registrant per il dominio example.com.

Questo comando viene eseguito solo nella Regione `us-east-1`. Se la Regione predefinita è impostata su `us-east-1`, è possibile omettere il parametro `region`.

```
aws route53domains resend-contact-reachability-email \
    --region us-east-1 \
    --domain-name example.com
```

Output:

```
{
    "domainName": "example.com",
    "emailAddress": "moliveira@example.com",
    "isAlreadyVerified": true
}
```

Se il valore di `isAlreadyVerified` è `true`, come in questo esempio, il contatto del registrant ha già confermato che l'indirizzo email specificato è raggiungibile.

Per ulteriori informazioni, consulta Reinvio di e-mail di autorizzazione e conferma nella Guida per gli sviluppatori di Amazon Route 53.

- Per i dettagli sull'API, consulta ResendContactReachabilityEmail AWS CLICommand Reference.

`retrieve-domain-auth-code`

Il seguente esempio di codice mostra come utilizzare`retrieve-domain-auth-code`.

AWS CLI

Come ottenere il codice di autorizzazione per un dominio in modo da poterlo trasferire a un altro registrar

Il comando `retrieve-domain-auth-code` seguente ottiene il codice di autorizzazione corrente per il dominio example.com. Questo valore deve essere fornito a un altro registrar di domini al fine di trasferire il dominio a quest'ultimo.

Questo comando viene eseguito solo nella Regione `us-east-1`. Se la Regione predefinita è impostata su `us-east-1`, è possibile omettere il parametro `region`.

```
aws route53domains retrieve-domain-auth-code \
    --region us-east-1 \
    --domain-name example.com
```

Output:

```
{
    "AuthCode": ")o!v3dJeXampLe"
}
```

Per ulteriori informazioni, consulta Trasferimento di un dominio da Amazon Route 53 a un altro registrar nella Guida per gli sviluppatori di Amazon Route 53.

- Per i dettagli sull'API, consulta RetrieveDomainAuthCode AWS CLICommand Reference.

`transfer-domain`

Il seguente esempio di codice mostra come utilizzare`transfer-domain`.

AWS CLI

Come trasferire un dominio ad Amazon Route 53

Il comando `transfer-domain` seguente trasferisce un dominio ad Amazon Route 53, con i parametri forniti dal file `C:\temp\transfer-domain.json` in formato JSON.

Questo comando viene eseguito solo nella Regione `us-east-1`. Se la Regione predefinita è impostata su `us-east-1`, è possibile omettere il parametro `region`.

```
aws route53domains transfer-domain \
    --region us-east-1 \
    --cli-input-json file://C:\temp\transfer-domain.json
```

Contenuto di `transfer-domain.json`:

```
{
    "DomainName": "example.com",
    "DurationInYears": 1,
    "Nameservers": [
        {
            "Name": "ns-2048.awsdns-64.com"
        },
        {
            "Name": "ns-2049.awsdns-65.net"
        },
        {
            "Name": "ns-2050.awsdns-66.org"
        },
        {
            "Name": "ns-2051.awsdns-67.co.uk"
        }
    ],
    "AuthCode": ")o!v3dJeXampLe",
    "AutoRenew": true,
    "AdminContact": {
        "FirstName": "Martha",
        "LastName": "Rivera",
        "ContactType": "PERSON",
        "OrganizationName": "Example",
        "AddressLine1": "1 Main Street",
        "City": "Anytown",
        "State": "WA",
        "CountryCode": "US",
        "ZipCode": "98101",
        "PhoneNumber": "+1.8005551212",
        "Email": "mrivera@example.com"
```

```
    },
    "RegistrantContact": {
        "FirstName": "Li",
        "LastName": "Juan",
        "ContactType": "PERSON",
        "OrganizationName": "Example",
        "AddressLine1": "1 Main Street",
        "City": "Anytown",
        "State": "WA",
        "CountryCode": "US",
        "ZipCode": "98101",
        "PhoneNumber": "+1.8005551212",
        "Email": "ljuan@example.com"
    },
    "TechContact": {
        "FirstName": "Mateo",
        "LastName": "Jackson",
        "ContactType": "PERSON",
        "OrganizationName": "Example",
        "AddressLine1": "1 Main Street",
        "City": "Anytown",
        "State": "WA",
        "CountryCode": "US",
        "ZipCode": "98101",
        "PhoneNumber": "+1.8005551212",
        "Email": "mjackson@example.com"
    },
    "PrivacyProtectAdminContact": true,
    "PrivacyProtectRegistrantContact": true,
    "PrivacyProtectTechContact": true
}
```

Output:

```
{
    "OperationId": "b114c44a-9330-47d1-a6e8-a0b11example"
}
```

Per confermare che l'operazione è riuscita, puoi eseguire il comando `get-operation-detail`. Per ulteriori informazioni, consulta get-operation-detail.

Per ulteriori informazioni, consulta Trasferimento della registrazione per un dominio ad Amazon Route 53 nella Guida per gli sviluppatori di Amazon Route 53.

- Per i dettagli sull'API, consulta TransferDomain AWS CLICommand Reference.

`update-domain-contact-privacy`

Il seguente esempio di codice mostra come utilizzare`update-domain-contact-privacy`.

AWS CLI

Come aggiornare le impostazioni sulla privacy dei contatti di un dominio

Il comando `update-domain-contact-privacy` seguente disattiva la protezione della privacy per il contatto amministrativo del dominio example.com. Questo comando viene eseguito solo nella Regione `us-east-1`.

Se la Regione predefinita è impostata su `us-east-1`, è possibile omettere il parametro `region`.

```
aws route53domains update-domain-contact-privacy \
    --region us-east-1 \
    --domain-name example.com \
    --no-admin-privacy
```

Output:

```
{
    "OperationId": "b3a219e9-d801-4244-b533-b7256example"
}
```

Per confermare che l'operazione è riuscita, puoi eseguire il comando `get-operation-detail`. Per ulteriori informazioni, consulta get-operation-detail.

Per ulteriori informazioni, consulta Abilitazione o disabilitazione della protezione della privacy per le informazioni di contatto per un dominio nella Guida per gli sviluppatori di Amazon Route 53.

- Per i dettagli sull'API, consulta UpdateDomainContactPrivacy AWS CLICommand Reference.

`update-domain-contact`

Il seguente esempio di codice mostra come utilizzare`update-domain-contact`.

AWS CLI

Come aggiornare le informazioni di contatto di un dominio

Il comando `update-domain-contact` seguente aggiorna le informazioni di contatto di un dominio, ottenendo i parametri dal file `C:\temp\update-domain-contact.json` in formato JSON.

Questo comando viene eseguito solo nella Regione `us-east-1`. Se la Regione predefinita è impostata su `us-east-1`, è possibile omettere il parametro `region`.

```
aws route53domains update-domain-contact \
    --region us-east-1 \
    --cli-input-json file://C:\temp\update-domain-contact.json
```

Contenuto di `update-domain-contact.json`:

```
{
    "AdminContact": {
        "AddressLine1": "101 Main Street",
        "AddressLine2": "Suite 1a",
        "City": "Seattle",
        "ContactType": "COMPANY",
        "CountryCode": "US",
        "Email": "w.xiulan@example.com",
        "FirstName": "Wang",
        "LastName": "Xiulan",
        "OrganizationName": "Example",
        "PhoneNumber": "+1.8005551212",
        "State": "WA",
        "ZipCode": "98101"
    },
    "DomainName": "example.com",
    "RegistrantContact": {
        "AddressLine1": "101 Main Street",
        "AddressLine2": "Suite 1a",
        "City": "Seattle",
        "ContactType": "COMPANY",
        "CountryCode": "US",
        "Email": "w.xiulan@example.com",
        "FirstName": "Wang",
        "LastName": "Xiulan",
        "OrganizationName": "Example",
        "PhoneNumber": "+1.8005551212",
        "State": "WA",
        "ZipCode": "98101"
    },
```

```
    "TechContact": {
        "AddressLine1": "101 Main Street",
        "AddressLine2": "Suite 1a",
        "City": "Seattle",
        "ContactType": "COMPANY",
        "CountryCode": "US",
        "Email": "w.xiulan@example.com",
        "FirstName": "Wang",
        "LastName": "Xiulan",
        "OrganizationName": "Example",
        "PhoneNumber": "+1.8005551212",
        "State": "WA",
        "ZipCode": "98101"
    }
}
```

Output:

```
{
    "OperationId": "b3a219e9-d801-4244-b533-b7256example"
}
```

Per confermare che l'operazione è riuscita, puoi eseguire get-domain-detail. Per ulteriori informazioni, consulta Aggiornamento delle informazioni di contatto per un dominio nella Guida per gli sviluppatori di Amazon Route 53.

- Per i dettagli sull'API, consulta UpdateDomainContact AWS CLICommand Reference.

`update-domain-nameservers`

Il seguente esempio di codice mostra come utilizzare`update-domain-nameservers`.

AWS CLI

Come aggiornare i server dei nomi di un dominio

Il comando `update-domain-nameservers` seguente aggiorna i server dei nomi per un dominio.

Questo comando viene eseguito solo nella Regione `us-east-1`. Se la Regione predefinita è impostata su `us-east-1`, è possibile omettere il parametro `region`.

```
aws route53domains update-domain-nameservers \
```

```
    --region us-east-1 \
    --domain-name example.com \
    --
nameservers Name=ns-1.awsdns-01.org Name=ns-2.awsdns-02.co.uk Name=ns-3.awsdns-03.net Name=n
```

Output:

```
{
    "OperationId": "f1691ec4-0e7a-489e-82e0-b19d3example"
}
```

Per confermare che l'operazione è riuscita, puoi eseguire get-domain-detail.

Per ulteriori informazioni, consulta Aggiunta o modifica di server dei nomi e glue record per un dominio nella Guida per gli sviluppatori di Amazon Route 53.

- Per i dettagli sull'API, consulta UpdateDomainNameservers AWS CLICommand Reference.

`update-tags-for-domain`

Il seguente esempio di codice mostra come utilizzare`update-tags-for-domain`.

AWS CLI

Come aggiungere o aggiornare i tag per un dominio

Il comando `update-tags-for-domain` seguente aggiunge o aggiorna due chiavi e i valori corrispondenti per il dominio example.com. Per aggiornare il valore di una chiave, è sufficiente includere la chiave e il nuovo valore. È possibile aggiungere o aggiornare i tag in un solo dominio alla volta.

Questo comando viene eseguito solo nella Regione `us-east-1`. Se la Regione predefinita è impostata su `us-east-1`, è possibile omettere il parametro `region`.

```
aws route53domains update-tags-for-domain \
    --region us-east-1 \
    --domain-name example.com \
    --tags-to-update "Key=key1,Value=value1" "Key=key2,Value=value2"
```

Questo comando non produce alcun output. Per confermare che i tag sono stati aggiunti o aggiornati, puoi eseguire list-tags-for-domain.

Per ulteriori informazioni, consulta Tagging delle risorse Amazon Route 53 nella Guida per gli sviluppatori di Amazon Route 53.

- Per i dettagli sull'API, consulta UpdateTagsForDomain AWS CLICommand Reference.

`view-billing`

Il seguente esempio di codice mostra come utilizzare`view-billing`.

AWS CLI

Per ottenere informazioni di fatturazione per gli addebiti di registrazione del dominio per l'account corrente AWS

Il comando `view-billing` seguente restituisce tutti i record di fatturazione relativi al dominio per l'account corrente per il periodo compreso tra il 1° gennaio 2018 (1514764800 nel formato orario Unix) e la mezzanotte del 31 dicembre 2019 (1577836800 nel formato orario Unix).

Questo comando viene eseguito solo nella Regione `us-east-1`. Se la Regione predefinita è impostata su `us-east-1`, è possibile omettere il parametro `region`.

```
aws route53domains view-billing \
    --region us-east-1 \
    --start-time 1514764800 \
    --end-time 1577836800
```

Output:

```
{
    "BillingRecords": [
        {
            "DomainName": "example.com",
            "Operation": "RENEW_DOMAIN",
            "InvoiceId": "149962827",
            "BillDate": 1536618063.181,
            "Price": 12.0
        },
        {
            "DomainName": "example.com",
            "Operation": "RENEW_DOMAIN",
            "InvoiceId": "290913289",
```

```
            "BillDate": 1568162630.884,
            "Price": 12.0
        }
    ]
}
```

Per ulteriori informazioni, consulta la ViewBillingpagina di riferimento dell'API Amazon Route 53.

- Per i dettagli sull'API, consulta ViewBilling AWS CLICommand Reference.

Esempi di profili Route 53 utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando i profili AWS Command Line Interface with Route 53.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`associate-profile`

Il seguente esempio di codice mostra come utilizzare`associate-profile`.

AWS CLI

Come associare un profilo

L'esempio `associate-profile` seguente associa un profilo a un VPC.

```
aws route53profiles associate-profile \
    --name test-association \
```

```
    --profile-id rp-4987774726example \
    --resource-id vpc-0af3b96b3example
```

Output:

```
{
    "ProfileAssociation": {
        "CreationTime": 1710851336.527,
        "Id": "rpassoc-489ce212fexample",
        "ModificationTime": 1710851336.527,
        "Name": "test-association",
        "OwnerId": "123456789012",
        "ProfileId": "rp-4987774726example",
        "ResourceId": "vpc-0af3b96b3example",
        "Status": "CREATING",
        "StatusMessage": "Creating Profile Association"
    }
}
```

Per ulteriori informazioni, consulta Utilizzo dei profili nella Guida per gli sviluppatori di Amazon Route 53.

- Per i dettagli sull'API, consulta AssociateProfile AWS CLICommand Reference.

`associate-resource-to-profile`

Il seguente esempio di codice mostra come utilizzare`associate-resource-to-profile`.

AWS CLI

Come associare una risorsa a un profilo

L'esempio `associate-resource-to-profile` seguente associa un gruppo di regole DNS Firewall con priorità 102 a un profilo.

```
aws route53profiles associate-resource-to-profile \
    --name test-resource-association \
    --profile-id rp-4987774726example \
    --resource-arn arn:aws:route53resolver:us-east-1:123456789012:firewall-rule-
group/rslvr-frg-cfe7f72example \
    --resource-properties "{\"priority\": 102}"
```

Output:

```
{
    "ProfileResourceAssociation": {
        "CreationTime": 1710851216.613,
        "Id": "rpr-001913120a7example",
        "ModificationTime": 1710851216.613,
        "Name": "test-resource-association",
        "OwnerId": "123456789012",
        "ProfileId": "rp-4987774726example",
        "ResourceArn": "arn:aws:route53resolver:us-east-1:123456789012:firewall-
rule-group/rslvr-frg-cfe7f72example",
        "ResourceProperties": "{\"priority\":102}",
        "ResourceType": "FIREWALL_RULE_GROUP",
        "Status": "UPDATING",
        "StatusMessage": "Updating the Profile to DNS Firewall rule group
 association"
    }
}
```

- Per i dettagli sull'API, consulta AssociateResourceToProfile AWS CLICommand Reference.

`create-profile`

Il seguente esempio di codice mostra come utilizzare`create-profile`.

AWS CLI

Come creare un profilo

L'esempio `create-profile` seguente crea un profilo.

```
aws route53profiles create-profile \
    --name test
```

Output:

```
{
    "Profile": {
        "Arn": "arn:aws:route53profiles:us-east-1:123456789012:profile/
rp-6ffe47d5example",
```

```
        "ClientToken": "2ca1a304-32b3-4f5f-bc4c-EXAMPLE11111",
        "CreationTime": 1710850903.578,
        "Id": "rp-6ffe47d5example",
        "ModificationTime": 1710850903.578,
        "Name": "test",
        "OwnerId": "123456789012",
        "ShareStatus": "NOT_SHARED",
        "Status": "COMPLETE",
        "StatusMessage": "Created Profile"
    }
}
```

- Per i dettagli sull'API, consulta CreateProfile AWS CLICommand Reference.

`delete-profile`

Il seguente esempio di codice mostra come utilizzare`delete-profile`.

AWS CLI

Come eliminare un profilo

L'esempio `delete-profile` seguente elimina un profilo.

```
aws route53profiles delete-profile \
    --profile-id rp-6ffe47d5example
```

Output:

```
{
    "Profile": {
        "Arn": "arn:aws:route53profiles:us-east-1:123456789012:profile/
rp-6ffe47d5example",
        "ClientToken": "0a15fec0-05d9-4f78-bec0-EXAMPLE11111",
        "CreationTime": 1710850903.578,
        "Id": "rp-6ffe47d5example",
        "ModificationTime": 1710850903.578,
        "Name": "test",
        "OwnerId": "123456789012",
        "ShareStatus": "NOT_SHARED",
        "Status": "DELETED",
        "StatusMessage": "Deleted Profile"
```

```
    }
}
```

- Per i dettagli sull'API, consulta DeleteProfile AWS CLICommand Reference.

`disassociate-profile`

Il seguente esempio di codice mostra come utilizzare`disassociate-profile`.

AWS CLI

Come dissociare un profilo

L'esempio `disassociate-profile` seguente dissocia un profilo da un VPC.

```
aws route53profiles disassociate-profile \
    --profile-id rp-4987774726example \
    --resource-id vpc-0af3b96b3example
```

Output:

```
{
    "ProfileAssociation": {
        "CreationTime": 1710851336.527,
        "Id": "rpassoc-489ce212fexample",
        "ModificationTime": 1710851401.362,
        "Name": "test-association",
        "OwnerId": "123456789012",
        "ProfileId": "rp-4987774726example",
        "ResourceId": "vpc-0af3b96b3example",
        "Status": "DELETING",
        "StatusMessage": "Deleting Profile Association"
    }
}
```

- Per i dettagli sull'API, consulta DisassociateProfile AWS CLICommand Reference.

`disassociate-resource-from-profile`

Il seguente esempio di codice mostra come utilizzare`disassociate-resource-from-profile`.

AWS CLI

Come dissociare una risorsa da un profilo

L'esempio `disassociate-resource-from-profile` seguente dissocia un gruppo di regole di DNS Firewall da un profilo.

```
aws route53profiles disassociate-resource-from-profile \
    --profile-id rp-4987774726example \
    --resource-arn arn:aws:route53resolver:us-east-1:123456789012:firewall-rule-group/rslvr-frg-cfe7f72example
```

Output:

```
{
    "ProfileResourceAssociation": {
        "CreationTime": 1710851216.613,
        "Id": "rpr-001913120a7example",
        "ModificationTime": 1710852624.36,
        "Name": "test-resource-association",
        "OwnerId": "123456789012",
        "ProfileId": "rp-4987774726example",
        "ResourceArn": "arn:aws:route53resolver:us-east-1:123456789012:firewall-rule-group/rslvr-frg-cfe7f72example",
        "ResourceProperties": "{\"priority\":105}",
        "ResourceType": "FIREWALL_RULE_GROUP",
        "Status": "DELETING",
        "StatusMessage": "Deleting the Profile to DNS Firewall rule group association"
    }
}
```

- Per i dettagli sull'API, consulta DisassociateResourceFromProfile AWS CLICommand Reference.

`get-profile-association`

Il seguente esempio di codice mostra come utilizzare`get-profile-association`.

AWS CLI

Come ottenere informazioni su un'associazione di profilo

Il comando `get-profile-association` seguente restituisce informazioni sull'associazione del profilo specificata.

```
aws route53profiles get-profile-association \
    --profile-association-id rpassoc-489ce212fexample
```

Output:

```
{
    "ProfileAssociation": {
        "CreationTime": 1709338817.148,
        "Id": "rrpassoc-489ce212fexample",
        "ModificationTime": 1709338974.772,
        "Name": "test-association",
        "OwnerId": "123456789012",
        "ProfileId": "rp-4987774726example",
        "ResourceId": "vpc-0af3b96b3example",
        "Status": "COMPLETE",
        "StatusMessage": "Created Profile Association"
    }
}
```

- Per i dettagli sull'API, consulta GetProfileAssociation AWS CLICommand Reference.

`get-profile-resource-association`

Il seguente esempio di codice mostra come utilizzare`get-profile-resource-association`.

AWS CLI

Come ottenere informazioni su una risorsa associata a un profilo

Il comando `get-profile-resource-association` seguente restituisce informazioni sull'associazione della risorsa specificata a un profilo.

```
aws route53profiles get-profile-resource-association \
    --profile-resource-association-id rpr-001913120a7example
```

Output:

```
{
```

```
    "ProfileResourceAssociation": {
        "CreationTime": 1710851216.613,
        "Id": "rpr-001913120a7example",
        "ModificationTime": 1710852303.798,
        "Name": "test-resource-association",
        "OwnerId": "123456789012",
        "ProfileId": "rp-4987774726example",
        "ResourceArn": "arn:aws:route53resolver:us-east-1:123456789012:firewall-
rule-group/rslvr-frg-cfe7f72example",
        "ResourceProperties": "{\"priority\":105}",
        "ResourceType": "FIREWALL_RULE_GROUP",
        "Status": "COMPLETE",
        "StatusMessage": "Completed creation of Profile to DNS Firewall rule group
 association"
    }
}
```

- Per i dettagli sull'API, consulta GetProfileResourceAssociation AWS CLICommand Reference.

`get-profile`

Il seguente esempio di codice mostra come utilizzare`get-profile`.

AWS CLI

Come ottenere informazioni su un profilo

Il comando `get-profile` seguente restituisce informazioni sul profilo specificato.

```
aws route53profiles get-profile \
    --profile-id rp-4987774726example
```

Output:

```
{
    "Profile": {
        "Arn": "arn:aws:route53profiles:us-east-1:123456789012:profile/
rp-4987774726example",
        "ClientToken": "0cbc5ae7-4921-4204-bea9-EXAMPLE11111",
        "CreationTime": 1710851044.288,
        "Id": "rp-4987774726example",
        "ModificationTime": 1710851044.288,
```

```
        "Name": "test",
        "OwnerId": "123456789012",
        "ShareStatus": "NOT_SHARED",
        "Status": "COMPLETE",
        "StatusMessage": "Created Profile"
    }
}
```

- Per i dettagli sull'API, consulta GetProfile AWS CLICommand Reference.

`list-profile-associations`

Il seguente esempio di codice mostra come utilizzare`list-profile-associations`.

AWS CLI

Come elencare le associazioni dei profili

Di seguito `list-profile-associations` sono elencate le associazioni di profili presenti nel tuo AWS account.

```
aws route53profiles list-profile-associations
```

Output:

```
{
    "ProfileAssociations": [
        {
            "CreationTime": 1709338817.148,
            "Id": "rpassoc-489ce212fexample",
            "ModificationTime": 1709338974.772,
            "Name": "test-association",
            "OwnerId": "123456789012",
            "ProfileId": "rp-4987774726example",
            "ResourceId": "vpc-0af3b96b3example",
            "Status": "COMPLETE",
            "StatusMessage": "Created Profile Association"
        }
    ]
}
```

- Per i dettagli sull'API, consulta ListProfileAssociations AWS CLICommand Reference.

`list-profile-resource-associations`

Il seguente esempio di codice mostra come utilizzare`list-profile-resource-associations`.

AWS CLI

Come elencare le associazioni di risorse del profilo

Il comando `list-profile-resource-associations` seguente elenca le associazioni di risorse del profilo per il profilo specificato.

```
aws route53profiles list-profile-resource-associations \
    --profile-id rp-4987774726example
```

Output:

```
{
     "ProfileResourceAssociations": [
         {
             "CreationTime": 1710851216.613,
             "Id": "rpr-001913120a7example",
             "ModificationTime": 1710851216.613,
             "Name": "test-resource-association",
             "OwnerId": "123456789012",
             "ProfileId": "rp-4987774726example",
             "ResourceArn": "arn:aws:route53resolver:us-
east-1:123456789012:firewall-rule-group/rslvr-frg-cfe7f72example",
             "ResourceProperties": "{\"priority\":102}",
             "ResourceType": "FIREWALL_RULE_GROUP",
             "Status": "COMPLETE",
             "StatusMessage": "Completed creation of Profile to DNS Firewall rule
 group association"
         }
     ]
 }
```

- Per i dettagli sull'API, consulta ListProfileResourceAssociations AWS CLICommand Reference.

`list-profiles`

Il seguente esempio di codice mostra come utilizzare`list-profiles`.

AWS CLI

Come elencare i profili

Di seguito `list-profiles` sono elencati i profili presenti nell' AWS account e vengono visualizzate informazioni aggiuntive su di essi.

```
aws route53profiles list-profiles
```

Output:

```
{
     "ProfileSummaries": [
         {
             "Arn": "arn:aws:route53profiles:us-east-1:123456789012:profile/
rp-4987774726example",
             "Id": "rp-4987774726example",
             "Name": "test",
             "ShareStatus": "NOT_SHARED"
         }
     ]
 }
```

- Per i dettagli sull'API, consulta ListProfiles AWS CLICommand Reference.

`list-tags-for-resource`

Il seguente esempio di codice mostra come utilizzare`list-tags-for-resource`.

AWS CLI

Come elencare i tag per una risorsa

Il comando `list-tags-for-resource` seguente elenca i tag della risorsa specificata.

```
aws route53profiles list-tags-for-resource \
    --resource-arn arn:aws:route53profiles:us-east-1:123456789012:profile/
rp-4987774726example
```

Output:

```
{
    "Tags": {
        "my-key-2": "my-value-2",
        "my-key-1": "my-value-1"
    }
}
```

- Per i dettagli sull'API, consulta ListTagsForResource AWS CLICommand Reference.

`update-profile-resource-association`

Il seguente esempio di codice mostra come utilizzare`update-profile-resource-association`.

AWS CLI

Come aggiornare una risorsa associata a un profilo

L'esempio `update-profile-resource-association` seguente aggiorna una priorità di un gruppo di regole DNS Firewall associato al profilo.

```
aws route53profiles update-profile-resource-association \
    --profile-resource-association-id rpr-001913120a7example \
    --resource-properties "{\"priority\": 105}"
```

Output:

```
{
    "ProfileResourceAssociation": {
        "CreationTime": 1710851216.613,
        "Id": "rpr-001913120a7example",
        "ModificationTime": 1710852303.798,
        "Name": "test-resource-association",
        "OwnerId": "123456789012",
        "ProfileId": "rp-4987774726example",
        "ResourceArn": "arn:aws:route53resolver:us-east-1:123456789012:firewall-
rule-group/rslvr-frg-cfe7f72example",
        "ResourceProperties": "{\"priority\":105}",
        "ResourceType": "FIREWALL_RULE_GROUP",
        "Status": "UPDATING",
        "StatusMessage": "Updating the Profile to DNS Firewall rule group
 association"
```

```
    }
}
```

- Per i dettagli sull'API, consulta UpdateProfileResourceAssociation AWS CLICommand Reference.

Esempi di Route 53 Resolver che utilizzano AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with Route 53 Resolver.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`associate-firewall-rule-group`

Il seguente esempio di codice mostra come utilizzare. `associate-firewall-rule-group`

AWS CLI

Come associare un gruppo di regole del firewall a un VPC

L'esempio `associate-firewall-rule-group` seguente associa un gruppo di regole di DNS Firewall a un VPC Amazon.

```
aws route53resolver associate-firewall-rule-group \
    --name test-association \
    --firewall-rule-group-id rslvr-frg-47f93271fexample \
    --vpc-id vpc-31e92222 \
    --priority 101
```

Output:

```
{
    "FirewallRuleGroupAssociation": {
        "Id": "rslvr-frgassoc-57e8873d7example",
        "Arn": "arn:aws:route53resolver:us-west-2:123456789012:firewall-rule-group-
association/rslvr-frgassoc-57e8873d7example",
        "FirewallRuleGroupId": "rslvr-frg-47f93271fexample",
        "VpcId": "vpc-31e92222",
        "Name": "test-association",
        "Priority": 101,
        "MutationProtection": "DISABLED",
        "Status": "UPDATING",
        "StatusMessage": "Creating Firewall Rule Group Association",
        "CreatorRequestId": "2ca1a304-32b3-4f5f-bc4c-EXAMPLE11111",
        "CreationTime": "2021-05-25T21:47:48.755768Z",
        "ModificationTime": "2021-05-25T21:47:48.755768Z"
    }
}
```

Per ulteriori informazioni, consulta Gestione delle associazioni tra il VPC e i gruppi di regole DNS Firewall per il risolutore Route 53 nella Guida per gli sviluppatori di Amazon Route 53.

- Per i dettagli sull'API, consulta AssociateFirewallRuleGroup AWS CLICommand Reference.

`associate-resolver-endpoint-ip-address`

Il seguente esempio di codice mostra come utilizzare`associate-resolver-endpoint-ip-address`.

AWS CLI

Come associare un altro indirizzo IP a un endpoint del risolutore

L'esempio `associate-resolver-endpoint-ip-address` seguente associa un altro indirizzo IP a un endpoint del risolutore in entrata. Se specifichi solo un ID di sottorete e ometti l'indirizzo IP dal parametro `--ip-address`, il risolutore sceglie automaticamente un indirizzo IP tra quelli disponibili nella sottorete specificata.

```
aws route53resolver associate-resolver-endpoint-ip-address \
    --resolver-endpoint-id rslvr-in-497098ad5example \
```

```
    --ip-address="SubnetId=subnet-12d8exam,Ip=192.0.2.118"
```

Output:

```
{
    "ResolverEndpoint": {
        "Id": "rslvr-in-497098ad5example",
        "CreatorRequestId": "AWSConsole.25.0123456789",
        "Arn": "arn:aws:route53resolver:us-west-2:111122223333:resolver-endpoint/
rslvr-in-497098ad5example",
        "Name": "my-inbound-endpoint",
        "SecurityGroupIds": [
            "sg-05cd7b25d6example"
        ],
        "Direction": "INBOUND",
        "IpAddressCount": 3,
        "HostVPCId": "vpc-304bexam",
        "Status": "UPDATING",
        "StatusMessage": "Updating the Resolver Endpoint",
        "CreationTime": "2020-01-02T23:25:45.538Z",
        "ModificationTime": "2020-01-02T23:25:45.538Z"
    }
}
```

Per ulteriori informazioni, consulta Valori che specifichi durante la creazione o la modifica di endpoint in entrata nella Guida per gli sviluppatori di Amazon Route 53.

- Per i dettagli sull'API, consulta AssociateResolverEndpointIpAddress AWS CLICommand Reference.

`associate-resolver-rule`

Il seguente esempio di codice mostra come utilizzare`associate-resolver-rule`.

AWS CLI

Come associare una regola del risolutore a un VPC

L'esempio `associate-resolver-rule` seguente associa una regola del risolutore a un VPC Amazon. Dopo l'esecuzione del comando, il risolutore inizia a inoltrare le query DNS alla rete in base alle impostazioni della regola, ad esempio il nome di dominio delle query inoltrate.

```
aws route53resolver associate-resolver-rule \
    --name my-resolver-rule-association \
    --resolver-rule-id rslvr-rr-42b60677c0example \
    --vpc-id vpc-304bexam
```

Output:

```
{
    "ResolverRuleAssociation": {
        "Id": "rslvr-rrassoc-d61cbb2c8bexample",
        "ResolverRuleId": "rslvr-rr-42b60677c0example",
        "Name": "my-resolver-rule-association",
        "VPCId": "vpc-304bexam",
        "Status": "CREATING",
        "StatusMessage": "[Trace id: 1-5dc5a8fa-ec2cc480d2ef07617example] Creating
 the association."
    }
}
```

Per ulteriori informazioni, consulta Inoltro di query DNS in uscita alla rete nella Guida per gli sviluppatori di Amazon Route 53.

- Per i dettagli sull'API, consulta AssociateResolverRule AWS CLICommand Reference.

`create-firewall-domain-list`

Il seguente esempio di codice mostra come utilizzare`create-firewall-domain-list`.

AWS CLI

Come creare un elenco di domini di DNS Firewall per il risolutore Route 53

L'`create-firewall-domain-list`esempio seguente crea un elenco di domini Route 53 Resolver DNS Firewall, denominato test, nel tuo account. AWS

```
aws route53resolver create-firewall-domain-list \
    --creator-request-id my-request-id \
    --name test
```

Output:

```
{
    "FirewallDomainList": {
        "Id": "rslvr-fdl-d61cbb2cbexample",
        "Arn": "arn:aws:route53resolver:us-west-2:123456789012:firewall-domain-list/
rslvr-fdl-d61cbb2cbexample",
        "Name": "test",
        "DomainCount": 0,
        "Status": "COMPLETE",
        "StatusMessage": "Created Firewall Domain List",
        "CreatorRequestId": "my-request-id",
        "CreationTime": "2021-05-25T15:55:51.115365Z",
        "ModificationTime": "2021-05-25T15:55:51.115365Z"
    }
}
```

Per ulteriori informazioni, consulta Gestione degli elenchi di domini personalizzati nella Guida per gli sviluppatori di Amazon Route 53.

- Per i dettagli sull'API, consulta Command CreateFirewallDomainListReference AWS CLI .

`create-firewall-rule-group`

Il seguente esempio di codice mostra come utilizzare`create-firewall-rule-group`.

AWS CLI

Come creare un gruppo di regole del firewall

L'esempio `create-firewall-rule-group` seguente crea un gruppo di regole di DNS Firewall.

```
aws route53resolver create-firewall-rule-group \
    --creator-request-id my-request-id \
    --name test
```

Output:

```
{
    "FirewallRuleGroup": {
        "Id": "rslvr-frg-47f93271fexample",
        "Arn": "arn:aws:route53resolver:us-west-2:123456789012:firewall-rule-group/
rslvr-frg-47f93271fexample",
```

```
        "Name": "test",
        "RuleCount": 0,
        "Status": "COMPLETE",
        "StatusMessage": "Created Firewall Rule Group",
        "OwnerId": "123456789012",
        "CreatorRequestId": "my-request-id",
        "ShareStatus": "NOT_SHARED",
        "CreationTime": "2021-05-25T18:59:26.490017Z",
        "ModificationTime": "2021-05-25T18:59:26.490017Z"
    }
}
```

Per ulteriori informazioni, consulta Gestione di gruppi di regole e regole in DNS Firewall nella Guida per gli sviluppatori di Amazon Route 53.

- Per i dettagli sull'API, consulta CreateFirewallRuleGroup AWS CLICommand Reference.

`create-firewall-rule`

Il seguente esempio di codice mostra come utilizzare`create-firewall-rule`.

AWS CLI

Come creare una regola del firewall

L'esempio `create-firewall-rule` seguente crea una regola del firewall in DNS Firewall per i domini elencati in un elenco di domini di DNS Firewall.

```
aws route53resolver create-firewall-rule \
    --name allow-rule \
    --firewall-rule-group-id rslvr-frg-47f93271fexample \
    --firewall-domain-list-id rslvr-fdl-9e956e9ffexample \
    --priority 101 \
    --action ALLOW
```

Output:

```
{
    "FirewallRule": {
        "FirewallRuleGroupId": "rslvr-frg-47f93271fexample",
        "FirewallDomainListId": "rslvr-fdl-9e956e9ffexample",
```

```
        "Name": "allow-rule",
        "Priority": 101,
        "Action": "ALLOW",
        "CreatorRequestId": "d81e3fb7-020b-415e-939f-EXAMPLE11111",
        "CreationTime": "2021-05-25T21:44:00.346093Z",
        "ModificationTime": "2021-05-25T21:44:00.346093Z"
    }
}
```

Per ulteriori informazioni, consulta Gestione di gruppi di regole e regole in DNS Firewall nella Guida per gli sviluppatori di Amazon Route 53.

- Per i dettagli sull'API, consulta CreateFirewallRule AWS CLICommand Reference.

`create-resolver-endpoint`

Il seguente esempio di codice mostra come utilizzare`create-resolver-endpoint`.

AWS CLI

Come creare un endpoint del risolutore in entrata

L'esempio `create-resolver-endpoint` seguente crea un endpoint del risolutore in entrata. È possibile utilizzare lo stesso comando per creare endpoint in entrata e in uscita.

aws route53resolver create-resolver-endpoint --name my-inbound-endpoint -- creator-request-id 2020-01-18:47 -- «sg-f62bexam» --direction INBOUND --ip-addresses =subnet-ba47exam, Ip=192.0.2.255 =subnet-12d8exam, security-group-ids Ip=192.0.2.254 SubnetId SubnetId

Output:

```
{
    "ResolverEndpoint": {
        "Id": "rslvr-in-f9ab8a03f1example",
        "CreatorRequestId": "2020-01-01-18:47",
        "Arn": "arn:aws:route53resolver:us-west-2:111122223333:resolver-endpoint/
rslvr-in-f9ab8a03f1example",
        "Name": "my-inbound-endpoint",
        "SecurityGroupIds": [
            "sg-f62bexam"
        ],
        "Direction": "INBOUND",
```

```
        "IpAddressCount": 2,
        "HostVPCId": "vpc-304examp",
        "Status": "CREATING",
        "StatusMessage": "[Trace id: 1-5dc1ff84-f3477826e4a190025example] Creating
 the Resolver Endpoint",
        "CreationTime": "2020-01-01T23:02:29.583Z",
        "ModificationTime": "2020-01-01T23:02:29.583Z"
    }
}
```

Come creare un endpoint del risolutore in uscita

L'esempio `create-resolver-endpoint` seguente crea un endpoint del risolutore in uscita utilizzando i valori riportati nel documento in formato JSON `create-outbound-resolver-endpoint.json`.

```
aws route53resolver create-resolver-endpoint \
    --cli-input-json file://c:\temp\create-outbound-resolver-endpoint.json
```

Contenuto di `create-outbound-resolver-endpoint.json`:

```
{
   "CreatorRequestId": "2020-01-01-18:47",
   "Direction": "OUTBOUND",
   "IpAddresses": [
      {
         "Ip": "192.0.2.255",
         "SubnetId": "subnet-ba47exam"
      },
      {
         "Ip": "192.0.2.254",
         "SubnetId": "subnet-12d8exam"
      }
   ],
   "Name": "my-outbound-endpoint",
   "SecurityGroupIds": [ "sg-05cd7b25d6example" ],
   "Tags": [
      {
         "Key": "my-key-name",
         "Value": "my-key-value"
      }
   ]
```

```
}
```

Per ulteriori informazioni, consulta Resolving DNS Queries Between VPCs and Your Network nella Amazon Route 53 Developer Guide.

- Per i dettagli sull'API, consulta Command CreateResolverEndpointReference AWS CLI .

`create-resolver-rule`

Il seguente esempio di codice mostra come utilizzare`create-resolver-rule`.

AWS CLI

Come creare una regola del risolutore

L'esempio `create-resolver-rule` seguente crea una regola di inoltro del risolutore. La regola utilizza l'endpoint in uscita rslvr-out-d 5e5920e37example per inoltrare le query DNS agli indirizzi IP 10.24.8.75 e 10.24.8.156. `example.com`

```
aws route53resolver create-resolver-rule \
    --creator-request-id 2020-01-02-18:47 \
    --domain-name example.com \
    --name my-rule \
    --resolver-endpoint-id rslvr-out-d5e5920e37example \
    --rule-type FORWARD \
    --target-ips "Ip=10.24.8.75" "Ip=10.24.8.156"
```

Output:

```
{
    "ResolverRule": {
        "Status": "COMPLETE",
        "RuleType": "FORWARD",
        "ResolverEndpointId": "rslvr-out-d5e5920e37example",
        "Name": "my-rule",
        "DomainName": "example.com.",
        "CreationTime": "2022-05-10T21:35:30.923187Z",
        "TargetIps": [
            {
                "Ip": "10.24.8.75",
                "Port": 53
```

```
            },
            {
                "Ip": "10.24.8.156",
                "Port": 53
            }
        ],
        "CreatorRequestId": "2022-05-10-16:33",
        "ModificationTime": "2022-05-10T21:35:30.923187Z",
        "ShareStatus": "NOT_SHARED",
        "Arn": "arn:aws:route53resolver:us-east-1:111117012054:resolver-rule/rslvr-
rr-b1e0b905e93611111",
        "OwnerId": "111111111111",
        "Id": "rslvr-rr-rslvr-rr-b1e0b905e93611111",
        "StatusMessage": "[Trace id: 1-22222222-3e56afcc71a3724664f22e24]
 Successfully created Resolver Rule."
    }
}
```

- Per i CreateResolverRuledettagli AWS CLI sull'API, consulta Command Reference.

`delete-firewall-domain-list`

Il seguente esempio di codice mostra come utilizzare`delete-firewall-domain-list`.

AWS CLI

Come eliminare un elenco di domini di DNS Firewall per il risolutore Route 53

L'`delete-firewall-domain-list`esempio seguente elimina un elenco di domini Route 53 Resolver DNS Firewall, denominato test, nell'account. AWS

```
aws route53resolver delete-firewall-domain-list \
    --firewall-domain-list-id rslvr-fdl-9e956e9ffexample
```

Output:

```
{
    "FirewallDomainList": {
        "Id": "rslvr-fdl-9e956e9ffexample",
        "Arn": "arn:aws:route53resolver:us-west-2:123456789012:firewall-domain-list/
rslvr-fdl-9e956e9ffexample",
```

```
        "Name": "test",
        "DomainCount": 6,
        "Status": "DELETING",
        "StatusMessage": "Deleting the Firewall Domain List",
        "CreatorRequestId": "my-request-id",
        "CreationTime": "2021-05-25T15:55:51.115365Z",
        "ModificationTime": "2021-05-25T18:58:05.588024Z"
    }
}
```

Per ulteriori informazioni, consulta Gestione degli elenchi di domini personalizzati nella Guida per gli sviluppatori di Amazon Route 53.

- Per i dettagli sull'API, consulta DeleteFirewallDomainListCommand Reference.AWS CLI

`delete-firewall-rule-group`

Il seguente esempio di codice mostra come utilizzare`delete-firewall-rule-group`.

AWS CLI

Come eliminare un gruppo di regole del firewall

L'esempio `delete-firewall-rule-group` seguente elimina un gruppo di regole del firewall.

```
aws route53resolver delete-firewall-rule-group \
    --firewall-rule-group-id rslvr-frg-47f93271fexample
```

Output:

```
{
    "FirewallRuleGroup": {
        "Id": "rslvr-frg-47f93271fexample",
        "Arn": "arn:aws:route53resolver:us-west-2:123456789012:firewall-rule-group/
rslvr-frg-47f93271fexample",
        "Name": "test",
        "RuleCount": 0,
        "Status": "UPDATING",
        "StatusMessage": "Updating Firewall Rule Group",
        "OwnerId": "123456789012",
        "CreatorRequestId": "my-request-id",
        "ShareStatus": "NOT_SHARED",
```

```
        "CreationTime": "2021-05-25T18:59:26.490017Z",
        "ModificationTime": "2021-05-25T21:51:53.028688Z"
    }
}
```

Per ulteriori informazioni, consulta Gestione di gruppi di regole e regole in DNS Firewall nella Guida per gli sviluppatori di Amazon Route 53.

- Per i dettagli sull'API, consulta DeleteFirewallRuleGroup AWS CLICommand Reference.

`delete-firewall-rule`

Il seguente esempio di codice mostra come utilizzare`delete-firewall-rule`.

AWS CLI

Come eliminare una regola del firewall

L'esempio `delete-firewall-rule` seguente elimina una regola del firewall specificata.

```
aws route53resolver delete-firewall-rule \
    --firewall-rule-group-id rslvr-frg-47f93271fexample \
    --firewall-domain-list-id rslvr-fdl-9e956e9ffexample
```

Output:

```
{
    "FirewallRule": {
        "FirewallRuleGroupId": "rslvr-frg-47f93271fexample",
        "FirewallDomainListId": "rslvr-fdl-9e956e9ffexample",
        "Name": "allow-rule",
        "Priority": 102,
        "Action": "ALLOW",
        "CreatorRequestId": "d81e3fb7-020b-415e-939f-EXAMPLE11111",
        "CreationTime": "2021-05-25T21:44:00.346093Z",
        "ModificationTime": "2021-05-25T21:45:59.611600Z"
    }
}
```

Per ulteriori informazioni, consulta Gestione di gruppi di regole e regole in DNS Firewall nella Guida per gli sviluppatori di Amazon Route 53.

- Per i dettagli sull'API, consulta DeleteFirewallRule AWS CLICommand Reference.

`delete-resolver-endpoint`

Il seguente esempio di codice mostra come utilizzare`delete-resolver-endpoint`.

AWS CLI

Come eliminare un endpoint del risolutore

L'esempio `delete-resolver-endpoint` seguente elimina l'endpoint specificato.

Importante Se elimini un endpoint in entrata, le query DNS dalla rete non vengono più inoltrate al risolutore nel VPC specificato nell'endpoint. Se elimini un endpoint in uscita, Resolver smette di inoltrare query DNS dal tuo VPC alla rete per le regole che specificano l'endpoint in uscita eliminato.

```
aws route53resolver delete-resolver-endpoint \
    --resolver-endpoint-id rslvr-in-497098ad59example
```

Output:

```
{
    "ResolverEndpoint": {
        "Id": "rslvr-in-497098ad59example",
        "CreatorRequestId": "AWSConsole.25.157290example",
        "Arn": "arn:aws:route53resolver:us-west-2:111122223333:resolver-endpoint/
rslvr-in-497098ad59example",
        "Name": "my-inbound-endpoint",
        "SecurityGroupIds": [
            "sg-05cd7b25d6example"
        ],
        "Direction": "INBOUND",
        "IpAddressCount": 5,
        "HostVPCId": "vpc-304bexam",
        "Status": "DELETING",
        "StatusMessage": "[Trace id: 1-5dc5b658-811b5be0922bbc382example] Deleting
 ResolverEndpoint.",
        "CreationTime": "2020-01-01T23:25:45.538Z",
        "ModificationTime": "2020-01-02T23:25:45.538Z"
    }
```

```
}
```

- Per i dettagli sull'API, consulta DeleteResolverEndpoint AWS CLICommand Reference.

`delete-resolver-rule`

Il seguente esempio di codice mostra come utilizzare`delete-resolver-rule`.

AWS CLI

Come eliminare una regola del risolutore

L'esempio `delete-resolver-rule` seguente elimina la regola specificata.

Nota Se una regola è associata a una regola VPCs, è necessario prima dissociare la regola dalla regola VPCs prima di poterla eliminare.

```
aws route53resolver delete-resolver-rule \
    --resolver-rule-id rslvr-rr-5b3809426bexample
```

Output:

```
{
    "ResolverRule": {
        "Id": "rslvr-rr-5b3809426bexample",
        "CreatorRequestId": "2020-01-03-18:47",
        "Arn": "arn:aws:route53resolver:us-west-2:111122223333:resolver-rule/rslvr-
rr-5b3809426bexample",
        "DomainName": "zenith.example.com.",
        "Status": "DELETING",
        "StatusMessage": "[Trace id: 1-5dc5e05b-602e67b052cb74f05example] Deleting
 Resolver Rule.",
        "RuleType": "FORWARD",
        "Name": "my-resolver-rule",
        "TargetIps": [
            {
                "Ip": "192.0.2.50",
                "Port": 53
            }
        ],
        "ResolverEndpointId": "rslvr-out-d5e5920e3example",
        "OwnerId": "111122223333",
```

```
        "ShareStatus": "NOT_SHARED"
    }
}
```

- Per i dettagli sull'API, consulta DeleteResolverRule AWS CLICommand Reference.

`disassociate-firewall-rule-group`

Il seguente esempio di codice mostra come utilizzare`disassociate-firewall-rule-group`.

AWS CLI

Come dissociare un gruppo di regole del firewall da un VPC

L'esempio `disassociate-firewall-rule-group` seguente dissocia un gruppo di regole di DNS Firewall da un VPC Amazon.

```
aws route53resolver disassociate-firewall-rule-group \
    --firewall-rule-group-association-id rslvr-frgassoc-57e8873d7example
```

Output:

```
{
    "FirewallRuleGroupAssociation": {
        "Id": "rslvr-frgassoc-57e8873d7example",
        "Arn": "arn:aws:route53resolver:us-west-2:123456789012:firewall-rule-group-
association/rslvr-frgassoc-57e8873d7example",
        "FirewallRuleGroupId": "rslvr-frg-47f93271fexample",
        "VpcId": "vpc-31e92222",
        "Name": "test-association",
        "Priority": 103,
        "MutationProtection": "DISABLED",
        "Status": "DELETING",
        "StatusMessage": "Deleting the Firewall Rule Group Association",
        "CreatorRequestId": "2ca1a304-32b3-4f5f-bc4c-EXAMPLE11111",
        "CreationTime": "2021-05-25T21:47:48.755768Z",
        "ModificationTime": "2021-05-25T21:51:02.377887Z"
    }
}
```

Per ulteriori informazioni, consulta Gestione delle associazioni tra il VPC e i gruppi di regole DNS Firewall per il risolutore Route 53 nella Guida per gli sviluppatori di Amazon Route 53.

- Per i dettagli sull'API, consulta DisassociateFirewallRuleGroup AWS CLICommand Reference.

`disassociate-resolver-endpoint-ip-address`

Il seguente esempio di codice mostra come utilizzare`disassociate-resolver-endpoint-ip-address`.

AWS CLI

Come dissociare un indirizzo IP da un endpoint del risolutore

L'esempio `disassociate-resolver-endpoint-ip-address` seguente rimuove un indirizzo IP da un endpoint del risolutore in entrata o in uscita specificato.

Nota Un endpoint deve avere almeno due indirizzi IP. Se un endpoint ha attualmente solo due indirizzi IP e desideri sostituire un indirizzo con un altro indirizzo, devi prima utilizzare associate-resolver-endpoint-ip-address per associare il nuovo indirizzo IP. Una volta eseguita questa operazione, puoi dissociare uno degli indirizzi IP originali dall'endpoint.

```
aws route53resolver disassociate-resolver-endpoint-ip-address \
    --resolver-endpoint-id rslvr-in-f9ab8a03f1example \
    --ip-address="SubnetId=subnet-12d8a459,Ip=172.31.40.121"
```

Output:

```
{
    "ResolverEndpoint": {
        "Id": "rslvr-in-f9ab8a03f1example",
        "CreatorRequestId": "2020-01-01-18:47",
        "Arn": "arn:aws:route53resolver:us-west-2:111122223333:resolver-endpoint/
rslvr-in-f9ab8a03f1example",
        "Name": "my-inbound-endpoint",
        "SecurityGroupIds": [
            "sg-f62bexam"
        ],
        "Direction": "INBOUND",
        "IpAddressCount": 3,
        "HostVPCId": "vpc-304bexam",
        "Status": "UPDATING",
        "StatusMessage": "Updating the Resolver Endpoint",
        "CreationTime": "2020-01-01T23:02:29.583Z",
```

```
        "ModificationTime": "2020-01-05T23:02:29.583Z"
    }
}
```

- Per i dettagli sull'API, consulta AWS CLI Command DisassociateResolverEndpointIpAddressReference.

`disassociate-resolver-rule`

Il seguente esempio di codice mostra come utilizzare`disassociate-resolver-rule`.

AWS CLI

Come dissociare una regola del risolutore da un VPC Amazon

L'esempio `disassociate-resolver-rule` seguente rimuove l'associazione tra la regola del risolutore specificata e il VPC specificato. Puoi rimuovere l'associazione di una regola da un VPC nei seguenti casi:

Per le query DNS che hanno origine in questo VPC, desideri che il risolutore smetta di inoltrare query alla tua rete per il nome di dominio specificato nella regola. Desideri eliminare la regola di inoltro. Se una regola è attualmente associata a una o più regole VPCs, è necessario dissociarla da tutte le regole VPCs prima di poterla eliminare.

```
aws route53resolver disassociate-resolver-rule \
    --resolver-rule-id rslvr-rr-4955cb98ceexample \
    --vpc-id vpc-304bexam
```

Output:

```
{
    "ResolverRuleAssociation": {
        "Id": "rslvr-rrassoc-322f4e8b9cexample",
        "ResolverRuleId": "rslvr-rr-4955cb98ceexample",
        "Name": "my-resolver-rule-association",
        "VPCId": "vpc-304bexam",
        "Status": "DELETING",
        "StatusMessage": "[Trace id: 1-5dc5ffa2-a26c38004c1f94006example] Deleting
 Association"
    }
```

```
}
```

- Per i dettagli sull'API, consulta DisassociateResolverRule AWS CLICommand Reference.

`get-firewall-config`

Il seguente esempio di codice mostra come utilizzare`get-firewall-config`.

AWS CLI

Come ottenere una configurazione del firewall per un VPC

L'esempio `get-firewall-config` seguente recupera il comportamento di DNS Firewall per il VPC specificato.

```
aws route53resolver get-firewall-config \
    --resource-id vpc-31e92222
```

Output:

```
{
    "FirewallConfig": {
        "Id": "rslvr-fc-86016850cexample",
        "ResourceId": "vpc-31e9222",
        "OwnerId": "123456789012",
        "FirewallFailOpen": "DISABLED"
    }
}
```

Per ulteriori informazioni, consulta Configurazione del VPC di DNS Firewall nella Guida per gli sviluppatori di Amazon Route 53.

- Per i dettagli sull'API, consulta GetFirewallConfig AWS CLICommand Reference.

`get-firewall-domain-list`

Il seguente esempio di codice mostra come utilizzare`get-firewall-domain-list`.

AWS CLI

Come ottenere un elenco di domini di DNS Firewall per il risolutore Route 53

L'esempio `get-firewall-domain-list` seguente recupera l'elenco dei domini con l'ID specificato.

```
aws route53resolver get-firewall-domain-list \
    --firewall-domain-list-id rslvr-fdl-42b60677cexample
```

Output:

```
{
    "FirewallDomainList": {
        "Id": "rslvr-fdl-9e956e9ffexample",
        "Arn": "arn:aws:route53resolver:us-west-2:123457689012:firewall-domain-list/
rslvr-fdl-42b60677cexample",
        "Name": "test",
        "DomainCount": 0,
        "Status": "COMPLETE",
        "StatusMessage": "Created Firewall Domain List",
        "CreatorRequestId": "my-request-id",
        "CreationTime": "2021-05-25T15:55:51.115365Z",
        "ModificationTime": "2021-05-25T15:55:51.115365Z"
    }
}
```

Per ulteriori informazioni, consulta Gestione degli elenchi di domini personalizzati nella Guida per gli sviluppatori di Amazon Route 53.

- Per i dettagli sull'API, consulta GetFirewallDomainList AWS CLICommand Reference.

`get-firewall-rule-group-association`

Il seguente esempio di codice mostra come utilizzare`get-firewall-rule-group-association`.

AWS CLI

Come ottenere un'associazione di gruppi di regole del firewall

L'esempio `get-firewall-rule-group-association` seguente recupera un'associazione di gruppi di regole del firewall.

```
aws route53resolver get-firewall-rule-group-association \
    --firewall-rule-group-association-id rslvr-frgassoc-57e8873d7example
```

Output:

```
{
    "FirewallRuleGroupAssociation": {
        "Id": "rslvr-frgassoc-57e8873d7example",
        "Arn": "arn:aws:route53resolver:us-west-2:123456789012:firewall-rule-group-
association/rslvr-frgassoc-57e8873d7example",
        "FirewallRuleGroupId": "rslvr-frg-47f93271fexample",
        "VpcId": "vpc-31e92222",
        "Name": "test-association",
        "Priority": 101,
        "MutationProtection": "DISABLED",
        "Status": "COMPLETE",
        "StatusMessage": "Finished rule group association update",
        "CreatorRequestId": "2ca1a304-32b3-4f5f-bc4c-EXAMPLE11111",
        "CreationTime": "2021-05-25T21:47:48.755768Z",
        "ModificationTime": "2021-05-25T21:47:48.755768Z"
    }
}
```

Per ulteriori informazioni, consulta Gestione delle associazioni tra il VPC e i gruppi di regole DNS Firewall per il risolutore Route 53 nella Guida per gli sviluppatori di Amazon Route 53.

- Per i dettagli sull'API, consulta GetFirewallRuleGroupAssociation AWS CLICommand Reference.

`get-firewall-rule-group-policy`

Il seguente esempio di codice mostra come utilizzare`get-firewall-rule-group-policy`.

AWS CLI

Per ottenere una AWS policy IAM

L'`get-firewall-rule-group-policy`esempio seguente ottiene la policy AWS Identity and Access Management (AWS IAM) per la condivisione del gruppo di regole specificato.

```
aws route53resolver get-firewall-rule-group-policy \
    --arn arn:aws:route53resolver:us-west-2:AWS_ACCOUNT_ID:firewall-rule-group/
rslvr-frg-47f93271fexample
```

Output:

```
{
    "FirewallRuleGroupPolicy": "{\"Version\":\"2012-10-17\",
\"Statement\":[{\"Sid\":\"test\",\"Effect\":\"Allow\",\"Principal
\":{\"AWS\":\"arn:aws:iam::AWS_ACCOUNT_ID:root\"},\"Action\":
[\"route53resolver:GetFirewallRuleGroup\",\"route53resolver:ListFirewallRuleGroups
\"],\"Resource\":\"arn:aws:route53resolver:us-east-1:AWS_ACCOUNT_ID:firewall-rule-
group/rslvr-frg-47f93271fexample\"}]}"
}
```

Per ulteriori informazioni, consulta Gestione di gruppi di regole e regole in DNS Firewall nella Guida per gli sviluppatori di Amazon Route 53.

- Per i dettagli sull'API, consulta GetFirewallRuleGroupPolicy AWS CLICommand Reference.

`get-firewall-rule-group`

Il seguente esempio di codice mostra come utilizzare`get-firewall-rule-group`.

AWS CLI

Come ottenere un gruppo di regole del firewall

L'esempio `get-firewall-rule-group` seguente recupera le informazioni su un gruppo di regole di DNS Firewall con l'ID specificato.

```
aws route53resolver get-firewall-rule-group \
    --firewall-rule-group-id rslvr-frg-47f93271fexample
```

Output:

```
{
    "FirewallRuleGroup": {
        "Id": "rslvr-frg-47f93271fexample",
        "Arn": "arn:aws:route53resolver:us-west-2:123456789012:firewall-rule-group/
rslvr-frg-47f93271fexample",
        "Name": "test",
        "RuleCount": 0,
        "Status": "COMPLETE",
        "StatusMessage": "Created Firewall Rule Group",
        "OwnerId": "123456789012",
        "CreatorRequestId": "my-request-id",
```

```
        "ShareStatus": "NOT_SHARED",
        "CreationTime": "2021-05-25T18:59:26.490017Z",
        "ModificationTime": "2021-05-25T18:59:26.490017Z"
    }
}
```

Per ulteriori informazioni, consulta Gestione di gruppi di regole e regole in DNS Firewall nella Guida per gli sviluppatori di Amazon Route 53.

- Per i dettagli sull'API, consulta GetFirewallRuleGroup AWS CLICommand Reference.

`get-resolver-endpoint`

Il seguente esempio di codice mostra come utilizzare`get-resolver-endpoint`.

AWS CLI

Come ottenere informazioni su un endpoint del risolutore

L'esempio `get-resolver-endpoint` seguente mostra i dettagli relativi all'endpoint in uscita specificato. È possibile utilizzare `get-resolver-endpoint` sia per gli endpoint in entrata sia per quelli in uscita specificando l'ID dell'endpoint appropriato.

```
aws route53resolver get-resolver-endpoint \
    --resolver-endpoint-id rslvr-out-d5e5920e37example
```

Output:

```
{
    "ResolverEndpoint": {
        "Id": "rslvr-out-d5e5920e37example",
        "CreatorRequestId": "2020-01-01-18:47",
        "Arn": "arn:aws:route53resolver:us-west-2:111122223333:resolver-endpoint/
rslvr-out-d5e5920e37example",
        "Name": "my-outbound-endpoint",
        "SecurityGroupIds": [
            "sg-05cd7b25d6example"
        ],
        "Direction": "OUTBOUND",
        "IpAddressCount": 2,
        "HostVPCId": "vpc-304bexam",
```

```
        "Status": "OPERATIONAL",
        "StatusMessage": "This Resolver Endpoint is operational.",
        "CreationTime": "2020-01-01T23:50:50.979Z",
        "ModificationTime": "2020-01-02T23:50:50.979Z"
    }
}
```

Per ulteriori informazioni, consulta Valori che specifichi durante la creazione o la modifica di endpoint in entrata nella Guida per gli sviluppatori di Amazon Route 53.

- Per i dettagli sull'API, consulta GetResolverEndpoint AWS CLICommand Reference.

`get-resolver-rule-association`

Il seguente esempio di codice mostra come utilizzare`get-resolver-rule-association`.

AWS CLI

Come ottenere informazioni sull'associazione tra una regola del risolutore e un VPC

L'esempio `get-resolver-rule-association` seguente mostra i dettagli sull'associazione tra una regola del risolutore specificata e un VPC. Associate una regola del resolver e un VPC utilizzando. associate-resolver-rule

```
aws route53resolver get-resolver-rule-association \
    --resolver-rule-association-id rslvr-rrassoc-d61cbb2c8bexample
```

Output:

```
{
    "ResolverRuleAssociation": {
        "Id": "rslvr-rrassoc-d61cbb2c8bexample",
        "ResolverRuleId": "rslvr-rr-42b60677c0example",
        "Name": "my-resolver-rule-association",
        "VPCId": "vpc-304bexam",
        "Status": "COMPLETE",
        "StatusMessage": ""
    }
}
```

- Per i dettagli sull'API, consulta Command GetResolverRuleAssociationReference AWS CLI .

`get-resolver-rule`

Il seguente esempio di codice mostra come utilizzare`get-resolver-rule`.

AWS CLI

Come ottenere informazioni su una regola del risolutore

L'esempio `get-resolver-rule` seguente mostra i dettagli sulla regola del risolutore specificata, ad esempio il nome del dominio per cui la regola inoltra le query DNS e l'ID dell'endpoint del risolutore in uscita a cui è associata la regola.

```
aws route53resolver get-resolver-rule \
    --resolver-rule-id rslvr-rr-42b60677c0example
```

Output:

```
{
    "ResolverRule": {
        "Id": "rslvr-rr-42b60677c0example",
        "CreatorRequestId": "2020-01-01-18:47",
        "Arn": "arn:aws:route53resolver:us-west-2:111122223333:resolver-rule/rslvr-
rr-42b60677c0example",
        "DomainName": "example.com.",
        "Status": "COMPLETE",
        "StatusMessage": "[Trace id: 1-5dc4b177-ff1d9d001a0f80005example]
 Successfully created Resolver Rule.",
        "RuleType": "FORWARD",
        "Name": "my-rule",
        "TargetIps": [
            {
                "Ip": "192.0.2.45",
                "Port": 53
            }
        ],
        "ResolverEndpointId": "rslvr-out-d5e5920e37example",
        "OwnerId": "111122223333",
        "ShareStatus": "NOT_SHARED"
    }
}
```

Per ulteriori informazioni, consulta Valori che specifichi durante la creazione o la modifica delle regole nella Guida per gli sviluppatori di Amazon Route 53.

- Per i dettagli sull'API, consulta GetResolverRule AWS CLICommand Reference.

`import-firewall-domains`

Il seguente esempio di codice mostra come utilizzare`import-firewall-domains`.

AWS CLI

Come importare domini in un elenco di domini

L'esempio `import-firewall-domains` seguente importa un set di domini da un file in un elenco di domini di DNS Firewall specificato.

```
aws route53resolver import-firewall-domains \
    --firewall-domain-list-id rslvr-fdl-d61cbb2cbexample \
    --operation REPLACE \
    --domain-file-url s3://PATH/TO/YOUR/FILE
```

Output:

```
{
    "Id": "rslvr-fdl-d61cbb2cbexample",
    "Name": "test",
    "Status": "IMPORTING",
    "StatusMessage": "Importing domains from provided file."
}
```

Per ulteriori informazioni, consulta Gestione degli elenchi di domini personalizzati nella Guida per gli sviluppatori di Amazon Route 53.

- Per i dettagli sull'API, consulta ImportFirewallDomains AWS CLICommand Reference.

`list-firewall-configs`

Il seguente esempio di codice mostra come utilizzare`list-firewall-configs`.

AWS CLI

Come elencare le configurazioni del firewall

L'esempio `list-firewall-configs` seguente elenca le configurazioni di DNS Firewall.

```
aws route53resolver list-firewall-configs
```

Output:

```
{
    "FirewallConfigs": [
        {
            "Id": "rslvr-fc-86016850cexample",
            "ResourceId": "vpc-31e92222",
            "OwnerId": "123456789012",
            "FirewallFailOpen": "DISABLED"
        }
    ]
}
```

Per ulteriori informazioni, consulta Configurazione del VPC di DNS Firewall nella Guida per gli sviluppatori di Amazon Route 53.

- Per i dettagli sull'API, consulta ListFirewallConfigs AWS CLICommand Reference.

`list-firewall-domain-lists`

Il seguente esempio di codice mostra come utilizzare`list-firewall-domain-lists`.

AWS CLI

Come elencare tutti gli elenchi di domini di DNS Firewall per il risolutore Route 53

L'esempio `list-firewall-domain-lists` seguente elenca tutti gli elenchi di domini.

```
aws route53resolver list-firewall-domain-lists
```

Output:

```
{
    "FirewallDomainLists": [
        {
            "Id": "rslvr-fdl-2c46f2ecfexample",
            "Arn": "arn:aws:route53resolver:us-west-2:123456789012:firewall-domain-
list/rslvr-fdl-2c46f2ecfexample",
            "Name": "AWSManagedDomainsMalwareDomainList",
            "CreatorRequestId": "AWSManagedDomainsMalwareDomainList",
```

```
            "ManagedOwnerName": "Route 53 Resolver DNS Firewall"
        },
        {
            "Id": "rslvr-fdl-aa970e9e1example",
            "Arn": "arn:aws:route53resolver:us-west-2:123456789012:firewall-domain-
list/rslvr-fdl-aa970e9e1example",
            "Name": "AWSManagedDomainsBotnetCommandandControl",
            "CreatorRequestId": "AWSManagedDomainsBotnetCommandandControl",
            "ManagedOwnerName": "Route 53 Resolver DNS Firewall"
        },
        {
            "Id": "rslvr-fdl-42b60677cexample",
            "Arn": "arn:aws:route53resolver:us-west-2:123456789111:firewall-domain-
list/rslvr-fdl-42b60677cexample",
            "Name": "test",
            "CreatorRequestId": "my-request-id"
        }
    ]
}
```

Per ulteriori informazioni, consulta Elenchi di domini di DNS Firewall per il risolutore Route 53 nella Guida per gli sviluppatori di Amazon Route 53.

- Per i dettagli sull'API, consulta ListFirewallDomainLists AWS CLICommand Reference.

`list-firewall-domains`

Il seguente esempio di codice mostra come utilizzare`list-firewall-domains`.

AWS CLI

Come elencare i domini inclusi in un elenco di domini

L'esempio `list-firewall-domains` seguente elenca i domini inclusi in un elenco di domini di DNS Firewall specificato.

```
aws route53resolver list-firewall-domains \
    --firewall-domain-list-id rslvr-fdl-d61cbb2cbexample
```

Output:

```
{
```

```
    "Domains": [
        "test1.com.",
        "test2.com.",
        "test3.com."
    ]
}
```

Per ulteriori informazioni, consulta Gestione degli elenchi di domini personalizzati nella Guida per gli sviluppatori di Amazon Route 53.

- Per i dettagli sull'API, consulta ListFirewallDomains AWS CLICommand Reference.

`list-firewall-rule-group-associations`

Il seguente esempio di codice mostra come utilizzare`list-firewall-rule-group-associations`.

AWS CLI

Come elencare le associazioni dei gruppi di regole di DNS Firewall

L'`list-firewall-rule-group-associations`esempio seguente elenca le associazioni dei gruppi di regole del firewall DNS con Amazon VPCs.

```
aws route53resolver list-firewall-rule-group-associations
```

Output:

```
{
    "FirewallRuleGroupAssociations": [
        {
            "Id": "rslvr-frgassoc-57e8873d7example",
            "Arn": "arn:aws:route53resolver:us-west-2:123456789012:firewall-rule-
group-association/rslvr-frgassoc-57e8873d7example",
            "FirewallRuleGroupId": "rslvr-frg-47f93271fexample",
            "VpcId": "vpc-31e92222",
            "Name": "test-association",
            "Priority": 101,
            "MutationProtection": "DISABLED",
            "Status": "UPDATING",
            "StatusMessage": "Creating Firewall Rule Group Association",
            "CreatorRequestId": "2ca1a304-32b3-4f5f-bc4c-EXAMPLE11111",
```

```
            "CreationTime": "2021-05-25T21:47:48.755768Z",
            "ModificationTime": "2021-05-25T21:47:48.755768Z"
        }
    ]
}
```

Per ulteriori informazioni, consulta Gestione delle associazioni tra il VPC e il gruppo di regole DNS Firewall per il risolutore Route 53 nella Guida per gli sviluppatori di Amazon Route 53.

- Per i dettagli sull'API, consulta ListFirewallRuleGroupAssociations AWS CLICommand Reference.

`list-firewall-rule-groups`

Il seguente esempio di codice mostra come utilizzare`list-firewall-rule-groups`.

AWS CLI

Come ottenere un elenco dei gruppi di regole del firewall

L'esempio `list-firewall-rule-groups` seguente elenca i gruppi di regole di DNS Firewall.

```
aws route53resolver list-firewall-rule-groups
```

Output:

```
{
    "FirewallRuleGroups": [
        {
            "Id": "rslvr-frg-47f93271fexample",
            "Arn": "arn:aws:route53resolver:us-west-2:123456789012:firewall-rule-
group/rslvr-frg-47f93271fexample",
            "Name": "test",
            "OwnerId": "123456789012",
            "CreatorRequestId": "my-request-id",
            "ShareStatus": "NOT_SHARED"
        }
    ]
}
```

Per ulteriori informazioni, consulta Gestione di gruppi di regole e regole in DNS Firewall nella Guida per gli sviluppatori di Amazon Route 53.

- Per i dettagli sull'API, consulta ListFirewallRuleGroups AWS CLICommand Reference.

`list-firewall-rules`

Il seguente esempio di codice mostra come utilizzare`list-firewall-rules`.

AWS CLI

Come elencare le regole del firewall

L'esempio `list-firewall-rules` seguente elenca tutte le regole di DNS Firewall incluse in un gruppo di regole del firewall.

```
aws route53resolver list-firewall-rules \
    --firewall-rule-group-id rslvr-frg-47f93271fexample
```

Output:

```
{
    "FirewallRules": [
        {
            "FirewallRuleGroupId": "rslvr-frg-47f93271fexample",
            "FirewallDomainListId": "rslvr-fdl-9e956e9ffexample",
            "Name": "allow-rule",
            "Priority": 101,
            "Action": "ALLOW",
            "CreatorRequestId": "d81e3fb7-020b-415e-939f-EXAMPLE11111",
            "CreationTime": "2021-05-25T21:44:00.346093Z",
            "ModificationTime": "2021-05-25T21:44:00.346093Z"
        }
    ]
}
```

Per ulteriori informazioni, consulta Gestione di gruppi di regole e regole in DNS Firewall nella Guida per gli sviluppatori di Amazon Route 53.

- Per i dettagli sull'API, consulta ListFirewallRules AWS CLICommand Reference.

`list-resolver-endpoint-ip-addresses`

Il seguente esempio di codice mostra come utilizzare`list-resolver-endpoint-ip-addresses`.

AWS CLI

Come elencare gli indirizzi IP per un endpoint in entrata o in uscita specificato

L'esempio `list-resolver-endpoint-ip-addresses` seguente elenca le informazioni sugli indirizzi IP associati all'endpoint in entrata `rslvr-in-f9ab8a03f1example`. È possibile utilizzare `list-resolver-endpoint-ip-addresses` anche per gli endpoint in uscita specificando l'ID dell'endpoint appropriato.

```
aws route53resolver list-resolver-endpoint-ip-addresses \
    --resolver-endpoint-id rslvr-in-f9ab8a03f1example
```

Output:

```
{
    "MaxResults": 10,
    "IpAddresses": [
        {
            "IpId": "rni-1de60cdbfeexample",
            "SubnetId": "subnet-ba47exam",
            "Ip": "192.0.2.44",
            "Status": "ATTACHED",
            "StatusMessage": "This IP address is operational.",
            "CreationTime": "2020-01-03T23:02:29.587Z",
            "ModificationTime": "2020-01-03T23:03:05.555Z"
        },
        {
            "IpId": "rni-aac7085e38example",
            "SubnetId": "subnet-12d8exam",
            "Ip": "192.0.2.45",
            "Status": "ATTACHED",
            "StatusMessage": "This IP address is operational.",
            "CreationTime": "2020-01-03T23:02:29.593Z",
            "ModificationTime": "2020-01-03T23:02:55.060Z"
        }
    ]
}
```

Per ulteriori informazioni sui valori nell'output, consulta Valori che specifichi durante la creazione o la modifica di endpoint in entrata e Valori che specifichi durante la creazione o la modifica di endpoint in uscita, entrambi nella Guida per gli sviluppatori di Amazon Route 53.

- Per i dettagli sull'API, consulta ListResolverEndpointIpAddresses AWS CLICommand Reference.

`list-resolver-endpoints`

Il seguente esempio di codice mostra come utilizzare`list-resolver-endpoints`.

AWS CLI

Per elencare gli endpoint Resolver in una regione AWS

L'esempio `list-resolver-endpoints` seguente elenca gli endpoint del risolutore in entrata e in uscita presenti nell'account corrente.

```
aws route53resolver list-resolver-endpoints
```

Output:

```
{
    "MaxResults": 10,
    "ResolverEndpoints": [
        {
            "Id": "rslvr-in-497098ad59example",
            "CreatorRequestId": "2020-01-01-18:47",
            "Arn": "arn:aws:route53resolver:us-west-2:111122223333:resolver-
endpoint/rslvr-in-497098ad59example",
            "Name": "my-inbound-endpoint",
            "SecurityGroupIds": [
                "sg-05cd7b25d6example"
            ],
            "Direction": "INBOUND",
            "IpAddressCount": 2,
            "HostVPCId": "vpc-304bexam",
            "Status": "OPERATIONAL",
            "StatusMessage": "This Resolver Endpoint is operational.",
            "CreationTime": "2020-01-01T23:25:45.538Z",
            "ModificationTime": "2020-01-01T23:25:45.538Z"
        },
        {
            "Id": "rslvr-out-d5e5920e37example",
            "CreatorRequestId": "2020-01-01-18:48",
```

```
            "Arn": "arn:aws:route53resolver:us-west-2:111122223333:resolver-
endpoint/rslvr-out-d5e5920e37example",
            "Name": "my-outbound-endpoint",
            "SecurityGroupIds": [
                "sg-05cd7b25d6example"
            ],
            "Direction": "OUTBOUND",
            "IpAddressCount": 2,
            "HostVPCId": "vpc-304bexam",
            "Status": "OPERATIONAL",
            "StatusMessage": "This Resolver Endpoint is operational.",
            "CreationTime": "2020-01-01T23:50:50.979Z",
            "ModificationTime": "2020-01-01T23:50:50.979Z"
        }
    ]
}
```

- Per i dettagli sull'API, consulta Command ListResolverEndpointsReference AWS CLI .

`list-resolver-rule-associations`

Il seguente esempio di codice mostra come utilizzare`list-resolver-rule-associations`.

AWS CLI

Per elencare le associazioni tra le regole del Resolver e VPCs

L'`list-resolver-rule-associations`esempio seguente elenca le associazioni tra le regole del resolver e quelle VPCs nell'account corrente. AWS

```
aws route53resolver list-resolver-rule-associations
```

Output:

```
{
    "MaxResults": 30,
    "ResolverRuleAssociations": [
        {
            "Id": "rslvr-autodefined-assoc-vpc-304bexam-internet-resolver",
            "ResolverRuleId": "rslvr-autodefined-rr-internet-resolver",
            "Name": "System Rule Association",
            "VPCId": "vpc-304bexam",
```

```
            "Status": "COMPLETE",
            "StatusMessage": ""
        },
        {
            "Id": "rslvr-rrassoc-d61cbb2c8bexample",
            "ResolverRuleId": "rslvr-rr-42b60677c0example",
            "Name": "my-resolver-rule-association",
            "VPCId": "vpc-304bexam",
            "Status": "COMPLETE",
            "StatusMessage": ""
        }
    ]
}
```

Per ulteriori informazioni, consulta How Route 53 Resolver inoltra le query DNS dalla tua rete VPCs alla tua rete nella Amazon Route 53 Developer Guide.

- Per i dettagli sull'API, consulta Command Reference. ListResolverRuleAssociationsAWS CLI

`list-resolver-rules`

Il seguente esempio di codice mostra come utilizzare`list-resolver-rules`.

AWS CLI

Come elencare le regole del risolutore

L'`list-resolver-rules`esempio seguente elenca tutte le regole Resolver nell'account corrente AWS .

```
aws route53resolver list-resolver-rules
```

Output:

```
{
    "MaxResults": 30,
    "ResolverRules": [
        {
            "Id": "rslvr-autodefined-rr-internet-resolver",
            "CreatorRequestId": "",
            "Arn": "arn:aws:route53resolver:us-west-2::autodefined-rule/rslvr-
autodefined-rr-internet-resolver",
```

```
            "DomainName": ".",
            "Status": "COMPLETE",
            "RuleType": "RECURSIVE",
            "Name": "Internet Resolver",
            "OwnerId": "Route 53 Resolver",
            "ShareStatus": "NOT_SHARED"
        },
        {
            "Id": "rslvr-rr-42b60677c0example",
            "CreatorRequestId": "2020-01-01-18:47",
            "Arn": "arn:aws:route53resolver:us-west-2:111122223333:resolver-rule/
rslvr-rr-42b60677c0bc4e299",
            "DomainName": "example.com.",
            "Status": "COMPLETE",
            "StatusMessage": "[Trace id: 1-5dc4b177-ff1d9d001a0f80005example]
 Successfully created Resolver Rule.",
            "RuleType": "FORWARD",
            "Name": "my-rule",
            "TargetIps": [
                {
                    "Ip": "192.0.2.45",
                    "Port": 53
                }
            ],
            "ResolverEndpointId": "rslvr-out-d5e5920e37example",
            "OwnerId": "111122223333",
            "ShareStatus": "NOT_SHARED"
        }
    ]
}
```

Per ulteriori informazioni, consulta How Route 53 Resolver inoltra le query DNS dalla tua rete VPCs alla tua rete nella Amazon Route 53 Developer Guide.

- Per i dettagli sull'API, consulta Command Reference. ListResolverRulesAWS CLI

`list-tags-for-resource`

Il seguente esempio di codice mostra come utilizzare`list-tags-for-resource`.

AWS CLI

Come elencare i tag per una risorsa del risolutore

L'esempio `list-tags-for-resource` seguente elenca i tag assegnati alla regola del risolutore specificata.

```
aws route53resolver list-tags-for-resource \
    --resource-arn "arn:aws:route53resolver:us-west-2:111122223333:resolver-rule/
rslvr-rr-42b60677c0example"
```

Output:

```
{
    "Tags": [
        {
            "Key": "my-key-1",
            "Value": "my-value-1"
        },
        {
            "Key": "my-key-2",
            "Value": "my-value-2"
        }
    ]
}
```

Per informazioni sull'utilizzo dei tag per l'allocazione dei costi, consulta Utilizzo dei tag per l'allocazione dei costi nella Guida per l'utente di Gestione costi e fatturazione AWS .

- Per i dettagli sull'API, consulta ListTagsForResource AWS CLICommand Reference.

`put-firewall-rule-group-policy`

Il seguente esempio di codice mostra come utilizzare`put-firewall-rule-group-policy`.

AWS CLI

Per allegare una policy AWS IAM per condividere una policy del gruppo di regole del firewall

L'`put-firewall-rule-group-policy`esempio seguente allega una policy di AWS Identity and Access Management (AWS IAM) per la condivisione del gruppo di regole.

```
aws route53resolver put-firewall-rule-group-policy \
    --firewall-rule-group-policy "{\"Version\":\"2012-10-17\",
\"Statement\":[{\"Sid\":\"test\",\"Effect\":\"Allow\",\"Principal
```

```
\":{\"AWS\":\"arn:aws:iam::AWS_ACCOUNT_ID:root\"},\"Action\":
[\"route53resolver:GetFirewallRuleGroup\",\"route53resolver:ListFirewallRuleGroups
\"],\"Resource\":\"arn:aws:route53resolver:us-east-1:AWS_ACCOUNT_ID:firewall-rule-
group/rslvr-frg-47f93271fexample\"}]}"
```

Output:

```
{
    "ReturnValue": true
}
```

Per ulteriori informazioni, consulta Gestione di gruppi di regole e regole in DNS Firewall nella Guida per gli sviluppatori di Amazon Route 53.

- Per i dettagli sull'API, consulta PutFirewallRuleGroupPolicy AWS CLICommand Reference.

`put-resolver-rule-policy`

Il seguente esempio di codice mostra come utilizzare`put-resolver-rule-policy`.

AWS CLI

Per condividere una regola Resolver con un altro account AWS

L'`put-resolver-rule-policy`esempio seguente specifica una regola Resolver che si desidera condividere con un altro AWS account, l'account con cui si desidera condividere la regola e le operazioni relative alle regole che si desidera che l'account sia in grado di eseguire sulle regole.

Nota È necessario eseguire questo comando utilizzando le credenziali dello stesso account che ha creato la regola.

```
aws route53resolver put-resolver-rule-policy \
    --region us-east-1 \
    --arn "arn:aws:route53resolver:us-east-1:111122223333:resolver-rule/rslvr-
rr-42b60677c0example" \
    --resolver-rule-policy "{\"Version\": \"2012-10-17\", \
        \"Statement\": [ { \
        \"Effect\" : \"Allow\", \
        \"Principal\" : {\"AWS\" : \"444455556666\" }, \
        \"Action\" : [ \
```

```
            \"route53resolver:GetResolverRule\", \
            \"route53resolver:AssociateResolverRule\", \
            \"route53resolver:DisassociateResolverRule\", \
            \"route53resolver:ListResolverRules\", \
            \"route53resolver:ListResolverRuleAssociations\" ], \
        \"Resource\" : [ \"arn:aws:route53resolver:us-east-1:111122223333:resolver-
rule/rslvr-rr-42b60677c0example\" ] } ] }"
```

Output:

```
{
    "ReturnValue": true
}
```

Dopo l'esecuzione di `put-resolver-rule-policy`, puoi eseguire i due comandi seguenti di Resource Access Manager (RAM). Devi utilizzare l'account con cui vuoi condividere la regola:

`get-resource-share-invitations` restituisce il valore `resourceShareInvitationArn`. Questo valore è necessario per accettare l'invito a utilizzare la regola condivisa. `accept-resource-share-invitation` accetta l'invito a utilizzare la regola condivisa.

Per ulteriori informazioni, consulta la seguente documentazione:

get-resource-share-invitationsaccept-resource-share-invitationsCondivisione delle regole di inoltro con altri AWS account e utilizzo di regole condivise nella Amazon Route 53 Developer Guide

- Per i dettagli sull'API, consulta AWS CLI Command PutResolverRulePolicyReference.

`tag-resource`

Il seguente esempio di codice mostra come utilizzare`tag-resource`.

AWS CLI

Come associare tag a una risorsa del risolutore

L'`tag-resource`esempio seguente associa due key/value coppie di tag alla regola Resolver specificata.

```
aws route53resolver tag-resource \
```

```
    --resource-arn "arn:aws:route53resolver:us-west-2:111122223333:resolver-rule/
rslvr-rr-42b60677c0example" \
    --tags "Key=my-key-1,Value=my-value-1" "Key=my-key-2,Value=my-value-2"
```

Questo comando non produce alcun output.

Per informazioni sull'utilizzo dei tag per l'allocazione dei costi, consulta Utilizzo dei tag per l'allocazione dei costi nella Guida per l'utente di Gestione costi e fatturazione AWS .

- Per i dettagli sull'API, vedere TagResourcein AWS CLI Command Reference.

`untag-resource`

Il seguente esempio di codice mostra come utilizzare`untag-resource`.

AWS CLI

Come rimuovere i tag da una risorsa del risolutore

L'esempio `untag-resource` seguente rimuove due tag dalla regola del risolutore specificata.

```
aws route53resolver untag-resource \
    --resource-arn "arn:aws:route53resolver:us-west-2:111122223333:resolver-rule/
rslvr-rr-42b60677c0example" \
    --tag-keys my-key-1 my-key-2
```

Questo comando non produce alcun output. Per confermare che i tag sono stati rimossi, puoi usare list-tags-for-resource.

Per informazioni sull'utilizzo dei tag per l'allocazione dei costi, consulta Utilizzo dei tag per l'allocazione dei costi nella Guida per l'utente di Gestione costi e fatturazione AWS .

- Per i dettagli sull'API, consulta UntagResource AWS CLICommand Reference.

`update-firewall-config`

Il seguente esempio di codice mostra come utilizzare`update-firewall-config`.

AWS CLI

Come aggiornare una configurazione del firewall

L'esempio `update-firewall-config` seguente aggiorna la configurazione di DNS Firewall.

```
aws route53resolver update-firewall-config \
    --resource-id vpc-31e92222 \
    --firewall-fail-open DISABLED
```

Output:

```
{
    "FirewallConfig": {
        "Id": "rslvr-fc-86016850cexample",
        "ResourceId": "vpc-31e92222",
        "OwnerId": "123456789012",
        "FirewallFailOpen": "DISABLED"
    }
}
```

Per ulteriori informazioni, consulta Configurazione del VPC di DNS Firewall nella Guida per gli sviluppatori di Amazon Route 53.

- Per i dettagli sull'API, consulta UpdateFirewallConfig AWS CLICommand Reference.

`update-firewall-domains`

Il seguente esempio di codice mostra come utilizzare`update-firewall-domains`.

AWS CLI

Come aggiornare un elenco di domini

L'esempio `update-firewall-domains` seguente aggiunge i domini a un elenco di domini con l'ID specificato.

```
aws route53resolver update-firewall-domains \
    --firewall-domain-list-id rslvr-fdl-42b60677cexampleb \
    --operation ADD \
    --domains test1.com test2.com test3.com
```

Output:

```
{
```

```
    "Id": "rslvr-fdl-42b60677cexample",
    "Name": "test",
    "Status": "UPDATING",
    "StatusMessage": "Updating the Firewall Domain List"
}
```

Per ulteriori informazioni, consulta Gestione degli elenchi di domini personalizzati nella Guida per gli sviluppatori di Amazon Route 53.

- Per i dettagli sull'API, consulta UpdateFirewallDomains AWS CLICommand Reference.

`update-firewall-rule-group-association`

Il seguente esempio di codice mostra come utilizzare`update-firewall-rule-group-association`.

AWS CLI

Come aggiornare un'associazione di gruppi di regole del firewall

L'esempio `update-firewall-rule-group-association` seguente aggiorna un'associazione di gruppi di regole del firewall.

```
aws route53resolver update-firewall-rule-group-association \
    --firewall-rule-group-association-id rslvr-frgassoc-57e8873d7example \
    --priority 103
```

Output:

```
{
    "FirewallRuleGroupAssociation": {
        "Id": "rslvr-frgassoc-57e8873d7example",
        "Arn": "arn:aws:route53resolver:us-west-2:123456789012:firewall-rule-group-
association/rslvr-frgassoc-57e8873d7example",
        "FirewallRuleGroupId": "rslvr-frg-47f93271fexample",
        "VpcId": "vpc-31e92222",
        "Name": "test-association",
        "Priority": 103,
        "MutationProtection": "DISABLED",
        "Status": "UPDATING",
        "StatusMessage": "Updating the Firewall Rule Group Association Attributes",
```

```
        "CreatorRequestId": "2ca1a304-32b3-4f5f-bc4c-EXAMPLE11111",
        "CreationTime": "2021-05-25T21:47:48.755768Z",
        "ModificationTime": "2021-05-25T21:50:09.272569Z"
    }
}
```

Per ulteriori informazioni, consulta Gestione delle associazioni tra il VPC e il gruppo di regole DNS Firewall per il risolutore Route 53 nella Guida per gli sviluppatori di Amazon Route 53.

- Per i dettagli sull'API, consulta UpdateFirewallRuleGroupAssociation AWS CLICommand Reference.

`update-firewall-rule`

Il seguente esempio di codice mostra come utilizzare`update-firewall-rule`.

AWS CLI

Come aggiornare una regola del firewall

L'esempio `update-firewall-rule` seguente aggiorna una regola del firewall con i parametri specificati.

```
aws route53resolver update-firewall-rule \
    --firewall-rule-group-id rslvr-frg-47f93271fexample \
    --firewall-domain-list-id rslvr-fdl-9e956e9ffexample \
    --priority 102
```

Output:

```
{
    "FirewallRule": {
        "FirewallRuleGroupId": "rslvr-frg-47f93271fexample",
        "FirewallDomainListId": "rslvr-fdl-9e956e9ffexample",
        "Name": "allow-rule",
        "Priority": 102,
        "Action": "ALLOW",
        "CreatorRequestId": "d81e3fb7-020b-415e-939f-EXAMPLE11111",
        "CreationTime": "2021-05-25T21:44:00.346093Z",
        "ModificationTime": "2021-05-25T21:45:59.611600Z"
    }
```

```
}
```

Per ulteriori informazioni, consulta Gestione di gruppi di regole e regole in DNS Firewall nella Guida per gli sviluppatori di Amazon Route 53.

- Per i dettagli sull'API, consulta UpdateFirewallRule AWS CLICommand Reference.

`update-resolver-endpoint`

Il seguente esempio di codice mostra come utilizzare`update-resolver-endpoint`.

AWS CLI

Come aggiornare il nome di un endpoint del risolutore

L'esempio `update-resolver-endpoint` seguente aggiorna il nome di un endpoint del risolutore. L'aggiornamento di altri valori non è supportato.

```
aws route53resolver update-resolver-endpoint \
    --resolver-endpoint-id rslvr-in-b5d45e32bdc445f09 \
    --name my-renamed-inbound-endpoint
```

Output:

```
{
    "ResolverEndpoint": {
        "Id": "rslvr-in-b5d45e32bdexample",
        "CreatorRequestId": "2020-01-02-18:48",
        "Arn": "arn:aws:route53resolver:us-west-2:111122223333:resolver-endpoint/
rslvr-in-b5d45e32bdexample",
        "Name": "my-renamed-inbound-endpoint",
        "SecurityGroupIds": [
            "sg-f62bexam"
        ],
        "Direction": "INBOUND",
        "IpAddressCount": 2,
        "HostVPCId": "vpc-304bexam",
        "Status": "OPERATIONAL",
        "StatusMessage": "This Resolver Endpoint is operational.",
        "CreationTime": "2020-01-01T18:33:59.265Z",
        "ModificationTime": "2020-01-08T18:33:59.265Z"
    }
```

```
}
```

- Per i dettagli sull'API, consulta UpdateResolverEndpoint AWS CLICommand Reference.

`update-resolver-rule`

Il seguente esempio di codice mostra come utilizzareu`pdate-resolver-rule`.

AWS CLI

Esempio 1: come aggiornare le impostazioni dell'endpoint del risolutore

L'esempio `update-resolver-rule` seguente aggiorna il nome della regola, gli indirizzi IP sulla rete on-premises a cui vengono inoltrate le query DNS e l'ID dell'endpoint del risolutore in uscita utilizzato per inoltrare le query alla rete.

Nota I valori esistenti di `TargetIps` vengono sovrascritti, quindi è necessario specificare tutti gli indirizzi IP che si desidera assegnare alla regola dopo l'aggiornamento.

```
aws route53resolver update-resolver-rule \
    --resolver-rule-id rslvr-rr-1247fa64f3example \
    --config Name="my-2nd-rule",TargetIps=[{Ip=192.0.2.45,Port=53},
{Ip=192.0.2.46,Port=53}],ResolverEndpointId=rslvr-out-7b89ed0d25example
```

Output:

```
{
    "ResolverRule": {
        "Id": "rslvr-rr-1247fa64f3example",
        "CreatorRequestId": "2020-01-02-18:47",
        "Arn": "arn:aws:route53resolver:us-west-2:111122223333:resolver-rule/rslvr-
rr-1247fa64f3example",
        "DomainName": "www.example.com.",
        "Status": "COMPLETE",
        "StatusMessage": "[Trace id: 1-5dcc90b9-8a8ee860aba1ebd89example]
 Successfully updated Resolver Rule.",
        "RuleType": "FORWARD",
        "Name": "my-2nd-rule",
        "TargetIps": [
            {
                "Ip": "192.0.2.45",
```

```
                "Port": 53
            },
            {
                "Ip": "192.0.2.46",
                "Port": 53
            }
        ],
        "ResolverEndpointId": "rslvr-out-7b89ed0d25example",
        "OwnerId": "111122223333",
        "ShareStatus": "NOT_SHARED"
    }
}
```

Esempio 2: come aggiornare le impostazioni dell'endpoint del risolutore utilizzando un file per le impostazioni ``config

In alternativa, è possibile includere le impostazioni `config` in un file JSON e quindi specificare tale file nella chiamata a `update-resolver-rule`.

```
aws route53resolver update-resolver-rule \
    --resolver-rule-id rslvr-rr-1247fa64f3example \
    --config file://c:\temp\update-resolver-rule.json
```

Contenuto di `update-resolver-rule.json`.

```
{
    "Name": "my-2nd-rule",
    "TargetIps": [
        {
            "Ip": "192.0.2.45",
            "Port": 53
        },
        {
            "Ip": "192.0.2.46",
            "Port": 53
        }
    ],
    "ResolverEndpointId": "rslvr-out-7b89ed0d25example"
}
```

Per ulteriori informazioni, consulta Valori che specifichi durante la creazione o la modifica delle regole nella Guida per gli sviluppatori di Amazon Route 53.

- Per i dettagli sull'API, consulta UpdateResolverRule AWS CLICommand Reference.

Esempi di utilizzo di Amazon S3 AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando Amazon S3. AWS Command Line Interface

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`abort-multipart-upload`

Il seguente esempio di codice mostra come usare`abort-multipart-upload`.

AWS CLI

Come annullare il caricamento in più parti specificato

Il comando `abort-multipart-upload` seguente annulla un caricamento in più parti per la chiave `multipart/01` nel bucket `amzn-s3-demo-bucket`.

```
aws s3api abort-multipart-upload \
    --bucket amzn-s3-demo-bucket \
    --key multipart/01 \
    --upload-
id dfRtDYU0WWCCcH43C3WFbkRONycyCpTJJvxu2i5GYkZljF.Yxwh6XG7WfS2vC4to6HiV6Yjlx.cph0gtNBtJ8P3UR
```

L'ID di caricamento richiesto da questo comando viene emesso da `create-multipart-upload` e può essere recuperato anche con `list-multipart-uploads`.

- Per i dettagli sull'API, consulta AbortMultipartUpload AWS CLICommand Reference.

`complete-multipart-upload`

Il seguente esempio di codice mostra come utilizzare`complete-multipart-upload`.

AWS CLI

Il comando seguente completa un caricamento in più parti per la chiave `multipart/01` nel bucket `amzn-s3-demo-bucket`.

```
aws s3api complete-multipart-upload --multipart-upload file://
mpustruct --bucket amzn-s3-demo-bucket --key 'multipart/01' --upload-
id dfRtDYU0WWCCcH43C3WFbkRONycyCpTJJvxu2i5GYkZljF.Yxwh6XG7WfS2vC4to6HiV6Yjlx.cph0gtNBtJ8P3UR
```

L'ID di caricamento richiesto da questo comando viene emesso da `create-multipart-upload` e può essere recuperato anche con `list-multipart-uploads`.

L'opzione di caricamento in più parti utilizzato nel precedente comando utilizza una struttura JSON che descrive le parti del caricamento in più parti che devono essere riassemblate nel file completo. In questo esempio, il prefisso `file://` è usato per caricare la struttura JSON da un file nella cartella locale chiamato `mpustruct`.

mpustruct:

```
{
  "Parts": [
    {
      "ETag": "e868e0f4719e394144ef36531ee6824c",
      "PartNumber": 1
    },
    {
      "ETag": "6bb2b12753d66fe86da4998aa33fffb0",
      "PartNumber": 2
    },
    {
      "ETag": "d0a0112e841abec9c9ec83406f0159c8",
      "PartNumber": 3
    }
  ]
}
```

Il ETag valore di ogni parte è upload, viene emesso ogni volta che si carica una parte utilizzando il `upload-part` comando e può anche essere recuperato chiamando `list-parts` o calcolato prendendo il MD5 checksum di ogni parte.

Output:

```
{
    "ETag": "\"3944a9f7a4faab7f78788ff6210f63f0-3\"",
    "Bucket": "amzn-s3-demo-bucket",
    "Location": "https://amzn-s3-demo-bucket.s3.amazonaws.com/multipart%2F01",
    "Key": "multipart/01"
}
```

- Per i dettagli sull'API, consulta AWS CLI Command CompleteMultipartUploadReference.

`copy-object`

Il seguente esempio di codice mostra come utilizzare`copy-object`.

AWS CLI

Il comando seguente copia un oggetto da `bucket-1` a `bucket-2`.

```
aws s3api copy-object --copy-source bucket-1/test.txt --key test.txt --
bucket bucket-2
```

Output:

```
{
    "CopyObjectResult": {
        "LastModified": "2015-11-10T01:07:25.000Z",
        "ETag": "\"589c8b79c230a6ecd5a7e1d040a9a030\""
    },
    "VersionId": "YdnYvTCVDqRRFA.NFJjy36p0hxifMlkA"
}
```

- Per i dettagli sull'API, consulta CopyObject AWS CLICommand Reference.

`cp`

Il seguente esempio di codice mostra come utilizzare`cp`.

AWS CLI

Esempio 1: copia di un file locale su S3

Il comando cp seguente copia un singolo file in un bucket e una chiave specificati:

```
aws s3 cp test.txt s3://amzn-s3-demo-bucket/test2.txt
```

Output:

```
upload: test.txt to s3://amzn-s3-demo-bucket/test2.txt
```

Esempio 2: copia di un file locale su S3 con una data di scadenza

Il comando cp seguente copia un singolo file in un bucket e in una chiave specificati che scadono al timestamp ISO 8601 specificato:

```
aws s3 cp test.txt s3://amzn-s3-demo-bucket/test2.txt \
    --expires 2014-10-01T20:30:00Z
```

Output:

```
upload: test.txt to s3://amzn-s3-demo-bucket/test2.txt
```

Esempio 3: copia di un file da S3 a S3

Il comando cp seguente copia un singolo oggetto s3 in un bucket e una chiave specificati:

```
aws s3 cp s3://amzn-s3-demo-bucket/test.txt s3://amzn-s3-demo-bucket/test2.txt
```

Output:

```
copy: s3://amzn-s3-demo-bucket/test.txt to s3://amzn-s3-demo-bucket/test2.txt
```

Esempio 4: copia di un oggetto S3 in un file locale

Il comando cp seguente copia localmente un singolo oggetto in un file specificato:

```
aws s3 cp s3://amzn-s3-demo-bucket/test.txt test2.txt
```

Output:

```
download: s3://amzn-s3-demo-bucket/test.txt to test2.txt
```

Esempio 5: copia di un oggetto S3 da un bucket a un altro

Il comando cp seguente copia un singolo oggetto in un bucket specificato mantenendo il nome originale:

```
aws s3 cp s3://amzn-s3-demo-bucket/test.txt s3://amzn-s3-demo-bucket2/
```

Output:

```
copy: s3://amzn-s3-demo-bucket/test.txt to s3://amzn-s3-demo-bucket2/test.txt
```

Esempio 6: copia in modo ricorsivo di oggetti S3 in una directory locale

Se passato con il parametro `--recursive`, il comando cp seguente copia in modo ricorsivo tutti gli oggetti con un prefisso e un bucket specificati in una directory specificata. In questo esempio, il bucket `amzn-s3-demo-bucket` contiene gli oggetti `test1.txt` e `test2.txt`:

```
aws s3 cp s3://amzn-s3-demo-bucket . \
    --recursive
```

Output:

```
download: s3://amzn-s3-demo-bucket/test1.txt to test1.txt
download: s3://amzn-s3-demo-bucket/test2.txt to test2.txt
```

Esempio 7: copia in modo ricorsivo di file locali su S3

Se passato con il parametro `--recursive`, il comando cp seguente copia in modo ricorsivo tutti i file in una directory specificata in un bucket e un prefisso specificati, escludendo alcuni file tramite un parametro `--exclude`. In questo esempio, la directory `myDir` contiene i file `test1.txt` e `test2.jpg`:

```
aws s3 cp myDir s3://amzn-s3-demo-bucket/ \
    --recursive \
```

```
    --exclude "*.jpg"
```

Output:

```
upload: myDir/test1.txt to s3://amzn-s3-demo-bucket/test1.txt
```

Esempio 8: copia in modo ricorsivo di oggetti S3 in un altro bucket

Se passato con il parametro `--recursive`, il comando `cp` seguente copia in modo ricorsivo tutti gli oggetti contenuti in un bucket specificato in un altro bucket, escludendo alcuni oggetti tramite un parametro `--exclude`. In questo esempio, il bucket `amzn-s3-demo-bucket` contiene gli oggetti `test1.txt` e `another/test1.txt`:

```
aws s3 cp s3://amzn-s3-demo-bucket/ s3://amzn-s3-demo-bucket2/ \
    --recursive \
    --exclude "another/*"
```

Output:

```
copy: s3://amzn-s3-demo-bucket/test1.txt to s3://amzn-s3-demo-bucket2/test1.txt
```

Puoi combinare le opzioni `--exclude` e `--include` per copiare solo gli oggetti che corrispondono a un modello, escludendo tutti gli altri:

```
aws s3 cp s3://amzn-s3-demo-bucket/logs/ s3://amzn-s3-demo-bucket2/logs/ \
    --recursive \
    --exclude "*" \
    --include "*.log"
```

Output:

```
copy: s3://amzn-s3-demo-bucket/logs/test/test.log to s3://amzn-s3-demo-bucket2/logs/
test/test.log
copy: s3://amzn-s3-demo-bucket/logs/test3.log to s3://amzn-s3-demo-bucket2/logs/
test3.log
```

Esempio 9: impostazione della lista di controllo degli accessi (ACL) durante la copia di un oggetto S3

Il comando cp seguente copia un singolo oggetto in un bucket e in una chiave specificati mentre si imposta l'ACL su public-read-write:

```
aws s3 cp s3://amzn-s3-demo-bucket/test.txt s3://amzn-s3-demo-bucket/test2.txt \
    --acl public-read-write
```

Output:

```
copy: s3://amzn-s3-demo-bucket/test.txt to s3://amzn-s3-demo-bucket/test2.txt
```

Nota: se stai utilizzando l'opzione --acl, assicurati che tutte le policy IAM associate includano l'azione "s3:PutObjectAcl":

```
aws iam get-user-policy \
    --user-name myuser \
    --policy-name mypolicy
```

Output:

```
{
    "UserName": "myuser",
    "PolicyName": "mypolicy",
    "PolicyDocument": {
        "Version":"2012-10-17",
        "Statement": [
            {
                "Action": [
                    "s3:PutObject",
                    "s3:PutObjectAcl"
                ],
                "Resource": [
                    "arn:aws:s3:::amzn-s3-demo-bucket/*"
                ],
                "Effect": "Allow",
                "Sid": "Stmt1234567891234"
            }
        ]
    }
}
```

Esempio 10: concessione delle autorizzazioni per un oggetto S3

Il comando `cp` seguente illustra l'utilizzo dell'opzione `--grants` per concedere l'accesso in lettura a tutti gli utenti identificati dall'URI e il controllo completo a un utente specifico identificato dal relativo ID canonico:

```
aws s3 cp file.txt s3://amzn-s3-demo-bucket/ --grants read=uri=http://
acs.amazonaws.com/groups/global/
AllUsers full=id=79a59df900b949e55d96a1e698fbacedfd6e09d98eacf8f8d5218e7cd47ef2be
```

Output:

```
upload: file.txt to s3://amzn-s3-demo-bucket/file.txt
```

Esempio 11: caricamento di un flusso di file locale su S3

PowerShell può alterare la codifica o aggiungere un CRLF all'input in piping.

Il comando `cp` seguente carica un flusso di file locale dall'input standard a un bucket e una chiave specificati:

```
aws s3 cp - s3://amzn-s3-demo-bucket/stream.txt
```

Esempio 12: caricamento di un flusso di file locale di dimensioni superiori a 50 GB su S3

Il comando `cp` seguente carica un flusso di file locale da 51 GB dall'input standard a un bucket e una chiave specificati. È necessario fornire l'opzione `--expected-size` oppure il caricamento potrebbe non riuscire quando raggiunge il limite predefinito di 10.000 parti:

```
aws s3 cp - s3://amzn-s3-demo-bucket/stream.txt --expected-size 54760833024
```

Esempio 13: download di un oggetto S3 come flusso di file locale

PowerShell può alterare la codifica o aggiungere un CRLF all'output reindirizzato o reindirizzato.

Il comando `cp` seguente scarica un oggetto S3 in locale come flusso verso l'output standard. Il download come flusso non è attualmente compatibile con il parametro `--recursive`:

```
aws s3 cp s3://amzn-s3-demo-bucket/stream.txt -
```

Esempio 14: caricamento su un punto di accesso S3

Il comando cp seguente carica un singolo file (mydoc.txt) nel punto di accesso (myaccesspoint) in corrispondenza della chiave (mykey):

```
aws s3 cp mydoc.txt s3://arn:aws:s3:us-west-2:123456789012:accesspoint/
myaccesspoint/mykey
```

Output:

```
upload: mydoc.txt to s3://arn:aws:s3:us-west-2:123456789012:accesspoint/
myaccesspoint/mykey
```

Esempio 15: download da un punto di accesso S3

Il comando cp seguente scarica un singolo oggetto (mykey) dal punto di accesso (myaccesspoint) al file locale (mydoc.txt):

```
aws s3 cp s3://arn:aws:s3:us-west-2:123456789012:accesspoint/myaccesspoint/
mykey mydoc.txt
```

Output:

```
download: s3://arn:aws:s3:us-west-2:123456789012:accesspoint/myaccesspoint/mykey to
 mydoc.txt
```

- Per informazioni dettagliate sull'API, consulta Cp in AWS CLI Command Reference.

create-bucket

Il seguente esempio di codice mostra come usare. create-bucket

AWS CLI

Esempio 1: come creare un bucket

L'esempio create-bucket seguente crea un bucket denominato amzn-s3-demo-bucket.

```
aws s3api create-bucket \
    --bucket amzn-s3-demo-bucket \
    --region us-east-1
```

Output:

```
{
    "Location": "/amzn-s3-demo-bucket"
}
```

Per ulteriori informazioni, consulta Creazione di un bucket nella Guida per l'utente di Amazon S3.

Esempio 2: come creare un bucket con il proprietario impostato

L'esempio `create-bucket` seguente crea un bucket denominato `amzn-s3-demo-bucket` che utilizza l'impostazione Proprietario del bucket applicato per la proprietà degli oggetti S3.

```
aws s3api create-bucket \
    --bucket amzn-s3-demo-bucket \
    --region us-east-1 \
    --object-ownership BucketOwnerEnforced
```

Output:

```
{
    "Location": "/amzn-s3-demo-bucket"
}
```

Per ulteriori informazioni, consulta la sezione Controllo della proprietà degli oggetti e disattivazione ACLs nella Guida per l'utente di Amazon S3.

Esempio 3: come creare un bucket al di fuori della Regione "us-east

L'esempio `create-bucket` seguente crea un bucket denominato `amzn-s3-demo-bucket` nella Regione `eu-west-1`. Le Regioni esterne a `us-east-1` richiedono la specifica del valore appropriato per `LocationConstraint` affinché il bucket venga creato nella Regione desiderata.

```
aws s3api create-bucket \
    --bucket amzn-s3-demo-bucket \
    --region eu-west-1 \
    --create-bucket-configuration LocationConstraint=eu-west-1
```

Output:

```
{
```

```
    "Location": "http://amzn-s3-demo-bucket.s3.amazonaws.com/"
}
```

Per ulteriori informazioni, consulta Creazione di un bucket nella Guida per l'utente di Amazon S3.

- Per i dettagli sull'API, consulta AWS CLI Command CreateBucketReference.

`create-multipart-upload`

Il seguente esempio di codice mostra come utilizzare`create-multipart-upload`.

AWS CLI

Il comando seguente completa un caricamento in più parti nel bucket `amzn-s3-demo-bucket` con la chiave `multipart/01`.

```
aws s3api create-multipart-upload --bucket amzn-s3-demo-bucket --key 'multipart/01'
```

Output:

```
{
    "Bucket": "amzn-s3-demo-bucket",
    "UploadId":
 "dfRtDYU0WWCCcH43C3WFbkRONycyCpTJJvxu2i5GYkZljF.Yxwh6XG7WfS2vC4to6HiV6Yjlx.cph0gtNBtJ8P3URC
    "Key": "multipart/01"
}
```

Il nome del file completato sarà `01`, disponibile in una cartella denominata `multipart` nel bucket `amzn-s3-demo-bucket`. Salva l'ID di caricamento, la chiave e il nome del bucket da utilizzare con il comando `upload-part`.

- Per i dettagli sull'API, consulta CreateMultipartUpload AWS CLICommand Reference.

`delete-bucket-analytics-configuration`

Il seguente esempio di codice mostra come utilizzare`delete-bucket-analytics-configuration`.

AWS CLI

Come eliminare una configurazione di analisi per un bucket

L'esempio `delete-bucket-analytics-configuration` seguente rimuove la configurazione di analisi per il bucket e l'ID specificati.

```
aws s3api delete-bucket-analytics-configuration \
    --bucket amzn-s3-demo-bucket \
    --id 1
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta DeleteBucketAnalyticsConfiguration AWS CLICommand Reference.

`delete-bucket-cors`

Il seguente esempio di codice mostra come utilizzare`delete-bucket-cors`.

AWS CLI

Il comando seguente elimina una configurazione di condivisione di risorse di origine incrociata (CORS) da un bucket denominato `amzn-s3-demo-bucket`.

```
aws s3api delete-bucket-cors --bucket amzn-s3-demo-bucket
```

- Per i dettagli sull'API, consulta DeleteBucketCors AWS CLICommand Reference.

`delete-bucket-encryption`

Il seguente esempio di codice mostra come utilizzare`delete-bucket-encryption`.

AWS CLI

Come eliminare la configurazione di crittografia lato server di un bucket

L'esempio `delete-bucket-encryption` seguente elimina la configurazione di crittografia lato server del bucket specificato.

```
aws s3api delete-bucket-encryption \
    --bucket amzn-s3-demo-bucket
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta DeleteBucketEncryption AWS CLICommand Reference.

`delete-bucket-intelligent-tiering-configuration`

Il seguente esempio di codice mostra come utilizzare`delete-bucket-intelligent-tiering-configuration`.

AWS CLI

Come rimuovere una configurazione S3 Intelligent-Tiering su un bucket

L'`delete-bucket-intelligent-tiering-configuration`esempio seguente rimuove una configurazione S3 Intelligent-Tiering, denominata ExampleConfig, su un bucket.

```
aws s3api delete-bucket-intelligent-tiering-configuration \
    --bucket amzn-s3-demo-bucket \
    --id ExampleConfig
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Utilizzare S3 Intelligent-Tiering nella Guida per l'utente di Amazon S3.

- Per i dettagli sull'API, consulta Command Reference. DeleteBucketIntelligentTieringConfigurationAWS CLI

`delete-bucket-inventory-configuration`

Il seguente esempio di codice mostra come utilizzare`delete-bucket-inventory-configuration`.

AWS CLI

Come eliminare la configurazione di inventario di un bucket

L'esempio `delete-bucket-inventory-configuration` seguente elimina la configurazione dell'inventario con ID 1 per il bucket specificato.

```
aws s3api delete-bucket-inventory-configuration \
    --bucket amzn-s3-demo-bucket \
    --id 1
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta DeleteBucketInventoryConfiguration AWS CLICommand Reference.

`delete-bucket-lifecycle`

Il seguente esempio di codice mostra come utilizzare`delete-bucket-lifecycle`.

AWS CLI

Il comando seguente elimina una configurazione del ciclo di vita da un bucket denominato `amzn-s3-demo-bucket`.

```
aws s3api delete-bucket-lifecycle --bucket amzn-s3-demo-bucket
```

- Per i dettagli sull'API, consulta DeleteBucketLifecycle AWS CLICommand Reference.

`delete-bucket-metrics-configuration`

Il seguente esempio di codice mostra come utilizzare`delete-bucket-metrics-configuration`.

AWS CLI

Come eliminare una configurazione delle metriche per un bucket

L'esempio `delete-bucket-metrics-configuration` seguente rimuove la configurazione delle metriche per il bucket e l'ID specificati.

```
aws s3api delete-bucket-metrics-configuration \
    --bucket amzn-s3-demo-bucket \
    --id 123
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta DeleteBucketMetricsConfiguration AWS CLICommand Reference.

`delete-bucket-ownership-controls`

Il seguente esempio di codice mostra come utilizzare`delete-bucket-ownership-controls`.

AWS CLI

Come rimuovere le impostazioni di proprietà del bucket di un bucket

L'esempio `delete-bucket-ownership-controls` seguente rimuove le impostazioni di proprietà del bucket di un bucket.

```
aws s3api delete-bucket-ownership-controls \
    --bucket amzn-s3-demo-bucket
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Impostazione della proprietà degli oggetti su un bucket esistente nella Guida per l'utente di Amazon S3.

- Per i dettagli sull'API, consulta DeleteBucketOwnershipControls AWS CLICommand Reference.

`delete-bucket-policy`

Il seguente esempio di codice mostra come utilizzare`delete-bucket-policy`.

AWS CLI

Il comando seguente elimina una policy di bucket denominata `amzn-s3-demo-bucket`.

```
aws s3api delete-bucket-policy --bucket amzn-s3-demo-bucket
```

- Per i dettagli sull'API, consulta DeleteBucketPolicy AWS CLICommand Reference.

`delete-bucket-replication`

Il seguente esempio di codice mostra come utilizzare`delete-bucket-replication`.

AWS CLI

Il comando seguente elimina una configurazione di replica da un bucket denominato `amzn-s3-demo-bucket`.

```
aws s3api delete-bucket-replication --bucket amzn-s3-demo-bucket
```

- Per i dettagli sull'API, consulta DeleteBucketReplication AWS CLICommand Reference.

`delete-bucket-tagging`

Il seguente esempio di codice mostra come utilizzare`delete-bucket-tagging`.

AWS CLI

Il comando seguente elimina una configurazione di tagging da un bucket denominato `amzn-s3-demo-bucket`.

```
aws s3api delete-bucket-tagging --bucket amzn-s3-demo-bucket
```

- Per i dettagli sull'API, consulta DeleteBucketTagging AWS CLICommand Reference.

`delete-bucket-website`

Il seguente esempio di codice mostra come utilizzare`delete-bucket-website`.

AWS CLI

Il comando seguente elimina una configurazione di siti web da un bucket denominato `amzn-s3-demo-bucket`.

```
aws s3api delete-bucket-website --bucket amzn-s3-demo-bucket
```

- Per i dettagli sull'API, consulta DeleteBucketWebsite AWS CLICommand Reference.

`delete-bucket`

Il seguente esempio di codice mostra come utilizzare`delete-bucket`.

AWS CLI

Il comando seguente elimina un bucket denominato `amzn-s3-demo-bucket`.

```
aws s3api delete-bucket --bucket amzn-s3-demo-bucket --region us-east-1
```

- Per i dettagli sull'API, consulta DeleteBucket AWS CLICommand Reference.

`delete-object-tagging`

Il seguente esempio di codice mostra come utilizzare`delete-object-tagging`.

AWS CLI

Come eliminare i set di tag di un oggetto

L'esempio `delete-object-tagging` seguente elimina il tag con la chiave specificata dall'oggetto `doc1.rtf`.

```
aws s3api delete-object-tagging \
    --bucket amzn-s3-demo-bucket \
    --key doc1.rtf
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta DeleteObjectTagging AWS CLICommand Reference.

`delete-object`

Il seguente esempio di codice mostra come utilizzare`delete-object`.

AWS CLI

Il comando seguente elimina un oggetto denominato `test.txt` da un bucket denominato `amzn-s3-demo-bucket`.

```
aws s3api delete-object --bucket amzn-s3-demo-bucket --key test.txt
```

Se il controllo delle versioni del bucket è abilitato, l'output conterrà l'ID della versione del contrassegno di eliminazione.

```
{
  "VersionId": "9_gKg5vG56F.TTEUdwkxGpJ3tNDlWlGq",
  "DeleteMarker": true
}
```

Per ulteriori informazioni sull'eliminazione degli oggetti, consulta "Eliminazione di oggetti" nella Guida per gli sviluppatori di Amazon S3.

- Per i dettagli sull'API, consulta DeleteObject AWS CLICommand Reference.

`delete-objects`

Il seguente esempio di codice mostra come utilizzare`delete-objects`.

AWS CLI

Il comando seguente elimina un oggetto da un bucket denominato `amzn-s3-demo-bucket`.

```
aws s3api delete-objects --bucket amzn-s3-demo-bucket --delete file://delete.json
```

`delete.json` è un documento JSON nella directory corrente che specifica l'oggetto da eliminare.

```
{
  "Objects": [
    {
      "Key": "test1.txt"
    }
  ],
  "Quiet": false
}
```

Output:

```
{
    "Deleted": [
        {
            "DeleteMarkerVersionId": "mYAT5Mc6F7aeUL8SS7FAAqUPO1koHwzU",
            "Key": "test1.txt",
            "DeleteMarker": true
        }
    ]
}
```

- Per i dettagli sull'API, consulta DeleteObjects AWS CLICommand Reference.

`delete-public-access-block`

Il seguente esempio di codice mostra come utilizzare`delete-public-access-block`.

AWS CLI

Come eliminare la configurazione di blocco dell'accesso pubblico per un bucket

L'esempio `delete-public-access-block` seguente rimuove la configurazione del blocco dell'accesso pubblico per il bucket e l'ID specificati.

```
aws s3api delete-public-access-block \
    --bucket amzn-s3-demo-bucket
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta DeletePublicAccessBlock AWS CLICommand Reference.

`get-bucket-accelerate-configuration`

Il seguente esempio di codice mostra come utilizzare`get-bucket-accelerate-configuration`.

AWS CLI

Come recuperare la configurazione di accelerazione di un bucket

L'esempio `get-bucket-accelerate-configuration` seguente recupera la configurazione di accelerazione per il bucket specificato.

```
aws s3api get-bucket-accelerate-configuration \
    --bucket amzn-s3-demo-bucket
```

Output:

```
{
    "Status": "Enabled"
}
```

- Per i dettagli sull'API, consulta GetBucketAccelerateConfiguration AWS CLICommand Reference.

`get-bucket-acl`

Il seguente esempio di codice mostra come utilizzare`get-bucket-acl`.

AWS CLI

Il comando seguente recupera una lista di controllo degli accessi (ACL) per un bucket denominato `amzn-s3-demo-bucket`.

```
aws s3api get-bucket-acl --bucket amzn-s3-demo-bucket
```

Output:

```
{
    "Owner": {
        "DisplayName": "my-username",
        "ID": "7009a8971cd538e11f6b6606438875e7c86c5b672f46db45460ddcd087d36c32"
    },
    "Grants": [
        {
            "Grantee": {
                "DisplayName": "my-username",
                "ID":
 "7009a8971cd538e11f6b6606438875e7c86c5b672f46db45460ddcd087d36c32"
            },
            "Permission": "FULL_CONTROL"
        }
    ]
}
```

- Per i dettagli sull'API, consulta GetBucketAcl AWS CLICommand Reference.

`get-bucket-analytics-configuration`

Il seguente esempio di codice mostra come utilizzare`get-bucket-analytics-configuration`.

AWS CLI

Come recuperare la configurazione di analisi per un bucket con un ID specifico

L'esempio `get-bucket-analytics-configuration` seguente visualizza la configurazione di analisi per il bucket e l'ID specificati.

```
aws s3api get-bucket-analytics-configuration \
    --bucket amzn-s3-demo-bucket \
    --id 1
```

Output:

```
{
    "AnalyticsConfiguration": {
        "StorageClassAnalysis": {},
        "Id": "1"
```

```
    }
}
```

- Per i dettagli sull'API, consulta GetBucketAnalyticsConfiguration AWS CLICommand Reference.

`get-bucket-cors`

Il seguente esempio di codice mostra come utilizzare`get-bucket-cors`.

AWS CLI

Il comando seguente recupera una configurazione di condivisione di risorse di origine incrociata (CORS) da un bucket denominato `amzn-s3-demo-bucket`.

```
aws s3api get-bucket-cors --bucket amzn-s3-demo-bucket
```

Output:

```
{
    "CORSRules": [
        {
            "AllowedHeaders": [
                "*"
            ],
            "ExposeHeaders": [
                "x-amz-server-side-encryption"
            ],
            "AllowedMethods": [
                "PUT",
                "POST",
                "DELETE"
            ],
            "MaxAgeSeconds": 3000,
            "AllowedOrigins": [
                "http://www.example.com"
            ]
        },
        {
            "AllowedHeaders": [
                "Authorization"
            ],
            "MaxAgeSeconds": 3000,
```

```
            "AllowedMethods": [
                "GET"
            ],
            "AllowedOrigins": [
                "*"
            ]
        }
    ]
}
```

- Per i dettagli sull'API, consulta GetBucketCors AWS CLICommand Reference.

`get-bucket-encryption`

Il seguente esempio di codice mostra come utilizzare`get-bucket-encryption`.

AWS CLI

Come recuperare la configurazione di crittografia lato server di un bucket

L'esempio `get-bucket-encryption` seguente restituisce la configurazione di crittografia lato server per il bucket `amzn-s3-demo-bucket`.

```
aws s3api get-bucket-encryption \
    --bucket amzn-s3-demo-bucket
```

Output:

```
{
    "ServerSideEncryptionConfiguration": {
        "Rules": [
            {
                "ApplyServerSideEncryptionByDefault": {
                    "SSEAlgorithm": "AES256"
                }
            }
        ]
    }
}
```

- Per i dettagli sull'API, consulta GetBucketEncryption AWS CLICommand Reference.

`get-bucket-intelligent-tiering-configuration`

Il seguente esempio di codice mostra come utilizzare`get-bucket-intelligent-tiering-configuration`.

AWS CLI

Per rimuovere una configurazione S3 Intelligent-Tiering su un bucket

L'`get-bucket-intelligent-tiering-configuration`esempio seguente recupera una configurazione S3 Intelligent-Tiering, denominata, su un bucket. ExampleConfig

```
aws s3api get-bucket-intelligent-tiering-configuration \
    --bucket amzn-s3-demo-bucket \
    --id ExampleConfig
```

Output:

```
{
    "IntelligentTieringConfiguration": {
        "Id": "ExampleConfig2",
        "Filter": {
            "Prefix": "images"
        },
        "Status": "Enabled",
        "Tierings": [
            {
                "Days": 90,
                "AccessTier": "ARCHIVE_ACCESS"
            },
            {
                "Days": 180,
                "AccessTier": "DEEP_ARCHIVE_ACCESS"
            }
        ]
    }
}
```

Per ulteriori informazioni, consulta Utilizzare S3 Intelligent-Tiering nella Guida per l'utente di Amazon S3.

- Per i dettagli sull'API, consulta Command Reference. GetBucketIntelligentTieringConfigurationAWS CLI

`get-bucket-inventory-configuration`

Il seguente esempio di codice mostra come utilizzare`get-bucket-inventory-configuration`.

AWS CLI

Come recuperare la configurazione di inventario per un bucket

L'esempio `get-bucket-inventory-configuration` seguente recupera la configurazione dell'inventario con ID 1 per il bucket specificato.

```
aws s3api get-bucket-inventory-configuration \
    --bucket amzn-s3-demo-bucket \
    --id 1
```

Output:

```
{
    "InventoryConfiguration": {
        "IsEnabled": true,
        "Destination": {
            "S3BucketDestination": {
                "Format": "ORC",
                "Bucket": "arn:aws:s3:::amzn-s3-demo-bucket",
                "AccountId": "123456789012"
            }
        },
        "IncludedObjectVersions": "Current",
        "Id": "1",
        "Schedule": {
            "Frequency": "Weekly"
        }
    }
}
```

- Per i dettagli sull'API, consulta GetBucketInventoryConfiguration AWS CLICommand Reference.

`get-bucket-lifecycle-configuration`

Il seguente esempio di codice mostra come utilizzare`get-bucket-lifecycle-configuration`.

AWS CLI

Il comando seguente recupera una configurazione del ciclo di vita per un bucket denominato `amzn-s3-demo-bucket`.

```
aws s3api get-bucket-lifecycle-configuration --bucket amzn-s3-demo-bucket
```

Output:

```
{
    "Rules": [
        {
            "ID": "Move rotated logs to Glacier",
            "Prefix": "rotated/",
            "Status": "Enabled",
            "Transitions": [
                {
                    "Date": "2015-11-10T00:00:00.000Z",
                    "StorageClass": "GLACIER"
                }
            ]
        },
        {
            "Status": "Enabled",
            "Prefix": "",
            "NoncurrentVersionTransitions": [
                {
                    "NoncurrentDays": 0,
                    "StorageClass": "GLACIER"
                }
            ],
            "ID": "Move old versions to Glacier"
        }
    ]
}
```

- Per i dettagli sull'API, consulta GetBucketLifecycleConfiguration AWS CLICommand Reference.

`get-bucket-lifecycle`

Il seguente esempio di codice mostra come utilizzare`get-bucket-lifecycle`.

AWS CLI

Il comando seguente recupera una configurazione del ciclo di vita per un bucket denominato `amzn-s3-demo-bucket`.

```
aws s3api get-bucket-lifecycle --bucket amzn-s3-demo-bucket
```

Output:

```
{
  "Rules": [
    {
      "ID": "Move to Glacier after sixty days (objects in logs/2015/)",
      "Prefix": "logs/2015/",
      "Status": "Enabled",
      "Transition": {
        "Days": 60,
        "StorageClass": "GLACIER"
      }
    },
    {
      "Expiration": {
        "Date": "2016-01-01T00:00:00.000Z"
      },
      "ID": "Delete 2014 logs in 2016.",
      "Prefix": "logs/2014/",
      "Status": "Enabled"
    }
  ]
}
```

- Per i dettagli sull'API, consulta GetBucketLifecycle AWS CLICommand Reference.

`get-bucket-location`

Il seguente esempio di codice mostra come utilizzare`get-bucket-location`.

AWS CLI

Il comando seguente recupera il vincolo di posizione per un bucket denominato `amzn-s3-demo-bucket`, se esiste un vincolo.

```
aws s3api get-bucket-location --bucket amzn-s3-demo-bucket
```

Output:

```
{
    "LocationConstraint": "us-west-2"
}
```

- Per i dettagli sull'API, consulta GetBucketLocation AWS CLICommand Reference.

`get-bucket-logging`

Il seguente esempio di codice mostra come utilizzare`get-bucket-logging`.

AWS CLI

Come recuperare lo stato della registrazione di log di un bucket

L'esempio `get-bucket-logging` seguente recupera lo stato di registrazione per il bucket specificato.

```
aws s3api get-bucket-logging \
    --bucket amzn-s3-demo-bucket
```

Output:

```
{
    "LoggingEnabled": {
        "TargetPrefix": "",
        "TargetBucket": "amzn-s3-demo-bucket-logs"
          }
}
```

- Per i dettagli sull'API, consulta GetBucketLogging AWS CLICommand Reference.

`get-bucket-metrics-configuration`

Il seguente esempio di codice mostra come utilizzare`get-bucket-metrics-configuration`.

AWS CLI

Come recuperare la configurazione delle metriche per un bucket con un ID specifico

L'esempio `get-bucket-metrics-configuration` seguente visualizza la configurazione delle metriche per il bucket e l'ID specificati.

```
aws s3api get-bucket-metrics-configuration \
    --bucket amzn-s3-demo-bucket \
    --id 123
```

Output:

```
{
    "MetricsConfiguration": {
        "Filter": {
            "Prefix": "logs"
        },
        "Id": "123"
    }
}
```

- Per i dettagli sull'API, consulta GetBucketMetricsConfiguration AWS CLICommand Reference.

`get-bucket-notification-configuration`

Il seguente esempio di codice mostra come utilizzare`get-bucket-notification-configuration`.

AWS CLI

Il comando seguente recupera la configurazione di notifica per un bucket denominato `amzn-s3-demo-bucket`.

```
aws s3api get-bucket-notification-configuration --bucket amzn-s3-demo-bucket
```

Output:

```
{
    "TopicConfigurations": [
        {
```

```
            "Id": "YmQzMmEwM2EjZWVlI0NGItNzVtZjI1MC00ZjgyLWZDBiZWNl",
            "TopicArn": "arn:aws:sns:us-west-2:123456789012:my-notification-topic",
            "Events": [
                "s3:ObjectCreated:*"
            ]
        }
    ]
}
```

- Per i dettagli sull'API, consulta GetBucketNotificationConfiguration AWS CLICommand Reference.

`get-bucket-notification`

Il seguente esempio di codice mostra come utilizzare`get-bucket-notification`.

AWS CLI

Il comando seguente recupera la configurazione di notifica per un bucket denominato `amzn-s3-demo-bucket`.

```
aws s3api get-bucket-notification --bucket amzn-s3-demo-bucket
```

Output:

```
{
    "TopicConfiguration": {
        "Topic": "arn:aws:sns:us-west-2:123456789012:my-notification-topic",
        "Id": "YmQzMmEwM2EjZWVlI0NGItNzVtZjI1MC00ZjgyLWZDBiZWNl",
        "Event": "s3:ObjectCreated:*",
        "Events": [
            "s3:ObjectCreated:*"
        ]
    }
}
```

- Per i dettagli sull'API, consulta GetBucketNotification AWS CLICommand Reference.

`get-bucket-ownership-controls`

Il seguente esempio di codice mostra come utilizzare`get-bucket-ownership-controls`.

AWS CLI

Come recuperare le impostazioni di proprietà del bucket di un bucket

L'esempio `get-bucket-ownership-controls` seguente recupera le impostazioni di proprietà del bucket di un bucket.

```
aws s3api get-bucket-ownership-controls \
    --bucket amzn-s3-demo-bucket
```

Output:

```
{
    "OwnershipControls": {
        "Rules": [
            {
                "ObjectOwnership": "BucketOwnerEnforced"
            }
        ]
    }
}
```

Per ulteriori informazioni, consulta Visualizzare l'impostazione di Object Ownership per un bucket S3 nella Guida per l'utente di Amazon S3.

- Per i dettagli sull'API, consulta GetBucketOwnershipControls AWS CLICommand Reference.

`get-bucket-policy-status`

Il seguente esempio di codice mostra come utilizzare`get-bucket-policy-status`.

AWS CLI

Come recuperare lo stato delle policy per un bucket, che indica se il bucket è pubblico

L'esempio `get-bucket-policy-status` seguente recupera lo stato della policy per il bucket `amzn-s3-demo-bucket`.

```
aws s3api get-bucket-policy-status \
    --bucket amzn-s3-demo-bucket
```

Output:

```
{
    "PolicyStatus": {
        "IsPublic": false
    }
}
```

- Per i dettagli sull'API, consulta GetBucketPolicyStatus AWS CLICommand Reference.

`get-bucket-policy`

Il seguente esempio di codice mostra come utilizzare`get-bucket-policy`.

AWS CLI

Il comando seguente recupera la policy di bucket per un bucket denominato `amzn-s3-demo-bucket`.

```
aws s3api get-bucket-policy --bucket amzn-s3-demo-bucket
```

Output:

```
{
    "Policy": "{\"Version\":\"2008-10-17\",\"Statement\":[{\"Sid\":\"\",\"Effect
\":\"Allow\",\"Principal\":\"*\",\"Action\":\"s3:GetObject\",\"Resource\":
\"arn:aws:s3:::amzn-s3-demo-bucket/*\"},{\"Sid\":\"\",\"Effect\":\"Deny\",
\"Principal\":\"*\",\"Action\":\"s3:GetObject\",\"Resource\":\"arn:aws:s3:::amzn-s3-
demo-bucket/secret/*\"}]}"
}
```

Come ottenere e implementare una policy di bucket L'esempio seguente mostra come scaricare una policy di bucket di Amazon S3, apportare modifiche al file e quindi utilizzare `put-bucket-policy` per applicare la policy di bucket modificata. Per scaricare la policy di bucket in un file, è possibile eseguire:

```
aws s3api get-bucket-policy --bucket amzn-s3-demo-bucket --query Policy --output
 text > policy.json
```

Sarà quindi possibile modificare il file `policy.json` in base alle esigenze. Sarà infine possibile riapplicare la policy modificata al bucket S3 eseguendo il file

`policy.json`, se necessario. Sarà infine possibile riapplicare la policy modificata al bucket S3 eseguendo il file

se necessario. Sarà infine possibile riapplicare la policy modificata al bucket S3 eseguendo il file

```
aws s3api put-bucket-policy --bucket amzn-s3-demo-bucket --policy file://policy.json
```

- Per i dettagli sull'API, consulta GetBucketPolicy AWS CLICommand Reference.

`get-bucket-replication`

Il seguente esempio di codice mostra come utilizzare`get-bucket-replication`.

AWS CLI

Il comando seguente recupera una configurazione di replica per un bucket denominato `amzn-s3-demo-bucket`:

```
aws s3api get-bucket-replication --bucket amzn-s3-demo-bucket
```

Output:

```
{
    "ReplicationConfiguration": {
        "Rules": [
            {
                "Status": "Enabled",
                "Prefix": "",
                "Destination": {
                    "Bucket": "arn:aws:s3:::amzn-s3-demo-bucket-backup",
                    "StorageClass": "STANDARD"
                },
                "ID": "ZmUwNzE4ZmQ4tMjVhOS00MTlkLOGI4NDkzZTIWJjNTUtYTA1"
            }
        ],
        "Role": "arn:aws:iam::123456789012:role/s3-replication-role"
    }
}
```

- Per i dettagli sull'API, consulta GetBucketReplication AWS CLICommand Reference.

`get-bucket-request-payment`

Il seguente esempio di codice mostra come utilizzare`get-bucket-request-payment`.

AWS CLI

Come recuperare la configurazione del pagamento delle richieste per un bucket

L'esempio `get-bucket-request-payment` seguente recupera la configurazione dei pagamenti a carico del richiedente per il bucket specificato.

```
aws s3api get-bucket-request-payment \
    --bucket amzn-s3-demo-bucket
```

Output:

```
{
    "Payer": "BucketOwner"
}
```

- Per i dettagli sull'API, consulta GetBucketRequestPayment AWS CLICommand Reference.

`get-bucket-tagging`

Il seguente esempio di codice mostra come utilizzare`get-bucket-tagging`.

AWS CLI

Il comando seguente recupera una configurazione di tagging per un bucket denominato `amzn-s3-demo-bucket`.

```
aws s3api get-bucket-tagging --bucket amzn-s3-demo-bucket
```

Output:

```
{
    "TagSet": [
        {
            "Value": "marketing",
            "Key": "organization"
        }
    ]
```

```
}
```

- Per i dettagli sull'API, consulta GetBucketTagging AWS CLICommand Reference.

`get-bucket-versioning`

Il seguente esempio di codice mostra come utilizzare`get-bucket-versioning`.

AWS CLI

Il comando seguente recupera una configurazione di controllo delle versioni per un bucket denominato `amzn-s3-demo-bucket`.

```
aws s3api get-bucket-versioning --bucket amzn-s3-demo-bucket
```

Output:

```
{
    "Status": "Enabled"
}
```

- Per i dettagli sull'API, consulta GetBucketVersioning AWS CLICommand Reference.

`get-bucket-website`

Il seguente esempio di codice mostra come utilizzare`get-bucket-website`.

AWS CLI

Il comando seguente recupera una configurazione di siti web statici per un bucket denominato `amzn-s3-demo-bucket`.

```
aws s3api get-bucket-website --bucket amzn-s3-demo-bucket
```

Output:

```
{
    "IndexDocument": {
        "Suffix": "index.html"
    },
```

```
    "ErrorDocument": {
        "Key": "error.html"
    }
}
```

- Per i dettagli sull'API, consulta GetBucketWebsite AWS CLICommand Reference.

`get-object-acl`

Il seguente esempio di codice mostra come utilizzare`get-object-acl`.

AWS CLI

Il comando seguente recupera una lista di controllo degli accessi (ACL) per un oggetto in un bucket denominato `amzn-s3-demo-bucket`.

```
aws s3api get-object-acl --bucket amzn-s3-demo-bucket --key index.html
```

Output:

```
{
    "Owner": {
        "DisplayName": "my-username",
        "ID": "7009a8971cd538e11f6b6606438875e7c86c5b672f46db45460ddcd087d36c32"
    },
    "Grants": [
        {
            "Grantee": {
                "DisplayName": "my-username",
                "ID":
 "7009a8971cd538e11f6b6606438875e7c86c5b672f46db45460ddcd087d36c32"
            },
            "Permission": "FULL_CONTROL"
        },
        {
            "Grantee": {
                "URI": "http://acs.amazonaws.com/groups/global/AllUsers"
            },
            "Permission": "READ"
        }
    ]
}
```

- Per i dettagli sull'API, consulta GetObjectAcl AWS CLICommand Reference.

`get-object-attributes`

Il seguente esempio di codice mostra come utilizzare`get-object-attributes`.

AWS CLI

Come recuperare i metadati da un oggetto senza restituire l'oggetto stesso

L'esempio `get-object-attributes` seguente recupera i metadati dall'oggetto `doc1.rtf`.

```
aws s3api get-object-attributes \
    --bucket amzn-s3-demo-bucket \
    --key doc1.rtf \
    --object-attributes "StorageClass" "ETag" "ObjectSize"
```

Output:

```
{
    "LastModified": "2022-03-15T19:37:31+00:00",
    "VersionId": "IuCPjXTDzHNfldAuitVBIKJpF2p1fg4P",
    "ETag": "b662d79adeb7c8d787ea7eafb9ef6207",
    "StorageClass": "STANDARD",
    "ObjectSize": 405
}
```

Per ulteriori informazioni, GetObjectAttributesconsulta Amazon S3 API Reference.

- Per i dettagli sull'API, consulta GetObjectAttributes AWS CLICommand Reference.

`get-object-legal-hold`

Il seguente esempio di codice mostra come utilizzare`get-object-legal-hold`.

AWS CLI

Come recuperare lo stato relativo alla conservazione a fini legali di un oggetto

L'esempio `get-object-legal-hold` seguente recupera lo stato relativo alla conservazione a fini legali per l'oggetto specificato.

```
aws s3api get-object-legal-hold \
    --bucket amzn-s3-demo-bucket-with-object-lock \
    --key doc1.rtf
```

Output:

```
{
    "LegalHold": {
        "Status": "ON"
    }
}
```

- Per i dettagli sull'API, consulta GetObjectLegalHold AWS CLICommand Reference.

`get-object-lock-configuration`

Il seguente esempio di codice mostra come utilizzare`get-object-lock-configuration`.

AWS CLI

Come recupera la configurazione Object Lock di un bucket

L'esempio `get-object-lock-configuration` seguente recupera la configurazione di Object Lock per il bucket specificato.

```
aws s3api get-object-lock-configuration \
    --bucket amzn-s3-demo-bucket-with-object-lock
```

Output:

```
{
    "ObjectLockConfiguration": {
        "ObjectLockEnabled": "Enabled",
        "Rule": {
            "DefaultRetention": {
                "Mode": "COMPLIANCE",
                "Days": 50
            }
        }
    }
}
```

- Per i dettagli sull'API, consulta GetObjectLockConfiguration AWS CLICommand Reference.

`get-object-retention`

Il seguente esempio di codice mostra come utilizzare`get-object-retention`.

AWS CLI

Come recuperare la configurazione di conservazione degli oggetti per un oggetto

L'esempio `get-object-retention` seguente recupera la configurazione di conservazione degli oggetti per il bucket specificato.

```
aws s3api get-object-retention \
    --bucket amzn-s3-demo-bucket-with-object-lock \
    --key doc1.rtf
```

Output:

```
{
    "Retention": {
        "Mode": "GOVERNANCE",
        "RetainUntilDate": "2025-01-01T00:00:00.000Z"
    }
}
```

- Per i dettagli sull'API, consulta GetObjectRetention AWS CLICommand Reference.

`get-object-tagging`

Il seguente esempio di codice mostra come utilizzare`get-object-tagging`.

AWS CLI

Come recuperare i tag associati a un oggetto

L'esempio `get-object-tagging` seguente recupera i valori della chiave specificata dall'oggetto specificato.

```
aws s3api get-object-tagging \
    --bucket amzn-s3-demo-bucket \
```

```
    --key doc1.rtf
```

Output:

```
{
    "TagSet": [
        {
            "Value": "confidential",
            "Key": "designation"
        }
    ]
}
```

L'esempio `get-object-tagging` seguente tenta di recuperare i set di tag dell'oggetto `doc2.rtf`, che non ha tag.

```
aws s3api get-object-tagging \
    --bucket amzn-s3-demo-bucket \
    --key doc2.rtf
```

Output:

```
{
    "TagSet": []
}
```

L'esempio `get-object-tagging` seguente recupera i set di tag dell'oggetto `doc3.rtf`, che ha più tag.

```
aws s3api get-object-tagging \
    --bucket amzn-s3-demo-bucket \
    --key doc3.rtf
```

Output:

```
{
    "TagSet": [
        {
            "Value": "confidential",
            "Key": "designation"
        },
```

```
        {
            "Value": "finance",
            "Key": "department"
        },
        {
            "Value": "payroll",
            "Key": "team"
        }
    ]
}
```

- Per i dettagli sull'API, consulta GetObjectTagging AWS CLICommand Reference.

`get-object-torrent`

Il seguente esempio di codice mostra come utilizzare`get-object-torrent`.

AWS CLI

Il seguente comando crea un file torrent per un oggetto in un bucket denominato `amzn-s3-demo-bucket`.

```
aws s3api get-object-torrent --bucket amzn-s3-demo-bucket --key large-video-file.mp4 large-video-file.torrent
```

Il file torrent viene salvato in locale nella cartella corrente. Nota: il nome del file di output (`large-video-file.torrent`) è specificato senza un nome di opzione e deve essere l'ultimo argomento del comando.

- Per i dettagli sull'API, consulta GetObjectTorrent AWS CLICommand Reference.

`get-object`

Il seguente esempio di codice mostra come utilizzare`get-object`.

AWS CLI

L'esempio seguente utilizza il comando `get-object` per scaricare un oggetto da Amazon S3.

```
aws s3api get-object --bucket text-content --key dir/my_images.tar.bz2 my_images.tar.bz2
```

Nota che il parametro outfile è specificato senza un nome di opzione, ad esempio "--outfile". Il nome del file di output deve essere l'ultimo parametro del comando.

L'esempio seguente mostra l'utilizzo di `--range` per scaricare un intervallo di byte specifico da un oggetto. Tieni presente che gli intervalli di byte devono essere preceduti dal prefisso "bytes=":

```
aws s3api get-object --bucket text-content --key dir/my_data --
range bytes=8888-9999 my_data_range
```

Per ulteriori informazioni sul recupero degli oggetti, consulta "Recupero degli oggetti" nella Guida per gli sviluppatori di Amazon S3.

- Per i dettagli sull'API, consulta GetObject AWS CLICommand Reference.

`get-public-access-block`

Il seguente esempio di codice mostra come utilizzare`get-public-access-block`.

AWS CLI

Come impostare o modificare la configurazione di blocco dell'accesso pubblico per un bucket

L'esempio `get-public-access-block` seguente visualizza la configurazione del blocco dell'accesso pubblico per il bucket specificato.

```
aws s3api get-public-access-block \
    --bucket amzn-s3-demo-bucket
```

Output:

```
{
    "PublicAccessBlockConfiguration": {
        "IgnorePublicAcls": true,
        "BlockPublicPolicy": true,
        "BlockPublicAcls": true,
        "RestrictPublicBuckets": true
    }
}
```

- Per i dettagli sull'API, consulta GetPublicAccessBlock AWS CLICommand Reference.

`head-bucket`

Il seguente esempio di codice mostra come utilizzare`head-bucket`.

AWS CLI

Il comando seguente verifica l'accesso a un bucket denominato `amzn-s3-demo-bucket`.

```
aws s3api head-bucket --bucket amzn-s3-demo-bucket
```

Se il bucket esiste e si dispone dell'accesso a esso, non viene restituito alcun output. In caso contrario, verrà visualizzato un messaggio di errore. Esempio:

```
A client error (404) occurred when calling the HeadBucket operation: Not Found
```

- Per i dettagli sull'API, consulta HeadBucket AWS CLICommand Reference.

`head-object`

Il seguente esempio di codice mostra come utilizzare`head-object`.

AWS CLI

Il comando seguente recupera i metadati di un oggetto in un bucket denominato `amzn-s3-demo-bucket`.

```
aws s3api head-object --bucket amzn-s3-demo-bucket --key index.html
```

Output:

```
{
    "AcceptRanges": "bytes",
    "ContentType": "text/html",
    "LastModified": "Thu, 16 Apr 2015 18:19:14 GMT",
    "ContentLength": 77,
    "VersionId": "null",
    "ETag": "\"30a6ec7e1a9ad79c203d05a589c8b400\"",
    "Metadata": {}
}
```

- Per i dettagli sull'API, consulta HeadObject AWS CLICommand Reference.

`list-bucket-analytics-configurations`

Il seguente esempio di codice mostra come utilizzare`list-bucket-analytics-configurations`.

AWS CLI

Come recuperare un elenco di configurazioni di analisi per un bucket

L'esempio `list-bucket-analytics-configurations` seguente recupera un elenco di configurazioni di analisi per il bucket specificato.

```
aws s3api list-bucket-analytics-configurations \
    --bucket amzn-s3-demo-bucket
```

Output:

```
{
    "AnalyticsConfigurationList": [
        {
            "StorageClassAnalysis": {},
            "Id": "1"
        }
    ],
    "IsTruncated": false
}
```

- Per i dettagli sull'API, consulta ListBucketAnalyticsConfigurations AWS CLICommand Reference.

`list-bucket-intelligent-tiering-configurations`

Il seguente esempio di codice mostra come utilizzare`list-bucket-intelligent-tiering-configurations`.

AWS CLI

Come recuperare tutte le configurazioni S3 Intelligent-Tiering su un bucket

L'esempio `list-bucket-intelligent-tiering-configurations` seguente recupera tutta la configurazione S3 Intelligent-Tiering su un bucket.

```
aws s3api list-bucket-intelligent-tiering-configurations \
    --bucket amzn-s3-demo-bucket
```

Output:

```
{
    "IsTruncated": false,
    "IntelligentTieringConfigurationList": [
        {
            "Id": "ExampleConfig",
            "Filter": {
                "Prefix": "images"
            },
            "Status": "Enabled",
            "Tierings": [
                {
                    "Days": 90,
                    "AccessTier": "ARCHIVE_ACCESS"
                },
                {
                    "Days": 180,
                    "AccessTier": "DEEP_ARCHIVE_ACCESS"
                }
            ]
        },
        {
            "Id": "ExampleConfig2",
            "Status": "Disabled",
            "Tierings": [
                {
                    "Days": 730,
                    "AccessTier": "ARCHIVE_ACCESS"
                }
            ]
        },
        {
            "Id": "ExampleConfig3",
            "Filter": {
                "Tag": {
                    "Key": "documents",
                    "Value": "taxes"
                }
            },
```

```
            "Status": "Enabled",
            "Tierings": [
                {
                    "Days": 90,
                    "AccessTier": "ARCHIVE_ACCESS"
                },
                {
                    "Days": 365,
                    "AccessTier": "DEEP_ARCHIVE_ACCESS"
                }
            ]
        }
    ]
}
```

Per ulteriori informazioni, consulta Utilizzare S3 Intelligent-Tiering nella Guida per l'utente di Amazon S3.

- Per i dettagli sull'API, consulta ListBucketIntelligentTieringConfigurations AWS CLICommand Reference.

`list-bucket-inventory-configurations`

Il seguente esempio di codice mostra come utilizzare`list-bucket-inventory-configurations`.

AWS CLI

Come recuperare un elenco di configurazioni di inventario per un bucket

L'esempio `list-bucket-inventory-configurations` seguente recupera la configurazione dell'inventario per il bucket specificato.

```
aws s3api list-bucket-inventory-configurations \
    --bucket amzn-s3-demo-bucket
```

Output:

```
{
    "InventoryConfigurationList": [
        {
```

```
            "IsEnabled": true,
            "Destination": {
                "S3BucketDestination": {
                    "Format": "ORC",
                    "Bucket": "arn:aws:s3:::amzn-s3-demo-bucket",
                    "AccountId": "123456789012"
                }
            },
            "IncludedObjectVersions": "Current",
            "Id": "1",
            "Schedule": {
                "Frequency": "Weekly"
            }
        },
        {
            "IsEnabled": true,
            "Destination": {
                "S3BucketDestination": {
                    "Format": "CSV",
                    "Bucket": "arn:aws:s3:::amzn-s3-demo-bucket",
                    "AccountId": "123456789012"
                }
            },
            "IncludedObjectVersions": "Current",
            "Id": "2",
            "Schedule": {
                "Frequency": "Daily"
            }
        }
    ],
    "IsTruncated": false
}
```

- Per i dettagli sull'API, consulta ListBucketInventoryConfigurations AWS CLICommand Reference.

`list-bucket-metrics-configurations`

Il seguente esempio di codice mostra come utilizzare`list-bucket-metrics-configurations`.

AWS CLI

Come recuperare un elenco di configurazioni di analisi per un bucket

L'esempio `list-bucket-metrics-configurations` seguente recupera un elenco di configurazioni di metriche per il bucket specificato.

```
aws s3api list-bucket-metrics-configurations \
    --bucket amzn-s3-demo-bucket
```

Output:

```
{
    "IsTruncated": false,
    "MetricsConfigurationList": [
        {
            "Filter": {
                "Prefix": "logs"
            },
            "Id": "123"
        },
        {
            "Filter": {
                "Prefix": "tmp"
            },
            "Id": "234"
        }
    ]
}
```

- Per i dettagli sull'API, consulta ListBucketMetricsConfigurations AWS CLICommand Reference.

`list-buckets`

Il seguente esempio di codice mostra come utilizzare`list-buckets`.

AWS CLI

Il comando seguente utilizza il comando `list-buckets` per visualizzare i nomi di tutti i bucket Amazon S3 (in tutte le Regioni).

```
aws s3api list-buckets --query "Buckets[].Name"
```

L'opzione query filtra l'output di `list-buckets` restituendo solo nomi dei bucket.

Per ulteriori informazioni sui bucket, consulta "Utilizzo dei bucket Amazon S3" nella Guida per gli sviluppatori di Amazon S3.

- Per i dettagli sull'API, consulta ListBuckets AWS CLICommand Reference.

`list-multipart-uploads`

Il seguente esempio di codice mostra come utilizzare`list-multipart-uploads`.

AWS CLI

Il comando seguente elenca tutti i caricamenti in più parti attivi per un bucket denominato `amzn-s3-demo-bucket`.

```
aws s3api list-multipart-uploads --bucket amzn-s3-demo-bucket
```

Output:

```
{
    "Uploads": [
        {
            "Initiator": {
                "DisplayName": "username",
                "ID": "arn:aws:iam::0123456789012:user/username"
            },
            "Initiated": "2015-06-02T18:01:30.000Z",
            "UploadId":
 "dfRtDYU0WWCCcH43C3WFbkRONycyCpTJJvxu2i5GYkZljF.Yxwh6XG7WfS2vC4to6HiV6Yjlx.cph0gtNBtJ8P3URC
            "StorageClass": "STANDARD",
            "Key": "multipart/01",
            "Owner": {
                "DisplayName": "aws-account-name",
                "ID":
 "100719349fc3b6dcd7c820a124bf7aecd408092c3d7b51b38494939801fc248b"
            }
        }
    ],
    "CommonPrefixes": []
}
```

I caricamenti in più parti in corso generano costi di archiviazione in Amazon S3. Completa o annulla un caricamento in più parti attivo per rimuoverne le parti dall'account corrente.

- Per i dettagli sull'API, consulta ListMultipartUploads AWS CLICommand Reference.

`list-object-versions`

Il seguente esempio di codice mostra come utilizzare`list-object-versions`.

AWS CLI

Il comando seguente recupera le informazioni sulla versione di un oggetto in un bucket denominato `amzn-s3-demo-bucket`.

```
aws s3api list-object-versions --bucket amzn-s3-demo-bucket --prefix index.html
```

Output:

```
{
    "DeleteMarkers": [
        {
            "Owner": {
                "DisplayName": "my-username",
                "ID":
 "7009a8971cd660687538875e7c86c5b672fe116bd438f46db45460ddcd036c32"
            },
            "IsLatest": true,
            "VersionId": "B2VsEK5saUNNHKcOAJj7hIE86RozToyq",
            "Key": "index.html",
            "LastModified": "2015-11-10T00:57:03.000Z"
        },
        {
            "Owner": {
                "DisplayName": "my-username",
                "ID":
 "7009a8971cd660687538875e7c86c5b672fe116bd438f46db45460ddcd036c32"
            },
            "IsLatest": false,
            "VersionId": ".FLQEZscLIcfxSq.jsFJ.szUkmng2Yw6",
            "Key": "index.html",
            "LastModified": "2015-11-09T23:32:20.000Z"
        }
    ],
    "Versions": [
        {
```

```
            "LastModified": "2015-11-10T00:20:11.000Z",
            "VersionId": "Rb_l2T8UHDkFEwCgJjhlgPOZC0qJ.vpD",
            "ETag": "\"0622528de826c0df5db1258a23b80be5\"",
            "StorageClass": "STANDARD",
            "Key": "index.html",
            "Owner": {
                "DisplayName": "my-username",
                "ID":
 "7009a8971cd660687538875e7c86c5b672fe116bd438f46db45460ddcd036c32"
            },
            "IsLatest": false,
            "Size": 38
        },
        {
            "LastModified": "2015-11-09T23:26:41.000Z",
            "VersionId": "rasWWGpgk9E4s0LyTJgusGeRQKLVIAFf",
            "ETag": "\"06225825b8028de826c0df5db1a23be5\"",
            "StorageClass": "STANDARD",
            "Key": "index.html",
            "Owner": {
                "DisplayName": "my-username",
                "ID":
 "7009a8971cd660687538875e7c86c5b672fe116bd438f46db45460ddcd036c32"
            },
            "IsLatest": false,
            "Size": 38
        },
        {
            "LastModified": "2015-11-09T22:50:50.000Z",
            "VersionId": "null",
            "ETag": "\"d1f45267a863c8392e07d24dd592f1b9\"",
            "StorageClass": "STANDARD",
            "Key": "index.html",
            "Owner": {
                "DisplayName": "my-username",
                "ID":
 "7009a8971cd660687538875e7c86c5b672fe116bd438f46db45460ddcd036c32"
            },
            "IsLatest": false,
            "Size": 533823
        }
    ]
}
```

- Per i dettagli sull'API, consulta ListObjectVersions AWS CLICommand Reference.

`list-objects-v2`

Il seguente esempio di codice mostra come utilizzare`list-objects-v2`.

AWS CLI

Come ottenere un elenco di oggetti in un bucket

L'esempio `list-objects-v2` seguente elenca gli oggetti nel bucket specificato.

```
aws s3api list-objects-v2 \
    --bucket amzn-s3-demo-bucket
```

Output:

```
{
    "Contents": [
        {
            "LastModified": "2019-11-05T23:11:50.000Z",
            "ETag": "\"621503c373607d548b37cff8778d992c\"",
            "StorageClass": "STANDARD",
            "Key": "doc1.rtf",
            "Size": 391
        },
        {
            "LastModified": "2019-11-05T23:11:50.000Z",
            "ETag": "\"a2cecc36ab7c7fe3a71a273b9d45b1b5\"",
            "StorageClass": "STANDARD",
            "Key": "doc2.rtf",
            "Size": 373
        },
        {
            "LastModified": "2019-11-05T23:11:50.000Z",
            "ETag": "\"08210852f65a2e9cb999972539a64d68\"",
            "StorageClass": "STANDARD",
            "Key": "doc3.rtf",
            "Size": 399
        },
        {
            "LastModified": "2019-11-05T23:11:50.000Z",
```

```
            "ETag": "\"d1852dd683f404306569471af106988e\"",
            "StorageClass": "STANDARD",
            "Key": "doc4.rtf",
            "Size": 6225
        }
    ]
}
```

- Per i dettagli sull'API, consulta ListObjectsV2 in AWS CLI Command Reference.

`list-objects`

Il seguente esempio di codice mostra come utilizzare`list-objects`.

AWS CLI

L'esempio seguente utilizza il comando `list-objects` per visualizzare i nomi di tutti gli oggetti del bucket specificato:

```
aws s3api list-objects --bucket text-content --query 'Contents[].{Key: Key, Size:
 Size}'
```

L'esempio utilizza l'argomento `--query` per filtrare l'output di `list-objects` fino al valore della chiave e alla dimensione per ogni oggetto

Per ulteriori informazioni sugli oggetti, consulta Operazioni con gli oggetti di Amazon S3 nella Guida per gli sviluppatori di Amazon S3.

- Per i dettagli sull'API, consulta ListObjects AWS CLICommand Reference.

`list-parts`

Il seguente esempio di codice mostra come utilizzare`list-parts`.

AWS CLI

Il comando seguente elenca tutte le parti che sono state caricate per un caricamento in più parti con chiave `multipart/01` nel bucket `amzn-s3-demo-bucket`:

```
aws s3api list-parts --bucket amzn-s3-demo-bucket --key 'multipart/01' --upload-
id dfRtDYU0WWCCcH43C3WFbkRONycyCpTJJvxu2i5GYkZljF.Yxwh6XG7WfS2vC4to6HiV6Yjlx.cph0gtNBtJ8P3UR
```

Output:

```
{
    "Owner": {
        "DisplayName": "aws-account-name",
        "ID": "100719349fc3b6dcd7c820a124bf7aecd408092c3d7b51b38494939801fc248b"
    },
    "Initiator": {
        "DisplayName": "username",
        "ID": "arn:aws:iam::0123456789012:user/username"
    },
    "Parts": [
        {
            "LastModified": "2015-06-02T18:07:35.000Z",
            "PartNumber": 1,
            "ETag": "\"e868e0f4719e394144ef36531ee6824c\"",
            "Size": 5242880
        },
        {
            "LastModified": "2015-06-02T18:07:42.000Z",
            "PartNumber": 2,
            "ETag": "\"6bb2b12753d66fe86da4998aa33fffb0\"",
            "Size": 5242880
        },
        {
            "LastModified": "2015-06-02T18:07:47.000Z",
            "PartNumber": 3,
            "ETag": "\"d0a0112e841abec9c9ec83406f0159c8\"",
            "Size": 5242880
        }
    ],
    "StorageClass": "STANDARD"
}
```

- Per i dettagli sull'API, consulta ListParts AWS CLICommand Reference.

`ls`

Il seguente esempio di codice mostra come utilizzare`ls`.

AWS CLI

Esempio 1: elenco di tutti i bucket di proprietà dell'utente

Il comando `ls` seguente elenca tutti i bucket di proprietà dell'utente. In questo esempio, l'utente possiede i bucket `amzn-s3-demo-bucket` e `amzn-s3-demo-bucket2`. Il timestamp è la data di creazione del bucket, mostrata nel fuso orario della macchina. Questa data può cambiare quando si apportano modifiche al bucket, ad esempio quando si modifica la policy di bucket. Tieni presente che se `s3://` viene utilizzato per l'argomento path `<S3Uri>`, elencherà anche tutti i bucket.

```
aws s3 ls
```

Output:

```
2013-07-11 17:08:50 amzn-s3-demo-bucket
2013-07-24 14:55:44 amzn-s3-demo-bucket2
```

Esempio 2: elenco di tutti i prefissi e gli oggetti in un bucket

Il comando `ls` seguente elenca gli oggetti e i prefissi comuni in un bucket e un prefisso specificati. In questo esempio, l'utente possiede il bucket `amzn-s3-demo-bucket` con gli oggetti `test.txt` e `somePrefix/test.txt`. `LastWriteTime` e `Length` sono arbitrari. Tieni presente che, poiché il comando `ls` non ha alcuna interazione con il filesystem locale, lo schema URI `s3://` non è necessario per risolvere l'ambiguità e può essere omesso.

```
aws s3 ls s3://amzn-s3-demo-bucket
```

Output:

```
                           PRE somePrefix/
2013-07-25 17:06:27         88 test.txt
```

Esempio 3: elenco di tutti i prefissi e gli oggetti in un bucket e un prefisso specifici

Il comando `ls` seguente elenca gli oggetti e i prefissi comuni in un bucket e un prefisso specificati. Tuttavia, non vi sono oggetti né prefissi comuni nel bucket e nel prefisso specificati.

```
aws s3 ls s3://amzn-s3-demo-bucket/noExistPrefix
```

Output:

```
None
```

Esempio 4: elenco ricorsivo di tutti i prefissi e gli oggetti in un bucket

Il comando `ls` seguente elencherà in modo ricorsivo gli oggetti in un bucket. Invece di visualizzare `PRE dirname/` nell'output, tutto il contenuto di un bucket verrà elencato in ordine.

```
aws s3 ls s3://amzn-s3-demo-bucket \
    --recursive
```

Output:

```
2013-09-02 21:37:53         10 a.txt
2013-09-02 21:37:53    2863288 foo.zip
2013-09-02 21:32:57         23 foo/bar/.baz/a
2013-09-02 21:32:58         41 foo/bar/.baz/b
2013-09-02 21:32:57        281 foo/bar/.baz/c
2013-09-02 21:32:57         73 foo/bar/.baz/d
2013-09-02 21:32:57        452 foo/bar/.baz/e
2013-09-02 21:32:57        896 foo/bar/.baz/hooks/bar
2013-09-02 21:32:57        189 foo/bar/.baz/hooks/foo
2013-09-02 21:32:57        398 z.txt
```

Esempio 5: riepilogo di tutti i prefissi e gli oggetti in un bucket

Il `ls` comando seguente illustra lo stesso comando utilizzando le opzioni --human-readable e --summarize. Bytes/MiB/KiB/GiB/TiB/PiB/EiB--human-readable visualizza la dimensione del file in. --summarize mostra il numero totale di oggetti e la dimensione totale alla fine dell'elenco dei risultati:

```
aws s3 ls s3://amzn-s3-demo-bucket \
    --recursive \
    --human-readable \
    --summarize
```

Output:

```
2013-09-02 21:37:53   10 Bytes a.txt
2013-09-02 21:37:53  2.9 MiB foo.zip
2013-09-02 21:32:57   23 Bytes foo/bar/.baz/a
2013-09-02 21:32:58   41 Bytes foo/bar/.baz/b
2013-09-02 21:32:57  281 Bytes foo/bar/.baz/c
2013-09-02 21:32:57   73 Bytes foo/bar/.baz/d
```

```
2013-09-02 21:32:57  452 Bytes foo/bar/.baz/e
2013-09-02 21:32:57  896 Bytes foo/bar/.baz/hooks/bar
2013-09-02 21:32:57  189 Bytes foo/bar/.baz/hooks/foo
2013-09-02 21:32:57  398 Bytes z.txt

Total Objects: 10
   Total Size: 2.9 MiB
```

Esempio 6: elenco da un punto di accesso S3

Il comando `ls` seguente elenca gli oggetti dal punto di accesso (`myaccesspoint`):

```
aws s3 ls s3://arn:aws:s3:us-west-2:123456789012:accesspoint/myaccesspoint/
```

Output:

```
                           PRE somePrefix/
2013-07-25 17:06:27         88 test.txt
```

- Per informazioni dettagliate sull'API, consulta Ls in AWS CLI Command Reference.

mb

Il seguente esempio di codice mostra come usare. `mb`

AWS CLI

Esempio 1: creare un bucket

Il comando `mb` seguente crea un gruppo di bucket. In questo esempio, l'utente crea il bucket `amzn-s3-demo-bucket`. Il bucket viene creato nella Regione specificata nel file di configurazione dell'utente:

```
aws s3 mb s3://amzn-s3-demo-bucket
```

Output:

```
make_bucket: s3://amzn-s3-demo-bucket
```

Esempio 2: crea un bucket nella Regione specificata

Il comando mb seguente crea un bucket in una Regione specificata dal parametro `--region`. In questo esempio, l'utente crea il bucket `amzn-s3-demo-bucket` nella Regione `us-west-1`:

```
aws s3 mb s3://amzn-s3-demo-bucket \
    --region us-west-1
```

Output:

```
make_bucket: s3://amzn-s3-demo-bucket
```

- Per informazioni dettagliate sull'API, consulta Mb in AWS CLI Command Reference.

mv

Il seguente esempio di codice mostra come utilizzaremv.

AWS CLI

Esempio 1: sposta un file locale nel bucket specificato

Il comando mv seguente sposta un singolo file in un bucket e una chiave specificati.

```
aws s3 mv test.txt s3://amzn-s3-demo-bucket/test2.txt
```

Output:

```
move: test.txt to s3://amzn-s3-demo-bucket/test2.txt
```

Esempio 2: sposta un oggetto nel bucket e nella chiave specificati

Il comando mv seguente sposta un singolo oggetto s3 in un bucket e una chiave specificati.

```
aws s3 mv s3://amzn-s3-demo-bucket/test.txt s3://amzn-s3-demo-bucket/test2.txt
```

Output:

```
move: s3://amzn-s3-demo-bucket/test.txt to s3://amzn-s3-demo-bucket/test2.txt
```

Esempio 3: sposta un oggetto S3 nella directory locale

Il comando mv seguente sposta in locale un singolo oggetto in un file specificato.

```
aws s3 mv s3://amzn-s3-demo-bucket/test.txt test2.txt
```

Output:

```
move: s3://amzn-s3-demo-bucket/test.txt to test2.txt
```

Esempio 4: sposta un oggetto con il suo nome originale nel bucket specificato

Il comando mv seguente sposta un singolo oggetto in un bucket specificato mantenendo il nome originale:

```
aws s3 mv s3://amzn-s3-demo-bucket/test.txt s3://amzn-s3-demo-bucket2/
```

Output:

```
move: s3://amzn-s3-demo-bucket/test.txt to s3://amzn-s3-demo-bucket2/test.txt
```

Esempio 5: sposta tutti gli oggetti e i prefissi di un bucket nella directory locale

Se passato con il parametro `--recursive`, il comando mv seguente sposta in modo ricorsivo tutti gli oggetti con un prefisso e un bucket specificati in una directory specificata. In questo esempio, il bucket `amzn-s3-demo-bucket` contiene gli oggetti `test1.txt` e `test2.txt`.

```
aws s3 mv s3://amzn-s3-demo-bucket . \
    --recursive
```

Output:

```
move: s3://amzn-s3-demo-bucket/test1.txt to test1.txt
move: s3://amzn-s3-demo-bucket/test2.txt to test2.txt
```

Esempio 6: sposta tutti gli oggetti e i prefissi di un bucket nella directory locale, ad eccezione dei file ".jpg"

Se passato con il parametro `--recursive`, il comando mv seguente sposta in modo ricorsivo tutti i file in una directory specificata in un bucket e un prefisso specificati, eccetto alcuni file

esclusi utilizzando un parametro `--exclude`. In questo esempio, la directory `myDir` contiene i file `test1.txt` e `test2.jpg`.

```
aws s3 mv myDir s3://amzn-s3-demo-bucket/ \
    --recursive \
    --exclude "*.jpg"
```

Output:

```
move: myDir/test1.txt to s3://amzn-s3-demo-bucket2/test1.txt
```

Esempio 7: sposta tutti gli oggetti e i prefissi in un bucket nella directory locale, eccetto il prefisso specificato

Se passato con il parametro `--recursive`, il comando `mv` seguente sposta in modo ricorsivo tutti gli oggetti contenuti in un bucket specificato in un altro bucket, eccetto alcuni oggetti esclusi utilizzando un parametro `--exclude`. In questo esempio, il bucket `amzn-s3-demo-bucket` contiene gli oggetti `test1.txt` e `another/test1.txt`.

```
aws s3 mv s3://amzn-s3-demo-bucket/ s3://amzn-s3-demo-bucket2/ \
    --recursive \
    --exclude "amzn-s3-demo-bucket/another/*"
```

Output:

```
move: s3://amzn-s3-demo-bucket/test1.txt to s3://amzn-s3-demo-bucket2/test1.txt
```

Esempio 8: sposta un oggetto nel bucket specificato e imposta l'ACL

Il comando `mv` seguente sposta un singolo oggetto in un bucket e una chiave specificati mentre si imposta l'ACL su `public-read-write`.

```
aws s3 mv s3://amzn-s3-demo-bucket/test.txt s3://amzn-s3-demo-bucket/test2.txt \
    --acl public-read-write
```

Output:

```
move: s3://amzn-s3-demo-bucket/test.txt to s3://amzn-s3-demo-bucket/test2.txt
```

Esempio 9: sposta un file locale nel bucket specificato e concedi le autorizzazioni

Il comando mv seguente illustra l'utilizzo dell'opzione `--grants` per concedere l'accesso in lettura a tutti gli utenti e il controllo completo a un utente specifico identificato dal relativo indirizzo e-mail.

```
aws s3 mv file.txt s3://amzn-s3-demo-bucket/ \
    --grants read=uri=http://acs.amazonaws.com/groups/global/
AllUsers full=emailaddress=user@example.com
```

Output:

```
move: file.txt to s3://amzn-s3-demo-bucket/file.txt
```

Esempio 10: sposta un file su un punto di accesso S3

Il comando mv seguente sposta un singolo file denominato `mydoc.txt` nel punto di accesso denominato `myaccesspoint` con la chiave denominata `mykey`.

```
aws s3 mv mydoc.txt s3://arn:aws:s3:us-west-2:123456789012:accesspoint/
myaccesspoint/mykey
```

Output:

```
move: mydoc.txt to s3://arn:aws:s3:us-west-2:123456789012:accesspoint/myaccesspoint/
mykey
```

- Per informazioni dettagliate sull'API, consulta Mv in AWS CLI Command Reference.

`presign`

Il seguente esempio di codice mostra come utilizzare`presign`.

AWS CLI

Esempio 1: come creare un URL prefirmato con la durata predefinita di un'ora che rimanda a un oggetto in un bucket S3

Il comando `presign` seguente genera un URL prefirmato, valido per un'ora, per un bucket e una chiave specificati.

```
aws s3 presign s3://amzn-s3-demo-bucket/test2.txt
```

Output:

```
https://amzn-s3-demo-bucket.s3.us-west-2.amazonaws.com/key?X-Amz-Algorithm=AWS4-
HMAC-SHA256&X-Amz-Credential=AKIAEXAMPLE123456789%2F20210621%2Fus-
west-2%2Fs3%2Faws4_request&X-Amz-Date=20210621T041609Z&X-Amz-Expires=3600&X-Amz-
SignedHeaders=host&X-Amz-
Signature=EXAMBLE1234494d5fba3fed607f98018e1dfc62e2529ae96d844123456
```

Esempio 2: come creare un URL prefirmato con una durata personalizzata che rimanda a un oggetto in un bucket S3

Il comando `presign` seguente genera un URL prefirmato per un bucket e una chiave specificati validi per una settimana.

```
aws s3 presign s3://amzn-s3-demo-bucket/test2.txt \
    --expires-in 604800
```

Output:

```
https://amzn-s3-demo-bucket.s3.us-west-2.amazonaws.com/key?X-Amz-Algorithm=AWS4-
HMAC-SHA256&X-Amz-Credential=AKIAEXAMPLE123456789%2F20210621%2Fus-
west-2%2Fs3%2Faws4_request&X-Amz-Date=20210621T041609Z&X-Amz-Expires=604800&X-Amz-
SignedHeaders=host&X-Amz-
Signature=EXAMBLE1234494d5fba3fed607f98018e1dfc62e2529ae96d844123456
```

Per ulteriori informazioni, consulta Condivisione di un oggetto con altri nella Guida per sviluppatori di S3.

- Per informazioni dettagliate sull'API, consulta Presign in AWS CLI Command Reference.

`put-bucket-accelerate-configuration`

Il seguente esempio di codice mostra come utilizzare`put-bucket-accelerate-configuration`.

AWS CLI

Come impostare la configurazione di accelerazione di un bucket

L'esempio `put-bucket-accelerate-configuration` seguente abilita la configurazione di accelerazione per il bucket specificato.

```
aws s3api put-bucket-accelerate-configuration \
    --bucket amzn-s3-demo-bucket \
    --accelerate-configuration Status=Enabled
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta PutBucketAccelerateConfiguration AWS CLICommand Reference.

`put-bucket-acl`

Il seguente esempio di codice mostra come utilizzare`put-bucket-acl`.

AWS CLI

Questo esempio concede `full control` a due AWS utenti (user1@example.com e user2@example.com) `read` l'autorizzazione a tutti:

```
aws s3api put-bucket-acl --bucket amzn-s3-demo-bucket --grant-full-
control emailaddress=user1@example.com,emailaddress=user2@example.com --grant-
read uri=http://acs.amazonaws.com/groups/global/AllUsers
```

Vedi http://docs.aws.amazon. com/AmazonS3/latest/API/RESTBucketPUTacl.html per i dettagli sulla personalizzazione ACLs (i comandi ACL s3api, ad esempio`put-bucket-acl`, utilizzano la stessa notazione abbreviata degli argomenti).

- Per i dettagli sull'API, consulta Command Reference. PutBucketAclAWS CLI

`put-bucket-analytics-configuration`

Il seguente esempio di codice mostra come utilizzare`put-bucket-analytics-configuration`.

AWS CLI

Come impostare una configurazione di analisi per il bucket

L'esempio `put-bucket-analytics-configuration` seguente configura l'analisi per il bucket specificato.

```
aws s3api put-bucket-analytics-configuration \
    --bucket amzn-s3-demo-bucket --id 1 \
```

```
    --analytics-configuration '{"Id": "1","StorageClassAnalysis": {}}'
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta PutBucketAnalyticsConfiguration AWS CLICommand Reference.

`put-bucket-cors`

Il seguente esempio di codice mostra come utilizzare`put-bucket-cors`.

AWS CLI

L'esempio seguente abilita le richieste `PUT`, `POST` e `DELETE` da www.example.com e abilita le richieste `GET` da qualsiasi dominio:

```
aws s3api put-bucket-cors --bucket amzn-s3-demo-bucket --cors-configuration file://
cors.json

cors.json:
{
  "CORSRules": [
    {
      "AllowedOrigins": ["http://www.example.com"],
      "AllowedHeaders": ["*"],
      "AllowedMethods": ["PUT", "POST", "DELETE"],
      "MaxAgeSeconds": 3000,
      "ExposeHeaders": ["x-amz-server-side-encryption"]
    },
    {
      "AllowedOrigins": ["*"],
      "AllowedHeaders": ["Authorization"],
      "AllowedMethods": ["GET"],
      "MaxAgeSeconds": 3000
    }
  ]
}
```

- Per i dettagli sull'API, consulta PutBucketCors AWS CLICommand Reference.

`put-bucket-encryption`

Il seguente esempio di codice mostra come utilizzare`put-bucket-encryption`.

AWS CLI

Come configurare la crittografia lato server per un bucket

L'`put-bucket-encryption`esempio seguente imposta la AES256 crittografia come predefinita per il bucket specificato.

```
aws s3api put-bucket-encryption \
    --bucket amzn-s3-demo-bucket \
    --server-side-encryption-configuration '{"Rules":
 [{"ApplyServerSideEncryptionByDefault": {"SSEAlgorithm": "AES256"}}]}'
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, vedere PutBucketEncryptionin AWS CLI Command Reference.

`put-bucket-intelligent-tiering-configuration`

Il seguente esempio di codice mostra come utilizzare`put-bucket-intelligent-tiering-configuration`.

AWS CLI

Come aggiornare una configurazione S3 Intelligent-Tiering su un bucket

L'`put-bucket-intelligent-tiering-configuration`esempio seguente aggiorna una configurazione S3 Intelligent-Tiering, denominata ExampleConfig, su un bucket. La configurazione passerà gli oggetti a cui non è stato effettuato l'accesso nelle immagini del prefisso al livello Accesso di archiviazione dopo 90 giorni e al livello Accesso di archiviazione profonda dopo 180 giorni.

```
aws s3api put-bucket-intelligent-tiering-configuration \
    --bucket amzn-s3-demo-bucket \
    --id "ExampleConfig" \
    --intelligent-tiering-configuration file://intelligent-tiering-
configuration.json
```

Contenuto di `intelligent-tiering-configuration.json`:

```
{
    "Id": "ExampleConfig",
```

```
    "Status": "Enabled",
    "Filter": {
        "Prefix": "images"
        },
    "Tierings": [
        {
            "Days": 90,
            "AccessTier": "ARCHIVE_ACCESS"
        },
        {
            "Days": 180,
            "AccessTier": "DEEP_ARCHIVE_ACCESS"
        }
    ]
}
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Impostazione della proprietà degli oggetti su un bucket esistente nella Guida per l'utente di Amazon S3.

- Per i dettagli sull'API, consulta Command Reference. PutBucketIntelligentTieringConfigurationAWS CLI

`put-bucket-inventory-configuration`

Il seguente esempio di codice mostra come utilizzare`put-bucket-inventory-configuration`.

AWS CLI

Esempio 1: come impostare una configurazione di inventario per un bucket

L'esempio `put-bucket-inventory-configuration` seguente imposta un report d'inventario settimanale in formato ORC per il bucket `amzn-s3-demo-bucket`.

```
aws s3api put-bucket-inventory-configuration \
    --bucket amzn-s3-demo-bucket \
    --id 1 \
    --inventory-configuration '{"Destination": { "S3BucketDestination":
 { "AccountId": "123456789012", "Bucket": "arn:aws:s3:::amzn-s3-demo-bucket",
 "Format": "ORC" }}, "IsEnabled": true, "Id": "1", "IncludedObjectVersions":
 "Current", "Schedule": { "Frequency": "Weekly" }}'
```

Questo comando non produce alcun output.

Esempio 2: come impostare una configurazione di inventario per un bucket

L'esempio `put-bucket-inventory-configuration` seguente imposta un report d'inventario giornaliero in formato CSV per il bucket `amzn-s3-demo-bucket`.

```
aws s3api put-bucket-inventory-configuration \
    --bucket amzn-s3-demo-bucket \
    --id 2 \
    --inventory-configuration '{"Destination": { "S3BucketDestination":
 { "AccountId": "123456789012", "Bucket": "arn:aws:s3:::amzn-s3-demo-bucket",
 "Format": "CSV" }}, "IsEnabled": true, "Id": "2", "IncludedObjectVersions":
 "Current", "Schedule": { "Frequency": "Daily" }}'
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta PutBucketInventoryConfiguration AWS CLICommand Reference.

`put-bucket-lifecycle-configuration`

Il seguente esempio di codice mostra come utilizzare`put-bucket-lifecycle-configuration`.

AWS CLI

Il comando seguente applica una configurazione del ciclo di vita da un bucket denominato `amzn-s3-demo-bucket`.

```
aws s3api put-bucket-lifecycle-configuration --bucket amzn-s3-demo-bucket --
lifecycle-configuration  file://lifecycle.json
```

Il file `lifecycle.json` è un documento JSON nella cartella corrente che specifica due regole:

```
{
    "Rules": [
        {
            "ID": "Move rotated logs to Glacier",
            "Prefix": "rotated/",
            "Status": "Enabled",
            "Transitions": [
                {
```

```
                    "Date": "2015-11-10T00:00:00.000Z",
                    "StorageClass": "GLACIER"
                }
            ]
        },
        {
            "Status": "Enabled",
            "Prefix": "",
            "NoncurrentVersionTransitions": [
                {
                    "NoncurrentDays": 2,
                    "StorageClass": "GLACIER"
                }
            ],
            "ID": "Move old versions to Glacier"
        }
    ]
}
```

La prima regola sposta i file con il prefisso `rotated` su Glacier nella data specificata. La seconda regola sposta le vecchie versioni degli oggetti su Glacier quando non sono più attuali. Per informazioni sui formati di timestamp accettabili, consulta "Specifica dei valori di parametro" nella Guida per l'utente dell'Interfaccia della linea di comando AWS (AWS CLI).

- Per i dettagli sull'API, consulta PutBucketLifecycleConfiguration AWS CLICommand Reference.

`put-bucket-lifecycle`

Il seguente esempio di codice mostra come utilizzare`put-bucket-lifecycle`.

AWS CLI

Il comando seguente applica una configurazione del ciclo di vita al bucket `amzn-s3-demo-bucket`:

```
aws s3api put-bucket-lifecycle --bucket amzn-s3-demo-bucket --lifecycle-
configuration file://lifecycle.json
```

Il file `lifecycle.json` è un documento JSON nella cartella corrente che specifica due regole:

```
{
  "Rules": [
```

```
    {
      "ID": "Move to Glacier after sixty days (objects in logs/2015/)",
      "Prefix": "logs/2015/",
      "Status": "Enabled",
      "Transition": {
        "Days": 60,
        "StorageClass": "GLACIER"
      }
    },
    {
      "Expiration": {
        "Date": "2016-01-01T00:00:00.000Z"
      },
      "ID": "Delete 2014 logs in 2016.",
      "Prefix": "logs/2014/",
      "Status": "Enabled"
    }
  ]
}
```

La prima regola sposta i file in Amazon Glacier dopo sessanta giorni. La seconda regola elimina i file da Amazon S3 alla data specificata. Per informazioni sui formati di timestamp accettabili, consulta "Specifica dei valori di parametro" nella Guida per l'utente dell'Interfaccia della linea di comando AWS (AWS CLI).

Ogni regola nell'esempio precedente specifica una policy (`Transition` o `Expiration`) e un prefisso del file (nome della cartella) a cui si applica. Puoi anche creare una regola che si applica a un intero bucket specificando un prefisso vuoto:

```
{
  "Rules": [
    {
      "ID": "Move to Glacier after sixty days (all objects in bucket)",
      "Prefix": "",
      "Status": "Enabled",
      "Transition": {
        "Days": 60,
        "StorageClass": "GLACIER"
      }
    }
  ]
}
```

- Per i dettagli sull'API, consulta PutBucketLifecycle AWS CLICommand Reference.

`put-bucket-logging`

Il seguente esempio di codice mostra come utilizzare`put-bucket-logging`.

AWS CLI

Esempio 1: come impostare la registrazione delle policy di bucket

L'esempio `put-bucket-logging` seguente imposta la policy di registrazione per amzn-s3-demo-bucket. Innanzitutto, concedi al servizio di registrazione di log l'autorizzazione principale nella policy di bucket con il comando `put-bucket-policy`.

```
aws s3api put-bucket-policy \
    --bucket amzn-s3-demo-bucket \
    --policy file://policy.json
```

Contenuto di `policy.json`:

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "S3ServerAccessLogsPolicy",
            "Effect": "Allow",
            "Principal": {"Service": "logging.s3.amazonaws.com"},
            "Action": "s3:PutObject",
            "Resource": "arn:aws:s3:::amzn-s3-demo-bucket/Logs/*",
            "Condition": {
                "ArnLike": {"aws:SourceARN": "arn:aws:s3:::SOURCE-BUCKET-NAME"},
                "StringEquals": {"aws:SourceAccount": "SOURCE-AWS-ACCOUNT-ID"}
            }
        }
    ]
}
```

Per applicare la policy di registrazione di log, utilizza `put-bucket-logging`.

```
aws s3api put-bucket-logging \
```

```
    --bucket amzn-s3-demo-bucket \
    --bucket-logging-status file://logging.json
```

Contenuto di `logging.json`:

```
{
     "LoggingEnabled": {
         "TargetBucket": "amzn-s3-demo-bucket",
         "TargetPrefix": "Logs/"
     }
 }
```

Il comando `put-bucket-policy` è necessario per concedere le autorizzazioni `s3:PutObject` al principale del servizio di registrazione di log.

Per ulteriori informazioni, consulta Registrazione degli accessi al server Amazon S3 nella Guida per l'utente di Amazon S3.

Esempio 2: come impostare una policy di bucket per la registrazione dell'accesso su un unico utente

L'esempio `put-bucket-logging` seguente imposta la policy di registrazione per amzn-s3-demo-bucket. L' AWS utente bob@example.com avrà il pieno controllo sui file di registro e nessun altro potrà accedervi. Concedi innanzitutto l'autorizzazione S3 con `put-bucket-acl`.

```
aws s3api put-bucket-acl \
    --bucket amzn-s3-demo-bucket \
    --grant-write URI=http://acs.amazonaws.com/groups/s3/LogDelivery \
    --grant-read-acp URI=http://acs.amazonaws.com/groups/s3/LogDelivery
```

Applica quindi la policy di registrazione di log con `put-bucket-logging`.

```
aws s3api put-bucket-logging \
    --bucket amzn-s3-demo-bucket \
    --bucket-logging-status file://logging.json
```

Contenuto di `logging.json`:

```
{
```

```
    "LoggingEnabled": {
        "TargetBucket": "amzn-s3-demo-bucket",
        "TargetPrefix": "amzn-s3-demo-bucket-logs/",
        "TargetGrants": [
            {
                "Grantee": {
                    "Type": "AmazonCustomerByEmail",
                    "EmailAddress": "bob@example.com"
                },
                "Permission": "FULL_CONTROL"
            }
        ]
    }
}
```

Il comando `put-bucket-acl` è necessario per concedere al sistema di distribuzione dei log S3 le autorizzazioni necessarie (autorizzazioni di scrittura e read-acp).

Per ulteriori informazioni, consulta Registrazione degli accessi al server Amazon S3 nella Guida per gli sviluppatori di Amazon S3.

- Per i dettagli sull'API, consulta PutBucketLogging AWS CLICommand Reference.

`put-bucket-metrics-configuration`

Il seguente esempio di codice mostra come utilizzare`put-bucket-metrics-configuration`.

AWS CLI

Come eliminare una configurazione delle metriche per un bucket

L'esempio `put-bucket-metrics-configuration` seguente imposta una configurazione della metrica con ID 123 per il bucket specificato.

```
aws s3api put-bucket-metrics-configuration \
    --bucket amzn-s3-demo-bucket \
    --id 123 \
    --metrics-configuration '{"Id": "123", "Filter": {"Prefix": "logs"}}'
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta PutBucketMetricsConfiguration AWS CLICommand Reference.

`put-bucket-notification-configuration`

Il seguente esempio di codice mostra come utilizzare`put-bucket-notification-configuration`.

AWS CLI

Come abilitare le notifiche specificate in un bucket

L'esempio `put-bucket-notification-configuration` seguente applica una configurazione di notifica a un bucket denominato `amzn-s3-demo-bucket`. Il file `notification.json` è un documento JSON nella cartella corrente che specifica un argomento SNS e un tipo di evento da monitorare.

```
aws s3api put-bucket-notification-configuration \
    --bucket amzn-s3-demo-bucket \
    --notification-configuration file://notification.json
```

Contenuto di `notification.json`:

```
{
    "TopicConfigurations": [
        {
            "TopicArn": "arn:aws:sns:us-west-2:123456789012:s3-notification-topic",
            "Events": [
                "s3:ObjectCreated:*"
            ]
        }
    ]
}
```

All'argomento SNS deve essere collegata una policy IAM che consenta ad Amazon S3 di effettuarvi pubblicazioni.

```
{
    "Version":"2012-10-17",
    "Id": "example-ID",
    "Statement": [
        {
            "Sid": "example-statement-ID",
            "Effect": "Allow",
            "Principal": {
```

```
                "Service": "s3.amazonaws.com"
            },
            "Action": [
                "SNS:Publish"
            ],
            "Resource": "arn:aws:sns:us-west-2:123456789012::s3-notification-topic",
            "Condition": {
                "ArnLike": {
                    "aws:SourceArn": "arn:aws:s3:*:*:amzn-s3-demo-bucket"
                }
            }
        }
    ]
}
```

- Per i dettagli sull'API, consulta PutBucketNotificationConfiguration AWS CLICommand Reference.

`put-bucket-notification`

Il seguente esempio di codice mostra come utilizzare`put-bucket-notification`.

AWS CLI

Applica una configurazione di notifica a un bucket denominato `amzn-s3-demo-bucket`:

```
aws s3api put-bucket-notification --bucket amzn-s3-demo-bucket --notification-
configuration file://notification.json
```

Il file `notification.json` è un documento JSON nella cartella corrente che specifica un argomento SNS e un tipo di evento da monitorare.

```
{
  "TopicConfiguration": {
    "Event": "s3:ObjectCreated:*",
    "Topic": "arn:aws:sns:us-west-2:123456789012:s3-notification-topic"
  }
}
```

All'argomento SNS deve essere collegata una policy IAM che consenta ad Amazon S3 di effettuarvi pubblicazioni.

```
{
 "Version":"2012-10-17",
 "Id": "example-ID",
 "Statement": [
  {
   "Sid": "example-statement-ID",
   "Effect": "Allow",
   "Principal": {
     "Service": "s3.amazonaws.com"
   },
   "Action": [
    "SNS:Publish"
   ],
   "Resource": "arn:aws:sns:us-west-2:123456789012:amzn-s3-demo-bucket",
   "Condition": {
      "ArnLike": {
      "aws:SourceArn": "arn:aws:s3:*:*:amzn-s3-demo-bucket"
    }
   }
  }
 ]
}
```

- Per i dettagli sull'API, consulta PutBucketNotification AWS CLICommand Reference.

`put-bucket-ownership-controls`

Il seguente esempio di codice mostra come utilizzare`put-bucket-ownership-controls`.

AWS CLI

Come aggiornare le impostazioni di proprietà del bucket di un bucket

L'esempio `put-bucket-ownership-controls` seguente aggiorna le impostazioni di proprietà del bucket di un bucket.

```
aws s3api put-bucket-ownership-controls \
    --bucket amzn-s3-demo-bucket \
    --ownership-controls="Rules=[{ObjectOwnership=BucketOwnerEnforced}]"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Impostazione della proprietà degli oggetti su un bucket esistente nella Guida per l'utente di Amazon S3.

- Per i dettagli sull'API, consulta PutBucketOwnershipControls AWS CLICommand Reference.

`put-bucket-policy`

Il seguente esempio di codice mostra come utilizzare`put-bucket-policy`.

AWS CLI

Questo esempio consente a tutti gli utenti di recuperare qualsiasi oggetto in amzn-s3-demo-bucket tranne quelli in. MySecretFolder Inoltre concede `put` l'autorizzazione all'utente root dell'account: `delete` AWS `1234-5678-9012`

```
aws s3api put-bucket-policy --bucket amzn-s3-demo-bucket --policy file://policy.json

policy.json:
{
   "Statement": [
      {
         "Effect": "Allow",
         "Principal": "*",
         "Action": "s3:GetObject",
         "Resource": "arn:aws:s3:::amzn-s3-demo-bucket/*"
      },
      {
         "Effect": "Deny",
         "Principal": "*",
         "Action": "s3:GetObject",
         "Resource": "arn:aws:s3:::amzn-s3-demo-bucket/MySecretFolder/*"
      },
      {
         "Effect": "Allow",
         "Principal": {
            "AWS": "arn:aws:iam::123456789012:root"
         },
         "Action": [
            "s3:DeleteObject",
            "s3:PutObject"
         ],
         "Resource": "arn:aws:s3:::amzn-s3-demo-bucket/*"
      }
```

```
  ]
}
```

- Per i dettagli sull'API, consulta PutBucketPolicy AWS CLICommand Reference.

`put-bucket-replication`

Il seguente esempio di codice mostra come utilizzare`put-bucket-replication`.

AWS CLI

Come configurare la replica per un bucket S3

L'esempio `put-bucket-replication` seguente applica una configurazione di replica al bucket S3 specificato.

```
aws s3api put-bucket-replication \
    --bucket amzn-s3-demo-bucket1 \
    --replication-configuration file://replication.json
```

Contenuto di `replication.json`:

```
{
    "Role": "arn:aws:iam::123456789012:role/s3-replication-role",
    "Rules": [
        {
            "Status": "Enabled",
            "Priority": 1,
            "DeleteMarkerReplication": { "Status": "Disabled" },
            "Filter" : { "Prefix": ""},
            "Destination": {
                "Bucket": "arn:aws:s3:::amzn-s3-demo-bucket2"
            }
        }
    ]
}
```

Il bucket di destinazione deve avere il controllo delle versioni abilitato. Il ruolo specificato deve disporre dell'autorizzazione di scrittura nel bucket di destinazione e di una relazione di attendibilità che consenta ad Amazon S3 di assumere il ruolo.

Esempio di policy di autorizzazione di un ruolo:

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "s3:GetReplicationConfiguration",
                "s3:ListBucket"
            ],
            "Resource": [
                "arn:aws:s3:::amzn-s3-demo-bucket1"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "s3:GetObjectVersion",
                "s3:GetObjectVersionAcl",
                "s3:GetObjectVersionTagging"
            ],
            "Resource": [
                "arn:aws:s3:::amzn-s3-demo-bucket1/*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "s3:ReplicateObject",
                "s3:ReplicateDelete",
                "s3:ReplicateTags"
            ],
            "Resource": "arn:aws:s3:::amzn-s3-demo-bucket2/*"
        }
    ]
}
```

Esempio di policy delle relazioni di attendibilità:

```
{
    "Version":"2012-10-17",
    "Statement": [
```

```
        {
            "Effect": "Allow",
            "Principal": {
                "Service": "s3.amazonaws.com"
            },
            "Action": "sts:AssumeRole"
        }
    ]
}
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta il seguente argomento nella Guida per l'utente della console Amazon Simple Storage Service.

- Per i dettagli sull'API, consulta PutBucketReplication AWS CLICommand Reference.

`put-bucket-request-payment`

Il seguente esempio di codice mostra come utilizzare`put-bucket-request-payment`.

AWS CLI

Esempio 1: come abilitare la configurazione "requester pays" per un bucket

L'esempio `put-bucket-request-payment` seguente abilita `requester pays` per il bucket specificato.

```
aws s3api put-bucket-request-payment \
    --bucket amzn-s3-demo-bucket \
    --request-payment-configuration '{"Payer":"Requester"}'
```

Questo comando non produce alcun output.

Esempio 2: come disabilitare la configurazione "requester pays" per un bucket

L'esempio `put-bucket-request-payment` seguente disabilita `requester pays` per il bucket specificato.

```
aws s3api put-bucket-request-payment \
    --bucket amzn-s3-demo-bucket \
```

```
    --request-payment-configuration '{"Payer":"BucketOwner"}'
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta PutBucketRequestPayment AWS CLICommand Reference.

put-bucket-tagging

Il seguente esempio di codice mostra come utilizzare`put-bucket-tagging`.

AWS CLI

Il comando seguente applica una configurazione di tagging da un bucket denominato `amzn-s3-demo-bucket`.

```
aws s3api put-bucket-tagging --bucket amzn-s3-demo-bucket --tagging file://
tagging.json
```

Il file `tagging.json` è un documento JSON nella cartella corrente in cui sono specificati i tag.

```
{
   "TagSet": [
     {
       "Key": "organization",
       "Value": "marketing"
     }
   ]
}
```

In alternativa, applica una configurazione di tagging ad `amzn-s3-demo-bucket` direttamente dalla riga di comando.

```
aws s3api put-bucket-tagging --bucket amzn-s3-demo-bucket --tagging
 'TagSet=[{Key=organization,Value=marketing}]'
```

- Per i dettagli sull'API, consulta PutBucketTagging AWS CLICommand Reference.

put-bucket-versioning

Il seguente esempio di codice mostra come utilizzare`put-bucket-versioning`.

AWS CLI

Il comando seguente abilita il controllo delle versioni sul bucket denominato `amzn-s3-demo-bucket`.

```
aws s3api put-bucket-versioning --bucket amzn-s3-demo-bucket --versioning-
configuration Status=Enabled
```

Il comando seguente abilita il controllo delle versioni e utilizza una modalità di autenticazione a più fattori (MFA).

```
aws s3api put-bucket-versioning --bucket amzn-s3-demo-bucket --versioning-
configuration Status=Enabled --mfa "SERIAL 123456"
```

- Per i dettagli sull'API, consulta PutBucketVersioning AWS CLICommand Reference.

`put-bucket-website`

Il seguente esempio di codice mostra come utilizzare`put-bucket-website`.

AWS CLI

Applica una configurazione statica del sito web a un bucket denominato `amzn-s3-demo-bucket`.

```
aws s3api put-bucket-website --bucket amzn-s3-demo-bucket --website-
configuration file://website.json
```

Il file `website.json` è un documento JSON nella cartella corrente che specifica le pagine dell'indice e degli errori per il sito web:

```
{
    "IndexDocument": {
        "Suffix": "index.html"
    },
    "ErrorDocument": {
        "Key": "error.html"
    }
}
```

- Per i dettagli sull'API, consulta PutBucketWebsite AWS CLICommand Reference.

`put-object-acl`

Il seguente esempio di codice mostra come utilizzare`put-object-acl`.

AWS CLI

Il comando seguente concede `full control` a due AWS utenti (user1@example.com e user2@example.com) `read` l'autorizzazione a tutti:

```
aws s3api put-object-acl --bucket amzn-s3-demo-bucket --key file.txt --grant-full-
control emailaddress=user1@example.com,emailaddress=user2@example.com --grant-
read uri=http://acs.amazonaws.com/groups/global/AllUsers
```

Vedi http://docs.aws.amazon. com/AmazonS3/latest/API/RESTBucketPUTacl.html per i dettagli sulla personalizzazione ACLs (i comandi ACL s3api, ad esempio`put-object-acl`, utilizzano la stessa notazione abbreviata degli argomenti).

- Per i dettagli sull'API, consulta Command Reference. PutObjectAclAWS CLI

`put-object-legal-hold`

Il seguente esempio di codice mostra come utilizzare`put-object-legal-hold`.

AWS CLI

Come applicare lo stato relativo alla conservazione a fini legali a un oggetto

L'esempio `put-object-legal-hold` seguente imposta un blocco a fini legali sull'oggetto `doc1.rtf`.

```
aws s3api put-object-legal-hold \
    --bucket amzn-s3-demo-bucket-with-object-lock \
    --key doc1.rtf \
    --legal-hold Status=ON
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta PutObjectLegalHold AWS CLICommand Reference.

`put-object-lock-configuration`

Il seguente esempio di codice mostra come utilizzare`put-object-lock-configuration`.

AWS CLI

Come impostare la configurazione Object Lock di un bucket

L'esempio `put-object-lock-configuration` seguente imposta un Object Lock di 50 giorni sul bucket specificato.

```
aws s3api put-object-lock-configuration \
    --bucket amzn-s3-demo-bucket-with-object-lock \
    --object-lock-configuration '{ "ObjectLockEnabled": "Enabled", "Rule":
 { "DefaultRetention": { "Mode": "COMPLIANCE", "Days": 50 }}}'
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta PutObjectLockConfiguration AWS CLICommand Reference.

`put-object-retention`

Il seguente esempio di codice mostra come utilizzare`put-object-retention`.

AWS CLI

Come impostare la configurazione di conservazione degli oggetti per un oggetto

L'esempio `put-object-retention` seguente imposta una configurazione di conservazione degli oggetti per l'oggetto specificato fino al giorno 01/01/2025.

```
aws s3api put-object-retention \
    --bucket amzn-s3-demo-bucket-with-object-lock \
    --key doc1.rtf \
    --retention '{ "Mode": "GOVERNANCE", "RetainUntilDate": "2025-01-01T00:00:00" }'
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta PutObjectRetention AWS CLICommand Reference.

`put-object-tagging`

Il seguente esempio di codice mostra come utilizzare`put-object-tagging`.

AWS CLI

Come impostare un tag su un oggetto

L'esempio `put-object-tagging` seguente assegna un tag con la chiave `designation` e il valore `confidential` sull'oggetto specificato.

```
aws s3api put-object-tagging \
    --bucket amzn-s3-demo-bucket \
    --key doc1.rtf \
    --tagging '{"TagSet": [{ "Key": "designation", "Value": "confidential" }]}'
```

Questo comando non produce alcun output.

L'esempio `put-object-tagging` seguente imposta diversi set di tag sull'oggetto specificato.

```
aws s3api put-object-tagging \
    --bucket amzn-s3-demo-bucket-example \
    --key doc3.rtf \
    --tagging '{"TagSet": [{ "Key": "designation", "Value": "confidential" },
 { "Key": "department", "Value": "finance" }, { "Key": "team", "Value":
 "payroll" } ]}'
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta PutObjectTagging AWS CLICommand Reference.

`put-object`

Il seguente esempio di codice mostra come utilizzare`put-object`.

AWS CLI

Esempio 1: carica un oggetto in Amazon S3

Il comando `put-object` seguente carica un oggetto in Amazon S3.

```
aws s3api put-object \
    --bucket amzn-s3-demo-bucket \
    --key my-dir/MySampleImage.png \
    --body MySampleImage.png
```

Per ulteriori informazioni sul caricamento di oggetti, vedete Uploading Objects < http://docs.aws.amazon. com/AmazonS3/latest/dev/UploadingObjects.html> nella Amazon S3 Developer Guide.

Esempio 2: carica un file video in Amazon S3

Il comando `put-object` seguente carica un file video.

```
aws s3api put-object \
    --bucket amzn-s3-demo-bucket \
    --key my-dir/big-video-file.mp4 \
    --body /media/videos/f-sharp-3-data-services.mp4
```

Per ulteriori informazioni sul caricamento di oggetti, consulta Uploading Objects < http://docs.aws.amazon. com/AmazonS3/latest/dev/UploadingObjects.html> nella Amazon S3 Developer Guide.

- Per i dettagli sull'API, consulta PutObjectCommand Reference.AWS CLI

`put-public-access-block`

Il seguente esempio di codice mostra come utilizzare`put-public-access-block`.

AWS CLI

Come impostare la configurazione di blocco dell'accesso pubblico per un bucket

L'esempio `put-public-access-block` seguente imposta una configurazione di blocco dell'accesso pubblico restrittiva per il bucket specificato.

```
aws s3api put-public-access-block \
    --bucket amzn-s3-demo-bucket \
    --public-access-block-
configuration "BlockPublicAcls=true,IgnorePublicAcls=true,BlockPublicPolicy=true,RestrictPub
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta PutPublicAccessBlock AWS CLICommand Reference.

`rb`

Il seguente esempio di codice mostra come utilizzare`rb`.

AWS CLI

Esempio 1: elimina un bucket

Il comando `rb` seguente rimuove un bucket. In questo esempio, il bucket dell'utente è `amzn-s3-demo-bucket`. Tieni presente che il bucket deve essere vuoto per essere rimosso:

```
aws s3 rb s3://amzn-s3-demo-bucket
```

Output:

```
remove_bucket: amzn-s3-demo-bucket
```

Esempio 2: forza l'eliminazione di un bucket

Il comando `rb` seguente utilizza il parametro `--force` per rimuovere prima tutti gli oggetti nel bucket, poi il bucket stesso. In questo esempio, il bucket dell'utente è `amzn-s3-demo-bucket` e gli oggetti in `amzn-s3-demo-bucket` sono `test1.txt` e `test2.txt`:

```
aws s3 rb s3://amzn-s3-demo-bucket \
    --force
```

Output:

```
delete: s3://amzn-s3-demo-bucket/test1.txt
delete: s3://amzn-s3-demo-bucket/test2.txt
remove_bucket: amzn-s3-demo-bucket
```

- Per informazioni dettagliate sull'API, consulta Rb in AWS CLI Command Reference.

`restore-object`

Il seguente esempio di codice mostra come utilizzare`restore-object`.

AWS CLI

Come creare una richiesta di ripristino per un oggetto

L'esempio `restore-object` seguente ripristina l'oggetto Amazon S3 Glacier specificato per il bucket `my-glacier-bucket` per 10 giorni.

```
aws s3api restore-object \
    --bucket my-glacier-bucket \
    --key doc1.rtf \
```

```
    --restore-request Days=10
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta RestoreObject AWS CLICommand Reference.

rm

Il seguente esempio di codice mostra come utilizzare`rm`.

AWS CLI

Esempio 1: elimina un oggetto S3

Il comando `rm` seguente elimina un singolo oggetto s3.

```
aws s3 rm s3://amzn-s3-demo-bucket/test2.txt
```

Output:

```
delete: s3://amzn-s3-demo-bucket/test2.txt
```

Esempio 2: elimina tutti i contenuti in un bucket

Il comando `rm` seguente elimina in modo ricorsivo tutti gli oggetti contenuti in un bucket e in un prefisso specificati se passato con il parametro `--recursive`. In questo esempio, il bucket `amzn-s3-demo-bucket` contiene gli oggetti `test1.txt` e `test2.txt`:

```
aws s3 rm s3://amzn-s3-demo-bucket \
    --recursive
```

Output:

```
delete: s3://amzn-s3-demo-bucket/test1.txt
delete: s3://amzn-s3-demo-bucket/test2.txt
```

Esempio 3: elimina tutti i contenuti in un bucket, ad eccezione dei file ``.jpg``

Il comando `rm` seguente elimina in modo ricorsivo tutti gli oggetti contenuti in un bucket e un prefisso specificati se passato con il parametro `--recursive`, eccetto alcuni oggetti esclusi

utilizzando un parametro `--exclude`. In questo esempio, il bucket `amzn-s3-demo-bucket` contiene gli oggetti `test1.txt` e `test2.jpg`:

```
aws s3 rm s3://amzn-s3-demo-bucket/ \
    --recursive \
    --exclude "*.jpg"
```

Output:

```
delete: s3://amzn-s3-demo-bucket/test1.txt
```

Esempio 4: elimina tutti i contenuti in un bucket, ad eccezione degli oggetti con il prefisso specificato

Il comando `rm` seguente elimina in modo ricorsivo tutti gli oggetti contenuti in un bucket e un prefisso specificati se passato con il parametro `--recursive`, eccetto alcuni oggetti con un prefisso particolare esclusi utilizzando un parametro `--exclude`. In questo esempio, il bucket `amzn-s3-demo-bucket` contiene gli oggetti `test1.txt` e `another/test.txt`:

```
aws s3 rm s3://amzn-s3-demo-bucket/ \
    --recursive \
    --exclude "another/*"
```

Output:

```
delete: s3://amzn-s3-demo-bucket/test1.txt
```

Esempio 5: elimina un oggetto da un punto di accesso S3

Il comando `rm` seguente elimina un singolo oggetto (`mykey`) dal punto di accesso (`myaccesspoint`). Il comando `rm` seguente elimina un singolo oggetto (`mykey`) dal punto di accesso (`myaccesspoint`).

```
aws s3 rm s3://arn:aws:s3:us-west-2:123456789012:accesspoint/myaccesspoint/mykey
```

Output:

```
delete: s3://arn:aws:s3:us-west-2:123456789012:accesspoint/myaccesspoint/mykey
```

- Per informazioni dettagliate sull'API, consulta Rm in AWS CLI Command Reference.

`select-object-content`

Il seguente esempio di codice mostra come utilizzare`select-object-content`.

AWS CLI

Come filtrare il contenuto di un oggetto Amazon S3 in base a un'istruzione SQL

L'esempio `select-object-content` seguente filtra l'oggetto `my-data-file.csv` con l'istruzione SQL specificata e invia l'output a un file.

```
aws s3api select-object-content \
    --bucket amzn-s3-demo-bucket \
    --key my-data-file.csv \
    --expression "select * from s3object limit 100" \
    --expression-type 'SQL' \
    --input-serialization '{"CSV": {}, "CompressionType": "NONE"}' \
    --output-serialization '{"CSV": {}}' "output.csv"
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta SelectObjectContent AWS CLICommand Reference.

`sync`

Il seguente esempio di codice mostra come utilizzare`sync`.

AWS CLI

Esempio 1: sincronizza tutti gli oggetti locali con il bucket specificato

Il comando `sync` seguente sincronizza gli oggetti da una directory locale al prefisso e al bucket specificati caricando i file locali su S3. Un file locale dovrà essere caricato se la sua dimensione è diversa da quella dell'oggetto S3, l'ora dell'ultima modifica del file locale è successiva all'ora dell'ultima modifica dell'oggetto S3 o il file locale non esiste nel bucket e nel prefisso specificati. In questo esempio, l'utente sincronizza il bucket `amzn-s3-demo-bucket` con la directory corrente locale. La directory corrente locale contiene i file `test.txt` e `test2.txt`. Il bucket `amzn-s3-demo-bucket` non contiene alcun oggetto.

```
aws s3 sync . s3://amzn-s3-demo-bucket
```

Output:

```
upload: test.txt to s3://amzn-s3-demo-bucket/test.txt
upload: test2.txt to s3://amzn-s3-demo-bucket/test2.txt
```

Esempio 2: sincronizza tutti gli oggetti S3 dal bucket S3 specificato con un altro bucket

Il comando `sync` seguente sincronizza gli oggetti con un prefisso e un bucket specificati con gli oggetti con un altro prefisso e un altro bucket specificati copiando gli oggetti S3. Un oggetto S3 richiederà la copia se le dimensioni dei due oggetti S3 sono diverse, l'ora dell'ultima modifica dell'origine è successiva all'ora dell'ultima modifica della destinazione o l'oggetto S3 non esiste nella destinazione del bucket e del prefisso specificati.

In questo esempio, l'utente sincronizza il bucket `amzn-s3-demo-bucket` con il bucket `amzn-s3-demo-bucket2`. Il bucket `amzn-s3-demo-bucket` contiene gli oggetti `test.txt` e `test2.txt`. Il bucket `amzn-s3-demo-bucket2` non contiene alcun oggetto:

```
aws s3 sync s3://amzn-s3-demo-bucket s3://amzn-s3-demo-bucket2
```

Output:

```
copy: s3://amzn-s3-demo-bucket/test.txt to s3://amzn-s3-demo-bucket2/test.txt
copy: s3://amzn-s3-demo-bucket/test2.txt to s3://amzn-s3-demo-bucket2/test2.txt
```

Esempio 3: sincronizza tutti gli oggetti S3 dal bucket S3 specificato alla directory locale

Il comando `sync` seguente sincronizza i file dal bucket S3 specificato alla directory locale scaricando oggetti S3. Un oggetto S3 dovrà essere scaricato se la sua dimensione è diversa da quella del file locale, l'ora dell'ultima modifica dell'oggetto S3 è più recente dell'ora dell'ultima modifica del file locale o l'oggetto S3 non esiste nella directory locale. Tieni presente che quando gli oggetti vengono scaricati da S3, l'ora dell'ultima modifica del file locale viene sostituita dall'ora dell'ultima modifica dell'oggetto S3. In questo esempio, l'utente sincronizza il bucket `amzn-s3-demo-bucket` con la directory locale corrente. Il bucket `amzn-s3-demo-bucket` contiene gli oggetti `test.txt` e `test2.txt`. La directory locale corrente non contiene file:

```
aws s3 sync s3://amzn-s3-demo-bucket .
```

Output:

```
download: s3://amzn-s3-demo-bucket/test.txt to test.txt
download: s3://amzn-s3-demo-bucket/test2.txt to test2.txt
```

Esempio 4: sincronizza tutti gli oggetti locali nel bucket specificato ed elimina tutti i file che non corrispondono

Il comando `sync` seguente sincronizza gli oggetti con un prefisso e un bucket specificati con i file in una directory locale caricando i file locali su S3. A causa del parametro `--delete`, tutti i file esistenti con il prefisso e il bucket specificati, ma non esistenti nella directory locale, verranno eliminati. In questo esempio, l'utente sincronizza il bucket `amzn-s3-demo-bucket` con la directory corrente locale. La directory corrente locale contiene i file `test.txt` e `test2.txt`. Il bucket `amzn-s3-demo-bucket` contiene l'oggetto `test3.txt`:

```
aws s3 sync . s3://amzn-s3-demo-bucket \
    --delete
```

Output:

```
upload: test.txt to s3://amzn-s3-demo-bucket/test.txt
upload: test2.txt to s3://amzn-s3-demo-bucket/test2.txt
delete: s3://amzn-s3-demo-bucket/test3.txt
```

Esempio 5: sincronizza tutti gli oggetti locali con il bucket specificato tranne i file ".jpg"

Il comando `sync` seguente sincronizza gli oggetti con un prefisso e un bucket specificati con i file in una directory locale caricando i file locali su S3. A causa del parametro `--exclude`, tutti i file che corrispondono al modello esistente, sia in S3 che localmente, verranno esclusi dalla sincronizzazione. In questo esempio, l'utente sincronizza il bucket `amzn-s3-demo-bucket` con la directory corrente locale. La directory corrente locale contiene i file `test.jpg` e `test2.txt`. Il bucket `amzn-s3-demo-bucket` contiene l'oggetto `test.jpg` di dimensioni diverse da quelle dell'oggetto `test.jpg` locale:

```
aws s3 sync . s3://amzn-s3-demo-bucket \
    --exclude "*.jpg"
```

Output:

```
upload: test2.txt to s3://amzn-s3-demo-bucket/test2.txt
```

Esempio 6: sincronizza tutti gli oggetti locali con il bucket specificato ad eccezione dei file di directory specificati

Il comando `sync` seguente sincronizza i file in una directory locale con gli oggetti con un prefisso e un bucket specificati scaricando oggetti S3. Questo esempio utilizza il flag del parametro `--exclude` per escludere una directory e un prefisso S3 specificati dal comando `sync`. In questo esempio, l'utente sincronizza la directory corrente locale con il bucket `amzn-s3-demo-bucket`. La directory corrente locale contiene i file `test.txt` e `another/test2.txt`. Il bucket `amzn-s3-demo-bucket` contiene gli oggetti `another/test5.txt` e `test1.txt`:

```
aws s3 sync s3://amzn-s3-demo-bucket/ . \
    --exclude "*another/*"
```

Output:

```
download: s3://amzn-s3-demo-bucket/test1.txt to test1.txt
```

Esempio 7: sincronizza tutti gli oggetti tra i bucket in diverse Regioni

Il comando `sync` seguente sincronizza i file tra due bucket in Regioni diverse:

```
aws s3 sync s3://my-us-west-2-bucket s3://my-us-east-1-bucket \
    --source-region us-west-2 \
    --region us-east-1
```

Output:

```
download: s3://my-us-west-2-bucket/test1.txt to s3://my-us-east-1-bucket/test1.txt
```

Esempio 8: sincronizza con un punto di accesso S3

Il comando `sync` seguente sincronizza la directory corrente con il punto di accesso (`myaccesspoint`):

```
aws s3 sync . s3://arn:aws:s3:us-west-2:123456789012:accesspoint/myaccesspoint/
```

Output:

```
upload: test.txt to s3://arn:aws:s3:us-west-2:123456789012:accesspoint/
myaccesspoint/test.txt
upload: test2.txt to s3://arn:aws:s3:us-west-2:123456789012:accesspoint/
myaccesspoint/test2.txt
```

- Per informazioni dettagliate sull'API, consulta Sync in AWS CLI Command Reference.

`upload-part-copy`

Il seguente esempio di codice mostra come utilizzare`upload-part-copy`.

AWS CLI

Come caricare parte di un oggetto copiando i dati da un oggetto esistente come origine dati

L'esempio `upload-part-copy` seguente carica una parte copiando i dati da un oggetto esistente come origine dati.

```
aws s3api upload-part-copy \
    --bucket amzn-s3-demo-bucket \
    --key "Map_Data_June.mp4" \
    --copy-source "amzn-s3-demo-bucket/copy_of_Map_Data_June.mp4" \
    --part-number 1 \
    --upload-
id "bq0tdE1CDpWQYRPLHuNG50xAT6pA5D.m_RiBy0ggOH6b13pVRY7QjvLlf75iFdJqp_2wztk5hvpUM2SesXgrzbeh
```

Output:

```
{
    "CopyPartResult": {
        "LastModified": "2019-12-13T23:16:03.000Z",
        "ETag": "\"711470fc377698c393d94aed6305e245\""
    }
}
```

- Per i dettagli sull'API, consulta UploadPartCopy AWS CLICommand Reference.

`upload-part`

Il seguente esempio di codice mostra come utilizzare`upload-part`.

AWS CLI

Il comando seguente carica la prima parte di un caricamento in più parti avviato con il comando `create-multipart-upload`.

```
aws s3api upload-part --bucket amzn-s3-demo-bucket --key
 'multipart/01' --part-number 1 --body part01 --upload-id
  "dfRtDYU0WWCCcH43C3WFbkRONycyCpTJJvxu2i5GYkZljF.Yxwh6XG7WfS2vC4to6HiV6Yjlx.cph0gtNBtJ8P3UR
```

L'opzione body richiede il nome o il percorso di un file locale per il caricamento (non utilizzare il prefisso file://). La dimensione minima della parte è 5 MB. L'ID di caricamento viene restituito da `create-multipart-upload` e può essere recuperato anche con `list-multipart-uploads`. Il bucket e la chiave vengono specificati quando si crea il caricamento in più parti.

Output:

```
{
    "ETag": "\"e868e0f4719e394144ef36531ee6824c\""
}
```

Salva il ETag valore di ogni parte per utilizzarlo in un secondo momento. Tale valore sarà necessario per completare un caricamento in più parti.

- Per i dettagli sull'API, consulta UploadPart AWS CLICommand Reference.

`website`

Il seguente esempio di codice mostra come utilizzare`website`.

AWS CLI

Configura un bucket S3 come un sito web statico

Il comando seguente configura un bucket denominato `amzn-s3-demo-bucket` come sito web statico. L'opzione del documento di indice specifica il file nel bucket `amzn-s3-demo-bucket` a cui verranno indirizzati i visitatori quando accedono all'URL del sito web. In questo caso, il bucket si trova nella Regione us-west-2, quindi il sito verrà visualizzato all'indirizzo `http://amzn-s3-demo-bucket.s3-website-us-west-2.amazonaws.com`.

Tutti i file nel bucket che appaiono nel sito statico devono essere configurati per consentire ai visitatori di aprirli. Le autorizzazioni relative ai file sono configurate separatamente dalla configurazione del sito web del bucket.

```
aws s3 website s3://amzn-s3-demo-bucket/ \
```

```
    --index-document index.html \
    --error-document error.html
```

Per informazioni su come ospitare un sito web statico su Amazon S3, consulta Hosting di un sito web statico nella Guida per gli sviluppatori di Amazon Simple Storage Service.

- Per informazioni dettagliate sull'API, consulta Website in AWS CLI Command Reference.

Esempi di Amazon S3 Control con AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando il controllo AWS Command Line Interface con Amazon S3.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`create-access-point`

Il seguente esempio di codice mostra come usare`create-access-point`.

AWS CLI

Come creare un punto di accesso

L'esempio `create-access-point` seguente crea un punto di accesso denominato `finance-ap` per il bucket `business-records` nell'account 123456789012. Prima di eseguire questo esempio, sostituisci il nome del punto di accesso, il nome del bucket e il numero di account con valori appropriati per il tuo caso d'uso.

```
aws s3control create-access-point \
```

```
    --account-id 123456789012 \
    --bucket business-records \
    --name finance-ap
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Creazione di punti di accesso nella Guida per gli sviluppatori di Amazon Simple Storage Service.

- Per i dettagli sull'API, consulta CreateAccessPoint AWS CLICommand Reference.

`create-job`

Il seguente esempio di codice mostra come utilizzare`create-job`.

AWS CLI

Come creare un processo Operazioni in batch Amazon S3

L'esempio `create-job` seguente crea un processo di operazioni in batch Amazon S3 con cui taggare agli oggetti come `confidential` in the bucket ``employee-records`.

```
aws s3control create-job \
    --account-id 123456789012 \
    --operation '{"S3PutObjectTagging": { "TagSet": [{"Key":"confidential",
 "Value":"true"}] }}' \
    --report '{"Bucket":"arn:aws:s3:::employee-records-logs","Prefix":"batch-op-
create-job",
 "Format":"Report_CSV_20180820","Enabled":true,"ReportScope":"AllTasks"}' \
    --manifest '{"Spec":{"Format":"S3BatchOperations_CSV_20180820","Fields":
["Bucket","Key"]},"Location":{"ObjectArn":"arn:aws:s3:::employee-records-logs/inv-
report/7a6a9be4-072c-407e-85a2-
ec3e982f773e.csv","ETag":"69f52a4e9f797e987155d9c8f5880897"}}' \
    --priority 42 \
    --role-arn arn:aws:iam::123456789012:role/S3BatchJobRole
```

Output:

```
{
    "JobId": "93735294-df46-44d5-8638-6356f335324e"
}
```

- Per i dettagli sull'API, consulta CreateJob AWS CLICommand Reference.

`delete-access-point-policy`

Il seguente esempio di codice mostra come utilizzare`delete-access-point-policy`.

AWS CLI

Come eliminare una policy del punto di accesso

L'esempio `delete-access-point-policy` seguente elimina la policy del punto di accesso dal punto di accesso indicato `finance-ap` nell'account 123456789012. Prima di eseguire questo esempio, sostituisci il nome del punto di accesso e il numero di account con valori appropriati per il tuo caso d'uso.

```
aws s3control delete-access-point-policy \
    --account-id 123456789012 \
    --name finance-ap
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Gestione dell'accesso ai dati con punti di accesso Amazon S3 nella Guida per gli sviluppatori di Amazon Simple Storage Service.

- Per i dettagli sull'API, consulta DeleteAccessPointPolicy AWS CLICommand Reference.

`delete-access-point`

Il seguente esempio di codice mostra come utilizzare`delete-access-point`.

AWS CLI

Come eliminare un punto di accesso

L'esempio `delete-access-point` seguente elimina un punto di accesso denominato `finance-ap` nell'account 123456789012. Prima di eseguire questo esempio, sostituisci il nome del punto di accesso e il numero di account con valori appropriati per il tuo caso d'uso.

```
aws s3control delete-access-point \
    --account-id 123456789012 \
```

```
    --name finance-ap
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Gestione dell'accesso ai dati con punti di accesso Amazon S3 nella Guida per gli sviluppatori di Amazon Simple Storage Service.

- Per i dettagli sull'API, consulta DeleteAccessPoint AWS CLICommand Reference.

`delete-public-access-block`

Il seguente esempio di codice mostra come utilizzare`delete-public-access-block`.

AWS CLI

Come eliminare le impostazioni di blocco dell'accesso pubblico dell'account

L'esempio `delete-public-access-block` seguente elimina le impostazioni di blocco dell'accesso pubblico per l'account specificato.

```
aws s3control delete-public-access-block \
    --account-id 123456789012
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta DeletePublicAccessBlock AWS CLICommand Reference.

`describe-job`

Il seguente esempio di codice mostra come utilizzare`describe-job`.

AWS CLI

Come descrivere un processo Operazioni in batch Amazon S3

L'esempio `describe-job` seguente fornisce i parametri di configurazione e lo stato per il processo di operazioni batch specificato.

```
aws s3control describe-job \
    --account-id 123456789012 \
    --job-id 93735294-df46-44d5-8638-6356f335324e
```

Output:

```
{
    "Job": {
        "TerminationDate": "2019-10-03T21:49:53.944Z",
        "JobId": "93735294-df46-44d5-8638-6356f335324e",
        "FailureReasons": [],
        "Manifest": {
            "Spec": {
                "Fields": [
                    "Bucket",
                    "Key"
                ],
                "Format": "S3BatchOperations_CSV_20180820"
            },
            "Location": {
                "ETag": "69f52a4e9f797e987155d9c8f5880897",
                "ObjectArn": "arn:aws:s3:::employee-records-logs/inv-
report/7a6a9be4-072c-407e-85a2-ec3e982f773e.csv"
            }
        },
        "Operation": {
            "S3PutObjectTagging": {
                "TagSet": [
                    {
                        "Value": "true",
                        "Key": "confidential"
                    }
                ]
            }
        },
        "RoleArn": "arn:aws:iam::123456789012:role/S3BatchJobRole",
        "ProgressSummary": {
            "TotalNumberOfTasks": 8,
            "NumberOfTasksFailed": 0,
            "NumberOfTasksSucceeded": 8
        },
        "Priority": 42,
        "Report": {
            "ReportScope": "AllTasks",
            "Format": "Report_CSV_20180820",
            "Enabled": true,
            "Prefix": "batch-op-create-job",
            "Bucket": "arn:aws:s3:::employee-records-logs"
```

```
        },
        "JobArn": "arn:aws:s3:us-west-2:123456789012:job/93735294-
df46-44d5-8638-6356f335324e",
        "CreationTime": "2019-10-03T21:48:48.048Z",
        "Status": "Complete"
    }
}
```

- Per i dettagli sull'API, consulta DescribeJob AWS CLICommand Reference.

`get-access-point-policy-status`

Il seguente esempio di codice mostra come utilizzare`get-access-point-policy-status`.

AWS CLI

Come recuperare lo stato della policy del punto di accesso

L'esempio `get-access-point-policy-status` seguente recupera lo stato della policy del punto di accesso per il punto di accesso denominato `finance-ap` nell'account 123456789012. Lo stato della policy del punto di accesso indica se la policy del punto di accesso consente l'accesso pubblico. Prima di eseguire questo esempio, sostituisci il nome del punto di accesso e il numero di account con valori appropriati per il tuo caso d'uso.

```
aws s3control get-access-point-policy-status \
    --account-id 123456789012 \
    --name finance-ap
```

Output:

```
{
    "PolicyStatus": {
        "IsPublic": false
    }
}
```

Per ulteriori informazioni su quando una policy del punto di accesso è considerata pubblica, consulta Significato di "pubblico" nella Guida per gli sviluppatori di Amazon Simple Storage Service.

- Per i dettagli sull'API, consulta GetAccessPointPolicyStatus AWS CLICommand Reference.

`get-access-point-policy`

Il seguente esempio di codice mostra come utilizzare`get-access-point-policy`.

AWS CLI

Come recuperare una policy del punto di accesso

L'esempio `get-access-point-policy` seguente recupera la policy del punto di accesso per il punto di accesso denominato `finance-ap` nell'account 123456789012. Prima di eseguire questo esempio, sostituisci il nome del punto di accesso e il numero di account con valori appropriati per il tuo caso d'uso.

```
aws s3control get-access-point-policy \
    --account-id 123456789012 \
    --name finance-ap
```

Output:

```
{
    "Policy": "{\"Version\":\"2012-10-17\",\"Statement\":[{\"Effect\":\"Allow
\",\"Principal\":{\"AWS\":\"arn:aws:iam::123456789012:role/Admin\"},\"Action\":
\"s3:GetObject\",\"Resource\":\"arn:aws:s3:us-west-2:123456789012:accesspoint/
finance-ap/object/records/*\"}]}"
}
```

Per ulteriori informazioni, consulta Gestione dell'accesso ai dati con punti di accesso Amazon S3 nella Guida per gli sviluppatori di Amazon Simple Storage Service.

- Per i dettagli sull'API, consulta GetAccessPointPolicy AWS CLICommand Reference.

`get-access-point`

Il seguente esempio di codice mostra come utilizzare`get-access-point`.

AWS CLI

Come recuperare i dettagli di configurazione del punto di accesso

L'esempio `get-access-point` seguente restituisce i dettagli di configurazione per il punto di accesso denominato `finance-ap` nell'account 123456789012. Prima di eseguire questo

esempio, sostituisci il nome del punto di accesso e il numero di account con valori appropriati per il tuo caso d'uso.

```
aws s3control get-access-point \
    --account-id 123456789012 \
    --name finance-ap
```

Output:

```
{
    "Name": "finance-ap",
    "Bucket": "business-records",
    "NetworkOrigin": "Internet",
    "PublicAccessBlockConfiguration": {
        "BlockPublicAcls": false,
        "IgnorePublicAcls": false,
        "BlockPublicPolicy": false,
        "RestrictPublicBuckets": false
    },
    "CreationDate": "2020-01-01T00:00:00Z"
}
```

Per ulteriori informazioni, consulta Gestione dell'accesso ai dati con punti di accesso Amazon S3 nella Guida per gli sviluppatori di Amazon Simple Storage Service.

- Per i dettagli sull'API, consulta GetAccessPoint AWS CLICommand Reference.

`get-multi-region-access-point-routes`

Il seguente esempio di codice mostra come utilizzare`get-multi-region-access-point-routes`.

AWS CLI

Come eseguire query sulla configurazione delle route di un punto di accesso multi-Regione

L'esempio `get-multi-region-access-point-routes` seguente recupera la configurazione dell'instradamento corrente del punto di accesso multi-Regione specificato.

```
aws s3control get-multi-region-access-point-routes \
    --region Region \
```

```
    --account-id 111122223333 \
    --mrap MultiRegionAccessPoint_ARN
```

Output:

```
{
    "Mrap": "arn:aws:s3::111122223333:accesspoint/0000000000000.mrap",
    "Routes": [
        {
            "Bucket": "amzn-s3-demo-bucket1",
            "Region": "ap-southeast-2",
            "TrafficDialPercentage": 100
        },
        {
            "Bucket": "amzn-s3-demo-bucket2",
            "Region": "us-west-1",
            "TrafficDialPercentage": 0
        }
    ]
}
```

- Per i dettagli sull'API, consulta GetMultiRegionAccessPointRoutes AWS CLICommand Reference.

`get-public-access-block`

Il seguente esempio di codice mostra come utilizzare`get-public-access-block`.

AWS CLI

Come elencare le impostazioni di accesso al blocco pubblico per un account

L'esempio `get-public-access-block` seguente visualizza le impostazioni di blocco dell'accesso pubblico per l'account specificato.

```
aws s3control get-public-access-block \
    --account-id 123456789012
```

Output:

```
{
```

```
   "PublicAccessBlockConfiguration": {
      "BlockPublicPolicy": true,
      "RestrictPublicBuckets": true,
      "IgnorePublicAcls": true,
      "BlockPublicAcls": true
   }
}
```

- Per i dettagli sull'API, consulta GetPublicAccessBlock AWS CLICommand Reference.

`list-access-points`

Il seguente esempio di codice mostra come utilizzare`list-access-points`.

AWS CLI

Esempio 1: come recuperare un elenco di tutti i punti di accesso per un account

L'esempio `list-access-points` seguente visualizza un elenco di tutti i punti di accesso collegati ai bucket di proprietà dell'account 123456789012.

```
aws s3control list-access-points \
    --account-id 123456789012
```

Output:

```
{
    "AccessPointList": [
        {
            "Name": "finance-ap",
            "NetworkOrigin": "Internet",
            "Bucket": "business-records"
        },
        {
            "Name": "managers-ap",
            "NetworkOrigin": "Internet",
            "Bucket": "business-records"
        },
        {
            "Name": "private-network-ap",
            "NetworkOrigin": "VPC",
            "VpcConfiguration": {
```

```
                "VpcId": "1a2b3c"
            },
            "Bucket": "business-records"
        },
        {
            "Name": "customer-ap",
            "NetworkOrigin": "Internet",
            "Bucket": "external-docs"
        },
        {
            "Name": "public-ap",
            "NetworkOrigin": "Internet",
            "Bucket": "external-docs"
        }
    ]
}
```

Esempio 2: come recuperare un elenco di tutti i punti di accesso per un bucket

L'esempio `list-access-points` seguente recupera un elenco di tutti i punti di accesso collegati ai bucket `external-docs` di proprietà dell'account 123456789012.

```
aws s3control list-access-points \
    --account-id 123456789012 \
    --bucket external-docs
```

Output:

```
{
    "AccessPointList": [
        {
            "Name": "customer-ap",
            "NetworkOrigin": "Internet",
            "Bucket": "external-docs"
        },
        {
            "Name": "public-ap",
            "NetworkOrigin": "Internet",
            "Bucket": "external-docs"
        }
    ]
}
```

Per ulteriori informazioni, consulta Gestione dell'accesso ai dati con punti di accesso Amazon S3 nella Guida per gli sviluppatori di Amazon Simple Storage Service.

- Per i dettagli sull'API, consulta ListAccessPoints AWS CLICommand Reference.

`list-jobs`

Il seguente esempio di codice mostra come utilizzare`list-jobs`.

AWS CLI

Come elencare i processi delle operazioni in batch Amazon S3 per un account

L'esempio `list-jobs` seguente elenca tutti i processi di operazioni in batch recenti per l'account specificato.

```
aws s3control list-jobs \
    --account-id 123456789012
```

Output:

```
{
    "Jobs": [
        {
            "Operation": "S3PutObjectTagging",
            "ProgressSummary": {
                "NumberOfTasksFailed": 0,
                "NumberOfTasksSucceeded": 8,
                "TotalNumberOfTasks": 8
            },
            "CreationTime": "2019-10-03T21:48:48.048Z",
            "Status": "Complete",
            "JobId": "93735294-df46-44d5-8638-6356f335324e",
            "Priority": 42
        },
        {
            "Operation": "S3PutObjectTagging",
            "ProgressSummary": {
                "NumberOfTasksFailed": 0,
                "NumberOfTasksSucceeded": 0,
                "TotalNumberOfTasks": 0
            },
```

```
            "CreationTime": "2019-10-03T21:46:07.084Z",
            "Status": "Failed",
            "JobId": "3f3c7619-02d3-4779-97f6-1d98dd313108",
            "Priority": 42
        },
    ]
}
```

- Per i dettagli sull'API, consulta ListJobs AWS CLICommand Reference.

`put-access-point-policy`

Il seguente esempio di codice mostra come utilizzare`put-access-point-policy`.

AWS CLI

Come modificare la policy di un punto di accesso

L'esempio `put-access-point-policy` seguente inserisce la policy del punto di accesso specificata per il punto di accesso `finance-ap` nell'account 123456789012. Se il punto di accesso `finance-ap` dispone già di una policy, questo comando sostituisce la policy esistente con quella specificata in questo comando. Prima di eseguire questo esempio, sostituisci il numero di account, il nome del punto di accesso e le istruzioni dei criteri con valori appropriati per il tuo caso d'uso.

```
aws s3control put-access-point-policy \
    --account-id 123456789012 \
    --name finance-ap \
    --policy file://ap-policy.json
```

Contenuto di `ap-policy.json`:

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::123456789012:user/Alice"
            },
            "Action": "s3:GetObject",
```

```
            "Resource": "arn:aws:s3:us-west-2:123456789012:accesspoint/finance-ap/
object/Alice/*"
        }
    ]
}
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Gestione dell'accesso ai dati con punti di accesso Amazon S3 nella Guida per gli sviluppatori di Amazon Simple Storage Service.

- Per i dettagli sull'API, consulta PutAccessPointPolicy AWS CLICommand Reference.

`put-public-access-block`

Il seguente esempio di codice mostra come utilizzare`put-public-access-block`.

AWS CLI

Come modificare le impostazioni del blocco dell'accesso pubblico per un account

L'esempio `put-public-access-block` seguente attiva tutte le impostazioni di blocco dell'accesso pubblico per l'account `true` specificato.

```
aws s3control put-public-access-block \
    --account-id 123456789012 \
    --public-access-block-configuration '{"BlockPublicAcls": true,
 "IgnorePublicAcls": true, "BlockPublicPolicy": true, "RestrictPublicBuckets":
 true}'
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta PutPublicAccessBlock AWS CLICommand Reference.

`submit-multi-region-access-point-routes`

Il seguente esempio di codice mostra come utilizzare`submit-multi-region-access-point-routes`.

AWS CLI

Come aggiornare la configurazione di instradamento del punto di accesso multi-Regione

L'esempio `submit-multi-region-access-point-routes` seguente aggiorna gli stati di instradamento di `amzn-s3-demo-bucket1` e `amzn-s3-demo-bucket2` nella `ap-southeast-2` Regione per il punto di accesso multi-Regione.

```
aws s3control submit-multi-region-access-point-routes \
    --region ap-southeast-2 \
    --account-id 111122223333 \
    --mrap MultiRegionAccessPoint_ARN \
    --route-updates Bucket=amzn-s3-demo-
bucket1,TrafficDialPercentage=100 Bucket=amzn-s3-demo-
bucket2,TrafficDialPercentage=0
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta SubmitMultiRegionAccessPointRoutes AWS CLICommand Reference.

`update-job-priority`

Il seguente esempio di codice mostra come utilizzare`update-job-priority`.

AWS CLI

Come aggiornare la priorità di un processo Operazioni in batch Amazon S3

L'esempio `update-job-priority` seguente aggiorna il processo specificato in base a una nuova priorità.

```
aws s3control update-job-priority \
    --account-id 123456789012 \
    --job-id 8d9a18fe-c303-4d39-8ccc-860d372da386 \
    --priority 52
```

Output:

```
{
    "JobId": "8d9a18fe-c303-4d39-8ccc-860d372da386",
    "Priority": 52
}
```

- Per i dettagli sull'API, consulta UpdateJobPriority AWS CLICommand Reference.

update-job-status

Il seguente esempio di codice mostra come utilizzareupdate-job-status.

AWS CLI

Come aggiornare lo stato di un processo Operazioni in batch Amazon S3

L'esempio update-job-status seguente annulla il processo specificato che è in attesa di approvazione.

```
aws s3control update-job-status \
    --account-id 123456789012 \
    --job-id 8d9a18fe-c303-4d39-8ccc-860d372da386 \
    --requested-job-status Cancelled
```

Output:

```
{
    "Status": "Cancelled",
    "JobId": "8d9a18fe-c303-4d39-8ccc-860d372da386"
}
```

L'esempio update-job-status seguente conferma ed esegue il processo specificato che è in attesa di approvazione.

```
aws s3control update-job-status \
    --account-id 123456789012 \
    --job-id 5782949f-3301-4fb3-be34-8d5bab54dbca \
    --requested-job-status Ready

Output::

{
    "Status": "Ready",
    "JobId": "5782949f-3301-4fb3-
be34-8d5bab54dbca"
}
```

L'esempio update-job-status seguente annulla il processo in esecuzione specificato.

```
 aws s3control update-job-status \
```

```
    --account-id 123456789012 \
    --job-id 5782949f-3301-4fb3-be34-8d5bab54dbca \
    --requested-job-status Cancelled

Output::
{
         "Status": "Cancelling",
         "JobId": "5782949f-3301-4fb3-be34-8d5bab54dbca"
}
```

- Per i dettagli sull'API, consulta UpdateJobStatus AWS CLICommand Reference.

Esempi di Secrets Manager che utilizzano AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with Secrets Manager.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`batch-get-secret-value`

Il seguente esempio di codice mostra come utilizzare`batch-get-secret-value`.

AWS CLI

Esempio 1: come recuperare il valore del segreto per un gruppo di segreti elencati per nome

L'esempio `batch-get-secret-value` seguente ottiene il valore di tre segreti.

```
aws secretsmanager batch-get-secret-value \
    --secret-id-list MySecret1 MySecret2 MySecret3
```

Output:

```
{
    "SecretValues": [
        {
            "ARN": "arn:aws:secretsmanager:us-west-2:123456789012:secret:MySecret1-
a1b2c3",
            "Name": "MySecret1",
            "VersionId": "a1b2c3d4-5678-90ab-cdef-EXAMPLEaaaaa",
            "SecretString": "{\"username\":\"diego_ramirez\",\"password\":\"EXAMPLE-
PASSWORD\",\"engine\":\"mysql\",\"host\":\"secretsmanagertutorial.cluster.us-
west-2.rds.amazonaws.com\",\"port\":3306,\"dbClusterIdentifier\":
\"secretsmanagertutorial\"}",
            "VersionStages": [
                "AWSCURRENT"
            ],
            "CreatedDate": "1523477145.729"
        },
        {
            "ARN": "arn:aws:secretsmanager:us-west-2:123456789012:secret:MySecret2-
a1b2c3",
            "Name": "MySecret2",
            "VersionId": "a1b2c3d4-5678-90ab-cdef-EXAMPLEbbbbb",
            "SecretString": "{\"username\":\"akua_mansa\",\"password\":\"EXAMPLE-
PASSWORD\"",
            "VersionStages": [
                "AWSCURRENT"
            ],
            "CreatedDate": "1673477781.275"
        },
        {
            "ARN": "arn:aws:secretsmanager:us-west-2:123456789012:secret:MySecret3-
a1b2c3",
            "Name": "MySecret3",
            "VersionId": "a1b2c3d4-5678-90ab-cdef-EXAMPLEccccc",
            "SecretString": "{\"username\":\"jie_liu\",\"password\":\"EXAMPLE-
PASSWORD\"",
            "VersionStages": [
                "AWSCURRENT"
            ],
            "CreatedDate": "1373477721.124"
        }
    ],
    "Errors": []
```

```
}
```

Per ulteriori informazioni, consulta Recuperare un gruppo di segreti in un batch nella Guida per l'utente di AWS Secrets Manager.

Esempio 2: come recuperare il valore del segreto per un gruppo di segreti selezionati in base al filtro

L'esempio `batch-get-secret-value` seguente ottiene i valori dei segreti nell'account in uso il cui nome contiene `MySecret`. Il filtro per nome fa distinzione tra maiuscole e minuscole.

```
aws secretsmanager batch-get-secret-value \
    --filters Key="name",Values="MySecret"
```

Output:

```
{
    "SecretValues": [
        {
            "ARN": "arn:aws:secretsmanager:us-west-2:123456789012:secret:MySecret1-
a1b2c3",
            "Name": "MySecret1",
            "VersionId": "a1b2c3d4-5678-90ab-cdef-EXAMPLEaaaaa",
            "SecretString": "{\"username\":\"diego_ramirez\",\"password\":\"EXAMPLE-
PASSWORD\",\"engine\":\"mysql\",\"host\":\"secretsmanagertutorial.cluster.us-
west-2.rds.amazonaws.com\",\"port\":3306,\"dbClusterIdentifier\":
\"secretsmanagertutorial\"}",
            "VersionStages": [
                "AWSCURRENT"
            ],
            "CreatedDate": "1523477145.729"
        },
        {
            "ARN": "arn:aws:secretsmanager:us-west-2:123456789012:secret:MySecret2-
a1b2c3",
            "Name": "MySecret2",
            "VersionId": "a1b2c3d4-5678-90ab-cdef-EXAMPLEbbbbb",
            "SecretString": "{\"username\":\"akua_mansa\",\"password\":\"EXAMPLE-
PASSWORD\"",
            "VersionStages": [
                "AWSCURRENT"
            ],
```

```
            "CreatedDate": "1673477781.275"
        },
        {
            "ARN": "arn:aws:secretsmanager:us-west-2:123456789012:secret:MySecret3-
a1b2c3",
            "Name": "MySecret3",
            "VersionId": "a1b2c3d4-5678-90ab-cdef-EXAMPLEccccc",
            "SecretString": "{\"username\":\"jie_liu\",\"password\":\"EXAMPLE-
PASSWORD\"",
            "VersionStages": [
                "AWSCURRENT"
            ],
            "CreatedDate": "1373477721.124"
        }
    ],
    "Errors": []
}
```

Per ulteriori informazioni, consulta Recuperare un gruppo di segreti in un batch nella Guida per l'utente di AWS Secrets Manager.

- Per i dettagli sull'API, consulta BatchGetSecretValue AWS CLICommand Reference.

`cancel-rotate-secret`

Il seguente esempio di codice mostra come utilizzare`cancel-rotate-secret`.

AWS CLI

Come disattivare la rotazione automatica per un segreto

L'esempio `cancel-rotate-secret` seguente disattiva la rotazione automatica per un segreto. Per riprendere la rotazione, chiama `rotate-secret`.

```
aws secretsmanager cancel-rotate-secret \
    --secret-id MyTestSecret
```

Output:

```
{
  "ARN": "arn:aws:secretsmanager:us-west-2:123456789012:secret:MyTestSecret-a1b2c3",
```

```
  "Name": "MyTestSecret"
}
```

Per ulteriori informazioni, consulta Rotazione di un segreto nella Guida per l'utente di Secrets Manager.

- Per i dettagli sull'API, consulta CancelRotateSecret AWS CLICommand Reference.

`create-secret`

Il seguente esempio di codice mostra come utilizzare`create-secret`.

AWS CLI

Esempio 1: come creare un segreto dalle credenziali in un file JSON

L'esempio di `create-secret` seguente mostra come creare un segreto partendo dalle credenziali in un file. Per ulteriori informazioni, consulta Caricamento dei parametri AWS CLI da un file nella Guida per l'utente della AWS CLI.

```
aws secretsmanager create-secret \
    --name MyTestSecret \
    --secret-string file://mycreds.json
```

Contenuto di `mycreds.json`:

```
{
  "engine": "mysql",
  "username": "saanvis",
  "password": "EXAMPLE-PASSWORD",
  "host": "my-database-endpoint.us-west-2.rds.amazonaws.com",
  "dbname": "myDatabase",
  "port": "3306"
}
```

Output:

```
{
  "ARN": "arn:aws:secretsmanager:us-west-2:123456789012:secret:MyTestSecret-a1b2c3",
  "Name": "MyTestSecret",
```

```
  "VersionId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
}
```

Per ulteriori informazioni, consulta Creazione di un segreto nella Guida per l'utente di Secrets Manager.

Esempio 2: come creare un segreto

L'esempio di `create-secret` seguente mostra come creare un segreto con due coppie chiave-valore. Quando immetti i comandi in una shell dei comandi, c'è il rischio che la cronologia dei comandi sia accessibile o che le utilità abbiano accesso ai parametri dei comandi. Questo è un problema se il comando include il valore di un segreto. Per ulteriori informazioni, consulta Mitigare i rischi dell'utilizzo degli strumenti della riga di comando per archiviare i segreti nella Guida per l'utente di Secrets Manager.

```
aws secretsmanager create-secret \
    --name MyTestSecret \
    --description "My test secret created with the CLI." \
    --secret-string "{\"user\":\"diegor\",\"password\":\"EXAMPLE-PASSWORD\"}"
```

Output:

```
{
  "ARN": "arn:aws:secretsmanager:us-west-2:123456789012:secret:MyTestSecret-a1b2c3",
  "Name": "MyTestSecret",
  "VersionId": "EXAMPLE1-90ab-cdef-fedc-ba987EXAMPLE"
}
```

Per ulteriori informazioni, consulta Creazione di un segreto nella Guida per l'utente di Secrets Manager.

- Per i dettagli sull'API, consulta AWS CLI Command CreateSecretReference.

`delete-resource-policy`

Il seguente esempio di codice mostra come utilizzare`delete-resource-policy`.

AWS CLI

Come eliminare una policy basata sulle risorse collegata a un segreto

L'esempio di `delete-resource-policy` seguente mostra come eliminare la policy basata su risorse collegata a un segreto.

```
aws secretsmanager delete-resource-policy \
    --secret-id MyTestSecret
```

Output:

```
{
    "ARN": "arn:aws:secretsmanager:us-west-2:123456789012:secret:MyTestSecret-
a1b2c3",
    "Name": "MyTestSecret"
}
```

Per ulteriori informazioni, consulta Autenticazione e controllo degli accessi nella Guida per l'utente di Secrets Manager.

- Per i dettagli sull'API, consulta DeleteResourcePolicy AWS CLICommand Reference.

`delete-secret`

Il seguente esempio di codice mostra come utilizzare`delete-secret`.

AWS CLI

Esempio 1: come eliminare un segreto

L'esempio di `delete-secret` seguente mostra come eliminare un segreto. È possibile recuperare il segreto con `DeletionDate` fino alla data e all'ora presenti nel campo di risposta `restore-secret`. Per eliminare un segreto replicato in altre regioni, è necessario dapprima rimuovere le relative repliche con `remove-regions-from-replication`, quindi chiamare `delete-secret`.

```
aws secretsmanager delete-secret \
    --secret-id MyTestSecret \
    --recovery-window-in-days 7
```

Output:

```
{
```

```
    "ARN": "arn:aws:secretsmanager:us-west-2:123456789012:secret:MyTestSecret-
a1b2c3",
    "Name": "MyTestSecret",
    "DeletionDate": 1524085349.095
}
```

Per ulteriori informazioni, consulta Eliminare un segreto nella Guida per l'utente di Secrets Manager.

Esempio 2: come eliminare immediatamente un segreto

L'esempio di `delete-secret` seguente mostra come eliminare immediatamente il secreto senza un intervallo di recupero. Non è possibile recuperare questo segreto.

```
aws secretsmanager delete-secret \
    --secret-id MyTestSecret \
    --force-delete-without-recovery
```

Output:

```
{
    "ARN": "arn:aws:secretsmanager:us-west-2:123456789012:secret:MyTestSecret-
a1b2c3",
    "Name": "MyTestSecret",
    "DeletionDate": 1508750180.309
}
```

Per ulteriori informazioni, consulta Eliminare un segreto nella Guida per l'utente di Secrets Manager.

- Per i dettagli sull'API, consulta DeleteSecret AWS CLICommand Reference.

`describe-secret`

Il seguente esempio di codice mostra come utilizzare`describe-secret`.

AWS CLI

Come recuperare i dettagli di un segreto

L'esempio `describe-secret` seguente mostra i dettagli di un segreto.

```
aws secretsmanager describe-secret \
    --secret-id MyTestSecret
```

Output:

```
{
    "ARN": "arn:aws:secretsmanager:us-west-2:123456789012:secret:MyTestSecret-
Ca8JGt",
    "Name": "MyTestSecret",
    "Description": "My test secret",
    "KmsKeyId": "arn:aws:kms:us-west-2:123456789012:key/EXAMPLE1-90ab-cdef-fedc-
ba987EXAMPLE",
    "RotationEnabled": true,
    "RotationLambdaARN": "arn:aws:lambda:us-
west-2:123456789012:function:MyTestRotationLambda",
    "RotationRules": {
        "AutomaticallyAfterDays": 2,
        "Duration": "2h",
        "ScheduleExpression": "cron(0 16 1,15 * ? *)"
    },
    "LastRotatedDate": 1525747253.72,
    "LastChangedDate": 1523477145.729,
    "LastAccessedDate": 1524572133.25,
    "Tags": [
        {
            "Key": "SecondTag",
            "Value": "AnotherValue"
        },
        {
            "Key": "FirstTag",
            "Value": "SomeValue"
        }
    ],
    "VersionIdsToStages": {
        "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111": [
            "AWSPREVIOUS"
        ],
        "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222": [
            "AWSCURRENT"
        ],
        "a1b2c3d4-5678-90ab-cdef-EXAMPLE33333": [
            "AWSPENDING"
        ]
```

```
    },
    "CreatedDate": 1521534252.66,
    "PrimaryRegion": "us-west-2",
    "ReplicationStatus": [
        {
            "Region": "eu-west-3",
            "KmsKeyId": "alias/aws/secretsmanager",
            "Status": "InSync",
            "StatusMessage": "Replication succeeded"
        }
    ]
}
```

Per ulteriori informazioni, consulta Segreto nella Guida per l'utente di Secrets Manager.

- Per i dettagli sull'API, consulta DescribeSecret AWS CLICommand Reference.

`get-random-password`

Il seguente esempio di codice mostra come utilizzare`get-random-password`.

AWS CLI

Come generare una password casuale

L'esempio `get-random-password` seguente genera una password casuale che contiene 20 caratteri e include almeno una lettera maiuscola, una lettera minuscola, un numero e un segno di punteggiatura.

```
aws secretsmanager get-random-password \
    --require-each-included-type \
    --password-length 20
```

Output:

```
{
    "RandomPassword": "EXAMPLE-PASSWORD"
}
```

Per ulteriori informazioni, consulta Creazione e gestione di segreti nella Guida per l'utente di Secrets Manager.

- Per i dettagli sull'API, consulta GetRandomPassword AWS CLICommand Reference.

`get-resource-policy`

Il seguente esempio di codice mostra come utilizzare`get-resource-policy`.

AWS CLI

Come recuperare la policy basata sulle risorse collegata a un segreto

L'esempio di `get-resource-policy` seguente mostra come recuperare la policy basata su risorse collegata a un segreto.

```
aws secretsmanager get-resource-policy \
    --secret-id MyTestSecret
```

Output:

```
{
    "ARN": "arn:aws:secretsmanager:us-west-2:123456789012:secret:MyTestSecret-
a1b2c3",
    "Name": "MyTestSecret",
    "ResourcePolicy": "{\n\"Version\":\"2012-10-17\",\n\"Statement\":[{\n\"Effect\":
\"Allow\",\n
    \"Principal\":{\n\"AWS\":\"arn:aws:iam::123456789012:root\"\n},\n\"Action\":
    \"secretsmanager:GetSecretValue\",\n\"Resource\":\"*\"\n}]\n}"
}
```

Per ulteriori informazioni, consulta Autenticazione e controllo degli accessi nella Guida per l'utente di Secrets Manager.

- Per i dettagli sull'API, consulta GetResourcePolicy AWS CLICommand Reference.

`get-secret-value`

Il seguente esempio di codice mostra come utilizzare`get-secret-value`.

AWS CLI

Esempio 1: come recuperare il valore crittografato di un segreto

L'esempio di `get-secret-value` seguente mostra come recuperare il valore corrente del segreto.

```
aws secretsmanager get-secret-value \
    --secret-id MyTestSecret
```

Output:

```
{
    "ARN": "arn:aws:secretsmanager:us-west-2:123456789012:secret:MyTestSecret-
a1b2c3",
    "Name": "MyTestSecret",
    "VersionId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "SecretString": "{\"user\":\"diegor\",\"password\":\"EXAMPLE-PASSWORD\"}",
    "VersionStages": [
        "AWSCURRENT"
    ],
    "CreatedDate": 1523477145.713
}
```

Per ulteriori informazioni, consulta Recupero di un segreto nella Guida per l'utente di Secrets Manager.

Esempio 2: come recuperare il valore del segreto precedente

L'esempio `get-secret-value` seguente mostra come recuperare il valore precedente del segreto:

```
aws secretsmanager get-secret-value \
    --secret-id MyTestSecret
    --version-stage AWSPREVIOUS
```

Output:

```
{
    "ARN": "arn:aws:secretsmanager:us-west-2:123456789012:secret:MyTestSecret-
a1b2c3",
    "Name": "MyTestSecret",
    "VersionId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
    "SecretString": "{\"user\":\"diegor\",\"password\":\"PREVIOUS-EXAMPLE-PASSWORD
\"}",
    "VersionStages": [
        "AWSPREVIOUS"
    ],
```

```
    "CreatedDate": 1523477145.713
}
```

Per ulteriori informazioni, consulta Recupero di un segreto nella Guida per l'utente di Secrets Manager.

- Per i dettagli sull'API, consulta GetSecretValue AWS CLICommand Reference.

`list-secret-version-ids`

Il seguente esempio di codice mostra come utilizzare`list-secret-version-ids`.

AWS CLI

Come elencare tutte le versioni del segreto associate a un segreto

L'esempio `list-secret-version-ids` seguente ottiene un elenco di tutte le versioni di un segreto.

```
aws secretsmanager list-secret-version-ids \
    --secret-id MyTestSecret
```

Output:

```
{
  "Versions": [
    {
        "VersionId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
        "VersionStages": [
            "AWSPREVIOUS"
        ],
        "LastAccessedDate": 1523477145.713,
        "CreatedDate": 1523477145.713
    },
    {
        "VersionId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
        "VersionStages": [
            "AWSCURRENT"
        ],
        "LastAccessedDate": 1523477145.713,
        "CreatedDate": 1523486221.391
    },
```

```
    {
        "CreatedDate": 1.51197446236E9,
        "VersionId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE33333;"
    }
    ],
    "ARN": "arn:aws:secretsmanager:us-west-2:123456789012:secret:MyTestSecret-
a1b2c3",
    "Name": "MyTestSecret"
}
```

Per ulteriori informazioni, consulta Versione nella Guida per l'utente di Secrets Manager.

- Per i dettagli sull'API, consulta ListSecretVersionIds AWS CLICommand Reference.

`list-secrets`

Il seguente esempio di codice mostra come utilizzare`list-secrets`.

AWS CLI

Esempio 1: come elencare i segreti nell'account

L'esempio di `list-secrets` seguente mostra come ottenere un elenco dei segreti del proprio account.

```
aws secretsmanager list-secrets
```

Output:

```
{
    "SecretList": [
        {
            "ARN": "arn:aws:secretsmanager:us-
west-2:123456789012:secret:MyTestSecret-a1b2c3",
            "Name": "MyTestSecret",
            "LastChangedDate": 1523477145.729,
            "SecretVersionsToStages": {
                "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111": [
                    "AWSCURRENT"
                ]
            }
        },
```

```
        {
            "ARN": "arn:aws:secretsmanager:us-
west-2:123456789012:secret:AnotherSecret-d4e5f6",
            "Name": "AnotherSecret",
            "LastChangedDate": 1523482025.685,
            "SecretVersionsToStages": {
                "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222": [
                    "AWSCURRENT"
                ]
            }
        }
    ]
}
```

Per ulteriori informazioni, consulta Cercare un segreto nella Guida per l'utente di Secrets Manager.

Esempio 2: come filtrare l'elenco dei segreti nell'account

L'esempio `list-secrets` seguente ottiene un elenco dei segreti nell'account in uso che contengono `Test` nel nome. Il filtro per nome fa distinzione tra maiuscole e minuscole.

```
aws secretsmanager list-secrets \
    --filter Key="name",Values="Test"
```

Output:

```
{
    "SecretList": [
        {
            "ARN": "arn:aws:secretsmanager:us-
west-2:123456789012:secret:MyTestSecret-a1b2c3",
            "Name": "MyTestSecret",
            "LastChangedDate": 1523477145.729,
            "SecretVersionsToStages": {
                "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111": [
                    "AWSCURRENT"
                ]
            }
        }
    ]
}
```

Per ulteriori informazioni, consulta Cercare un segreto nella Guida per l'utente di Secrets Manager.

Esempio 3: come elencare i segreti presenti in un account gestito da un altro servizio

L'esempio `list-secrets` seguente restituisce i segreti nel tuo account che sono gestiti da Amazon RDS.

```
aws secretsmanager list-secrets \
    --filter Key="owning-service",Values="rds"
```

Output:

```
{
    "SecretList": [
        {
            "Name": "rds!cluster-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
            "Tags": [
                {
                    "Value": "arn:aws:rds:us-
west-2:123456789012:cluster:database-1",
                    "Key": "aws:rds:primaryDBClusterArn"
                },
                {
                    "Value": "rds",
                    "Key": "aws:secretsmanager:owningService"
                }
            ],
            "RotationRules": {
                "AutomaticallyAfterDays": 1
            },
            "LastChangedDate": 1673477781.275,
            "LastRotatedDate": 1673477781.26,
            "SecretVersionsToStages": {
                "a1b2c3d4-5678-90ab-cdef-EXAMPLEaaaaa": [
                    "AWSPREVIOUS"
                ],
                "a1b2c3d4-5678-90ab-cdef-EXAMPLEbbbbb": [
                    "AWSCURRENT",
                    "AWSPENDING"
                ]
            },
            "OwningService": "rds",
```

```
            "RotationEnabled": true,
            "CreatedDate": 1673467300.7,
            "LastAccessedDate": 1673395200.0,
            "ARN": "arn:aws:secretsmanager:us-west-2:123456789012:secret:rds!
cluster-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111-a1b2c3",
            "Description": "Secret associated with primary RDS DB cluster:
 arn:aws:rds:us-west-2:123456789012:cluster:database-1"
        }
    ]
}
```

Per ulteriori informazioni, consulta Segreti gestiti da altri servizi nella Guida per l'utente di Secrets Manager.

- Per i dettagli sull'API, consulta ListSecrets AWS CLICommand Reference.

`put-resource-policy`

Il seguente esempio di codice mostra come utilizzare`put-resource-policy`.

AWS CLI

Come aggiungere una policy basata sulle risorse a un segreto

L'esempio di `put-resource-policy` seguente mostra come aggiungere una policy di autorizzazioni a un segreto, verificando innanzitutto che la policy non fornisca un accesso ampio al segreto. La policy viene letta da un file. Per ulteriori informazioni, consulta Caricamento dei parametri AWS CLI da un file nella Guida per l'utente della AWS CLI.

```
aws secretsmanager put-resource-policy \
    --secret-id MyTestSecret \
    --resource-policy file://mypolicy.json \
    --block-public-policy
```

Contenuto di `mypolicy.json`:

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
```

```
                "AWS": "arn:aws:iam::123456789012:role/MyRole"
            },
            "Action": "secretsmanager:GetSecretValue",
            "Resource": "*"
        }
    ]
}
```

Output:

```
{
    "ARN": "arn:aws:secretsmanager:us-west-2:123456789012:secret:MyTestSecret-
a1b2c3",
    "Name": "MyTestSecret"
}
```

Per ulteriori informazioni, consulta Allegare una policy di autorizzazione a un segreto nella Guida per l'utente di Secrets Manager.

- Per i dettagli sull'API, consulta AWS CLI Command PutResourcePolicyReference.

`put-secret-value`

Il seguente esempio di codice mostra come utilizzare`put-secret-value`.

AWS CLI

Esempio 1: come archiviare il nuovo valore di un segreto in un segreto

L'esempio `put-secret-value` seguente mostra come creare una nuova versione di un segreto con due coppie chiave-valore.

```
aws secretsmanager put-secret-value \
    --secret-id MyTestSecret \
    --secret-string "{\"user\":\"diegor\",\"password\":\"EXAMPLE-PASSWORD\"}"
```

Output:

```
{
    "ARN": "arn:aws:secretsmanager:us-
west-2:123456789012:secret:MyTestSecret-1a2b3c",
    "Name": "MyTestSecret",
```

```
    "VersionId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "VersionStages": [
        "AWSCURRENT"
    ]
}
```

Per ulteriori informazioni, consulta Modificare un segreto nella Guida per l'utente di Secrets Manager.

Esempio 2: come archiviare il nuovo valore di un segreto dalle credenziali in un file JSON

L'esempio di `put-secret-value` seguente mostra come creare una nuova versione di un segreto partendo dalle credenziali in un file. Per ulteriori informazioni, consulta Caricamento dei parametri AWS CLI da un file nella Guida per l'utente della AWS CLI.

```
aws secretsmanager put-secret-value \
    --secret-id MyTestSecret \
    --secret-string file://mycreds.json
```

Contenuto di `mycreds.json`:

```
{
  "engine": "mysql",
  "username": "saanvis",
  "password": "EXAMPLE-PASSWORD",
  "host": "my-database-endpoint.us-west-2.rds.amazonaws.com",
  "dbname": "myDatabase",
  "port": "3306"
}
```

Output:

```
{
    "ARN": "arn:aws:secretsmanager:us-west-2:123456789012:secret:MyTestSecret-
a1b2c3",
    "Name": "MyTestSecret",
    "VersionId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "VersionStages": [
        "AWSCURRENT"
    ]
}
```

Per ulteriori informazioni, consulta Modificare un segreto nella Guida per l'utente di Secrets Manager.

- Per i dettagli sull'API, consulta AWS CLI Command PutSecretValueReference.

`remove-regions-from-replication`

Il seguente esempio di codice mostra come utilizzare`remove-regions-from-replication`.

AWS CLI

Come eliminare un segreto di replica

L'esempio di `remove-regions-from-replication` seguente mostra come eliminare un segreto di replica nella Regione eu-west-3. Per eliminare un segreto primario replicato in altre Regioni, è necessario dapprima eliminare le relative repliche e poi chiamare `delete-secret`.

```
aws secretsmanager remove-regions-from-replication \
    --secret-id MyTestSecret \
    --remove-replica-regions eu-west-3
```

Output:

```
{
    "ARN": "arn:aws:secretsmanager:us-
west-2:123456789012:secret:MyTestSecret-1a2b3c",
    "ReplicationStatus": []
}
```

Per ulteriori informazioni, consulta Eliminazione di un segreto di replica nella Guida per l'utente di Secrets Manager.

- Per i dettagli sull'API, consulta RemoveRegionsFromReplication AWS CLICommand Reference.

`replicate-secret-to-regions`

Il seguente esempio di codice mostra come utilizzare`replicate-secret-to-regions`.

AWS CLI

Come replicare un segreto in un'altra Regione

Nell'esempio `replicate-secret-to-regions` seguente, un segreto viene replicato nella Regione eu-west-3. La replica è crittografata con la chiave AWS `aws/secretsmanager` gestita.

```
aws secretsmanager replicate-secret-to-regions \
    --secret-id MyTestSecret \
    --add-replica-regions Region=eu-west-3
```

Output:

```
{
    "ARN": "arn:aws:secretsmanager:us-
west-2:123456789012:secret:MyTestSecret-1a2b3c",
    "ReplicationStatus": [
        {
            "Region": "eu-west-3",
            "KmsKeyId": "alias/aws/secretsmanager",
            "Status": "InProgress"
        }
    ]
}
```

Per ulteriori informazioni, consulta Replica di un segreto in un'altra Regione nella Guida per l'utente di Secrets Manager.

- Per i dettagli sull'API, consulta ReplicateSecretToRegions AWS CLICommand Reference.

`restore-secret`

Il seguente esempio di codice mostra come utilizzare`restore-secret`.

AWS CLI

Come ripristinare un segreto precedentemente eliminato

L'esempio di `restore-secret` seguente mostra il ripristino di un segreto per il quale in precedenza era stata pianificata l'eliminazione.

```
aws secretsmanager restore-secret \
    --secret-id MyTestSecret
```

Output:

```
{
    "ARN": "arn:aws:secretsmanager:us-west-2:123456789012:secret:MyTestSecret-
a1b2c3",
    "Name": "MyTestSecret"
}
```

Per ulteriori informazioni, consulta Eliminare un segreto nella Guida per l'utente di Secrets Manager.

- Per i dettagli sull'API, consulta RestoreSecret AWS CLICommand Reference.

`rotate-secret`

Il seguente esempio di codice mostra come utilizzare`rotate-secret`.

AWS CLI

Esempio 1: come configurare e avviare la rotazione automatica di un segreto

L'esempio `rotate-secret` seguente configura e avvia la rotazione automatica per un segreto. Secrets Manager ruota immediatamente il segreto una volta e successivamente lo ruota ogni otto ore in una finestra di due ore. L'output mostra il valore `VersionId` della nuova versione del segreto creata dalla rotazione.

```
aws secretsmanager rotate-secret \
    --secret-id MyTestDatabaseSecret \
    --rotation-lambda-arn arn:aws:lambda:us-
west-2:1234566789012:function:SecretsManagerTestRotationLambda \
    --rotation-rules "{\"ScheduleExpression\": \"cron(0 8/8 * * ? *)\", \"Duration
\": \"2h\"}"
```

Output:

```
{
    "ARN": "aws:arn:secretsmanager:us-
west-2:123456789012:secret:MyTestDatabaseSecret-a1b2c3",
    "Name": "MyTestDatabaseSecret",
    "VersionId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
}
```

Per ulteriori informazioni, consulta Ruotare i segreti nella Guida per l'utente di Secrets Manager.

Esempio 2: come configurare e avviare la rotazione automatica su un intervallo di rotazione

L'esempio `rotate-secret` seguente configura e avvia la rotazione automatica per un segreto. Secrets Manager ruota immediatamente il segreto una volta e successivamente lo ruota ogni dieci giorni. L'output mostra il valore `VersionId` della nuova versione del segreto creata dalla rotazione.

```
aws secretsmanager rotate-secret \
    --secret-id MyTestDatabaseSecret \
    --rotation-lambda-arn arn:aws:lambda:us-
west-2:1234566789012:function:SecretsManagerTestRotationLambda \
    --rotation-rules "{\"ScheduleExpression\": \"rate(10 days)\"}"
```

Output:

```
{
    "ARN": "aws:arn:secretsmanager:us-
west-2:123456789012:secret:MyTestDatabaseSecret-a1b2c3",
    "Name": "MyTestDatabaseSecret",
    "VersionId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
}
```

Per ulteriori informazioni, consulta Ruotare i segreti nella Guida per l'utente di Secrets Manager.

Esempio 3: come ruotare immediatamente un segreto

L'esempio di `rotate-secret` seguente mostra come avviare una rotazione immediata. L'output mostra il valore `VersionId` della nuova versione del segreto creata dalla rotazione. Il segreto deve avere già la rotazione configurata.

```
aws secretsmanager rotate-secret \
    --secret-id MyTestDatabaseSecret
```

Output:

```
{
    "ARN": "aws:arn:secretsmanager:us-
west-2:123456789012:secret:MyTestDatabaseSecret-a1b2c3",
    "Name": "MyTestDatabaseSecret",
    "VersionId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
```

```
}
```

Per ulteriori informazioni, consulta Ruotare i segreti nella Guida per l'utente di Secrets Manager.

- Per i dettagli sull'API, consulta RotateSecret AWS CLICommand Reference.

`stop-replication-to-replica`

Il seguente esempio di codice mostra come utilizzare`stop-replication-to-replica`.

AWS CLI

Come promuovere un segreto di replica a primario

L'esempio di `stop-replication-to-replica` seguente mostra come rimuovere il collegamento tra un segreto di replica e quello primario. Il segreto di replica viene promosso a segreto primario nella Regione della replica. È necessario effettuare una chiamata `stop-replication-to-replica` dall'interno della Regione della replica.

```
aws secretsmanager stop-replication-to-replica \
    --secret-id MyTestSecret
```

Output:

```
{
    "ARN": "arn:aws:secretsmanager:us-west-2:123456789012:secret:MyTestSecret-
a1b2c3"
}
```

Per ulteriori informazioni, consulta Promuovere un segreto di replica nella Guida per l'utente di Secrets Manager.

- Per i dettagli sull'API, consulta StopReplicationToReplica AWS CLICommand Reference.

`tag-resource`

Il seguente esempio di codice mostra come utilizzare`tag-resource`.

AWS CLI

Esempio 1: come aggiungere un tag a un segreto

L'esempio seguente mostra come collegare un tag con una sintassi abbreviata.

```
aws secretsmanager tag-resource \
    --secret-id MyTestSecret \
    --tags Key=FirstTag,Value=FirstValue
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Taggare i segreti nella Guida per l'utente di Secrets Manager.

Esempio 2: come aggiungere tag a un segreto

L'esempio di `tag-resource` seguente mostra come collegare due tag chiave-valore a un segreto.

```
aws secretsmanager tag-resource \
    --secret-id MyTestSecret \
    --tags '[{"Key": "FirstTag", "Value": "FirstValue"}, {"Key": "SecondTag",
 "Value": "SecondValue"}]'
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Taggare i segreti nella Guida per l'utente di Secrets Manager.

- Per i dettagli sull'API, consulta TagResource AWS CLICommand Reference.

`untag-resource`

Il seguente esempio di codice mostra come utilizzare`untag-resource`.

AWS CLI

Come rimuovere tag da un segreto

L'esempio di `untag-resource` seguente mostra come rimuovere due tag da un segreto. Per ogni tag, vengono rimossi sia la chiave che il valore.

```
aws secretsmanager untag-resource \
    --secret-id MyTestSecret \
    --tag-keys '[ "FirstTag", "SecondTag"]'
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Taggare i segreti nella Guida per l'utente di Secrets Manager.

- Per i dettagli sull'API, consulta UntagResource AWS CLICommand Reference.

`update-secret-version-stage`

Il seguente esempio di codice mostra come utilizzare`update-secret-version-stage`.

AWS CLI

Esempio 1: come ripristinare la versione precedente di un segreto

L'`update-secret-version-stage`esempio seguente sposta l'etichetta staging AWS CURRENT alla versione precedente di un segreto, che ripristina il segreto alla versione precedente. Per trovare l'ID della versione precedente, utilizza `list-secret-version-ids`. Per questo esempio, la versione con l'etichetta AWS CURRENT è a1b2c3d4-5678-90ab-cdef- e la versione con l'etichetta PREVIOUS è a1b2c3d4-5678-90ab-cdef-. EXAMPLE11111 AWS EXAMPLE22222 In questo esempio, si sposta AWS l'etichetta CURRENT dalla versione 11111 alla 22222. Poiché l'etichetta AWS CURRENT viene rimossa da una versione, sposta `update-secret-version-stage` automaticamente l'etichetta AWS PREVIOUS a quella versione (11111). L'effetto è che le versioni AWS CURRENT e AWS PREVIOUS vengono scambiate.

```
aws secretsmanager update-secret-version-stage \
    --secret-id MyTestSecret \
    --version-stage AWSCURRENT \
    --move-to-version-id a1b2c3d4-5678-90ab-cdef-EXAMPLE22222 \
    --remove-from-version-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Output:

```
{
    "ARN": "arn:aws:secretsmanager:us-west-2:123456789012:secret:MyTestSecret-
a1b2c3",
    "Name": "MyTestSecret"
}
```

Per ulteriori informazioni, consulta Versione nella Guida per l'utente di Secrets Manager.

Esempio 2: come aggiungere un'etichetta temporanea collegata a una versione di un segreto

L'esempio `update-secret-version-stage` seguente aggiunge un'etichetta temporanea a una versione di un segreto. Puoi esaminare i risultati eseguendo `list-secret-version-ids` e visualizzando il campo di risposta `VersionStages` per la versione interessata.

```
aws secretsmanager update-secret-version-stage \
    --secret-id MyTestSecret \
    --version-stage STAGINGLABEL1 \
    --move-to-version-id EXAMPLE1-90ab-cdef-fedc-ba987EXAMPLE
```

Output:

```
{
    "ARN": "arn:aws:secretsmanager:us-west-2:123456789012:secret:MyTestSecret-
a1b2c3",
    "Name": "MyTestSecret"
}
```

Per ulteriori informazioni, consulta Versione nella Guida per l'utente di Secrets Manager.

Esempio 3: come eliminare un'etichetta temporanea collegata a una versione di un segreto

L'esempio `update-secret-version-stage` seguente elimina un'etichetta temporanea collegata a una versione di un segreto. Puoi esaminare i risultati eseguendo `list-secret-version-ids` e visualizzando il campo di risposta `VersionStages` per la versione interessata.

```
aws secretsmanager update-secret-version-stage \
    --secret-id MyTestSecret \
    --version-stage STAGINGLABEL1 \
    --remove-from-version-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Output:

```
{
    "ARN": "arn:aws:secretsmanager:us-west-2:123456789012:secret:MyTestSecret-
a1b2c3",
    "Name": "MyTestSecret"
}
```

Per ulteriori informazioni, consulta Versione nella Guida per l'utente di Secrets Manager.

- Per i dettagli sull'API, vedere UpdateSecretVersionStagein AWS CLI Command Reference.

`update-secret`

Il seguente esempio di codice mostra come utilizzare`update-secret`.

AWS CLI

Esempio 1: come aggiornare la descrizione di un segreto

L'esempio di `update-secret` seguente mostra come aggiornare la descrizione di un segreto.

```
aws secretsmanager update-secret \
    --secret-id MyTestSecret \
    --description "This is a new description for the secret."
```

Output:

```
{
    "ARN": "arn:aws:secretsmanager:us-west-2:123456789012:secret:MyTestSecret-
a1b2c3",
    "Name": "MyTestSecret"
}
```

Per ulteriori informazioni, consulta Modificare un segreto nella Guida per l'utente di Secrets Manager.

Esempio 2: come aggiornare la chiave di crittografia associata a un segreto

L'esempio di `update-secret` seguente mostra come aggiornare la chiave KMS utilizzata per crittografare il valore del segreto. La chiave KMS deve trovarsi nella stessa Regione del segreto.

```
aws secretsmanager update-secret \
    --secret-id MyTestSecret \
    --kms-key-id arn:aws:kms:us-west-2:123456789012:key/EXAMPLE1-90ab-cdef-fedc-
ba987EXAMPLE
```

Output:

```
{
    "ARN": "arn:aws:secretsmanager:us-west-2:123456789012:secret:MyTestSecret-
a1b2c3",
```

```
    "Name": "MyTestSecret"
}
```

Per ulteriori informazioni, consulta Modificare un segreto nella Guida per l'utente di Secrets Manager.

- Per i dettagli sull'API, consulta UpdateSecret AWS CLICommand Reference.

`validate-resource-policy`

Il seguente esempio di codice mostra come utilizzare`validate-resource-policy`.

AWS CLI

Come convalidare un policy delle risorse

L'esempio `validate-resource-policy` seguente verifica che una policy delle risorse non conceda un ampio accesso a un segreto. La policy viene letta da un file su disco. Per ulteriori informazioni, consulta Caricamento dei parametri AWS CLI da un file nella Guida per l'utente della AWS CLI.

```
aws secretsmanager validate-resource-policy \
    --resource-policy file://mypolicy.json
```

Contenuto di `mypolicy.json`:

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::123456789012:role/MyRole"
            },
            "Action": "secretsmanager:GetSecretValue",
            "Resource": "*"
        }
    ]
}
```

Output:

```
{
    "PolicyValidationPassed": true,
    "ValidationErrors": []
}
```

Per ulteriori informazioni, consulta Riferimento per le autorizzazioni per Secrets Manager nella Guida per l'utente di Secrets Manager.

- Per i dettagli sull'API, consulta AWS CLI Command ValidateResourcePolicyReference.

Esempi di CSPM di Security Hub utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando il CSPM AWS Command Line Interface with Security Hub.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`accept-administrator-invitation`

Il seguente esempio di codice mostra come utilizzare. `accept-administrator-invitation`

AWS CLI

Come accettare un invito da un account amministratore

L'esempio `accept-administrator-invitation` seguente accetta l'invito dall'account amministratore specificato.

```
aws securityhub accept-invitation \
    --administrator-id 123456789012 \
```

```
    --invitation-id 7ab938c5d52d7904ad09f9e7c20cc4eb
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Gestione degli account degli amministratori e dei membri nella Guida per l'utente di Centrale di sicurezza AWS .

- Per i dettagli sull'API, consulta AcceptAdministratorInvitation AWS CLICommand Reference.

`accept-invitation`

Il seguente esempio di codice mostra come utilizzare`accept-invitation`.

AWS CLI

Come accettare un invito da un account amministratore

L'esempio `accept-invitation` seguente accetta l'invito dall'account amministratore specificato.

```
aws securityhub accept-invitation \
    --master-id 123456789012 \
    --invitation-id 7ab938c5d52d7904ad09f9e7c20cc4eb
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Gestione degli account degli amministratori e dei membri nella Guida per l'utente di Centrale di sicurezza AWS .

- Per i dettagli sull'API, consulta AcceptInvitation AWS CLICommand Reference.

`batch-delete-automation-rules`

Il seguente esempio di codice mostra come utilizzare`batch-delete-automation-rules`.

AWS CLI

Come eliminare le regole di automazione

L'esempio `batch-delete-automation-rules` seguente elimina la regola di automazione specificata. È possibile eliminare una o più regole con un solo comando. Solo l'account amministratore di Centrale di sicurezza può eseguire questo comando.

```
aws securityhub batch-delete-automation-rules \
    --automation-rules-arns '["arn:aws:securityhub:us-
east-1:123456789012:automation-rule/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"]'
```

Output:

```
{
    "ProcessedAutomationRules": [
        "arn:aws:securityhub:us-east-1:123456789012:automation-rule/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
    ],
    "UnprocessedAutomationRules": []
}
```

Per ulteriori informazioni, consulta Eliminare le regole di automazione nella Guida per l'utente di Centrale di sicurezza AWS .

- Per i dettagli sull'API, consulta BatchDeleteAutomationRules AWS CLICommand Reference.

`batch-disable-standards`

Il seguente esempio di codice mostra come utilizzare`batch-disable-standards`.

AWS CLI

Come disabilitare uno standard

L'esempio `batch-disable-standards` seguente disabilita lo standard associato all'ARN della sottoscrizione specificata.

```
aws securityhub batch-disable-standards \
    --standards-subscription-arns "arn:aws:securityhub:us-
west-1:123456789012:subscription/pci-dss/v/3.2.1"
```

Output:

```
{
    "StandardsSubscriptions": [
        {
            "StandardsArn": "arn:aws:securityhub:eu-central-1::standards/pci-dss/
v/3.2.1",
```

```
            "StandardsInput": { },
            "StandardsStatus": "DELETING",
            "StandardsSubscriptionArn": "arn:aws:securityhub:us-
west-1:123456789012:subscription/pci-dss/v/3.2.1"
        }
    ]
}
```

Per ulteriori informazioni, consulta Disabilitazione o abilitazione di uno standard di sicurezza nella Guida per l'utente di Centrale di sicurezza AWS .

- Per i dettagli sull'API, consulta BatchDisableStandards AWS CLICommand Reference.

`batch-enable-standards`

Il seguente esempio di codice mostra come utilizzare`batch-enable-standards`.

AWS CLI

Come abilitare uno standard

L'esempio `batch-enable-standards` seguente abilita lo standard PCI DSS per l'account richiedente.

```
aws securityhub batch-enable-standards \
    --standards-subscription-requests '{"StandardsArn":"arn:aws:securityhub:us-
west-1::standards/pci-dss/v/3.2.1"}'
```

Output:

```
{
    "StandardsSubscriptions": [
        {
            "StandardsArn": "arn:aws:securityhub:us-west-1::standards/pci-dss/
v/3.2.1",
            "StandardsInput": { },
            "StandardsStatus": "PENDING",
            "StandardsSubscriptionArn": "arn:aws:securityhub:us-
west-1:123456789012:subscription/pci-dss/v/3.2.1"
        }
    ]
}
```

Per ulteriori informazioni, consulta Disabilitazione o abilitazione di uno standard di sicurezza nella Guida per l'utente di Centrale di sicurezza AWS .

- Per i dettagli sull'API, consulta BatchEnableStandards AWS CLICommand Reference.

`batch-get-automation-rules`

Il seguente esempio di codice mostra come utilizzare`batch-get-automation-rules`.

AWS CLI

Come ottenere i dettagli sulle regole di automazione

L'esempio `batch-get-automation-rules` seguente ottiene i dettagli per la regola di automazione specificata. È possibile ottenere dettagli per una o più regole con un solo comando.

```
aws securityhub batch-get-automation-rules \
    --automation-rules-arns '["arn:aws:securityhub:us-
east-1:123456789012:automation-rule/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"]'
```

Output:

```
{
    "Rules": [
        {
            "RuleArn": "arn:aws:securityhub:us-east-1:123456789012:automation-rule/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
            "RuleStatus": "ENABLED",
            "RuleOrder": 1,
            "RuleName": "Suppress informational findings",
            "Description": "Suppress GuardDuty findings with Informational
 severity",
            "IsTerminal": false,
            "Criteria": {
                "ProductName": [
                    {
                        "Value": "GuardDuty",
                        "Comparison": "EQUALS"
                    }
                ],
                "SeverityLabel": [
                    {
```

```
                        "Value": "INFORMATIONAL",
                        "Comparison": "EQUALS"
                    }
                ],
                "WorkflowStatus": [
                    {
                        "Value": "NEW",
                        "Comparison": "EQUALS"
                    }
                ],
                "RecordState": [
                    {
                        "Value": "ACTIVE",
                        "Comparison": "EQUALS"
                    }
                ]
            },
            "Actions": [
                {
                    "Type": "FINDING_FIELDS_UPDATE",
                    "FindingFieldsUpdate": {
                        "Note": {
                            "Text": "Automatically suppress GuardDuty findings with
 Informational severity",
                            "UpdatedBy": "sechub-automation"
                        },
                        "Workflow": {
                            "Status": "SUPPRESSED"
                        }
                    }
                }
            ],
            "CreatedAt": "2023-05-31T17:56:14.837000+00:00",
            "UpdatedAt": "2023-05-31T17:59:38.466000+00:00",
            "CreatedBy": "arn:aws:iam::123456789012:role/Admin"
        }
    ],
    "UnprocessedAutomationRules": []
}
```

Per ulteriori informazioni, consulta Visualizzazione delle regole di automazione nella Guida per l'utente di Centrale di sicurezza AWS .

- Per i dettagli sull'API, consulta BatchGetAutomationRules AWS CLICommand Reference.

batch-get-configuration-policy-associations

Il seguente esempio di codice mostra come utilizzarebatch-get-configuration-policy-associations.

AWS CLI

Come ottenere i dettagli dell'associazione di configurazione per un batch di destinazioni

L'esempio batch-get-configuration-policy-associations seguente recupera i dettagli dell'associazione per le destinazioni specificate. È possibile fornire l'account IDs IDs, l'unità organizzativa o l'ID root per l'obiettivo.

```
aws securityhub batch-get-configuration-policy-associations \
    --target '{"OrganizationalUnitId": "ou-6hi7-8j91kl2m"}'
```

Output:

```
{
    "ConfigurationPolicyId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE33333",
    "TargetId": "ou-6hi7-8j91kl2m",
    "TargetType": "ORGANIZATIONAL_UNIT",
    "AssociationType": "APPLIED",
    "UpdatedAt": "2023-09-26T21:13:01.816000+00:00",
    "AssociationStatus": "SUCCESS",
    "AssociationStatusMessage": "Association applied successfully on this target."
}
```

Per ulteriori informazioni, consulta Visualizzazione delle policy di configurazione di Centrale di sicurezza nella Guida per l'utente di Centrale di sicurezza AWS .

- Per i dettagli sull'API, consulta BatchGetConfigurationPolicyAssociations AWS CLICommand Reference.

batch-get-security-controls

Il seguente esempio di codice mostra come utilizzarebatch-get-security-controls.

AWS CLI

Come ottenere i dettagli del controllo di sicurezza

L'`batch-get-security-controls`esempio seguente ottiene i dettagli per i controlli di sicurezza ACM.1 e IAM.1 nell'account e nella regione correnti AWS . AWS

```
aws securityhub batch-get-security-controls \
    --security-control-ids '["ACM.1", "IAM.1"]'
```

Output:

```
{
    "SecurityControls": [
        {
            "SecurityControlId": "ACM.1",
            "SecurityControlArn": "arn:aws:securityhub:us-
east-2:123456789012:security-control/ACM.1",
            "Title": "Imported and ACM-issued certificates should be renewed after a
 specified time period",
            "Description": "This control checks whether an AWS Certificate Manager
 (ACM) certificate is renewed within the specified time period. It checks both
 imported certificates and certificates provided by ACM. The control fails if the
 certificate isn't renewed within the specified time period. Unless you provide a
 custom parameter value for the renewal period, Security Hub uses a default value of
 30 days.",
            "RemediationUrl": "https://docs.aws.amazon.com/console/securityhub/
ACM.1/remediation",
            "SeverityRating": "MEDIUM",
            "SecurityControlStatus": "ENABLED"
            "UpdateStatus": "READY",
            "Parameters": {
                "daysToExpiration": {
                    "ValueType": CUSTOM,
                    "Value": {
                        "Integer": 15
                    }
                }
            },
            "LastUpdateReason": "Updated control parameter"
        },
        {
            "SecurityControlId": "IAM.1",
            "SecurityControlArn": "arn:aws:securityhub:us-
east-2:123456789012:security-control/IAM.1",
            "Title": "IAM policies should not allow full \"*\" administrative
 privileges",
```

```
            "Description": "This AWS control checks whether the default version of
 AWS Identity and Access Management (IAM) policies (also known as customer managed
 policies) do not have administrator access with a statement that has \"Effect\":
 \"Allow\" with \"Action\": \"*\" over \"Resource\": \"*\". It only checks for
 the Customer Managed Policies that you created, but not inline and AWS Managed
 Policies.",
            "RemediationUrl": "https://docs.aws.amazon.com/console/securityhub/
IAM.1/remediation",
            "SeverityRating": "HIGH",
            "SecurityControlStatus": "ENABLED"
            "UpdateStatus": "READY",
            "Parameters": {}
        }
    ]
}
```

Per ulteriori informazioni, consulta Visualizzazione dei dettagli per un controllo nella Guida per l'utente di Centrale di sicurezza AWS .

- Per i dettagli sull'API, vedere BatchGetSecurityControlsin Command Reference.AWS CLI

`batch-get-standards-control-associations`

Il seguente esempio di codice mostra come utilizzare`batch-get-standards-control-associations`.

AWS CLI

Come ottenere lo stato di abilitazione di un controllo

L'esempio `batch-get-standards-control-associations` seguente identifica se i controlli specificati sono abilitati negli standard indicati.

```
aws securityhub batch-get-standards-control-associations \
    --standards-control-association-ids '[{"SecurityControlId":
 "Config.1","StandardsArn": "arn:aws:securityhub:us-east-1:123456789012:ruleset/cis-
aws-foundations-benchmark/v/1.2.0"}, {"SecurityControlId": "IAM.6","StandardsArn":
 "arn:aws:securityhub:us-east-1:123456789012:standards/aws-foundational-security-
best-practices/v/1.0.0"}]'
```

Output:

```
{
```

```
    "StandardsControlAssociationDetails": [
        {
            "StandardsArn": "arn:aws:securityhub:::ruleset/cis-aws-foundations-
benchmark/v/1.2.0",
            "SecurityControlId": "Config.1",
            "SecurityControlArn": "arn:aws:securityhub:us-
east-1:068873283051:security-control/Config.1",
            "AssociationStatus": "ENABLED",
            "RelatedRequirements": [
                "CIS AWS Foundations 2.5"
            ],
            "UpdatedAt": "2022-10-27T16:07:12.960000+00:00",
            "StandardsControlTitle": "Ensure AWS Config is enabled",
            "StandardsControlDescription": "AWS Config is a web service that
 performs configuration management of supported AWS resources within your account
 and delivers log files to you. The recorded information includes the configuration
 item (AWS resource), relationships between configuration items (AWS resources), and
 any configuration changes between resources. It is recommended to enable AWS Config
 in all regions.",
            "StandardsControlArns": [
                "arn:aws:securityhub:us-east-1:068873283051:control/cis-aws-
foundations-benchmark/v/1.2.0/2.5"
            ]
        },
        {
            "StandardsArn": "arn:aws:securityhub:us-east-1::standards/aws-
foundational-security-best-practices/v/1.0.0",
            "SecurityControlId": "IAM.6",
            "SecurityControlArn": "arn:aws:securityhub:us-
east-1:068873283051:security-control/IAM.6",
            "AssociationStatus": "DISABLED",
            "RelatedRequirements": [],
            "UpdatedAt": "2022-11-22T21:30:35.080000+00:00",
            "UpdatedReason": "test",
            "StandardsControlTitle": "Hardware MFA should be enabled for the root
 user",
            "StandardsControlDescription": "This AWS control checks whether your AWS
 account is enabled to use a hardware multi-factor authentication (MFA) device to
 sign in with root user credentials.",
            "StandardsControlArns": [
                "arn:aws:securityhub:us-east-1:068873283051:control/aws-
foundational-security-best-practices/v/1.0.0/IAM.6"
            ]
        }
```

```
    ]
}
```

Per ulteriori informazioni, consulta Abilitazione e disabilitazione dei controlli con standard specifici nella Guida per l'utente di Centrale di sicurezza AWS .

- Per i dettagli sull'API, consulta BatchGetStandardsControlAssociations AWS CLICommand Reference.

`batch-import-findings`

Il seguente esempio di codice mostra come utilizzare`batch-import-findings`.

AWS CLI

Come aggiornare un esito

L'esempio `batch-import-findings` seguente aggiorna un esito.

```
aws securityhub batch-import-findings \
     --findings '
        [{
            "AwsAccountId": "123456789012",
            "CreatedAt": "2020-05-27T17:05:54.832Z",
            "Description": "Vulnerability in a CloudTrail trail",
            "FindingProviderFields": {
                "Severity": {
                    "Label": "LOW",
                    "Original": "10"
                },
                "Types": [
                    "Software and Configuration Checks/Vulnerabilities/CVE"
                ]
            },
            "GeneratorId": "TestGeneratorId",
            "Id": "Id1",
            "ProductArn": "arn:aws:securityhub:us-
west-1:123456789012:product/123456789012/default",
            "Resources": [
                {
                    "Id": "arn:aws:cloudtrail:us-west-1:123456789012:trail/
TrailName",
                    "Partition": "aws",
```

```
                    "Region": "us-west-1",
                    "Type": "AwsCloudTrailTrail"
                }
            ],
            "SchemaVersion": "2018-10-08",
            "Title": "CloudTrail trail vulnerability",
            "UpdatedAt": "2020-06-02T16:05:54.832Z"
        }]'
```

Output:

```
{
    "FailedCount": 0,
    "SuccessCount": 1,
    "FailedFindings": []
}
```

Per ulteriori informazioni, consulta Utilizzare BatchImportFindings per creare e aggiornare i risultati nella Guida per l'utente AWS di Security Hub.

- Per i dettagli sull'API, vedere BatchImportFindingsin AWS CLI Command Reference.

`batch-update-automation-rules`

Il seguente esempio di codice mostra come utilizzare`batch-update-automation-rules`.

AWS CLI

Come aggiornare le regole di automazione

L'esempio `batch-update-automation-rules` seguente aggiornala regola di automazione specificata. È possibile aggiornare una o più regole con un solo comando. Solo l'account amministratore di Centrale di sicurezza può eseguire questo comando.

```
aws securityhub batch-update-automation-rules \
    --update-automation-rules-request-items '[ \
        { \
            "Actions": [{ \
                "Type": "FINDING_FIELDS_UPDATE", \
                "FindingFieldsUpdate": { \
                    "Note": { \
                        "Text": "Known issue that is a risk", \
```

```
                    "UpdatedBy": "sechub-automation" \
                }, \
                "Workflow": { \
                    "Status": "NEW" \
                } \
            } \
        }], \
        "Criteria": { \
            "SeverityLabel": [{ \
                "Value": "LOW", \
                "Comparison": "EQUALS" \
            }] \
        }, \
        "RuleArn": "arn:aws:securityhub:us-east-1:123456789012:automation-rule/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111", \
        "RuleOrder": 1, \
        "RuleStatus": "DISABLED" \
    } \
]'
```

Output:

```
{
    "ProcessedAutomationRules": [
        "arn:aws:securityhub:us-east-1:123456789012:automation-rule/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
    ],
    "UnprocessedAutomationRules": []
}
```

Per ulteriori informazioni, consulta Modifica delle regole di automazione nella Guida per l'utente di Centrale di sicurezza AWS .

- Per i dettagli sull'API, consulta BatchUpdateAutomationRules AWS CLICommand Reference.

`batch-update-findings`

Il seguente esempio di codice mostra come utilizzare`batch-update-findings`.

AWS CLI

Esempio 1: come aggiornare un esito

L'esempio `batch-update-findings` seguente aggiorna due esiti per aggiungere una nota, modificare l'etichetta di gravità e risolvere il flusso di lavoro.

```
aws securityhub batch-update-findings \
    --finding-identifiers '[{"Id": "arn:aws:securityhub:us-
west-1:123456789012:subscription/pci-dss/v/3.2.1/PCI.Lambda.2/finding/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111", "ProductArn": "arn:aws:securityhub:us-
west-1::product/aws/securityhub"}, {"Id": "arn:aws:securityhub:us-
west-1:123456789012:subscription/pci-dss/v/3.2.1/PCI.Lambda.2/finding/
a1b2c3d4-5678-90ab-cdef-EXAMPLE22222", "ProductArn": "arn:aws:securityhub:us-
west-1::product/aws/securityhub"}]' \
    --note '{"Text": "Known issue that is not a risk.", "UpdatedBy": "user1"}' \
    --severity '{"Label": "LOW"}' \
    --workflow '{"Status": "RESOLVED"}'
```

Output:

```
{
    "ProcessedFindings": [
        {
            "Id": "arn:aws:securityhub:us-west-1:123456789012:subscription/pci-dss/
v/3.2.1/PCI.Lambda.2/finding/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
            "ProductArn": "arn:aws:securityhub:us-west-1::product/aws/securityhub"
        },
        {
            "Id": "arn:aws:securityhub:us-west-1:123456789012:subscription/pci-dss/
v/3.2.1/PCI.Lambda.2/finding/a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
            "ProductArn": "arn:aws:securityhub:us-west-1::product/aws/securityhub"
        }
    ],
    "UnprocessedFindings": []
}
```

Per ulteriori informazioni, consulta Utilizzare BatchUpdateFindings per aggiornare un risultato nella Guida per l'utente AWS di Security Hub.

Esempio 2: come aggiornare un esito utilizzando la sintassi abbreviata

L'esempio `batch-update-findings` seguente aggiorna due esiti per aggiungere una nota, modificare l'etichetta di gravità e risolvere il flusso di lavoro utilizzando la sintassi abbreviata.

```
aws securityhub batch-update-findings \
```

```
    --finding-identifiers Id="arn:aws:securityhub:us-
west-1:123456789012:subscription/pci-dss/v/3.2.1/PCI.Lambda.2/finding/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",ProductArn="arn:aws:securityhub:us-
west-1::product/aws/securityhub" Id="arn:aws:securityhub:us-
west-1:123456789012:subscription/pci-dss/v/3.2.1/PCI.Lambda.2/finding/
a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",ProductArn="arn:aws:securityhub:us-
west-1::product/aws/securityhub" \
    --note Text="Known issue that is not a risk.",UpdatedBy="user1" \
    --severity Label="LOW" \
    --workflow Status="RESOLVED"
```

Output:

```
{
    "ProcessedFindings": [
        {
            "Id": "arn:aws:securityhub:us-west-1:123456789012:subscription/pci-dss/
v/3.2.1/PCI.Lambda.2/finding/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
            "ProductArn": "arn:aws:securityhub:us-west-1::product/aws/securityhub"
        },
        {
            "Id": "arn:aws:securityhub:us-west-1:123456789012:subscription/pci-dss/
v/3.2.1/PCI.Lambda.2/finding/a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
            "ProductArn": "arn:aws:securityhub:us-west-1::product/aws/securityhub"
        }
    ],
    "UnprocessedFindings": []
}
```

Per ulteriori informazioni, consulta Utilizzare BatchUpdateFindings per aggiornare un risultato nella Guida per l'utente AWS di Security Hub.

- Per i dettagli sull'API, vedere BatchUpdateFindingsin AWS CLI Command Reference.

`batch-update-standards-control-associations`

Il seguente esempio di codice mostra come utilizzare`batch-update-standards-control-associations`.

AWS CLI

Come aggiornare lo stato di abilitazione di un controllo negli standard abilitati

L'`batch-update-standards-control-associations`esempio seguente disabilita CloudTrail .1 negli standard specificati.

```
aws securityhub batch-update-standards-control-associations \
    --standards-control-association-updates '[{"SecurityControlId": "CloudTrail.1",
 "StandardsArn": "arn:aws:securityhub:::ruleset/cis-aws-foundations-benchmark/
v/1.2.0", "AssociationStatus": "DISABLED", "UpdatedReason": "Not applicable
 to environment"}, {"SecurityControlId": "CloudTrail.1", "StandardsArn":
 "arn:aws:securityhub:::standards/cis-aws-foundations-benchmark/v/1.4.0",
 "AssociationStatus": "DISABLED", "UpdatedReason": "Not applicable to
 environment"}]'
```

Questo comando non produce alcun output se ha esito positivo.

Per ulteriori informazioni, consulta Abilitazione e disabilitazione dei controlli con standard specifici e Abilitazione e disabilitazione dei controlli con tutti gli standard nella Guida per l'utente di Centrale di sicurezza AWS .

- Per i dettagli sull'API, vedere BatchUpdateStandardsControlAssociationsin AWS CLI Command Reference.

`create-action-target`

Il seguente esempio di codice mostra come utilizzare`create-action-target`.

AWS CLI

Come creare un'azione personalizzata

L'esempio `create-action-target` seguente crea un vocabolario personalizzato. Fornisce il nome, la descrizione e l'identificatore dell'operazione.

```
aws securityhub create-action-target \
    --name "Send to remediation" \
    --description "Action to send the finding for remediation tracking" \
    --id "Remediation"
```

Output:

```
{
```

```
    "ActionTargetArn": "arn:aws:securityhub:us-west-1:123456789012:action/custom/
Remediation"
}
```

Per ulteriori informazioni, vedere Creazione di un'azione personalizzata e associazione a una regola CloudWatch Events nella Guida per l'utente AWS di Security Hub.

- Per i dettagli sull'API, consulta CreateActionTarget AWS CLICommand Reference.

`create-automation-rule`

Il seguente esempio di codice mostra come utilizzare`create-automation-rule`.

AWS CLI

Come creare una regola di automazione

L'`create-automation-rule`esempio seguente crea una regola di automazione nell' AWS account corrente e AWS nella regione. Centrale di sicurezza filtra gli esiti in base ai criteri specificati e applica le operazioni agli esiti corrispondenti. Solo l'account amministratore di Centrale di sicurezza può eseguire questo comando.

```
aws securityhub create-automation-rule \
    --actions '[{ \
        "Type": "FINDING_FIELDS_UPDATE", \
        "FindingFieldsUpdate": { \
            "Severity": { \
                "Label": "HIGH" \
            }, \
            "Note": { \
                "Text": "Known issue that is a risk. Updated by automation rules", \
                "UpdatedBy": "sechub-automation" \
            } \
        } \
    }]' \
    --criteria '{ \
        "SeverityLabel": [{ \
            "Value": "INFORMATIONAL", \
            "Comparison": "EQUALS" \
        }] \
    }' \
    --description "A sample rule" \
```

```
    --no-is-terminal \
    --rule-name "sample rule" \
    --rule-order 1 \
    --rule-status "ENABLED"
```

Output:

```
{
    "RuleArn": "arn:aws:securityhub:us-east-1:123456789012:automation-rule/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
}
```

Per ulteriori informazioni, consulta Creazione di regole di automazione nella Guida per l'utente di Centrale di sicurezza AWS .

- Per i dettagli sull'API, consulta CreateAutomationRule AWS CLICommand Reference.

`create-configuration-policy`

Il seguente esempio di codice mostra come utilizzare`create-configuration-policy`.

AWS CLI

Come creare una policy di configurazione

L'esempio `create-configuration-policy` seguente crea una policy di configurazione con le impostazioni specificate.

```
aws securityhub create-configuration-policy \
    --name "SampleConfigurationPolicy" \
    --description "SampleDescription" \
    --configuration-policy '{"SecurityHub": {"ServiceEnabled":
 true, "EnabledStandardIdentifiers": ["arn:aws:securityhub:eu-
central-1::standards/aws-foundational-security-best-practices/
v/1.0.0","arn:aws:securityhub:::ruleset/cis-aws-foundations-benchmark/
v/1.2.0"],"SecurityControlsConfiguration":{"DisabledSecurityControlIdentifiers":
 ["CloudTrail.2"], "SecurityControlCustomParameters": [{"SecurityControlId":
 "ACM.1", "Parameters": {"daysToExpiration": {"ValueType": "CUSTOM", "Value":
 {"Integer": 15}}}}]}}}' \
    --tags '{"Environment": "Prod"}'
```

Output:

```
{
    "Arn": "arn:aws:securityhub:eu-central-1:123456789012:configuration-policy/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "Id": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "Name": "SampleConfigurationPolicy",
    "Description": "SampleDescription",
    "UpdatedAt": "2023-11-28T20:28:04.494000+00:00",
    "CreatedAt": "2023-11-28T20:28:04.494000+00:00",
    "ConfigurationPolicy": {
        "SecurityHub": {
            "ServiceEnabled": true,
            "EnabledStandardIdentifiers": [
                "arn:aws:securityhub:eu-central-1::standards/aws-foundational-
security-best-practices/v/1.0.0",
                "arn:aws:securityhub:::ruleset/cis-aws-foundations-benchmark/
v/1.2.0"
            ],
            "SecurityControlsConfiguration": {
                "DisabledSecurityControlIdentifiers": [
                    "CloudTrail.2"
                ],
                "SecurityControlCustomParameters": [
                    {
                        "SecurityControlId": "ACM.1",
                        "Parameters": {
                            "daysToExpiration": {
                                "ValueType": "CUSTOM",
                                "Value": {
                                    "Integer": 15
                                }
                            }
                        }
                    }
                ]
            }
        }
    }
}
```

Per ulteriori informazioni, consulta Creazione e associazione di policy di configurazione di Centrale di sicurezza nella Guida per l'utente di Centrale di sicurezza AWS .

- Per i dettagli sull'API, consulta CreateConfigurationPolicy AWS CLICommand Reference.

`create-finding-aggregator`

Il seguente esempio di codice mostra come utilizzare`create-finding-aggregator`.

AWS CLI

Come abilitare l'aggregazione degli esiti

L'esempio `create-finding-aggregator` seguente configura l'aggregazione degli esiti. Il comando viene eseguito dalla Regione Stati Uniti orientali (Virginia) e designa la Regione Stati Uniti orientali (Virginia) come Regione di aggregazione. Indica di collegare solo Regioni specifiche e di non collegare automaticamente nuove Regioni. Seleziona Stati Uniti occidentali (California settentrionale) e Stati Uniti occidentali (Oregon) come Regioni collegate.

```
aws securityhub create-finding-aggregator \
    --region us-east-1 \
    --region-linking-mode SPECIFIED_REGIONS \
    --regions us-west-1,us-west-2
```

Output:

```
{
    "FindingAggregatorArn": "arn:aws:securityhub:us-east-1:222222222222:finding-
aggregator/123e4567-e89b-12d3-a456-426652340000",
    "FindingAggregationRegion": "us-east-1",
    "RegionLinkingMode": "SPECIFIED_REGIONS",
    "Regions": "us-west-1,us-west-2"
}
```

Per ulteriori informazioni, consulta Abilitazione dell'aggregazione degli esiti nella Guida per l'utente di Centrale di sicurezza AWS .

- Per i dettagli sull'API, consulta CreateFindingAggregator AWS CLICommand Reference.

`create-insight`

Il seguente esempio di codice mostra come utilizzare`create-insight`.

AWS CLI

Come creare un approfondimento personalizzato

L'`create-insight`esempio seguente crea un'analisi personalizzata denominata Critical role finding che restituisce risultati critici correlati ai AWS ruoli.

```
aws securityhub create-insight \
    --filters '{"ResourceType": [{ "Comparison": "EQUALS", "Value": "AwsIamRole"}],
 "SeverityLabel": [{"Comparison": "EQUALS", "Value": "CRITICAL"}]}' \
    --group-by-attribute "ResourceId" \
    --name "Critical role findings"
```

Output:

```
{
    "InsightArn": "arn:aws:securityhub:us-west-1:123456789012:insight/123456789012/
custom/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
}
```

Per ulteriori informazioni, consulta Gestione di informazioni dettagliate personalizzate nella Guida per l'utente di Centrale di sicurezza AWS .

- Per i dettagli sull'API, consulta CreateInsight AWS CLICommand Reference.

`create-members`

Il seguente esempio di codice mostra come utilizzare`create-members`.

AWS CLI

Come aggiungere account come account membri

L'esempio `create-members` seguente aggiunge due account come account membri all'account amministratore richiedente.

```
aws securityhub create-members \
    --account-details '[{"AccountId": "123456789111"}, {"AccountId":
 "123456789222"}]'
```

Output:

```
{
    "UnprocessedAccounts": []
```

```
}
```

Per ulteriori informazioni, consulta Gestione degli account degli amministratori e dei membri nella Guida per l'utente di Centrale di sicurezza AWS .

- Per i dettagli sull'API, consulta CreateMembers AWS CLICommand Reference.

`decline-invitations`

Il seguente esempio di codice mostra come utilizzare`decline-invitations`.

AWS CLI

Come rifiutare un invito a diventare un account membro

L'esempio `decline-invitations` seguente rifiuta un invito a diventare un account membro dell'account amministratore specificato. L'account membro è l'account richiedente.

```
aws securityhub decline-invitations \
    --account-ids "123456789012"
```

Output:

```
{
    "UnprocessedAccounts": []
}
```

Per ulteriori informazioni, consulta Gestione degli account degli amministratori e dei membri nella Guida per l'utente di Centrale di sicurezza AWS .

- Per i dettagli sull'API, consulta DeclineInvitations AWS CLICommand Reference.

`delete-action-target`

Il seguente esempio di codice mostra come utilizzare`delete-action-target`.

AWS CLI

Come eliminare un'azione personalizzata

L'esempio `delete-action-target` seguente elimina l'operazione identificata dall'ARN specificato.

```
aws securityhub delete-action-target \
    --action-target-arn "arn:aws:securityhub:us-west-1:123456789012:action/custom/
Remediation"
```

Output:

```
{
    "ActionTargetArn": "arn:aws:securityhub:us-west-1:123456789012:action/custom/
Remediation"
}
```

Per ulteriori informazioni, vedere Creazione di un'azione personalizzata e associazione a una regola CloudWatch Events nella Guida per l'utente AWS di Security Hub.

- Per i dettagli sull'API, consulta DeleteActionTarget AWS CLICommand Reference.

`delete-configuration-policy`

Il seguente esempio di codice mostra come utilizzare`delete-configuration-policy`.

AWS CLI

Come eliminare una policy di configurazione

L'esempio `delete-configuration-policy` seguente elimina la policy di configurazione specificata.

```
aws securityhub delete-configuration-policy \
    --identifier "arn:aws:securityhub:eu-central-1:123456789012:configuration-
policy/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Eliminazione e dissociazione di policy di configurazione di Centrale di sicurezza nella Guida per l'utente di Centrale di sicurezza AWS .

- Per i dettagli sull'API, consulta DeleteConfigurationPolicy AWS CLICommand Reference.

`delete-finding-aggregator`

Il seguente esempio di codice mostra come utilizzare`delete-finding-aggregator`.

AWS CLI

Come arrestare l'aggregazione di esiti

L'esempio `delete-finding-aggregator` seguente arresta l'aggregazione degli esiti. Il comando viene eseguito dalla Regione Stati Uniti orientali (Virginia), ovvero la Regione di aggregazione.

```
aws securityhub delete-finding-aggregator \
    --region us-east-1 \
    --finding-aggregator-arn arn:aws:securityhub:us-east-1:222222222222:finding-aggregator/123e4567-e89b-12d3-a456-426652340000
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Arresto dell'aggregazione degli esiti nella Guida per l'utente di Centrale di sicurezza AWS .

- Per i dettagli sull'API, consulta DeleteFindingAggregator AWS CLICommand Reference.

`delete-insight`

Il seguente esempio di codice mostra come utilizzare`delete-insight`.

AWS CLI

Come eliminare un approfondimento personalizzato

L'esempio `delete-insight` seguente elimina l'approfondimento personalizzato con l'ARN specificato.

```
aws securityhub delete-insight \
    --insight-arn "arn:aws:securityhub:us-west-1:123456789012:insight/123456789012/custom/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
```

Output:

```
{
   "InsightArn": "arn:aws:securityhub:eu-central-1:123456789012:insight/123456789012/custom/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
```

```
}
```

Per ulteriori informazioni, consulta Gestione di informazioni dettagliate personalizzate nella Guida per l'utente di Centrale di sicurezza AWS .

- Per i dettagli sull'API, consulta DeleteInsight AWS CLICommand Reference.

`delete-invitations`

Il seguente esempio di codice mostra come utilizzare`delete-invitations`.

AWS CLI

Come eliminare un invito a diventare un account membro

L'esempio `delete-invitations` seguente elimina un invito a diventare un account membro per l'account amministratore specificato. L'account membro è l'account richiedente.

```
aws securityhub delete-invitations \
    --account-ids "123456789012"
```

Output:

```
{
    "UnprocessedAccounts": []
}
```

Per ulteriori informazioni, consulta Gestione degli account degli amministratori e dei membri nella Guida per l'utente di Centrale di sicurezza AWS .

- Per i dettagli sull'API, consulta DeleteInvitations AWS CLICommand Reference.

`delete-members`

Il seguente esempio di codice mostra come utilizzare`delete-members`.

AWS CLI

Come eliminare account membri

L'esempio `delete-members` seguente elimina gli account membri specificati dall'account amministratore richiedente.

```
aws securityhub delete-members \
    --account-ids "123456789111" "123456789222"
```

Output:

```
{
    "UnprocessedAccounts": []
}
```

Per ulteriori informazioni, consulta Gestione degli account degli amministratori e dei membri nella Guida per l'utente di Centrale di sicurezza AWS .

- Per i dettagli sull'API, consulta DeleteMembers AWS CLICommand Reference.

`describe-action-targets`

Il seguente esempio di codice mostra come utilizzare`describe-action-targets`.

AWS CLI

Come recuperare dettagli sulle operazioni personalizzate

L'esempio `describe-action-targets` seguente recupera informazioni sull'operazione personalizzata identificata dall'ARN specificato.

```
aws securityhub describe-action-targets \
    --action-target-arns "arn:aws:securityhub:us-west-1:123456789012:action/custom/
Remediation"
```

Output:

```
{
    "ActionTargets": [
        {
            "ActionTargetArn": "arn:aws:securityhub:us-west-1:123456789012:action/
custom/Remediation",
            "Description": "Action to send the finding for remediation tracking",
            "Name": "Send to remediation"
        }
    ]
}
```

Per ulteriori informazioni, vedere Creazione di un'azione personalizzata e associazione a una regola CloudWatch Events nella Guida per l'utente AWS di Security Hub.

- Per i dettagli sull'API, consulta DescribeActionTargets AWS CLICommand Reference.

`describe-hub`

Il seguente esempio di codice mostra come utilizzare`describe-hub`.

AWS CLI

Come ottenere informazioni su una risorsa hub

L'`describe-hub`esempio seguente restituisce la data di sottoscrizione e altre impostazioni di configurazione per la risorsa hub specificata. La risorsa hub viene identificata mediante il rispettivo ARN.

```
aws securityhub describe-hub \
    --hub-arn "arn:aws:securityhub:us-west-1:123456789012:hub/default"
```

Output:

```
{
    "HubArn": "arn:aws:securityhub:us-west-1:123456789012:hub/default",
    "SubscribedAt": "2019-11-19T23:15:10.046Z",
    "AutoEnableControls": true,
    "ControlFindingGenerator": "SECURITY_CONTROL"
}
```

Per ulteriori informazioni, vedere AWS::SecurityHub: :Hub nella Guida per l'AWS CloudFormation utente.

- Per i dettagli sull'API, consulta DescribeHub AWS CLICommand Reference.

`describe-organization-configuration`

Il seguente esempio di codice mostra come utilizzare`describe-organization-configuration`.

AWS CLI

Come visualizzare la configurazione di Centrale di sicurezza per un'organizzazione

L'esempio `describe-organization-configuration` seguente restituisce informazioni sul modo in cui un'organizzazione è configurata in Centrale di sicurezza. In questo esempio, l'organizzazione utilizza una configurazione centrale. Solo l'account amministratore di Centrale di sicurezza può eseguire questo comando.

```
aws securityhub describe-organization-configuration
```

Output:

```
{
    "AutoEnable": false,
    "MemberAccountLimitReached": false,
    "AutoEnableStandards": "NONE",
    "OrganizationConfiguration": {
        "ConfigurationType": "LOCAL",
        "Status": "ENABLED",
        "StatusMessage": "Central configuration has been enabled successfully"
    }
}
```

Per ulteriori informazioni, consulta Managing accounts with AWS Organizations nella AWS Security Hub User Guide.

- Per i dettagli sull'API, vedere DescribeOrganizationConfigurationin AWS CLI Command Reference.

`describe-products`

Il seguente esempio di codice mostra come utilizzare`describe-products`.

AWS CLI

Come restituire informazioni sulle integrazioni di prodotti disponibili

L'esempio `describe-products` seguente restituisce singolarmente le integrazioni di prodotti disponibili.

```
aws securityhub describe-products \
    --max-results 1
```

Output:

```
{
    "NextToken": "U2FsdGVkX18vvPlOqb7RDrWRWVFBJI46MOIAb+nZmRJmR15NoRi2gm13sdQEn3O/
pq/78dGs+bKpgA+7HMPHO0qX33/zoRI+uIG/F9yLNhcOrOWzFUdy36JcXLQji3Rpnn/
cD1SVkGA98qI3zPOSDg==",
    "Products": [
        {
            "ProductArn": "arn:aws:securityhub:us-west-1:123456789333:product/
crowdstrike/crowdstrike-falcon",
            "ProductName": "CrowdStrike Falcon",
            "CompanyName": "CrowdStrike",
            "Description": "CrowdStrike Falcon's single lightweight sensor unifies
 next-gen antivirus, endpoint detection and response, and 24/7 managed hunting, via
 the cloud.",
            "Categories": [
                "Endpoint Detection and Response (EDR)",
                "AV Scanning and Sandboxing",
                "Threat Intelligence Feeds and Reports",
                "Endpoint Forensics",
                "Network Forensics"
            ],
            "IntegrationTypes": [
                "SEND_FINDINGS_TO_SECURITY_HUB"
            ],
            "MarketplaceUrl": "https://aws.amazon.com/marketplace/seller-profile?
id=a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
            "ActivationUrl": "https://falcon.crowdstrike.com/support/documentation",
            "ProductSubscriptionResourcePolicy": "{\"Version\":
\"2012-10-17\",\"Statement\":[{\"Effect\":\"Allow\",\"Principal\":{\"AWS\":
\"123456789333\"},\"Action\":[\"securityhub:BatchImportFindings\"],\"Resource\":
\"arn:aws:securityhub:us-west-1:123456789012:product-subscription/crowdstrike/
crowdstrike-falcon\",\"Condition\":{\"StringEquals\":{\"securityhub:TargetAccount
\":\"123456789012\"}}},{\"Effect\":\"Allow\",\"Principal\":{\"AWS\":
\"123456789012\"},\"Action\":[\"securityhub:BatchImportFindings\"],\"Resource
\":\"arn:aws:securityhub:us-west-1:123456789333:product/crowdstrike/crowdstrike-
falcon\",\"Condition\":{\"StringEquals\":{\"securityhub:TargetAccount\":
\"123456789012\"}}}]}"
        }
   ]
}
```


Per ulteriori informazioni, consulta Gestione delle integrazioni di prodotti nella Guida per l'utente di Centrale di sicurezza AWS .

- Per i dettagli sull'API, consulta DescribeProducts AWS CLICommand Reference.

`describe-standards-controls`

Il seguente esempio di codice mostra come utilizzare`describe-standards-controls`.

AWS CLI

Come richiedere l'elenco di controlli in uno standard abilitato

L'esempio `describe-standards-controls` seguente richiede l'elenco dei controlli nella sottoscrizione dell'account richiedente allo standard PCI DSS. La richiesta restituisce due controlli alla volta.

```
aws securityhub describe-standards-controls \
    --standards-subscription-arn "arn:aws:securityhub:us-
west-1:123456789012:subscription/pci-dss/v/3.2.1" \
    --max-results 2
```

Output:

```
{
    "Controls": [
        {
            "StandardsControlArn": "arn:aws:securityhub:us-
west-1:123456789012:control/pci-dss/v/3.2.1/PCI.AutoScaling.1",
            "ControlStatus": "ENABLED",
            "ControlStatusUpdatedAt": "2020-05-15T18:49:04.473000+00:00",
            "ControlId": "PCI.AutoScaling.1",
            "Title": "Auto scaling groups associated with a load balancer should use
 health checks",
            "Description": "This AWS control checks whether your Auto Scaling groups
 that are associated with a load balancer are using Elastic Load Balancing health
 checks.",
            "RemediationUrl": "https://docs.aws.amazon.com/console/securityhub/
PCI.AutoScaling.1/remediation",
            "SeverityRating": "LOW",
            "RelatedRequirements": [
                "PCI DSS 2.2"
            ]
        },
        {
```

```
            "StandardsControlArn": "arn:aws:securityhub:us-
west-1:123456789012:control/pci-dss/v/3.2.1/PCI.CW.1",
            "ControlStatus": "ENABLED",
            "ControlStatusUpdatedAt": "2020-05-15T18:49:04.498000+00:00",
            "ControlId": "PCI.CW.1",
            "Title": "A log metric filter and alarm should exist for usage of the
 \"root\" user",
            "Description": "This control checks for the CloudWatch metric
 filters using the following pattern { $.userIdentity.type = \"Root\" &&
 $.userIdentity.invokedBy NOT EXISTS && $.eventType != \"AwsServiceEvent\" }
 It checks that the log group name is configured for use with active multi-
region CloudTrail, that there is at least one Event Selector for a Trail with
 IncludeManagementEvents set to true and ReadWriteType set to All, and that there is
 at least one active subscriber to an SNS topic associated with the alarm.",
            "RemediationUrl": "https://docs.aws.amazon.com/console/securityhub/
PCI.CW.1/remediation",
            "SeverityRating": "MEDIUM",
            "RelatedRequirements": [
                "PCI DSS 7.2.1"
            ]
        }
    ],
    "NextToken": "U2FsdGVkX1+eNkPoZHVl11ip5HUYQPWSWZGmftcmJiHL8JoKEsCDuaKayiPDyLK
+LiTkShveoOdvfxXCkOBaGhohIXhsIedN+LSjQV/
l7kfCfJcq4PziNC1N9xe9aq2pjlLVZnznTfSImrodT5bRNHe4fELCQq/z+5ka
+5Lzmc11axcwTd5lKgQyQqmUVoeriHZhyIiBgWKf7oNYdBVG8OEortVWvSkoUTt
+B2ThcnC7l43kI0UNxlkZ6sc64AsW"
}
```

Per ulteriori informazioni, consulta Visualizzazione dei dettagli per i controlli nella Guida per l'utente di Centrale di sicurezza AWS .

- Per i dettagli sull'API, consulta DescribeStandardsControls AWS CLICommand Reference.

`describe-standards`

Il seguente esempio di codice mostra come utilizzare`describe-standards`.

AWS CLI

Come restituire un elenco di standard disponibili

L'esempio `describe-standards` seguente restituisce l'elenco degli standard disponibili.

```
aws securityhub describe-standards
```

Output:

```
{
    "Standards": [
        {
            "StandardsArn": "arn:aws:securityhub:us-west-1::standards/aws-
foundational-security-best-practices/v/1.0.0",
            "Name": "AWS Foundational Security Best Practices v1.0.0",
            "Description": "The AWS Foundational Security Best Practices standard
 is a set of automated security checks that detect when AWS accounts and deployed
 resources do not align to security best practices. The standard is defined by AWS
 security experts. This curated set of controls helps improve your security posture
 in AWS, and cover AWS's most popular and foundational services.",
            "EnabledByDefault": true
        },
        {
            "StandardsArn": "arn:aws:securityhub:::ruleset/cis-aws-foundations-
benchmark/v/1.2.0",
            "Name": "CIS AWS Foundations Benchmark v1.2.0",
            "Description": "The Center for Internet Security (CIS) AWS Foundations
 Benchmark v1.2.0 is a set of security configuration best practices for AWS. This
 Security Hub standard automatically checks for your compliance readiness against a
 subset of CIS requirements.",
            "EnabledByDefault": true
        },
        {
            "StandardsArn": "arn:aws:securityhub:us-west-1::standards/pci-dss/
v/3.2.1",
            "Name": "PCI DSS v3.2.1",
            "Description": "The Payment Card Industry Data Security Standard (PCI
 DSS) v3.2.1 is an information security standard for entities that store, process,
 and/or transmit cardholder data. This Security Hub standard automatically checks
 for your compliance readiness against a subset of PCI DSS requirements.",
            "EnabledByDefault": false
        }
    ]
}
```

Per ulteriori informazioni, consulta gli standard di sicurezza in AWS Security Hub nella Guida per l'utente AWS di Security Hub.

- Per i dettagli sull'API, vedere DescribeStandardsin AWS CLI Command Reference.

`disable-import-findings-for-product`

Il seguente esempio di codice mostra come utilizzare`disable-import-findings-for-product`.

AWS CLI

Come interrompere la ricezione di esiti da un'integrazione di prodotti

L'esempio `disable-import-findings-for-product` seguente disabilita il flusso di esiti per la sottoscrizione specificata a un'integrazione di prodotti.

```
aws securityhub disable-import-findings-for-product \
    --product-subscription-arn "arn:aws:securityhub:us-west-1:123456789012:product-subscription/crowdstrike/crowdstrike-falcon"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Gestione delle integrazioni di prodotti nella Guida per l'utente di Centrale di sicurezza AWS .

- Per i dettagli sull'API, consulta DisableImportFindingsForProduct AWS CLICommand Reference.

`disable-organization-admin-account`

Il seguente esempio di codice mostra come utilizzare`disable-organization-admin-account`.

AWS CLI

Come rimuovere un account amministratore di Centrale di sicurezza

L'`disable-organization-admin-account`esempio seguente revoca l'assegnazione dell'account specificato come account amministratore di Security Hub per Organizations. AWS

```
aws securityhub disable-organization-admin-account \
    --admin-account-id 777788889999
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Designazione di un account amministratore di Centrale di sicurezza nella Guida per l'utente di Centrale di sicurezza AWS .

- Per i dettagli sull'API, vedere DisableOrganizationAdminAccountin AWS CLI Command Reference.

`disable-security-hub`

Il seguente esempio di codice mostra come utilizzare`disable-security-hub`.

AWS CLI

Per disabilitare AWS Security Hub

L'`disable-security-hub`esempio seguente disattiva AWS Security Hub per l'account richiedente.

```
aws securityhub disable-security-hub
```

Questo comando non produce alcun output.

Per ulteriori informazioni, vedere Disabling AWS Security Hub nella Security Hub User Guide.AWS

- Per i dettagli sull'API, vedere DisableSecurityHubin AWS CLI Command Reference.

`disassociate-from-administrator-account`

Il seguente esempio di codice mostra come utilizzare`disassociate-from-administrator-account`.

AWS CLI

Come annullare l'associazione da un account amministratore

L'esempio `disassociate-from-administrator-account` seguente annulla l'associazione tra l'account richiedente e il relativo account amministratore corrente.

```
aws securityhub disassociate-from-administrator-account
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Gestione degli account degli amministratori e dei membri nella Guida per l'utente di Centrale di sicurezza AWS .

- Per i dettagli sull'API, consulta DisassociateFromAdministratorAccount AWS CLICommand Reference.

`disassociate-from-master-account`

Il seguente esempio di codice mostra come utilizzare`disassociate-from-master-account`.

AWS CLI

Come annullare l'associazione da un account amministratore

L'esempio `disassociate-from-master-account` seguente annulla l'associazione tra l'account richiedente e il relativo account amministratore corrente.

```
aws securityhub disassociate-from-master-account
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Gestione degli account degli amministratori e dei membri nella Guida per l'utente di Centrale di sicurezza AWS .

- Per i dettagli sull'API, consulta DisassociateFromMasterAccount AWS CLICommand Reference.

`disassociate-members`

Il seguente esempio di codice mostra come utilizzare`disassociate-members`.

AWS CLI

Come annullare l'associazione di account membri

L'esempio `disassociate-members` seguente annulla l'associazione tra gli account membri specificati e l'account amministratore richiedente.

```
aws securityhub disassociate-members  \
    --account-ids "123456789111" "123456789222"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Gestione degli account degli amministratori e dei membri nella Guida per l'utente di Centrale di sicurezza AWS .

- Per i dettagli sull'API, consulta DisassociateMembers AWS CLICommand Reference.

`enable-import-findings-for-product`

Il seguente esempio di codice mostra come utilizzare`enable-import-findings-for-product`.

AWS CLI

Come iniziare a ricevere gli esiti da un'integrazione di prodotti

L'esempio `enable-import-findings-for-product` seguente abilita il flusso di esiti dall'integrazione di prodotti.

```
aws securityhub enable-import-findings-for-product \
    --product-arn "arn:aws:securityhub:us-east-1:123456789333:product/crowdstrike/
crowdstrike-falcon"
```

Output:

```
{
    "ProductSubscriptionArn": "arn:aws:securityhub:us-east-1:123456789012:product-
subscription/crowdstrike/crowdstrike-falcon"
}
```

Per ulteriori informazioni, consulta Gestione delle integrazioni di prodotti nella Guida per l'utente di Centrale di sicurezza AWS .

- Per i dettagli sull'API, consulta EnableImportFindingsForProduct AWS CLICommand Reference.

`enable-organization-admin-account`

Il seguente esempio di codice mostra come utilizzare`enable-organization-admin-account`.

AWS CLI

Come designare un account dell'organizzazione come account amministratore di Centrale di sicurezza

L'esempio `enable-organization-admin-account` seguente designa l'account specificato come account amministratore di Centrale di sicurezza.

```
aws securityhub enable-organization-admin-account \
    --admin-account-id 777788889999
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Designazione di un account amministratore di Centrale di sicurezza nella Guida per l'utente di Centrale di sicurezza AWS .

- Per i dettagli sull'API, consulta EnableOrganizationAdminAccount AWS CLICommand Reference.

`enable-security-hub`

Il seguente esempio di codice mostra come utilizzare`enable-security-hub`.

AWS CLI

Per abilitare AWS Security Hub

L'`enable-security-hub`esempio seguente abilita AWS Security Hub per l'account richiedente. Il comando configura Centrale di sicurezza per abilitare gli standard predefiniti. Per la risorsa hub, assegna il valore `Security` al tag `Department`.

```
aws securityhub enable-security-hub \
    --enable-default-standards \
    --tags '{"Department": "Security"}'
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Abilitazione di Centrale di sicurezza nella Guida per l'utente di Centrale di sicurezza AWS .

- Per i dettagli sull'API, vedere EnableSecurityHubin AWS CLI Command Reference.

`get-administrator-account`

Il seguente esempio di codice mostra come utilizzare`get-administrator-account`.

AWS CLI

Come recuperare informazioni su un account amministratore

L'esempio `get-administrator-account` seguente recupera informazioni sull'account amministratore per l'account richiedente.

```
aws securityhub get-administrator-account
```

Output:

```
{
   "Master": {
      "AccountId": "123456789012",
      "InvitationId": "7ab938c5d52d7904ad09f9e7c20cc4eb",
      "InvitedAt": 2020-06-01T20:21:18.042000+00:00,
      "MemberStatus": "ASSOCIATED"
   }
}
```

Per ulteriori informazioni, consulta Gestione degli account degli amministratori e dei membri nella Guida per l'utente di Centrale di sicurezza AWS .

- Per i dettagli sull'API, consulta GetAdministratorAccount AWS CLICommand Reference.

`get-configuration-policy-association`

Il seguente esempio di codice mostra come utilizzare`get-configuration-policy-association`.

AWS CLI

Come ottenere i dettagli dell'associazione di configurazione per una destinazione

L'esempio `get-configuration-policy-association` seguente recupera i dettagli dell'associazione per la destinazione specificata. È possibile fornire un ID account, l'ID delle unità organizzative o l'ID root per la destinazione.

```
aws securityhub get-configuration-policy-association \
    --target '{"OrganizationalUnitId": "ou-6hi7-8j91kl2m"}'
```

Output:

```
{
    "ConfigurationPolicyId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE33333",
    "TargetId": "ou-6hi7-8j91kl2m",
    "TargetType": "ORGANIZATIONAL_UNIT",
    "AssociationType": "APPLIED",
    "UpdatedAt": "2023-09-26T21:13:01.816000+00:00",
    "AssociationStatus": "SUCCESS",
    "AssociationStatusMessage": "Association applied successfully on this target."
}
```

Per ulteriori informazioni, consulta Visualizzazione delle policy di configurazione di Centrale di sicurezza nella Guida per l'utente di Centrale di sicurezza AWS .

- Per i dettagli sull'API, consulta GetConfigurationPolicyAssociation AWS CLICommand Reference.

`get-configuration-policy`

Il seguente esempio di codice mostra come utilizzare`get-configuration-policy`.

AWS CLI

Come visualizzare i dettagli della policy di configurazione

L'esempio `get-configuration-policy` seguente recupera i dettagli sulla policy di configurazione specificata.

```
aws securityhub get-configuration-policy \
   --identifier "arn:aws:securityhub:eu-central-1:123456789012:configuration-policy/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
```

Output:

```
{
    "Arn": "arn:aws:securityhub:eu-central-1:123456789012:configuration-policy/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "Id": "ce5ed1e7-9639-4e2f-9313-fa87fcef944b",
    "Name": "SampleConfigurationPolicy",
    "Description": "SampleDescription",
    "UpdatedAt": "2023-11-28T20:28:04.494000+00:00",
    "CreatedAt": "2023-11-28T20:28:04.494000+00:00",
```

```
    "ConfigurationPolicy": {
        "SecurityHub": {
            "ServiceEnabled": true,
            "EnabledStandardIdentifiers": [
                "arn:aws:securityhub:eu-central-1::standards/aws-foundational-
security-best-practices/v/1.0.0",
                "arn:aws:securityhub:::ruleset/cis-aws-foundations-benchmark/
v/1.2.0"
            ],
            "SecurityControlsConfiguration": {
                "DisabledSecurityControlIdentifiers": [
                    "CloudTrail.2"
                ],
                "SecurityControlCustomParameters": [
                    {
                        "SecurityControlId": "ACM.1",
                        "Parameters": {
                            "daysToExpiration": {
                                "ValueType": "CUSTOM",
                                "Value": {
                                    "Integer": 15
                                }
                            }
                        }
                    }
                ]
            }
        }
    }
}
```

Per ulteriori informazioni, consulta Visualizzazione delle policy di configurazione di Centrale di sicurezza nella Guida per l'utente di Centrale di sicurezza AWS .

- Per i dettagli sull'API, consulta GetConfigurationPolicy AWS CLICommand Reference.

`get-enabled-standards`

Il seguente esempio di codice mostra come utilizzare`get-enabled-standards`.

AWS CLI

Come recuperare informazioni su uno standard abilitato

L'esempio `get-enabled-standards` seguente recupera le informazioni sullo standard PCI DSS.

```
aws securityhub get-enabled-standards \
    --standards-subscription-arn "arn:aws:securityhub:us-
west-1:123456789012:subscription/pci-dss/v/3.2.1"
```

Output:

```
{
    "StandardsSubscriptions": [
        {
            "StandardsArn": "arn:aws:securityhub:us-west-1::standards/pci-dss/
v/3.2.1",
            "StandardsInput": { },
            "StandardsStatus": "READY",
            "StandardsSubscriptionArn": "arn:aws:securityhub:us-
west-1:123456789012:subscription/pci-dss/v/3.2.1"
        }
    ]
}
```

Per ulteriori informazioni, consulta gli standard di sicurezza in AWS Security Hub nella Guida per l'utente AWS di Security Hub.

- Per i dettagli sull'API, vedere GetEnabledStandardsin AWS CLI Command Reference.

`get-finding-aggregator`

Il seguente esempio di codice mostra come utilizzare`get-finding-aggregator`.

AWS CLI

Come recuperare la configurazione di aggregazione degli esiti corrente

L'esempio `get-finding-aggregator` seguente recupera la configurazione di aggregazione degli esiti corrente.

```
aws securityhub get-finding-aggregator \
    --finding-aggregator-arn arn:aws:securityhub:us-east-1:222222222222:finding-
aggregator/123e4567-e89b-12d3-a456-426652340000
```

Output:

```
{
    "FindingAggregatorArn": "arn:aws:securityhub:us-east-1:222222222222:finding-
aggregator/123e4567-e89b-12d3-a456-426652340000",
    "FindingAggregationRegion": "us-east-1",
    "RegionLinkingMode": "SPECIFIED_REGIONS",
    "Regions": "us-west-1,us-west-2"
}
```

Per ulteriori informazioni, consulta Visualizzazione della configurazione di aggregazione degli esiti corrente nella Guida per l'utente di Centrale di sicurezza AWS .

- Per i dettagli sull'API, consulta GetFindingAggregator AWS CLICommand Reference.

`get-finding-history`

Il seguente esempio di codice mostra come utilizzare`get-finding-history`.

AWS CLI

Come visualizzare la cronologia degli esiti

L'esempio `get-finding-history` seguente recupera gli ultimi 90 giorni della cronologia per l'esito specificato. In questo esempio, i risultati sono limitati a due record della cronologia degli esiti.

```
aws securityhub get-finding-history \
    --finding-identifier Id="arn:aws:securityhub:us-
east-1:123456789012:security-control/S3.17/finding/a1b2c3d4-5678-90ab-cdef-
EXAMPLE11111",ProductArn="arn:aws:securityhub:us-east-1::product/aws/securityhub"
```

Output:

```
{
    "Records": [
        {
            "FindingIdentifier": {
                "Id": "arn:aws:securityhub:us-east-1:123456789012:security-control/
S3.17/finding/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
                "ProductArn": "arn:aws:securityhub:us-east-1::product/aws/
securityhub"
```

```
            },
            "UpdateTime": "2023-06-02T03:15:25.685000+00:00",
            "FindingCreated": false,
            "UpdateSource": {
                "Type": "BATCH_IMPORT_FINDINGS",
                "Identity": "arn:aws:securityhub:us-east-1::product/aws/securityhub"
            },
            "Updates": [
                {
                    "UpdatedField": "Compliance.RelatedRequirements",
                    "OldValue": "[\"NIST.800-53.r5 SC-12(2)\",\"NIST.800-53.r5
 SC-12(3)\",\"NIST.800-53.r5 SC-12(6)\",\"NIST.800-53.r5 CM-3(6)\",\"NIST.800-53.r5
 SC-13\",\"NIST.800-53.r5 SC-28\",\"NIST.800-53.r5 SC-28(1)\",\"NIST.800-53.r5
 SC-7(10)\"]",
                    "NewValue": "[\"NIST.800-53.r5 SC-12(2)\",\"NIST.800-53.r5
 CM-3(6)\",\"NIST.800-53.r5 SC-13\",\"NIST.800-53.r5 SC-28\",\"NIST.800-53.r5
 SC-28(1)\",\"NIST.800-53.r5 SC-7(10)\",\"NIST.800-53.r5 CA-9(1)\",\"NIST.800-53.r5
 SI-7(6)\",\"NIST.800-53.r5 AU-9\"]"
                },
                {
                    "UpdatedField": "LastObservedAt",
                    "OldValue": "2023-06-01T09:15:38.587Z",
                    "NewValue": "2023-06-02T03:15:22.946Z"
                },
                {
                    "UpdatedField": "UpdatedAt",
                    "OldValue": "2023-06-01T09:15:31.049Z",
                    "NewValue": "2023-06-02T03:15:14.861Z"
                },
                {
                    "UpdatedField": "ProcessedAt",
                    "OldValue": "2023-06-01T09:15:41.058Z",
                    "NewValue": "2023-06-02T03:15:25.685Z"
                }
            ]
        },
        {
            "FindingIdentifier": {
                "Id": "arn:aws:securityhub:us-east-1:123456789012:security-control/
S3.17/finding/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
                "ProductArn": "arn:aws:securityhub:us-east-1::product/aws/
securityhub"
            },
            "UpdateTime": "2023-05-23T02:06:51.518000+00:00",
```

```
            "FindingCreated": "true",
            "UpdateSource": {
                "Type": "BATCH_IMPORT_FINDINGS",
                "Identity": "arn:aws:securityhub:us-east-1::product/aws/securityhub"
            },
            "Updates": []
        }
    ]
}
```

Per ulteriori informazioni, consulta Cronologia degli esiti nella Guida per l'utente di Centrale di sicurezza AWS .

- Per i dettagli sull'API, consulta GetFindingHistory AWS CLICommand Reference.

`get-findings`

Il seguente esempio di codice mostra come utilizzare`get-findings`.

AWS CLI

Esempio 1: come restituire gli esiti generati per uno standard specifico

L'esempio `get-findings` seguente restituisce gli esiti per lo standard PCI DSS.

```
aws securityhub get-findings \
    --filters '{"GeneratorId":[{"Value": "pci-dss","Comparison":"PREFIX"}]}' \
    --max-items 1
```

Output:

```
{
    "Findings": [
        {
            "SchemaVersion": "2018-10-08",
            "Id": "arn:aws:securityhub:eu-central-1:123456789012:subscription/pci-
dss/v/3.2.1/PCI.Lambda.2/finding/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
            "ProductArn": "arn:aws:securityhub:us-west-1::product/aws/securityhub",
            "GeneratorId": "pci-dss/v/3.2.1/PCI.Lambda.2",
            "AwsAccountId": "123456789012",
            "Types": [
                "Software and Configuration Checks/Industry and Regulatory
 Standards/PCI-DSS"
```

```
            ],
            "FindingProviderFields": {
                "Severity": {
                    "Original": 0,
                    "Label": "INFORMATIONAL"
                },
                "Types": [
                    "Software and Configuration Checks/Industry and Regulatory
 Standards/PCI-DSS"
                ]
            },
            "FirstObservedAt": "2020-06-02T14:02:49.159Z",
            "LastObservedAt": "2020-06-02T14:02:52.397Z",
            "CreatedAt": "2020-06-02T14:02:49.159Z",
            "UpdatedAt": "2020-06-02T14:02:52.397Z",
            "Severity": {
                "Original": 0,
                "Label": "INFORMATIONAL",
                "Normalized": 0
            },
            "Title": "PCI.Lambda.2 Lambda functions should be in a VPC",
            "Description": "This AWS control checks whether a Lambda function is in
 a VPC.",
            "Remediation": {
                "Recommendation": {
                    "Text": "For directions on how to fix this issue, please consult
 the AWS Security Hub PCI DSS documentation.",
                    "Url": "https://docs.aws.amazon.com/console/securityhub/
PCI.Lambda.2/remediation"
                }
            },
            "ProductFields": {
                "StandardsArn": "arn:aws:securityhub:::standards/pci-dss/v/3.2.1",
                "StandardsSubscriptionArn": "arn:aws:securityhub:us-
west-1:123456789012:subscription/pci-dss/v/3.2.1",
                "ControlId": "PCI.Lambda.2",
                "RecommendationUrl": "https://docs.aws.amazon.com/console/
securityhub/PCI.Lambda.2/remediation",
                "RelatedAWSResources:0/name": "securityhub-lambda-inside-
vpc-0e904a3b",
                "RelatedAWSResources:0/type": "AWS::Config::ConfigRule",
                "StandardsControlArn": "arn:aws:securityhub:us-
west-1:123456789012:control/pci-dss/v/3.2.1/PCI.Lambda.2",
                "aws/securityhub/SeverityLabel": "INFORMATIONAL",
```

```
                "aws/securityhub/ProductName": "Security Hub",
                "aws/securityhub/CompanyName": "AWS",
                "aws/securityhub/FindingId": "arn:aws:securityhub:eu-
central-1::product/aws/securityhub/arn:aws:securityhub:eu-
central-1:123456789012:subscription/pci-dss/v/3.2.1/PCI.Lambda.2/finding/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
        },
            "Resources": [
                {
                    "Type": "AwsAccount",
                    "Id": "AWS::::Account:123456789012",
                    "Partition": "aws",
                    "Region": "us-west-1"
                }
            ],
            "Compliance": {
                "Status": "PASSED",
                "RelatedRequirements": [
                    "PCI DSS 1.2.1",
                    "PCI DSS 1.3.1",
                    "PCI DSS 1.3.2",
                    "PCI DSS 1.3.4"
                ]
            },
            "WorkflowState": "NEW",
            "Workflow": {
                "Status": "NEW"
            },
            "RecordState": "ARCHIVED"
        }
    ],
    "NextToken": "eyJOZXh0VG9rZW4iOiBudWxsLCAiYm90b190cnVuY2F0ZV9hbW91bnQiOiAxfQ=="
}
```

Esempio 2: come restituire gli esiti con gravità critica con stato NOTIFIED del flusso di lavoro

L'esempio `get-findings` seguente restituisce gli esiti con un valore dell'etichetta di gravità pari a CRITICAL e lo stato del flusso di lavoro impostato su NOTIFIED. I risultati vengono visualizzati in ordine decrescente in base al valore di Confidence.

```
aws securityhub get-findings \
```

```
    --filters '{"SeverityLabel":[{"Value":
 "CRITICAL","Comparison":"EQUALS"}],"WorkflowStatus":
 [{"Value":"NOTIFIED","Comparison":"EQUALS"}]}' \
    --sort-criteria '{ "Field": "Confidence", "SortOrder": "desc"}' \
    --max-items 1
```

Output:

```
{
    "Findings": [
        {
            "SchemaVersion": "2018-10-08",
            "Id": "arn:aws:securityhub:us-west-1: 123456789012:subscription/cis-aws-
foundations-benchmark/v/1.2.0/1.13/finding/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
            "ProductArn": "arn:aws:securityhub:us-west-2::product/aws/securityhub",
            "GeneratorId": "arn:aws:securityhub:::ruleset/cis-aws-foundations-
benchmark/v/1.2.0/rule/1.13",
            "AwsAccountId": "123456789012",
            "Types": [
                "Software and Configuration Checks/Industry and Regulatory
 Standards/CIS AWS Foundations Benchmark"
            ],
            "FindingProviderFields" {
                "Severity": {
                    "Original": 90,
                    "Label": "CRITICAL"
                },
                "Types": [
                    "Software and Configuration Checks/Industry and Regulatory
 Standards/CIS AWS Foundations Benchmark"
                ]
            },
            "FirstObservedAt": "2020-05-21T20:16:34.752Z",
            "LastObservedAt": "2020-06-09T08:16:37.171Z",
            "CreatedAt": "2020-05-21T20:16:34.752Z",
            "UpdatedAt": "2020-06-09T08:16:36.430Z",
            "Severity": {
                "Original": 90,
                "Label": "CRITICAL",
                "Normalized": 90
            },
            "Title": "1.13 Ensure MFA is enabled for the \"root\" account",
```

```
            "Description": "The root account is the most privileged user in an AWS
 account. MFA adds an extra layer of protection on top of a user name and password.
 With MFA enabled, when a user signs in to an AWS website, they will be prompted for
 their user name and password as well as for an authentication code from their AWS
 MFA device.",
            "Remediation": {
                "Recommendation": {
                    "Text": "For directions on how to fix this issue, please consult
 the AWS Security Hub CIS documentation.",
                    "Url": "https://docs.aws.amazon.com/console/securityhub/
standards-cis-1.13/remediation"
                }
            },
            "ProductFields": {
                "StandardsGuideArn": "arn:aws:securityhub:::ruleset/cis-aws-
foundations-benchmark/v/1.2.0",
                "StandardsGuideSubscriptionArn": "arn:aws:securityhub:us-
west-1:123456789012:subscription/cis-aws-foundations-benchmark/v/1.2.0",
                "RuleId": "1.13",
                "RecommendationUrl": "https://docs.aws.amazon.com/console/
securityhub/standards-cis-1.13/remediation",
                "RelatedAWSResources:0/name": "securityhub-root-account-mfa-
enabled-5pftha",
                "RelatedAWSResources:0/type": "AWS::Config::ConfigRule",
                "StandardsControlArn": "arn:aws:securityhub:us-
west-1:123456789012:control/cis-aws-foundations-benchmark/v/1.2.0/1.13",
                "aws/securityhub/SeverityLabel": "CRITICAL",
                "aws/securityhub/ProductName": "Security Hub",
                "aws/securityhub/CompanyName": "AWS",
                "aws/securityhub/FindingId": "arn:aws:securityhub:us-
west-1::product/aws/securityhub/arn:aws:securityhub:us-
west-1:123456789012:subscription/cis-aws-foundations-benchmark/v/1.2.0/1.13/finding/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
            },
            "Resources": [
                {
                    "Type": "AwsAccount",
                    "Id": "AWS::::Account:123456789012",
                    "Partition": "aws",
                    "Region": "us-west-1"
                }
            ],
            "Compliance": {
                "Status": "FAILED"
```

```
            },
            "WorkflowState": "NEW",
            "Workflow": {
                "Status": "NOTIFIED"
            },
            "RecordState": "ACTIVE"
        }
    ]
}
```

Per ulteriori informazioni, consulta Filtro e raggruppamento degli esiti nella Guida per l'utente di Centrale di sicurezza AWS .

- Per i dettagli sull'API, consulta GetFindings AWS CLICommand Reference.

`get-insight-results`

Il seguente esempio di codice mostra come utilizzare`get-insight-results`.

AWS CLI

Come recuperare i risultati per un approfondimento

L'esempio `get-insight-results` seguente restituisce l'elenco dei risultati di approfondimento con l'ARN specificato.

```
aws securityhub get-insight-results \
    --insight-arn "arn:aws:securityhub:us-west-1:123456789012:insight/123456789012/
custom/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
```

Output:

```
{
    "InsightResults": {
        "GroupByAttribute": "ResourceId",
        "InsightArn": "arn:aws:securityhub:us-
west-1:123456789012:insight/123456789012/custom/a1b2c3d4-5678-90ab-cdef-
EXAMPLE11111",
        "ResultValues": [
            {
                "Count": 10,
                "GroupByAttributeValue": "AWS::::Account:123456789111"
```

```
            },
            {
                "Count": 3,
                "GroupByAttributeValue": "AWS::::Account:123456789222"
            }
        ]
    }
}
```

Per ulteriori informazioni, consulta Visualizzazione e azioni su risultati e risultati di informazione dettagliata nella Guida per l'utente di Centrale di sicurezza AWS .

- Per i dettagli sull'API, consulta GetInsightResults AWS CLICommand Reference.

`get-insights`

Il seguente esempio di codice mostra come utilizzare`get-insights`.

AWS CLI

Come recuperare i dettagli su un approfondimento

L'esempio `get-insights` seguente recupera i dettagli di configurazione per l'approfondimento con l'ARN specificato.

```
aws securityhub get-insights \
    --insight-arns "arn:aws:securityhub:us-west-1:123456789012:insight/123456789012/
custom/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
```

Output:

```
{
    "Insights": [
        {
            "Filters": {
               "ResourceType": [
                    {
                        "Comparison": "EQUALS",
                        "Value": "AwsIamRole"
                    }
                ],
                "SeverityLabel": [
                    {
```

```
                        "Comparison": "EQUALS",
                        "Value": "CRITICAL"
                    }
                ],
            },
            "GroupByAttribute": "ResourceId",
            "InsightArn": "arn:aws:securityhub:us-
west-1:123456789012:insight/123456789012/custom/a1b2c3d4-5678-90ab-cdef-
EXAMPLE11111",
            "Name": "Critical role findings"
        }
    ]
}
```

Per ulteriori informazioni, consulta Insights in AWS Security Hub nella Guida per l'utente AWS di Security Hub.

- Per i dettagli sull'API, consulta GetInsights AWS CLICommand Reference.

`get-invitations-count`

Il seguente esempio di codice mostra come utilizzare`get-invitations-count`.

AWS CLI

Come recuperare il numero di inviti non accettati

L'esempio `get-invitations-count` seguente recupera il numero di inviti rifiutati o senza risposta da parte dell'account richiedente.

```
aws securityhub get-invitations-count
```

Output:

```
{
  "InvitationsCount": 3
}
```

Per ulteriori informazioni, consulta Gestione degli account degli amministratori e dei membri nella Guida per l'utente di Centrale di sicurezza AWS .

- Per i dettagli sull'API, consulta GetInvitationsCount AWS CLICommand Reference.

`get-master-account`

Il seguente esempio di codice mostra come utilizzare`get-master-account`.

AWS CLI

Come recuperare informazioni su un account amministratore

L'esempio `get-master-account` seguente recupera informazioni sull'account amministratore per l'account richiedente.

```
aws securityhub get-master-account
```

Output:

```
{
   "Master": {
      "AccountId": "123456789012",
      "InvitationId": "7ab938c5d52d7904ad09f9e7c20cc4eb",
      "InvitedAt": 2020-06-01T20:21:18.042000+00:00,
      "MemberStatus": "ASSOCIATED"
   }
}
```

Per ulteriori informazioni, consulta Gestione degli account degli amministratori e dei membri nella Guida per l'utente di Centrale di sicurezza AWS .

- Per i dettagli sull'API, consulta GetMasterAccount AWS CLICommand Reference.

`get-members`

Il seguente esempio di codice mostra come utilizzare`get-members`.

AWS CLI

Come recuperare informazioni sugli account membri selezionati

L'esempio `get-members` seguente recupera le informazioni relative agli account membri specificati.

```
aws securityhub get-members \
    --account-ids "444455556666" "777788889999"
```

Output:

```
{
    "Members": [
        {
            "AccountId": "123456789111",
            "AdministratorId": "123456789012",
            "InvitedAt": 2020-06-01T20:15:15.289000+00:00,
            "MasterId": "123456789012",
            "MemberStatus": "ASSOCIATED",
            "UpdatedAt": 2020-06-01T20:15:15.289000+00:00
        },
        {
            "AccountId": "123456789222",
            "AdministratorId": "123456789012",
            "InvitedAt": 2020-06-01T20:15:15.289000+00:00,
            "MasterId": "123456789012",
            "MemberStatus": "ASSOCIATED",
            "UpdatedAt": 2020-06-01T20:15:15.289000+00:00
        }
    ],
    "UnprocessedAccounts": [ ]
}
```

Per ulteriori informazioni, consulta Gestione degli account degli amministratori e dei membri nella Guida per l'utente di Centrale di sicurezza AWS .

- Per i dettagli sull'API, consulta GetMembers AWS CLICommand Reference.

`get-security-control-definition`

Il seguente esempio di codice mostra come utilizzare`get-security-control-definition`.

AWS CLI

Come ottenere i dettagli della definizione del controllo di sicurezza

L'esempio `get-security-control-definition` seguente recupera i dettagli della definizione per un controllo di Centrale di sicurezza. I dettagli includono il titolo e la descrizione del controllo, la disponibilità della Regione, i parametri e altre informazioni.

```
aws securityhub get-security-control-definition \
```

```
    --security-control-id ACM.1
```

Output:

```
{
    "SecurityControlDefinition": {
        "SecurityControlId": "ACM.1",
        "Title": "Imported and ACM-issued certificates should be renewed after a
 specified time period",
        "Description": "This control checks whether an AWS Certificate Manager
 (ACM) certificate is renewed within the specified time period. It checks both
 imported certificates and certificates provided by ACM. The control fails if the
 certificate isn't renewed within the specified time period. Unless you provide a
 custom parameter value for the renewal period, Security Hub uses a default value of
 30 days.",
        "RemediationUrl": "https://docs.aws.amazon.com/console/securityhub/ACM.1/
remediation",
        "SeverityRating": "MEDIUM",
        "CurrentRegionAvailability": "AVAILABLE",
        "ParameterDefinitions": {
            "daysToExpiration": {
                "Description": "Number of days within which the ACM certificate must
 be renewed",
                "ConfigurationOptions": {
                    "Integer": {
                        "DefaultValue": 30,
                        "Min": 14,
                        "Max": 365
                    }
                }
            }
        }
    }
}
```

Per ulteriori informazioni, consulta Parametri di controllo personalizzati nella Guida per l'utente di Centrale di sicurezza AWS .

- Per i dettagli sull'API, consulta GetSecurityControlDefinition AWS CLICommand Reference.

`invite-members`

Il seguente esempio di codice mostra come utilizzare`invite-members`.

AWS CLI

Come inviare inviti agli account membro

L'esempio `invite-members` seguente invia gli inviti agli account membri specificati.

```
aws securityhub invite-members \
    --account-ids "123456789111" "123456789222"
```

Output:

```
{
    "UnprocessedAccounts": []
}
```

Per ulteriori informazioni, consulta Gestione degli account degli amministratori e dei membri nella Guida per l'utente di Centrale di sicurezza AWS .

- Per i dettagli sull'API, consulta InviteMembers AWS CLICommand Reference.

`list-automation-rules`

Il seguente esempio di codice mostra come utilizzare`list-automation-rules`.

AWS CLI

Come visualizzare un elenco di regole di automazione

L'`list-automation-rules`esempio seguente elenca le regole di automazione per un AWS account. Solo l'account amministratore di Centrale di sicurezza può eseguire questo comando.

```
aws securityhub list-automation-rules \
    --max-results 3 \
    --next-token NULL
```

Output:

```
{
    "AutomationRulesMetadata": [
```

```
        {
            "RuleArn": "arn:aws:securityhub:us-east-1:123456789012:automation-rule/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
            "RuleStatus": "ENABLED",
            "RuleOrder": 1,
            "RuleName": "Suppress informational findings",
            "Description": "Suppress GuardDuty findings with Informational
 severity",
            "IsTerminal": false,
            "CreatedAt": "2023-05-31T17:56:14.837000+00:00",
            "UpdatedAt": "2023-05-31T17:59:38.466000+00:00",
            "CreatedBy": "arn:aws:iam::123456789012:role/Admin"
        },
        {
            "RuleArn": "arn:aws:securityhub:us-east-1:123456789012:automation-rule/
a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
            "RuleStatus": "ENABLED",
            "RuleOrder": 1,
            "RuleName": "sample rule",
            "Description": "A sample rule",
            "IsTerminal": false,
            "CreatedAt": "2023-07-15T23:37:20.223000+00:00",
            "UpdatedAt": "2023-07-15T23:37:20.223000+00:00",
            "CreatedBy": "arn:aws:iam::123456789012:role/Admin"
        },
        {
            "RuleArn": "arn:aws:securityhub:us-east-1:123456789012:automation-rule/
a1b2c3d4-5678-90ab-cdef-EXAMPLE33333",
            "RuleStatus": "ENABLED",
            "RuleOrder": 1,
            "RuleName": "sample rule",
            "Description": "A sample rule",
            "IsTerminal": false,
            "CreatedAt": "2023-07-15T23:45:25.126000+00:00",
            "UpdatedAt": "2023-07-15T23:45:25.126000+00:00",
            "CreatedBy": "arn:aws:iam::123456789012:role/Admin"
        }
    ]
}
```

Per ulteriori informazioni, consulta Visualizzazione delle regole di automazione nella Guida per l'utente di Centrale di sicurezza AWS .

- Per i dettagli sull'API, consulta ListAutomationRules AWS CLICommand Reference.

`list-configuration-policies`

Il seguente esempio di codice mostra come utilizzare`list-configuration-policies`.

AWS CLI

Come elencare i riepiloghi delle policy di configurazione

L'esempio `list-configuration-policies` seguente elenca un riepilogo delle policy di configurazione per l'organizzazione.

```
aws securityhub list-configuration-policies \
    --max-items 3
```

Output:

```
{
    "ConfigurationPolicySummaries": [
        {
            "Arn": "arn:aws:securityhub:eu-central-1:123456789012:configuration-
policy/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
            "Id": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
            "Name": "SampleConfigurationPolicy1",
            "Description": "SampleDescription1",
            "UpdatedAt": "2023-09-26T21:08:36.214000+00:00",
            "ServiceEnabled": true
        },
        {
            "Arn": "arn:aws:securityhub:eu-central-1:123456789012:configuration-
policy/a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
            "Id": "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
            "Name": "SampleConfigurationPolicy2",
            "Description": "SampleDescription2"
            "UpdatedAt": "2023-11-28T19:26:25.207000+00:00",
            "ServiceEnabled": true
        },
        {
            "Arn": "arn:aws:securityhub:eu-central-1:123456789012:configuration-
policy/a1b2c3d4-5678-90ab-cdef-EXAMPLE33333",
            "Id": "a1b2c3d4-5678-90ab-cdef-EXAMPLE33333",
            "Name": "SampleConfigurationPolicy3",
            "Description": "SampleDescription3",
            "UpdatedAt": "2023-11-28T20:28:04.494000+00:00",
```

```
            "ServiceEnabled": true
        }
}
```

Per ulteriori informazioni, consulta Visualizzazione delle policy di configurazione di Centrale di sicurezza nella Guida per l'utente di Centrale di sicurezza AWS .

- Per i dettagli sull'API, consulta ListConfigurationPolicies AWS CLICommand Reference.

`list-configuration-policy-associations`

Il seguente esempio di codice mostra come utilizzare`list-configuration-policy-associations`.

AWS CLI

Come elencare le associazioni di configurazione

L'esempio `list-configuration-policy-associations` seguente elenca un riepilogo delle associazioni di configurazione per l'organizzazione. La risposta include associazioni con policy di configurazione e comportamenti autogestiti.

```
aws securityhub list-configuration-policy-associations \
    --filters '{"AssociationType": "APPLIED"}' \
    --max-items 4
```

Output:

```
{
    "ConfigurationPolicyAssociationSummaries": [
        {
            "ConfigurationPolicyId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
            "TargetId": "r-1ab2",
            "TargetType": "ROOT",
            "AssociationType": "APPLIED",
            "UpdatedAt": "2023-11-28T19:26:49.417000+00:00",
            "AssociationStatus": "FAILED",
            "AssociationStatusMessage": "Policy association failed because 2
 organizational units or accounts under this root failed."
        },
        {
```

```
            "ConfigurationPolicyId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
            "TargetId": "ou-1ab2-c3de4f5g",
            "TargetType": "ORGANIZATIONAL_UNIT",
            "AssociationType": "APPLIED",
            "UpdatedAt": "2023-09-26T21:14:05.283000+00:00",
            "AssociationStatus": "FAILED",
            "AssociationStatusMessage": "One or more children under this target
 failed association."
        },
        {
            "ConfigurationPolicyId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE33333",
            "TargetId": "ou-6hi7-8j91kl2m",
            "TargetType": "ORGANIZATIONAL_UNIT",
            "AssociationType": "APPLIED",
            "UpdatedAt": "2023-09-26T21:13:01.816000+00:00",
            "AssociationStatus": "SUCCESS",
            "AssociationStatusMessage": "Association applied successfully on this
 target."
        },
        {
            "ConfigurationPolicyId": "SELF_MANAGED_SECURITY_HUB",
            "TargetId": "111122223333",
            "TargetType": "ACCOUNT",
            "AssociationType": "APPLIED",
            "UpdatedAt": "2023-11-28T22:01:26.409000+00:00",
            "AssociationStatus": "SUCCESS"
    }
}
```

Per ulteriori informazioni, consulta Visualizzazione dello stato e dei dettagli delle policy di configurazione nella Guida per l'utente di Centrale di sicurezza AWS .

- Per i dettagli sull'API, consulta ListConfigurationPolicyAssociations AWS CLICommand Reference.

`list-enabled-products-for-import`

Il seguente esempio di codice mostra come utilizzare`list-enabled-products-for-import`.

AWS CLI

Come restituire l'elenco delle integrazioni di prodotti abilitate

L'esempio `list-enabled-products-for-import` seguente restituisce l'elenco degli ARN delle sottoscrizioni per le integrazioni di prodotti attualmente abilitate.

```
aws securityhub list-enabled-products-for-import
```

Output:

```
{
    "ProductSubscriptions": [ "arn:aws:securityhub:us-west-1:123456789012:product-subscription/crowdstrike/crowdstrike-falcon", "arn:aws:securityhub:us-west-1:123456789012:product-subscription/aws/securityhub" ]
}
```

Per ulteriori informazioni, consulta Gestione delle integrazioni di prodotti nella Guida per l'utente di Centrale di sicurezza AWS .

- Per i dettagli sull'API, consulta ListEnabledProductsForImport AWS CLICommand Reference.

`list-finding-aggregators`

Il seguente esempio di codice mostra come utilizzare`list-finding-aggregators`.

AWS CLI

Come elencare i widget disponibili

L'esempio `list-finding-aggregators` seguente restituisce l'ARN della configurazione di aggregazione degli esiti.

```
aws securityhub list-finding-aggregators
```

Output:

```
{
    "FindingAggregatorArn": "arn:aws:securityhub:us-east-1:222222222222:finding-aggregator/123e4567-e89b-12d3-a456-426652340000"
}
```

Per ulteriori informazioni, consulta Visualizzazione della configurazione di aggregazione degli esiti corrente nella Guida per l'utente di Centrale di sicurezza AWS .

- Per i dettagli sull'API, consulta ListFindingAggregators AWS CLICommand Reference.

`list-invitations`

Il seguente esempio di codice mostra come utilizzare`list-invitations`.

AWS CLI

Come visualizzare un elenco di inviti

L'esempio `list-invitations` seguente recupera l'elenco degli inviti inviati all'account richiedente.

```
aws securityhub list-invitations
```

Output:

```
{
    "Invitations": [
        {
            "AccountId": "123456789012",
            "InvitationId": "7ab938c5d52d7904ad09f9e7c20cc4eb",
            "InvitedAt": 2020-06-01T20:21:18.042000+00:00,
            "MemberStatus": "ASSOCIATED"
        }
    ],
}
```

Per ulteriori informazioni, consulta Gestione degli account degli amministratori e dei membri nella Guida per l'utente di Centrale di sicurezza AWS .

- Per i dettagli sull'API, consulta ListInvitations AWS CLICommand Reference.

`list-members`

Il seguente esempio di codice mostra come utilizzare`list-members`.

AWS CLI

Come recuperare un elenco di account membri

L'esempio `list-members` seguente restituisce l'elenco di account membri per l'account amministratore richiedente.

```
aws securityhub list-members
```

Output:

```
{
    "Members": [
        {
            "AccountId": "123456789111",
            "AdministratorId": "123456789012",
            "InvitedAt": 2020-06-01T20:15:15.289000+00:00,
            "MasterId": "123456789012",
            "MemberStatus": "ASSOCIATED",
            "UpdatedAt": 2020-06-01T20:15:15.289000+00:00
        },
        {
            "AccountId": "123456789222",
            "AdministratorId": "123456789012",
            "InvitedAt": 2020-06-01T20:15:15.289000+00:00,
            "MasterId": "123456789012",
            "MemberStatus": "ASSOCIATED",
            "UpdatedAt": 2020-06-01T20:15:15.289000+00:00
        }
    ],
}
```

Per ulteriori informazioni, consulta Gestione degli account degli amministratori e dei membri nella Guida per l'utente di Centrale di sicurezza AWS .

- Per i dettagli sull'API, consulta ListMembers AWS CLICommand Reference.

`list-organization-admin-accounts`

Il seguente esempio di codice mostra come utilizzare`list-organization-admin-accounts`.

AWS CLI

Come elencare gli account amministrativi designati per Centrale di sicurezza

L'esempio `list-organization-admin-accounts` seguente elenca gli account amministratore di Centrale di sicurezza per un'organizzazione.

```
aws securityhub list-organization-admin-accounts
```

Output:

```
{
    AdminAccounts": [
        { "AccountId": "777788889999" },
        { "Status": "ENABLED" }
    ]
}
```

Per ulteriori informazioni, consulta Designazione di un account amministratore di Centrale di sicurezza nella Guida per l'utente di Centrale di sicurezza AWS .

- Per i dettagli sull'API, consulta ListOrganizationAdminAccounts AWS CLICommand Reference.

`list-security-control-definitions`

Il seguente esempio di codice mostra come utilizzare`list-security-control-definitions`.

AWS CLI

Esempio 1: come elencare tutti i controlli di sicurezza disponibili

L'esempio `list-security-control-definitions` seguente elenca i controlli di sicurezza disponibili in tutti gli standard di Centrale di sicurezza. Questo esempio limita i risultati a tre controlli.

```
aws securityhub list-security-control-definitions \
    --max-items 3
```

Output:

```
{
    "SecurityControlDefinitions": [
        {
            "SecurityControlId": "ACM.1",
            "Title": "Imported and ACM-issued certificates should be renewed after a
 specified time period",
            "Description": "This control checks whether an AWS Certificate Manager
 (ACM) certificate is renewed within the specified time period. It checks both
 imported certificates and certificates provided by ACM. The control fails if the
 certificate isn't renewed within the specified time period. Unless you provide a
```

```
 custom parameter value for the renewal period, Security Hub uses a default value of
 30 days.",
            "RemediationUrl": "https://docs.aws.amazon.com/console/securityhub/
ACM.1/remediation",
            "SeverityRating": "MEDIUM",
            "CurrentRegionAvailability": "AVAILABLE",
            "CustomizableProperties": [
                "Parameters"
            ]
        },
        {
            "SecurityControlId": "ACM.2",
            "Title": "RSA certificates managed by ACM should use a key length of at
 least 2,048 bits",
            "Description": "This control checks whether RSA certificates managed by
 AWS Certificate Manager use a key length of at least 2,048 bits. The control fails
 if the key length is smaller than 2,048 bits.",
            "RemediationUrl": "https://docs.aws.amazon.com/console/securityhub/
ACM.2/remediation",
            "SeverityRating": "HIGH",
            "CurrentRegionAvailability": "AVAILABLE",
            "CustomizableProperties": []
        },
        {
            "SecurityControlId": "APIGateway.1",
            "Title": "API Gateway REST and WebSocket API execution logging should be
 enabled",
            "Description": "This control checks whether all stages of an Amazon
 API Gateway REST or WebSocket API have logging enabled. The control fails if
 the 'loggingLevel' isn't 'ERROR' or 'INFO' for all stages of the API. Unless you
 provide custom parameter values to indicate that a specific log type should be
 enabled, Security Hub produces a passed finding if the logging level is either
 'ERROR' or 'INFO'.",
            "RemediationUrl": "https://docs.aws.amazon.com/console/securityhub/
APIGateway.1/remediation",
            "SeverityRating": "MEDIUM",
            "CurrentRegionAvailability": "AVAILABLE",
            "CustomizableProperties": [
                "Parameters"
            ]
        }
    ],
```

```
    "NextToken": "U2FsdGVkX1/UprCPzxVbkDeHikDXbDxfgJZ1w2RG1XWsFPTMTIQPVE0m/
FduIGxS7ObRtAbaUt/8/RCQcg2PU0YXI20hH/GrhoOTgv+TSm0qvQVFhkJepWmqh
+NYawjocVBeos6xzn/8qnbF9IuwGg=="
}
```

Per ulteriori informazioni, consulta Visualizzazione dei dettagli per uno standard nella Guida per l'utente di Centrale di sicurezza AWS .

Esempio 2: come elencare i controlli di sicurezza disponibili per uno standard specifico

L'`list-security-control-definitions`esempio seguente elenca i controlli di sicurezza disponibili per il CIS AWS Foundations Benchmark v1.4.0. Questo esempio limita i risultati a tre controlli.

```
aws securityhub list-security-control-definitions \
    --standards-arn "arn:aws:securityhub:us-east-1::standards/cis-aws-foundations-
benchmark/v/1.4.0" \
    --max-items 3
```

Output:

```
{
    "SecurityControlDefinitions": [
        {
            "SecurityControlId": "CloudTrail.1",
            "Title": "CloudTrail should be enabled and configured with at least one
 multi-Region trail that includes read and write management events",
            "Description": "This AWS control checks that there is at least one
 multi-region AWS CloudTrail trail includes read and write management events.",
            "RemediationUrl": "https://docs.aws.amazon.com/console/securityhub/
CloudTrail.1/remediation",
            "SeverityRating": "HIGH",
            "CurrentRegionAvailability": "AVAILABLE",
            "CustomizableProperties": []
        },
        {
            "SecurityControlId": "CloudTrail.2",
            "Title": "CloudTrail should have encryption at-rest enabled",
            "Description": "This AWS control checks whether AWS CloudTrail is
 configured to use the server side encryption (SSE) AWS Key Management Service (AWS
 KMS) customer master key (CMK) encryption. The check will pass if the KmsKeyId is
 defined.",
```

```
            "RemediationUrl": "https://docs.aws.amazon.com/console/securityhub/
CloudTrail.2/remediation",
            "SeverityRating": "MEDIUM",
            "CurrentRegionAvailability": "AVAILABLE",
            "CustomizableProperties": []
        },
        {
            "SecurityControlId": "CloudTrail.4",
            "Title": "CloudTrail log file validation should be enabled",
            "Description": "This AWS control checks whether CloudTrail log file
 validation is enabled.",
            "RemediationUrl": "https://docs.aws.amazon.com/console/securityhub/
CloudTrail.4/remediation",
            "SeverityRating": "MEDIUM",
            "CurrentRegionAvailability": "AVAILABLE",
            "CustomizableProperties": []
        }
    ],
    "NextToken": "eyJOZXh0VG9rZW4iOiBudWxsLCAiYm90b190cnVuY2F0ZV9hbW91bnQiOiAzfQ=="
}
```

Per ulteriori informazioni, consulta Visualizzazione dei dettagli per uno standard nella Guida per l'utente di Centrale di sicurezza AWS .

- Per i dettagli sull'API, vedere ListSecurityControlDefinitionsin Command Reference.AWS CLI

`list-standards-control-associations`

Il seguente esempio di codice mostra come utilizzare`list-standards-control-associations`.

AWS CLI

Come ottenere lo stato di abilitazione di un controllo in ogni standard abilitato

L'`list-standards-control-associations`esempio seguente elenca lo stato di abilitazione di CloudTrail .1 in ogni standard abilitato.

```
aws securityhub list-standards-control-associations \
    --security-control-id CloudTrail.1
```

Output:

```
{
```

```
    "StandardsControlAssociationSummaries": [
        {
            "StandardsArn": "arn:aws:securityhub:us-east-2::standards/nist-800-53/
v/5.0.0",
            "SecurityControlId": "CloudTrail.1",
            "SecurityControlArn": "arn:aws:securityhub:us-
east-2:123456789012:security-control/CloudTrail.1",
            "AssociationStatus": "ENABLED",
            "RelatedRequirements": [
                "NIST.800-53.r5 AC-2(4)",
                "NIST.800-53.r5 AC-4(26)",
                "NIST.800-53.r5 AC-6(9)",
                "NIST.800-53.r5 AU-10",
                "NIST.800-53.r5 AU-12",
                "NIST.800-53.r5 AU-2",
                "NIST.800-53.r5 AU-3",
                "NIST.800-53.r5 AU-6(3)",
                "NIST.800-53.r5 AU-6(4)",
                "NIST.800-53.r5 AU-14(1)",
                "NIST.800-53.r5 CA-7",
                "NIST.800-53.r5 SC-7(9)",
                "NIST.800-53.r5 SI-3(8)",
                "NIST.800-53.r5 SI-4(20)",
                "NIST.800-53.r5 SI-7(8)",
                "NIST.800-53.r5 SA-8(22)"
            ],
            "UpdatedAt": "2023-05-15T17:52:21.304000+00:00",
            "StandardsControlTitle": "CloudTrail should be enabled and configured
 with at least one multi-Region trail that includes read and write management
 events",
            "StandardsControlDescription": "This AWS control checks that there is
 at least one multi-region AWS CloudTrail trail includes read and write management
 events."
        },
        {
            "StandardsArn": "arn:aws:securityhub:::ruleset/cis-aws-foundations-
benchmark/v/1.2.0",
            "SecurityControlId": "CloudTrail.1",
            "SecurityControlArn": "arn:aws:securityhub:us-
east-2:123456789012:security-control/CloudTrail.1",
            "AssociationStatus": "ENABLED",
            "RelatedRequirements": [
                "CIS AWS Foundations 2.1"
            ],
```

```
            "UpdatedAt": "2020-02-10T21:22:53.998000+00:00",
            "StandardsControlTitle": "Ensure CloudTrail is enabled in all regions",
            "StandardsControlDescription": "AWS CloudTrail is a web service that
 records AWS API calls for your account and delivers log files to you. The recorded
 information includes the identity of the API caller, the time of the API call,
 the source IP address of the API caller, the request parameters, and the response
 elements returned by the AWS service."
        },
        {
            "StandardsArn": "arn:aws:securityhub:us-east-2::standards/aws-
foundational-security-best-practices/v/1.0.0",
            "SecurityControlId": "CloudTrail.1",
            "SecurityControlArn": "arn:aws:securityhub:us-
east-2:123456789012:security-control/CloudTrail.1",
            "AssociationStatus": "DISABLED",
            "RelatedRequirements": [],
            "UpdatedAt": "2023-05-15T19:31:52.671000+00:00",
            "UpdatedReason": "Alternative compensating controls are in place",
            "StandardsControlTitle": "CloudTrail should be enabled and configured
 with at least one multi-Region trail that includes read and write management
 events",
            "StandardsControlDescription": "This AWS control checks that there is
 at least one multi-region AWS CloudTrail trail includes read and write management
 events."
        },
        {
            "StandardsArn": "arn:aws:securityhub:us-east-2::standards/cis-aws-
foundations-benchmark/v/1.4.0",
            "SecurityControlId": "CloudTrail.1",
            "SecurityControlArn": "arn:aws:securityhub:us-
east-2:123456789012:security-control/CloudTrail.1",
            "AssociationStatus": "ENABLED",
            "RelatedRequirements": [
                "CIS AWS Foundations Benchmark v1.4.0/3.1"
            ],
            "UpdatedAt": "2022-11-10T15:40:36.021000+00:00",
            "StandardsControlTitle": "Ensure CloudTrail is enabled in all regions",
            "StandardsControlDescription": "AWS CloudTrail is a web service that
 records AWS API calls for your account and delivers log files to you. The recorded
 information includes the identity of the API caller, the time of the API call,
 the source IP address of the API caller, the request parameters, and the response
 elements returned by the AWS service. CloudTrail provides a history of AWS API
 calls for an account, including API calls made via the Management Console, SDKs,
 command line tools, and higher-level AWS services (such as CloudFormation)."
```

```
        }
    ]
}
```

Per ulteriori informazioni, consulta Abilitazione e disabilitazione dei controlli con standard specifici nella Guida per l'utente di Centrale di sicurezza AWS .

- Per i dettagli sull'API, vedere ListStandardsControlAssociationsin AWS CLI Command Reference.

`list-tags-for-resource`

Il seguente esempio di codice mostra come utilizzare`list-tags-for-resource`.

AWS CLI

Come recuperare i tag assegnati a una risorsa

L'esempio `list-tags-for-resource` seguente restituisce i tag associati alla risorsa hub specificata.

```
aws securityhub list-tags-for-resource \
    --resource-arn "arn:aws:securityhub:us-west-1:123456789012:hub/default"
```

Output:

```
{
    "Tags": {
        "Department" : "Operations",
        "Area" : "USMidwest"
    }
}
```

Per ulteriori informazioni, vedere AWS::SecurityHub: :Hub nella Guida per l'AWS CloudFormation utente.

- Per i dettagli sull'API, consulta ListTagsForResource AWS CLICommand Reference.

`start-configuration-policy-association`

Il seguente esempio di codice mostra come utilizzare`start-configuration-policy-association`.

AWS CLI

Esempio 1: come associare una policy di configurazione

L'esempio `start-configuration-policy-association` seguente associa la policy di configurazione specificata all'unità organizzativa indicata. Una configurazione può essere associata a un account di destinazione, a un'unità organizzativa o alla radice.

```
aws securityhub start-configuration-policy-association \
    --configuration-policy-identifier "arn:aws:securityhub:eu-central-1:123456789012:configuration-policy/a1b2c3d4-5678-90ab-cdef-EXAMPLE33333" \
    --target '{"OrganizationalUnitId": "ou-6hi7-8j91kl2m"}'
```

Output:

```
{
    "ConfigurationPolicyId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE33333",
    "TargetId": "ou-6hi7-8j91kl2m",
    "TargetType": "ORGANIZATIONAL_UNIT",
    "AssociationType": "APPLIED",
    "UpdatedAt": "2023-11-29T17:40:52.468000+00:00",
    "AssociationStatus": "PENDING"
}
```

Per ulteriori informazioni, consulta Creazione e associazione di policy di configurazione di Centrale di sicurezza nella Guida per l'utente di Centrale di sicurezza AWS .

Esempio 2: come associare una configurazione autogestita

L'esempio `start-configuration-policy-association` seguente associa una configurazione autogestita all'account specificato.

```
aws securityhub start-configuration-policy-association \
    --configuration-policy-identifier "SELF_MANAGED_SECURITY_HUB" \
    --target '{"OrganizationalUnitId": "123456789012"}'
```

Output:

```
{
    "ConfigurationPolicyId": "SELF_MANAGED_SECURITY_HUB",
    "TargetId": "123456789012",
```

```
    "TargetType": "ACCOUNT",
    "AssociationType": "APPLIED",
    "UpdatedAt": "2023-11-29T17:40:52.468000+00:00",
    "AssociationStatus": "PENDING"
}
```

Per ulteriori informazioni, consulta Creazione e associazione di policy di configurazione di Centrale di sicurezza nella Guida per l'utente di Centrale di sicurezza AWS .

- Per i dettagli sull'API, consulta StartConfigurationPolicyAssociation AWS CLICommand Reference.

`start-configuration-policy-disassociation`

Il seguente esempio di codice mostra come utilizzare`start-configuration-policy-disassociation`.

AWS CLI

Esempio 1: come annullare l'associazione per una policy di configurazione

L'esempio `start-configuration-policy-disassociation` seguente annulla l'associazione tra la policy di configurazione e l'unità organizzativa specificata. Per una configurazione, l'associazione può essere annullata da un account di destinazione, da un'unità organizzativa o dalla root.

```
aws securityhub start-configuration-policy-disassociation \
    --configuration-policy-identifier "arn:aws:securityhub:eu-
central-1:123456789012:configuration-policy/a1b2c3d4-5678-90ab-cdef-EXAMPLE33333" \
    --target '{"OrganizationalUnitId": "ou-6hi7-8j91kl2m"}'
```

Questo comando non produce alcun output.

Per ulteriori informazioni, vedere Dissociazione di una configurazione dagli account e OUs nella Guida per l'utente AWS di Security Hub.

Esempio 2: come annullare l'associazione per una configurazione autogestita

L'esempio `start-configuration-policy-disassociation` seguente annulla l'associazione tra una configurazione autogestita e l'account specificato.

```
aws securityhub start-configuration-policy-disassociation \
```

```
    --configuration-policy-identifier "SELF_MANAGED_SECURITY_HUB" \
    --target '{"AccountId": "123456789012"}'
```

Questo comando non produce alcun output.

Per ulteriori informazioni, vedere Dissociazione di una configurazione dagli account e OUs nella Guida per l'utente AWS di Security Hub.

- Per i dettagli sull'API, vedere StartConfigurationPolicyDisassociationin AWS CLI Command Reference.

`tag-resource`

Il seguente esempio di codice mostra come utilizzare`tag-resource`.

AWS CLI

Come assegnare un tag a una risorsa

L'esempio `tag-resource` seguente assegna valori per i tag Department e Area alla risorsa hub specificata.

```
aws securityhub tag-resource \
    --resource-arn "arn:aws:securityhub:us-west-1:123456789012:hub/default" \
    --tags '{"Department":"Operations", "Area":"USMidwest"}'
```

Questo comando non produce alcun output.

Per ulteriori informazioni, vedere AWS::SecurityHub: :Hub nella Guida per l'AWS CloudFormation utente.

- Per i dettagli sull'API, consulta TagResource AWS CLICommand Reference.

`untag-resource`

Il seguente esempio di codice mostra come utilizzare`untag-resource`.

AWS CLI

Come rimuovere un tag da una risorsa

L'esempio `untag-resource` seguente rimuove il tag Department dalla risorsa hub specificata.

```
aws securityhub untag-resource \
    --resource-arn "arn:aws:securityhub:us-west-1:123456789012:hub/default" \
    --tag-keys "Department"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, vedere AWS::SecurityHub: :Hub nella Guida per l'AWS CloudFormation utente.

- Per i dettagli sull'API, consulta UntagResource AWS CLICommand Reference.

`update-action-target`

Il seguente esempio di codice mostra come utilizzare`update-action-target`.

AWS CLI

Come aggiornare un'operazione personalizzata

L'esempio `update-action-target` seguente elimina il nome dell'operazione personalizzata identificata dall'ARN specificato.

```
aws securityhub update-action-target \
    --action-target-arn "arn:aws:securityhub:us-west-1:123456789012:action/custom/
Remediation" \
    --name "Send to remediation"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, vedere Creazione di un'azione personalizzata e associazione a una regola CloudWatch Events nella Guida per l'utente AWS di Security Hub.

- Per i dettagli sull'API, consulta UpdateActionTarget AWS CLICommand Reference.

`update-configuration-policy`

Il seguente esempio di codice mostra come utilizzare`update-configuration-policy`.

AWS CLI

Come aggiornare una policy di configurazione

L'esempio `update-configuration-policy` seguente aggiorna una policy di configurazione esistente per utilizzare le impostazioni specificate.

```
aws securityhub update-configuration-policy \
    --identifier "arn:aws:securityhub:eu-central-1:508236694226:configuration-policy/09f37766-57d8-4ede-9d33-5d8b0fecf70e" \
    --name "SampleConfigurationPolicyUpdated" \
    --description "SampleDescriptionUpdated" \
    --configuration-policy '{"SecurityHub": {"ServiceEnabled": true, "EnabledStandardIdentifiers": ["arn:aws:securityhub:eu-central-1::standards/aws-foundational-security-best-practices/v/1.0.0","arn:aws:securityhub:::ruleset/cis-aws-foundations-benchmark/v/1.2.0"],"SecurityControlsConfiguration":{"DisabledSecurityControlIdentifiers": ["CloudWatch.1"], "SecurityControlCustomParameters": [{"SecurityControlId": "ACM.1", "Parameters": {"daysToExpiration": {"ValueType": "CUSTOM", "Value": {"Integer": 21}}}}]}}}' \
    --updated-reason "Disabling CloudWatch.1 and changing parameter value"
```

Output:

```
{
    "Arn": "arn:aws:securityhub:eu-central-1:123456789012:configuration-policy/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "Id": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "Name": "SampleConfigurationPolicyUpdated",
    "Description": "SampleDescriptionUpdated",
    "UpdatedAt": "2023-11-28T20:28:04.494000+00:00",
    "CreatedAt": "2023-11-28T20:28:04.494000+00:00",
    "ConfigurationPolicy": {
        "SecurityHub": {
            "ServiceEnabled": true,
            "EnabledStandardIdentifiers": [
                "arn:aws:securityhub:eu-central-1::standards/aws-foundational-security-best-practices/v/1.0.0",
                "arn:aws:securityhub:::ruleset/cis-aws-foundations-benchmark/v/1.2.0"
            ],
            "SecurityControlsConfiguration": {
                "DisabledSecurityControlIdentifiers": [
                    "CloudWatch.1"
                ],
                "SecurityControlCustomParameters": [
                    {
```

```
                        "SecurityControlId": "ACM.1",
                        "Parameters": {
                            "daysToExpiration": {
                                "ValueType": "CUSTOM",
                                "Value": {
                                    "Integer": 21
                                }
                            }
                        }
                    }
                ]
            }
        }
    }
}
```

Per ulteriori informazioni, consulta Aggiornamento delle policy di configurazione di Centrale di sicurezza nella Guida per l'utente di Centrale di sicurezza AWS .

- Per i dettagli sull'API, consulta UpdateConfigurationPolicy AWS CLICommand Reference.

`update-finding-aggregator`

Il seguente esempio di codice mostra come utilizzare`update-finding-aggregator`.

AWS CLI

Come aggiornare la configurazione di aggregazione degli esiti corrente

L'esempio `update-finding-aggregator` seguente modifica la configurazione dell'aggregazione degli esiti in base al collegamento da Regioni selezionate. Il comando viene eseguito dalla Regione Stati Uniti orientali (Virginia), ovvero la Regione di aggregazione. Seleziona Stati Uniti occidentali (California settentrionale) e Stati Uniti occidentali (Oregon) come Regioni collegate.

```
aws securityhub update-finding-aggregator \
    --region us-east-1 \
    --finding-aggregator-arn arn:aws:securityhub:us-east-1:222222222222:finding-aggregator/123e4567-e89b-12d3-a456-426652340000 \
    --region-linking-mode SPECIFIED_REGIONS \
    --regions us-west-1,us-west-2
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Aggiornamento della configurazione di aggregazione degli esiti nella Guida per l'utente di Centrale di sicurezza AWS .

- Per i dettagli sull'API, consulta UpdateFindingAggregator AWS CLICommand Reference.

`update-insight`

Il seguente esempio di codice mostra come utilizzare`update-insight`.

AWS CLI

Esempio 1: come modificare il filtro per un approfondimento personalizzato

L'esempio `update-insight` seguente modifica il filtro per un approfondimento personalizzato. L'analisi aggiornata cerca risultati con una severità elevata correlati ai AWS ruoli.

```
aws securityhub update-insight \
    --insight-arn "arn:aws:securityhub:us-west-1:123456789012:insight/123456789012/
custom/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111" \
    --filters '{"ResourceType": [{ "Comparison": "EQUALS", "Value": "AwsIamRole"}],
 "SeverityLabel": [{"Comparison": "EQUALS", "Value": "HIGH"}]}' \
    --name "High severity role findings"
```

Esempio 2: come modificare l'attributo di raggruppamento per un approfondimento personalizzato

L'esempio `update-insight` seguente modifica l'attributo di raggruppamento per l'approfondimento personalizzato con l'ARN specificato. Il nuovo attributo di raggruppamento è l'ID della risorsa.

```
aws securityhub update-insight \
    --insight-arn "arn:aws:securityhub:us-west-1:123456789012:insight/123456789012/
custom/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111" \
    --group-by-attribute "ResourceId" \
    --name "Critical role findings"
```

Output:

```
{
    "Insights": [
```

```
        {
            "InsightArn": "arn:aws:securityhub:us-
west-1:123456789012:insight/123456789012/custom/a1b2c3d4-5678-90ab-cdef-
EXAMPLE11111",
            "Name": "Critical role findings",
            "Filters": {
                "SeverityLabel": [
                    {
                        "Value": "CRITICAL",
                        "Comparison": "EQUALS"
                    }
                ],
                "ResourceType": [
                    {
                        "Value": "AwsIamRole",
                        "Comparison": "EQUALS"
                    }
                ]
            },
            "GroupByAttribute": "ResourceId"
        }
    ]
}
```

Per ulteriori informazioni, consulta Gestione di informazioni dettagliate personalizzate nella Guida per l'utente di Centrale di sicurezza AWS .

- Per i dettagli sull'API, consulta UpdateInsight AWS CLICommand Reference.

`update-organization-configuration`

Il seguente esempio di codice mostra come utilizzare`update-organization-configuration`.

AWS CLI

Come aggiornare la configurazione di Centrale di sicurezza per un'organizzazione

L'esempio `update-organization-configuration` seguente specifica che Centrale di sicurezza deve utilizzare la configurazione centrale per un'organizzazione. Dopo aver eseguito questo comando, l'amministratore di Centrale di sicurezza delegato può creare e gestire le policy di configurazione per configurare l'organizzazione. L'amministratore può utilizzare questo comando anche per passare dalla configurazione centrale a quella locale. Se il tipo di

configurazione è locale, l'amministratore può scegliere se abilitare automaticamente Centrale di sicurezza e gli standard di sicurezza predefiniti nei nuovi account dell'organizzazione.

```
aws securityhub update-organization-configuration \
    --no-auto-enable \
    --organization-configuration '{"ConfigurationType": "CENTRAL"}'
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Managing accounts with AWS Organizations nella AWS Security Hub User Guide.

- Per i dettagli sull'API, vedere UpdateOrganizationConfigurationin AWS CLI Command Reference.

`update-security-control`

Il seguente esempio di codice mostra come utilizzare`update-security-control`.

AWS CLI

Come aggiornare le proprietà del controllo di sicurezza

L'esempio `update-security-control` seguente specifica valori personalizzati per un parametro di controllo di sicurezza di Centrale di sicurezza.

```
aws securityhub update-security-control \
    --security-control-id ACM.1 \
    --parameters '{"daysToExpiration": {"ValueType": "CUSTOM", "Value": {"Integer":
 15}}}' \
    --last-update-reason "Internal compliance requirement"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Parametri di controllo personalizzati nella Guida per l'utente di Centrale di sicurezza AWS .

- Per i dettagli sull'API, consulta UpdateSecurityControl AWS CLICommand Reference.

`update-security-hub-configuration`

Il seguente esempio di codice mostra come utilizzare`update-security-hub-configuration`.

AWS CLI

Come aggiornare la configurazione di Centrale di sicurezza

L'esempio `update-security-hub-configuration` seguente configura Centrale di sicurezza per abilitare automaticamente nuovi controlli per gli standard abilitati.

```
aws securityhub update-security-hub-configuration \
    --auto-enable-controls
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Abilitazione automatica di nuovi controlli nella Guida per l'utente di Centrale di sicurezza AWS .

- Per i dettagli sull'API, consulta UpdateSecurityHubConfiguration AWS CLICommand Reference.

`update-standards-control`

Il seguente esempio di codice mostra come utilizzare`update-standards-control`.

AWS CLI

Esempio 1: come disabilitare un controllo

L'`update-standards-control`esempio seguente disattiva il PCI. AutoScaling1. Controllo.

```
aws securityhub update-standards-control \
    --standards-control-arn "arn:aws:securityhub:us-west-1:123456789012:control/pci-dss/v/3.2.1/PCI.AutoScaling.1" \
    --control-status "DISABLED" \
    --disabled-reason "Not applicable for my service"
```

Questo comando non produce alcun output.

Esempio 2: come abilitare un controllo

L'`update-standards-control`esempio seguente abilita il PCI. AutoScaling1. controllo.

```
aws securityhub update-standards-control \
    --standards-control-arn "arn:aws:securityhub:us-west-1:123456789012:control/pci-dss/v/3.2.1/PCI.AutoScaling.1" \
```

```
    --control-status "ENABLED"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Disabilitazione e abilitazione dei controlli individuali nella Guida per l'utente di Centrale di sicurezza AWS .

- Per i dettagli sull'API, consulta UpdateStandardsControl AWS CLICommand Reference.

Esempi di utilizzo di Security Lake AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando il AWS Command Line Interface con Security Lake.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`create-aws-log-source`

Il seguente esempio di codice mostra come utilizzare`create-aws-log-source`.

AWS CLI

Come aggiungere un Amazon Web Service con supporto nativo come sorgente Amazon Security Lake

L'esempio `create-aws-logsource` seguente aggiunge i log di flusso VPC come origine di Security Lake negli account e nelle Regioni designati.

```
aws securitylake create-aws-log-source \
```

```
    --sources '[{"regions": ["us-east-1"], "accounts": ["123456789012"],
 "sourceName": "SH_FINDINGS", "sourceVersion": "2.0"}]'
```

Output:

```
{
    "failed": [
        "123456789012"
    ]
}
```

Per ulteriori informazioni, consulta Aggiungere un AWS servizio come fonte nella Guida per l'utente di Amazon Security Lake.

- Per i dettagli sull'API, consulta CreateAwsLogSource AWS CLICommand Reference.

`create-custom-log-source`

Il seguente esempio di codice mostra come utilizzare`create-custom-log-source`.

AWS CLI

Come aggiungere un'origine personalizzata come origine Amazon Security Lake

L'esempio `create-custom-logsource` seguente aggiunge un'origine personalizzata come origine di Security Lake nell'account del provider di log e nella Regione designati.

```
aws securitylake create-custom-log-source \
    --source-name "VPC_FLOW" \
    --event-classes '["DNS_ACTIVITY", "NETWORK_ACTIVITY"]' \
    --configuration '{"crawlerConfiguration": {"roleArn": "arn:aws:glue:eu-
west-2:123456789012:crawler/E1WG1ZNPRXT0D4"},"providerIdentity": {"principal":
 "029189416600","externalId": "123456789012"}}' --region "us-east-1"
```

Output:

```
{
    "customLogSource": {
        "attributes": {
            "crawlerArn": "arn:aws:glue:eu-west-2:123456789012:crawler/
E1WG1ZNPRXT0D4",
```

```
            "databaseArn": "arn:aws:glue:eu-west-2:123456789012:database/
E1WG1ZNPRXT0D4",
            "tableArn": "arn:aws:glue:eu-west-2:123456789012:table/E1WG1ZNPRXT0D4"
        },
        "provider": {
            "location": "amzn-s3-demo-bucket--usw2-az1--x-s3",
            "roleArn": "arn:aws:iam::123456789012:role/AmazonSecurityLake-Provider-
testCustom2-eu-west-2"
        },
        "sourceName": "testCustom2"
        "sourceVersion": "2.0"
    }
}
```

Per ulteriori informazioni, consulta Aggiungere un'origine personalizzata nella Guida per l'utente di Amazon Security Lake.

- Per i dettagli sull'API, consulta CreateCustomLogSource AWS CLICommand Reference.

`create-data-lake-exception-subscription`

Il seguente esempio di codice mostra come utilizzare`create-data-lake-exception-subscription`.

AWS CLI

Come inviare notifiche relative alle eccezioni di Security Lake

L'esempio `create-data-lake-exception-subscription` seguente invia notifiche relative alle eccezioni di Security Lake all'account specificato tramite invio di SMS. Il messaggio di eccezione rimane attivo per il periodo di tempo specificato.

```
aws securitylake create-data-lake-exception-subscription \
    --notification-endpoint "123456789012" \
    --exception-time-to-live 30 \
    --subscription-protocol "sms"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Risoluzione dei problemi di Amazon Security Lake nella Guida per l'utente di Amazon Security Lake.

- Per i dettagli sull'API, consulta CreateDataLakeExceptionSubscription AWS CLICommand Reference.

`create-data-lake-organization-configuration`

Il seguente esempio di codice mostra come utilizzare`create-data-lake-organization-configuration`.

AWS CLI

Come configurare Security Lake nei nuovi account dell'organizzazione

L'esempio `create-data-lake-organization-configuration` seguente abilita Security Lake e la raccolta degli eventi di origine specificati e registra i logo dei nuovi account dell'organizzazione.

```
aws securitylake create-data-lake-organization-configuration \
    --auto-enable-new-account '[{"region":"us-east-1","sources":
[{"sourceName":"SH_FINDINGS","sourceVersion": "1.0"}]}]'
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Managing multiple account with AWS Organizations nella Amazon Security Lake User Guide.

- Per i dettagli sull'API, consulta CreateDataLakeOrganizationConfiguration AWS CLICommand Reference.

`create-data-lake`

Il seguente esempio di codice mostra come utilizzare`create-data-lake`.

AWS CLI

Esempio 1: come configurare il data lake in più Regioni

L'`create-data-lake`esempio seguente abilita Amazon Security Lake in più AWS regioni e configura il tuo data lake.

```
aws securitylake create-data-lake \
```

```
    --configurations '[{"encryptionConfiguration":
 {"kmsKeyId":"S3_MANAGED_KEY"},"region":"us-east-1","lifecycleConfiguration":
 {"expiration":{"days":365},"transitions":
[{"days":60,"storageClass":"ONEZONE_IA"}]}}, {"encryptionConfiguration":
 {"kmsKeyId":"S3_MANAGED_KEY"},"region":"us-east-2","lifecycleConfiguration":
 {"expiration":{"days":365},"transitions":
[{"days":60,"storageClass":"ONEZONE_IA"}]}}]' \
    --meta-store-manager-role-arn "arn:aws:iam:us-east-1:123456789012:role/service-
role/AmazonSecurityLakeMetaStoreManager"
```

Output:

```
{
    "dataLakes": [
        {
            "createStatus": "COMPLETED",
            "dataLakeArn": "arn:aws:securitylake:us-east-1:522481757177:data-lake/
default",
            "encryptionConfiguration": {
                "kmsKeyId": "S3_MANAGED_KEY"
            },
            "lifecycleConfiguration": {
                "expiration": {
                    "days": 365
                },
                "transitions": [
                    {
                        "days": 60,
                        "storageClass": "ONEZONE_IA"
                    }
                ]
            },
            "region": "us-east-1",
            "replicationConfiguration": {
                "regions": [
                    "ap-northeast-3"
                ],
                "roleArn": "arn:aws:securitylake:ap-northeast-3:522481757177:data-
lake/default"
            },
            "s3BucketArn": "arn:aws:s3:::aws-security-data-lake-us-east-1-
gnevt6s8z7bzby8oi3uiaysbr8v2ml",
            "updateStatus": {
```

```
                "exception": {},
                "requestId": "f20a6450-d24a-4f87-a6be-1d4c075a59c2",
                "status": "INITIALIZED"
            }
        },
        {
            "createStatus": "COMPLETED",
            "dataLakeArn": "arn:aws:securitylake:us-east-2:522481757177:data-lake/
default",
            "encryptionConfiguration": {
                "kmsKeyId": "S3_MANAGED_KEY"
            },
            "lifecycleConfiguration": {
                "expiration": {
                    "days": 365
                },
                "transitions": [
                    {
                        "days": 60,
                        "storageClass": "ONEZONE_IA"
                    }
                ]
            },
            "region": "us-east-2",
            "replicationConfiguration": {
                "regions": [
                    "ap-northeast-3"
                ],
                "roleArn": "arn:aws:securitylake:ap-northeast-3:522481757177:data-
lake/default"
            },
            "s3BucketArn": "arn:aws:s3:::aws-security-data-lake-us-east-2-
cehuifzl5rwmhm6m62h7zhvtseogr9",
            "updateStatus": {
                "exception": {},
                "requestId": "f20a6450-d24a-4f87-a6be-1d4c075a59c2",
                "status": "INITIALIZED"
            }
        }
    ]
}
```

Per ulteriori informazioni, consulta Guida introduttiva ad Amazon Security Lake nella Guida per l'utente di Amazon Security Lake.

Esempio 2: come configurare il data lake in un'unica Regione

L'`create-data-lake`esempio seguente abilita Amazon Security Lake in una singola AWS regione e configura il tuo data lake.

```
aws securitylake create-data-lake \
    --configurations '[{"encryptionConfiguration":
 {"kmsKeyId":"1234abcd-12ab-34cd-56ef-1234567890ab"},"region":"us-
east-2","lifecycleConfiguration": {"expiration":{"days":500},"transitions":
[{"days":30,"storageClass":"GLACIER"}]}}]' \
    --meta-store-manager-role-arn "arn:aws:iam:us-east-1:123456789012:role/service-
role/AmazonSecurityLakeMetaStoreManager"
```

Output:

```
{
    "dataLakes": [
        {
            "createStatus": "COMPLETED",
            "dataLakeArn": "arn:aws:securitylake:us-east-2:522481757177:data-lake/
default",
            "encryptionConfiguration": {
                "kmsKeyId": "1234abcd-12ab-34cd-56ef-1234567890ab"
            },
            "lifecycleConfiguration": {
                "expiration": {
                    "days": 500
                },
                "transitions": [
                    {
                        "days": 30,
                        "storageClass": "GLACIER"
                    }
                ]
            },
            "region": "us-east-2",
            "replicationConfiguration": {
                "regions": [
                    "ap-northeast-3"
                ],
```

```
                "roleArn": "arn:aws:securitylake:ap-northeast-3:522481757177:data-
lake/default"
            },
            "s3BucketArn": "arn:aws:s3:::aws-security-data-lake-us-east-2-
cehuifzl5rwmhm6m62h7zhvtseogr9",
            "updateStatus": {
                "exception": {},
                "requestId": "77702a53-dcbf-493e-b8ef-518e362f3003",
                "status": "INITIALIZED"
            }
        }
    ]
}
```

Per ulteriori informazioni, consulta Guida introduttiva ad Amazon Security Lake nella Guida per l'utente di Amazon Security Lake.

- Per i dettagli sull'API, consulta CreateDataLake AWS CLICommand Reference.

`create-subscriber-notification`

Il seguente esempio di codice mostra come utilizzare`create-subscriber-notification`.

AWS CLI

Come creare una notifica per gli abbonati

L'esempio `create-subscriber-notification` seguente mostra come specificare la notifica dell'abbonato per creare una notifica quando nuovi dati vengono scritti nel data lake.

```
aws securitylake create-subscriber-notification \
    --subscriber-id "12345ab8-1a34-1c34-1bd4-12345ab9012" \
    --configuration '{"httpsNotificationConfiguration":
 {"targetRoleArn":"arn:aws:iam::XXX:role/service-role/RoleName",
 "endpoint":"https://account-management.$3.$2.securitylake.aws.dev/v1/datalake"}}'
```

Output:

```
{
    "subscriberEndpoint": [
        "https://account-management.$3.$2.securitylake.aws.dev/v1/datalake"
    ]
```

```
}
```

Per ulteriori informazioni, consulta Gestione degli abbonati nella Guida per l'utente di Amazon Security Lake.

- Per i dettagli sull'API, consulta CreateSubscriberNotification AWS CLICommand Reference.

`create-subscriber`

Il seguente esempio di codice mostra come utilizzare`create-subscriber`.

AWS CLI

Esempio 1: come creare un abbonato con accesso ai dati

L'`create-subscriber`esempio seguente crea un sottoscrittore in Security Lake con accesso ai dati nella AWS regione corrente per l'identità di sottoscrittore specificata per un' AWS origine.

```
aws securitylake create-subscriber \
    --access-types "S3" \
    --sources '[{"awsLogSource": {"sourceName": "VPC_FLOW","sourceVersion":
 "2.0"}}]' \
    --subscriber-name 'opensearch-s3' \
    --subscriber-identity '{"principal": "029189416600","externalId":
 "123456789012"}'
```

Output:

```
{
    "subscriber": {
        "accessTypes": [
            "S3"
        ],
        "createdAt": "2024-07-17T19:08:26.787000+00:00",
        "roleArn": "arn:aws:iam::773172568199:role/AmazonSecurityLake-896f218b-
cfba-40be-a255-8b49a65d0407",
        "s3BucketArn": "arn:aws:s3:::aws-security-data-lake-us-east-1-
um632ufwpvxkyz0bc5hkb64atycnf3",
        "sources": [
            {
                "awsLogSource": {
                    "sourceName": "VPC_FLOW",
```

```
                    "sourceVersion": "2.0"
                }
            }
        ],
        "subscriberArn": "arn:aws:securitylake:us-
east-1:773172568199:subscriber/896f218b-cfba-40be-a255-8b49a65d0407",
        "subscriberId": "896f218b-cfba-40be-a255-8b49a65d0407",
        "subscriberIdentity": {
            "externalId": "123456789012",
            "principal": "029189416600"
        },
        "subscriberName": "opensearch-s3",
        "subscriberStatus": "ACTIVE",
        "updatedAt": "2024-07-17T19:08:27.133000+00:00"
    }
}
```

Per ulteriori informazioni, consulta Creare un abbonato con accesso ai dati nella Guida per l'utente di Amazon Security Lake.

Esempio 2: come creare un abbonato con accesso alle query

L'`create-subscriber`esempio seguente crea un sottoscrittore in Security Lake con accesso tramite query nella AWS regione corrente per l'identità di sottoscrittore specificata.

```
aws securitylake create-subscriber \
    --access-types "LAKEFORMATION" \
    --sources '[{"awsLogSource": {"sourceName": "VPC_FLOW","sourceVersion":
 "2.0"}}]' \
    --subscriber-name 'opensearch-s3' \
    --subscriber-identity '{"principal": "029189416600","externalId":
 "123456789012"}'
```

Output:

```
{
    "subscriber": {
        "accessTypes": [
            "LAKEFORMATION"
        ],
        "createdAt": "2024-07-18T01:05:55.853000+00:00",
        "resourceShareArn": "arn:aws:ram:us-east-1:123456789012:resource-
share/8c31da49-c224-4f1e-bb12-37ab756d6d8a",
```

```
        "resourceShareName": "LakeFormation-V2-NAMENAMENA-123456789012",
        "sources": [
            {
                "awsLogSource": {
                    "sourceName": "VPC_FLOW",
                    "sourceVersion": "2.0"
                }
            }
        ],
        "subscriberArn": "arn:aws:securitylake:us-east-1:123456789012:subscriber/
e762aabb-ce3d-4585-beab-63474597845d",
        "subscriberId": "e762aabb-ce3d-4585-beab-63474597845d",
        "subscriberIdentity": {
            "externalId": "123456789012",
            "principal": "029189416600"
        },
        "subscriberName": "opensearch-s3",
        "subscriberStatus": "ACTIVE",
        "updatedAt": "2024-07-18T01:05:58.393000+00:00"
    }
}
```

Per ulteriori informazioni, consulta Creare un abbonato con accesso alle query nella Guida per l'utente di Amazon Security Lake.

- Per i dettagli sull'API, vedere CreateSubscriberin AWS CLI Command Reference.

`delete-aws-log-source`

Il seguente esempio di codice mostra come utilizzare`delete-aws-log-source`.

AWS CLI

Per rimuovere un servizio supportato nativamente AWS .

L'esempio `delete-aws-logsource` seguente elimina i log di flusso VPC come origine di Security Lake negli account e nelle Regioni designati.

```
aws securitylake delete-aws-log-source \
    --sources '[{"regions": ["us-east-1"], "accounts": ["123456789012"],
 "sourceName": "SH_FINDINGS", "sourceVersion": "2.0"}]'
```

Output:

```
{
    "failed": [
        "123456789012"
    ]
}
```

Per ulteriori informazioni, consulta Rimuovere un AWS servizio come fonte nella Guida per l'utente di Amazon Security Lake.

- Per i dettagli sull'API, consulta DeleteAwsLogSource AWS CLICommand Reference.

`delete-custom-log-source`

Il seguente esempio di codice mostra come utilizzare`delete-custom-log-source`.

AWS CLI

Come rimuovere un'origine personalizzata.

L'esempio `delete-custom-logsource` seguente elimina un'origine personalizzata nell'account del provider di log designato nella Regione designata.

```
aws securitylake delete-custom-log-source \
    --source-name "CustomSourceName"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Eliminazione di una fonte personalizzata nella Guida per l'utente di Amazon Security Lake.

- Per i dettagli sull'API, consulta DeleteCustomLogSource AWS CLICommand Reference.

`delete-data-lake-organization-configuration`

Il seguente esempio di codice mostra come utilizzare`delete-data-lake-organization-configuration`.

AWS CLI

Come arrestare la raccolta automatica delle origini negli account membro

L'`delete-data-lake-organization-configuration`esempio seguente interrompe la raccolta automatica dei risultati del AWS Security Hub dai nuovi account membri che entrano a far parte dell'organizzazione. Solo l'account amministratore di Security Lake delegato può eseguire questo comando. Impedisce ai nuovi account membro di fornire automaticamente dati al data lake.

```
aws securitylake delete-data-lake-organization-configuration \
    --auto-enable-new-account '[{"region":"us-east-1","sources":
[{"sourceName":"SH_FINDINGS"}]}]'
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Managing multiple account with AWS Organizations nella Amazon Security Lake User Guide.

- Per i dettagli sull'API, consulta DeleteDataLakeOrganizationConfiguration AWS CLICommand Reference.

`delete-data-lake`

Il seguente esempio di codice mostra come utilizzare`delete-data-lake`.

AWS CLI

Come disabilitare il data lake

L'`delete-data-lake`esempio seguente disabilita il data lake nelle AWS regioni specificate. Nelle Regioni specificate, le origini non forniscono più dati al data lake. Per una distribuzione di Security Lake che utilizza AWS Organizations, solo l'amministratore delegato di Security Lake dell'organizzazione può disabilitare Security Lake per gli account dell'organizzazione.

```
aws securitylake delete-data-lake \
    --regions "ap-northeast-1" "eu-central-1"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Disabilitazione di Amazon Security Lake nella Guida per l'utente di Amazon Security Lake.

- Per i dettagli sull'API, vedere DeleteDataLakein AWS CLI Command Reference.

`delete-subscriber-notification`

Il seguente esempio di codice mostra come utilizzare`delete-subscriber-notification`.

AWS CLI

Come eliminare un abbonato da una notifica

L'esempio `delete-subscriber-notification` seguente mostra come eliminare la notifica di sottoscrizione per un abbonato specifico di Security Lake.

```
aws securitylake delete-subscriber-notification \
    --subscriber-id "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Gestione degli abbonati nella Guida per l'utente di Amazon Security Lake.

- Per i dettagli sull'API, consulta DeleteSubscriberNotification AWS CLICommand Reference.

`delete-subscriber`

Il seguente esempio di codice mostra come utilizzare`delete-subscriber`.

AWS CLI

Come eliminare un abbonato

L'esempio `delete-subscriber` seguente mostra come rimuovere un abbonato se non desideri più che questo utilizzi i dati di Security Lake.

```
aws securitylake delete-subscriber \
    --subscriber-id "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Gestione degli abbonati nella Guida per l'utente di Amazon Security Lake.

- Per i dettagli sull'API, consulta DeleteSubscriber AWS CLICommand Reference.

`get-data-lake-exception-subscription`

Il seguente esempio di codice mostra come utilizzare`get-data-lake-exception-subscription`.

AWS CLI

Come ottenere dettagli su un abbonamento eccezionale

L'esempio `get-data-lake-exception-subscription` seguente fornisce dettagli su un abbonamento alle eccezioni di Security Lake. In questo esempio, l'utente dell' AWS account specificato riceve una notifica degli errori tramite l'invio di SMS. Il messaggio di eccezione rimane attivo nell'account per il periodo di tempo specificato. Un abbonamento alle eccezioni notifica a un utente di Security Lake un errore tramite il protocollo preferito del richiedente.

```
aws securitylake get-data-lake-exception-subscription
```

Output:

```
{
    "exceptionTimeToLive": 30,
    "notificationEndpoint": "123456789012",
    "subscriptionProtocol": "sms"
}
```

Per ulteriori informazioni, consulta Risoluzione dei problemi con gli stati del data lake nella Guida per l'utente di Amazon Security Lake.

- Per i dettagli sull'API, consulta GetDataLakeExceptionSubscription AWS CLICommand Reference.

`get-data-lake-organization-configuration`

Il seguente esempio di codice mostra come utilizzare`get-data-lake-organization-configuration`.

AWS CLI

Come ottenere dettagli sulla configurazione dei nuovi account dell'organizzazione

L'esempio `get-data-lake-organization-configuration` seguente recupera i dettagli sui log di origine che i nuovi account dell'organizzazione invieranno dopo l'onboarding su Amazon Security Lake.

```
aws securitylake get-data-lake-organization-configuration
```

Output:

```
{
    "autoEnableNewAccount": [
        {
            "region": "us-east-1",
            "sources": [
                {
                    "sourceName": "VPC_FLOW",
                    "sourceVersion": "1.0"
                },
                {
                    "sourceName": "ROUTE53",
                    "sourceVersion": "1.0"
                },
                {
                    "sourceName": "SH_FINDINGS",
                    "sourceVersion": "1.0"
                }
            ]
        }
    ]
}
```

Per ulteriori informazioni, consulta Managing multiple account with AWS Organizations nella Amazon Security Lake User Guide.

- Per i dettagli sull'API, consulta GetDataLakeOrganizationConfiguration AWS CLICommand Reference.

`get-data-lake-sources`

Il seguente esempio di codice mostra come utilizzare`get-data-lake-sources`.

AWS CLI

Come ottenere lo stato della raccolta dei log

L'`get-data-lake-sources`esempio seguente ottiene un'istantanea della raccolta dei registri per l'account specificato nella AWS regione corrente. L'account ha Amazon Security Lake abilitato.

```
aws securitylake get-data-lake-sources \
    --accounts "123456789012"
```

Output:

```
{
    "dataLakeSources": [
        {
            "account": "123456789012",
            "sourceName": "SH_FINDINGS",
            "sourceStatuses": [
                {
                    "resource": "vpc-1234567890abcdef0",
                    "status": "COLLECTING"
                }
            ]
        },
        {
            "account": "123456789012",
            "sourceName": "VPC_FLOW",
            "sourceStatuses": [
                {
                    "resource": "vpc-1234567890abcdef0",
                    "status": "NOT_COLLECTING"
                }
            ]
        },
        {
            "account": "123456789012",
            "sourceName": "LAMBDA_EXECUTION",
            "sourceStatuses": [
                {
                    "resource": "vpc-1234567890abcdef0",
                    "status": "COLLECTING"
                }
```

```
            ]
        },
        {
            "account": "123456789012",
            "sourceName": "ROUTE53",
            "sourceStatuses": [
                {
                    "resource": "vpc-1234567890abcdef0",
                    "status": "COLLECTING"
                }
            ]
        },
        {
            "account": "123456789012",
            "sourceName": "CLOUD_TRAIL_MGMT",
            "sourceStatuses": [
                {
                    "resource": "vpc-1234567890abcdef0",
                    "status": "COLLECTING"
                }
            ]
        }
    ],
    "dataLakeArn": null
}
```

Per ulteriori informazioni, consulta la sezione Raccolta di dati dai AWS servizi nella Guida per l'utente di Amazon Security Lake.

- Per i dettagli sull'API, consulta GetDataLakeSources AWS CLICommand Reference.

`get-subscriber`

Il seguente esempio di codice mostra come utilizzare`get-subscriber`.

AWS CLI

Come recuperare le informazioni sull'abbonamento

L'esempio `get-subscriber` seguente recupera le informazioni sull'abbonamento per l'abbonato Securiy Lake specificato.

```
aws securitylake get-subscriber \
```

```
    --subscriber-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Output:


```
{
    "subscriber": {
        "accessTypes": [
            "LAKEFORMATION"
        ],
        "createdAt": "2024-04-19T15:19:44.421803+00:00",
        "resourceShareArn": "arn:aws:ram:eu-west-2:123456789012:resource-share/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
        "resourceShareName": "LakeFormation-V3-TKJGBHCKTZ-123456789012",
        "sources": [
            {
                "awsLogSource": {
                    "sourceName": "LAMBDA_EXECUTION",
                    "sourceVersion": "1.0"
                }
            },
            {
                "awsLogSource": {
                    "sourceName": "EKS_AUDIT",
                    "sourceVersion": "2.0"
                }
            },
            {
                "awsLogSource": {
                    "sourceName": "ROUTE53",
                    "sourceVersion": "1.0"
                }
            },
            {
                "awsLogSource": {
                    "sourceName": "SH_FINDINGS",
                    "sourceVersion": "1.0"
                }
            },
            {
                "awsLogSource": {
                    "sourceName": "VPC_FLOW",
                    "sourceVersion": "1.0"
                }
```

```
            },
            {
                "customLogSource": {
                    "attributes": {
                        "crawlerArn": "arn:aws:glue:eu-west-2:123456789012:crawler/
testCustom2",
                        "databaseArn": "arn:aws:glue:eu-
west-2:123456789012:database/amazon_security_lake_glue_db_eu_west_2",
                        "tableArn": "arn:aws:glue:eu-west-2:123456789012:table/
amazon_security_lake_table_eu_west_2_ext_testcustom2"
                    },
                    "provider": {
                        "location": "s3://aws-security-data-lake-eu-
west-2-8ugsus4ztnsfpjbldwbgf4vge98av9/ext/testCustom2/",
                        "roleArn": "arn:aws:iam::123456789012:role/
AmazonSecurityLake-Provider-testCustom2-eu-west-2"
                    },
                    "sourceName": "testCustom2"
                }
            },
            {
                "customLogSource": {
                    "attributes": {
                        "crawlerArn": "arn:aws:glue:eu-west-2:123456789012:crawler/
TestCustom",
                        "databaseArn": "arn:aws:glue:eu-
west-2:123456789012:database/amazon_security_lake_glue_db_eu_west_2",
                        "tableArn": "arn:aws:glue:eu-west-2:123456789012:table/
amazon_security_lake_table_eu_west_2_ext_testcustom"
                    },
                    "provider": {
                        "location": "s3://aws-security-data-lake-eu-
west-2-8ugsus4ztnsfpjbldwbgf4vge98av9/ext/TestCustom/",
                        "roleArn": "arn:aws:iam::123456789012:role/
AmazonSecurityLake-Provider-TestCustom-eu-west-2"
                    },
                    "sourceName": "TestCustom"
                }
            }
        ],
        "subscriberArn": "arn:aws:securitylake:eu-west-2:123456789012:subscriber/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
        "subscriberId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
        "subscriberIdentity": {
```

```
            "externalId": "123456789012",
            "principal": "123456789012"
        },
        "subscriberName": "test",
        "subscriberStatus": "ACTIVE",
        "updatedAt": "2024-04-19T15:19:55.230588+00:00"
    }
}
```

Per ulteriori informazioni, consulta Gestione degli abbonati nella Guida per l'utente di Amazon Security Lake.

- Per i dettagli sull'API, consulta GetSubscriber AWS CLICommand Reference.

`list-data-lake-exceptions`

Il seguente esempio di codice mostra come utilizzare`list-data-lake-exceptions`.

AWS CLI

Come elencare i problemi che riguardano il tuo data lake

L'`list-data-lake-exceptions`esempio seguente elenca i problemi che hanno interessato il data lake negli ultimi 14 giorni nelle AWS regioni specificate.

```
aws securitylake list-data-lake-exceptions \
    --regions "us-east-1" "eu-west-3"
```

Output:

```
{
    "exceptions": [
        {
            "exception": "The account does not have the required role permissions.
 Update your role permissions to use the new data source version.",
            "region": "us-east-1",
            "timestamp": "2024-02-29T12:24:15.641725+00:00"
        },
        {
            "exception": "The account does not have the required role permissions.
 Update your role permissions to use the new data source version.",
            "region": "eu-west-3",
```

```
            "timestamp": "2024-02-29T12:24:15.641725+00:00"
        }
    ]
}
```

Per ulteriori informazioni, consulta Risoluzione dei problemi di Amazon Security Lake nella Guida per l'utente di Amazon Security Lake.

- Per i dettagli sull'API, consulta ListDataLakeExceptions AWS CLICommand Reference.

`list-data-lakes`

Il seguente esempio di codice mostra come utilizzare`list-data-lakes`.

AWS CLI

Come elencare l'oggetto di configurazione Security Lake

L'`list-data-lakes`esempio seguente elenca l'oggetto di configurazione Amazon Security Lake per la AWS regione specificata. È possibile utilizzare questo comando per determinare se Security Lake è abilitato in una o più Regioni specificate.

```
aws securitylake list-data-lakes \
    --regions "us-east-1"
```

Output:

```
{
    "dataLakes": [
        {
            "createStatus": "COMPLETED",
            "dataLakeArn": "arn:aws:securitylake:us-east-1:123456789012:data-lake/
default",
            "encryptionConfiguration": {
                "kmsKeyId": "S3_MANAGED_KEY"
            },
            "lifecycleConfiguration": {
                "expiration": {
                    "days": 365
                },
                "transitions": [
```

```
                    {
                        "days": 60,
                        "storageClass": "ONEZONE_IA"
                    }
                ]
            },
            "region": "us-east-1",
            "replicationConfiguration": {
                "regions": [
                    "ap-northeast-3"
                ],
                "roleArn": "arn:aws:securitylake:ap-northeast-3:123456789012:data-
lake/default"
            },
            "s3BucketArn": "arn:aws:s3:::aws-security-data-lake-us-
east-1-1234567890abcdef0",
            "updateStatus": {
                "exception": {
                    "code": "software.amazon.awssdk.services.s3.model.S3Exception",
                    "reason": ""
                },
                "requestId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
                "status": "FAILED"
            }
        }
    ]
}
```

Per ulteriori informazioni, consulta Verifica dello stato della regione nella Guida per l'utente di Amazon Security Lake.

- Per i dettagli sull'API, consulta ListDataLakes AWS CLICommand Reference.

`list-log-sources`

Il seguente esempio di codice mostra come utilizzare`list-log-sources`.

AWS CLI

Come recuperare le origini dei log di Amazon Security Lake

L'esempio `list-log-sources` seguente elenca le sorgenti di log di Amazon Security Lake in un account specificato.

```
aws securitylake list-log-sources \
    --accounts "123456789012"
```

Output:

```
{
    "account": "123456789012",
    "region": "xy-region-1",
    "sources": [
        {
               "awsLogSource": {
                "sourceName": "VPC_FLOW",
                "sourceVersion": "2.0"
            }
        },
        {
            "awsLogSource": {
                "sourceName": "SH_FINDINGS",
                "sourceVersion": "2.0"
            }
        }
    ]
}
```

Per ulteriori informazioni, consulta Gestione delle fonti nella Guida per l'utente di Amazon Security Lake.

- Per i dettagli sull'API, consulta ListLogSources AWS CLICommand Reference.

`list-subscribers`

Il seguente esempio di codice mostra come utilizzare`list-subscribers`.

AWS CLI

Come recuperare le origini degli abbonati di Amazon Security Lake

L'esempio `list-subscribers` seguente elenca tutti gli abbonati a Amazon Security Lake in un account specifico.

```
aws securitylake list-subscribers
```

Output:

```
{
    "subscribers": [
        {
            "accessTypes": [
                "S3"
            ],
            "createdAt": "2024-06-04T15:02:28.921000+00:00",
            "roleArn": "arn:aws:iam::123456789012:role/AmazonSecurityLake-
E1WG1ZNPRXT0D4",
            "s3BucketArn": "amzn-s3-demo-bucket--usw2-az1--x-s3",
            "sources": [
                {
                    "awsLogSource": {
                        "sourceName": "CLOUD_TRAIL_MGMT",
                        "sourceVersion": "2.0"
                    }
                },
                {
                    "awsLogSource": {
                        "sourceName": "LAMBDA_EXECUTION",
                        "sourceVersion": "1.0"
                    }
                },
                {
                    "customLogSource": {
                        "attributes": {
                            "crawlerArn": "arn:aws:glue:eu-
west-2:123456789012:crawler/E1WG1ZNPRXT0D4",
                            "databaseArn": "arn:aws:glue:eu-
west-2:123456789012:database/E1WG1ZNPRXT0D4",
                            "tableArn": "arn:aws:glue:eu-west-2:123456789012:table/
E1WG1ZNPRXT0D4"
                        },
                        "provider": {
                            "location": "amzn-s3-demo-bucket--usw2-az1--x-s3",
                            "roleArn": "arn:aws:iam::123456789012:role/
AmazonSecurityLake-E1WG1ZNPRXT0D4"
                        },
                        "sourceName": "testCustom2"
                    }
                }
            ],
```

```
            "subscriberArn": "arn:aws:securitylake:eu-
west-2:123456789012:subscriber/E1WG1ZNPRXT0D4",
            "subscriberEndpoint": "arn:aws:sqs:eu-
west-2:123456789012:AmazonSecurityLake-a1b2c3d4-5678-90ab-cdef-EXAMPLE11111-Main-
Queue",
            "subscriberId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
            "subscriberIdentity": {
                "externalId": "ext123456789012",
                "principal": "123456789012"
            },
            "subscriberName": "Test",
            "subscriberStatus": "ACTIVE",
            "updatedAt": "2024-06-04T15:02:35.617000+00:00"
        }
    ]
}
```

Per ulteriori informazioni, consulta Gestione degli abbonati nella Guida per l'utente di Amazon Security Lake.

- Per i dettagli sull'API, consulta ListSubscribers AWS CLICommand Reference.

`list-tags-for-resource`

Il seguente esempio di codice mostra come utilizzare`list-tags-for-resource`.

AWS CLI

Come elencare i tag di una risorsa esistente

L'esempio `list-tags-for-resource` seguente elenca i tag associati all'abbonato ad Amazon Security Lake specificato. In questo esempio, la chiave del tag Proprietario ha un valore di tag associato. È possibile utilizzare questa operazione per elencare i tag anche per altre risorse di Security Lake esistenti.

```
aws securitylake list-tags-for-resource \
    --resource-arn "arn:aws:securitylake:us-
east-1:123456789012:subscriber/1234abcd-12ab-34cd-56ef-1234567890ab"
```

Output:

```
{
```

```
    "tags": [
        {
            "key": "Environment",
            "value": "Cloud"
        },
        {
            "key": "CostCenter",
            "value": "12345"
        },
        {
            "key": "Owner",
            "value": ""
        }
    ]
}
```

Per ulteriori informazioni, consulta Associazione di tag alle risorse di Amazon Security Lake nella Guida per l'utente di Amazon Security Lake.

- Per i dettagli sull'API, consulta ListTagsForResource AWS CLICommand Reference.

`register-data-lake-delegated-administrator`

Il seguente esempio di codice mostra come utilizzare`register-data-lake-delegated-administrator`.

AWS CLI

Come designare l'amministratore delegato

L'`register-data-lake-delegated-administrator`esempio seguente designa l' AWS account specificato come amministratore delegato di Amazon Security Lake.

```
aws securitylake register-data-lake-delegated-administrator \
    --account-id 123456789012
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Managing multiple account with AWS Organizations nella Amazon Security Lake User Guide.

- Per i dettagli sull'API, consulta RegisterDataLakeDelegatedAdministrator AWS CLICommand Reference.

`tag-resource`

Il seguente esempio di codice mostra come utilizzare`tag-resource`.

AWS CLI

Come aggiungere tag a una risorsa esistente

L'esempio `tag-resource` seguente aggiunge un tag a una risorsa di abbonato esistente. Per creare una nuova risorsa e aggiungervi uno o più tag, non utilizzare questa operazione. Utilizza invece l'operazione Crea appropriata per il tipo di risorsa da creare.

```
aws securitylake tag-resource \
    --resource-arn "arn:aws:securitylake:us-
east-1:123456789012:subscriber/1234abcd-12ab-34cd-56ef-1234567890ab" \
    --tags key=Environment,value=Cloud
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Associazione di tag alle risorse di Amazon Security Lake nella Guida per l'utente di Amazon Security Lake.

- Per i dettagli sull'API, consulta TagResource AWS CLICommand Reference.

`untag-resource`

Il seguente esempio di codice mostra come utilizzare`untag-resource`.

AWS CLI

Come rimuovere i tag da una risorsa esistente

L'esempio `untag-resource` seguente rimuove i tag specificati da una risorsa di un abbonato esistente.

```
aws securitylake untag-resource \
    --resource-arn "arn:aws:securitylake:us-
east-1:123456789012:subscriber/1234abcd-12ab-34cd-56ef-1234567890ab" \
    --tags Environment Owner
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Associazione di tag alle risorse di Amazon Security Lake nella Guida per l'utente di Amazon Security Lake.

- Per i dettagli sull'API, consulta UntagResource AWS CLICommand Reference.

`update-data-lake-exception-subscription`

Il seguente esempio di codice mostra come utilizzare`update-data-lake-exception-subscription`.

AWS CLI

Come aggiornare l'abbonamento alle notifiche per le eccezioni di Security Lake

L'esempio `update-data-lake-exception-subscription` seguente aggiorna la sottoscrizione alle notifiche che notifica agli utenti le eccezioni di Security Lake.

```
aws securitylake update-data-lake-exception-subscription \
    --notification-endpoint "123456789012" \
    --exception-time-to-live 30 \
    --subscription-protocol "email"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Risoluzione dei problemi di Amazon Security Lake nella Guida per l'utente di Amazon Security Lake.

- Per i dettagli sull'API, consulta UpdateDataLakeExceptionSubscription AWS CLICommand Reference.

`update-data-lake`

Il seguente esempio di codice mostra come utilizzare`update-data-lake`.

AWS CLI

Esempio 1: come aggiornare le impostazioni del data lake

L'esempio `update-data-lake` seguente aggiorna le impostazioni del tuo data lake Amazon Security Lake. È possibile utilizzare questa operazione per specificare le impostazioni di crittografia dei dati, archiviazione e Regione di rollup.

```
aws securitylake update-data-lake \
    --configurations '[{"encryptionConfiguration":
 {"kmsKeyId":"S3_MANAGED_KEY"},"region":"us-east-1","lifecycleConfiguration":
 {"expiration":{"days":365},"transitions":
[{"days":60,"storageClass":"ONEZONE_IA"}]}}, {"encryptionConfiguration":
 {"kmsKeyId":"S3_MANAGED_KEY"},"region":"us-east-2","lifecycleConfiguration":
 {"expiration":{"days":365},"transitions":
[{"days":60,"storageClass":"ONEZONE_IA"}]}}]' \
    --meta-store-manager-role-arn "arn:aws:iam:us-east-1:123456789012:role/service-
role/AmazonSecurityLakeMetaStoreManager"
```

Output:

```
{
    "dataLakes": [
        {
            "createStatus": "COMPLETED",
            "dataLakeArn": "arn:aws:securitylake:us-east-1:522481757177:data-lake/
default",
            "encryptionConfiguration": {
                "kmsKeyId": "S3_MANAGED_KEY"
            },
            "lifecycleConfiguration": {
                "expiration": {
                    "days": 365
                },
                "transitions": [
                    {
                        "days": 60,
                        "storageClass": "ONEZONE_IA"
                    }
                ]
            },
            "region": "us-east-1",
            "replicationConfiguration": {
                "regions": [
                    "ap-northeast-3"
                ],
                "roleArn": "arn:aws:securitylake:ap-northeast-3:522481757177:data-
lake/default"
            },
            "s3BucketArn": "arn:aws:s3:::aws-security-data-lake-us-east-1-
gnevt6s8z7bzby8oi3uiaysbr8v2ml",
```

```
            "updateStatus": {
                "exception": {},
                "requestId": "f20a6450-d24a-4f87-a6be-1d4c075a59c2",
                "status": "INITIALIZED"
            }
        },
        {
            "createStatus": "COMPLETED",
            "dataLakeArn": "arn:aws:securitylake:us-east-2:522481757177:data-lake/
default",
            "encryptionConfiguration": {
                "kmsKeyId": "S3_MANAGED_KEY"
            },
            "lifecycleConfiguration": {
                "expiration": {
                    "days": 365
                },
                "transitions": [
                    {
                        "days": 60,
                        "storageClass": "ONEZONE_IA"
                    }
                ]
            },
            "region": "us-east-2",
            "replicationConfiguration": {
                "regions": [
                    "ap-northeast-3"
                ],
                "roleArn": "arn:aws:securitylake:ap-northeast-3:522481757177:data-
lake/default"
            },
            "s3BucketArn": "arn:aws:s3:::aws-security-data-lake-us-east-2-
cehuifzl5rwmhm6m62h7zhvtseogr9",
            "updateStatus": {
                "exception": {},
                "requestId": "f20a6450-d24a-4f87-a6be-1d4c075a59c2",
                "status": "INITIALIZED"
            }
        }
    ]
}
```

Per ulteriori informazioni, consulta Guida introduttiva ad Amazon Security Lake nella Guida per l'utente di Amazon Security Lake.

Esempio 2: come configurare il data lake in un'unica Regione

L'`create-data-lake`esempio seguente abilita Amazon Security Lake in una singola AWS regione e configura il tuo data lake.

```
aws securitylake create-data-lake \
    --configurations '[{"encryptionConfiguration":
 {"kmsKeyId":"1234abcd-12ab-34cd-56ef-1234567890ab"},"region":"us-
east-2","lifecycleConfiguration": {"expiration":{"days":500},"transitions":
[{"days":30,"storageClass":"GLACIER"}]}}]' \
    --meta-store-manager-role-arn "arn:aws:iam:us-east-1:123456789012:role/service-
role/AmazonSecurityLakeMetaStoreManager"
```

Output:

```
{
    "dataLakes": [
        {
            "createStatus": "COMPLETED",
            "dataLakeArn": "arn:aws:securitylake:us-east-2:522481757177:data-lake/
default",
            "encryptionConfiguration": {
                "kmsKeyId": "1234abcd-12ab-34cd-56ef-1234567890ab"
            },
            "lifecycleConfiguration": {
                "expiration": {
                    "days": 500
                },
                "transitions": [
                    {
                        "days": 30,
                        "storageClass": "GLACIER"
                    }
                ]
            },
            "region": "us-east-2",
            "replicationConfiguration": {
                "regions": [
                    "ap-northeast-3"
                ],
```

```
                "roleArn": "arn:aws:securitylake:ap-northeast-3:522481757177:data-
lake/default"
            },
            "s3BucketArn": "arn:aws:s3:::aws-security-data-lake-us-east-2-
cehuifzl5rwmhm6m62h7zhvtseogr9",
            "updateStatus": {
                "exception": {},
                "requestId": "77702a53-dcbf-493e-b8ef-518e362f3003",
                "status": "INITIALIZED"
            }
        }
    ]
}
```

Per ulteriori informazioni, consulta Guida introduttiva ad Amazon Security Lake nella Guida per l'utente di Amazon Security Lake.

- Per i dettagli sull'API, consulta UpdateDataLake AWS CLICommand Reference.

`update-subscriber-notification`

Il seguente esempio di codice mostra come utilizzare`update-subscriber-notification`.

AWS CLI

Come aggiornare una notifica dell'abbonato

L'esempio `update-subscriber-notification` seguente mostra come aggiornare il metodo di notifica per un abbonato.

```
aws securitylake update-subscriber-notification \
    --subscriber-id "12345ab8-1a34-1c34-1bd4-12345ab9012" \
    --configuration '{"httpsNotificationConfiguration":
 {"targetRoleArn":"arn:aws:iam::XXX:role/service-role/RoleName",
 "endpoint":"https://account-management.$3.$2.securitylake.aws.dev/v1/datalake"}}'
```

Output:

```
{
    "subscriberEndpoint": [
        "https://account-management.$3.$2.securitylake.aws.dev/v1/datalake"
    ]
```

```
}
```

Per ulteriori informazioni, consulta Gestione degli abbonati nella Guida per l'utente di Amazon Security Lake.

- Per i dettagli sull'API, consulta UpdateSubscriberNotification AWS CLICommand Reference.

`update-subscriber`

Il seguente esempio di codice mostra come utilizzare`update-subscriber`.

AWS CLI

Come aggiornare un abbonato Amazon Security Lake.

L'esempio `update-subscriber` seguente aggiorna le fonti di accesso ai dati Security Lake per uno specifico abbonato a Security Lake.

```
aws securitylake update-subscriber \
    --subscriber-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Output:

```
{
    "subscriber": {
        "accessTypes": [
            "LAKEFORMATION"
        ],
        "createdAt": "2024-04-19T15:19:44.421803+00:00",
        "resourceShareArn": "arn:aws:ram:eu-west-2:123456789012:resource-share/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
        "resourceShareName": "LakeFormation-V3-TKJGBHCKTZ-123456789012",
        "sources": [
            {
                "awsLogSource": {
                    "sourceName": "LAMBDA_EXECUTION",
                    "sourceVersion": "1.0"
                }
            },
            {
                "awsLogSource": {
                    "sourceName": "EKS_AUDIT",
                    "sourceVersion": "2.0"
```

```
                }
            },
            {
                "awsLogSource": {
                    "sourceName": "ROUTE53",
                    "sourceVersion": "1.0"
                }
            },
            {
                "awsLogSource": {
                    "sourceName": "SH_FINDINGS",
                    "sourceVersion": "1.0"
                }
            },
            {
                "awsLogSource": {
                    "sourceName": "VPC_FLOW",
                    "sourceVersion": "1.0"
                }
            },
            {
                "customLogSource": {
                    "attributes": {
                        "crawlerArn": "arn:aws:glue:eu-west-2:123456789012:crawler/
E1WG1ZNPRXT0D4",
                        "databaseArn": "arn:aws:glue:eu-
west-2:123456789012:database/E1WG1ZNPRXT0D4",
                        "tableArn": "arn:aws:glue:eu-west-2:123456789012:table/
E1WG1ZNPRXT0D4"
                    },
                    "provider": {
                        "location": "amzn-s3-demo-bucket--usw2-az1--x-s3",
                        "roleArn": "arn:aws:iam::123456789012:role/
AmazonSecurityLake-E1WG1ZNPRXT0D4"
                    },
                    "sourceName": "testCustom2"
                }
            }
        ],
        "subscriberArn": "arn:aws:securitylake:eu-west-2:123456789012:subscriber/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
        "subscriberId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
        "subscriberIdentity": {
            "externalId": "123456789012",
```

```
            "principal": "123456789012"
        },
        "subscriberName": "test",
        "subscriberStatus": "ACTIVE",
        "updatedAt": "2024-07-18T20:47:37.098000+00:00"
    }
}
```

Per ulteriori informazioni, consulta Gestione degli abbonati nella Guida per l'utente di Amazon Security Lake.

- Per i dettagli sull'API, consulta UpdateSubscriber AWS CLICommand Reference.

AWS Serverless Application Repository esempi utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with AWS Serverless Application Repository.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`put-application-policy`

Il seguente esempio di codice mostra come utilizzare`put-application-policy`.

AWS CLI

Esempio 1: come condividere pubblicamente un'applicazione

Quanto segue `put-application-policy` condivide un'applicazione pubblicamente, in modo che chiunque possa trovare e distribuire l'applicazione nel AWS Serverless Application Repository.

```
aws serverlessrepo put-application-policy \
    --application-id arn:aws:serverlessrepo:us-east-1:123456789012:applications/my-test-application \
    --statements Principals='*',Actions=Deploy
```

Output:

```
{
    "Statements": [
        {
            "Actions": [
                "Deploy"
            ],
            "Principals": [
                ""
            ],
            "StatementId": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE"
        }
    ]
}
```

Esempio 2: come condividere privatamente un'applicazione

Quanto segue `put-application-policy` condivide un'applicazione privatamente, quindi solo AWS account specifici possono trovare e distribuire l'applicazione nel AWS Serverless Application Repository.

```
aws serverlessrepo put-application-policy \
    --application-id arn:aws:serverlessrepo:us-east-1:123456789012:applications/my-test-application \
    --statements Principals=111111111111,222222222222,Actions=Deploy
```

Output:

```
{
    "Statements": [
        {
            "Actions": [
                "Deploy"
            ],
            "Principals": [
```

```
                "111111111111",
                "222222222222"
            ],
            "StatementId": "a1b2c3d4-5678-90ab-cdef-11111EXAMPLE"
        }
    ]
}
```

Per ulteriori informazioni, consulta Condivisione di un'applicazione tramite la console nella Guida per gli sviluppatori del Repository dell'applicazione serverless AWS

- Per i dettagli sull'API, consulta PutApplicationPolicyCommand Reference.AWS CLI

Esempi di Service Catalog utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with Service Catalog.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`accept-portfolio-share`

Il seguente esempio di codice mostra come utilizzare`accept-portfolio-share`.

AWS CLI

Come accettare una condivisione di portafoglio

L'esempio `accept-portfolio-share` seguente accetta un'offerta, effettuata da un altro utente, per condividere il portafoglio specificato.

```
aws servicecatalog accept-portfolio-share \
    --portfolio-id port-2s6wuabcdefghijk
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta AcceptPortfolioShare AWS CLICommand Reference.

associate-principal-with-portfolio

Il seguente esempio di codice mostra come utilizzare`associate-principal-with-portfolio`.

AWS CLI

Come associare un principale a un portafoglio

L'esempio `associate-principal-with-portfolio` seguente associa un utente al portafoglio specificato.

```
aws servicecatalog associate-principal-with-portfolio \
    --portfolio-id port-2s6abcdefwdh4 \
    --principal-arn arn:aws:iam::123456789012:user/usertest \
    --principal-type IAM
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta AssociatePrincipalWithPortfolio AWS CLICommand Reference.

associate-product-with-portfolio

Il seguente esempio di codice mostra come utilizzare`associate-product-with-portfolio`.

AWS CLI

Come associare un prodotto a un portafoglio

L'esempio `associate-product-with-portfolio` seguente associa il prodotto indicato al portafoglio specificato.

```
aws servicecatalog associate-product-with-portfolio
     --product-id prod-3p5abcdef3oyk
     --portfolio-id port-2s6abcdef5wdh4
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta AssociateProductWithPortfolio AWS CLICommand Reference.

`associate-tag-option-with-resource`

Il seguente esempio di codice mostra come utilizzare`associate-tag-option-with-resource`.

AWS CLI

Associare un a TagOption a una risorsa

L'`associate-tag-option-with-resource`esempio seguente associa la risorsa specificata TagOption alla risorsa specificata.

```
aws servicecatalog associate-tag-option-with-resource \
    --resource-id port-2s6abcdq5wdh4 \
    --tag-option-id tag-p3abc2pkpz5qc
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, vedere AssociateTagOptionWithResourcein AWS CLI Command Reference.

`copy-product`

Il seguente esempio di codice mostra come utilizzare`copy-product`.

AWS CLI

Come copiare un prodotto

L'esempio `copy-product` seguente crea una copia del prodotto specificato, utilizzando un file JSON per passare i parametri.

```
aws servicecatalog copy-product --cli-input-json file://copy-product-input.json
```

Contenuto di `copy-product-input.json`:

```
{
    "SourceProductArn": "arn:aws:catalog:us-west-2:123456789012:product/prod-
tcabcd3syn2xy",
```

```
    "TargetProductName": "copy-of-myproduct",
    "CopyOptions": [
        "CopyTags"
    ]
}
```

Output:

```
{
    "CopyProductToken": "copyproduct-abc5defgjkdji"
}
```

- Per i dettagli sull'API, consulta CopyProduct AWS CLICommand Reference.

`create-portfolio-share`

Il seguente esempio di codice mostra come utilizzare`create-portfolio-share`.

AWS CLI

Come condividere un portafoglio con un account

L'esempio `create-portfolio-share` seguente condivide il portafoglio specificato con l'account specificato.

```
aws servicecatalog create-portfolio-share \
    --portfolio-id port-2s6abcdef5wdh4 \
    --account-id 794123456789
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta CreatePortfolioShare AWS CLICommand Reference.

`create-portfolio`

Il seguente esempio di codice mostra come utilizzare`create-portfolio`.

AWS CLI

Come creare un portafoglio

L'esempio `create-portfolio` seguente crea un portafoglio.

```
aws servicecatalog create-portfolio  \
    --provider-name my-provider \
    --display-name my-portfolio
```

Output:

```
{
    "PortfolioDetail": {
        "ProviderName": "my-provider",
        "DisplayName": "my-portfolio",
        "CreatedTime": 1571337221.555,
        "ARN": "arn:aws:catalog:us-east-2:123456789012:portfolio/
port-2s6xmplq5wdh4",
        "Id": "port-2s6xmplq5wdh4"
    }
}
```

- Per i dettagli sull'API, consulta CreatePortfolio AWS CLICommand Reference.

`create-product`

Il seguente esempio di codice mostra come utilizzare`create-product`.

AWS CLI

Come creare un prodotto

L'esempio `create-product` seguente crea un prodotto, utilizzando un file JSON per passare i parametri.

```
aws servicecatalog create-product \
    --cli-input-json file://create-product-input.json
```

Contenuto di `create-product-input.json`:

```
{
    "AcceptLanguage": "en",
    "Name": "test-product",
    "Owner": "test-owner",
    "Description": "test-description",
    "Distributor": "test-distributor",
```

```
    "SupportDescription": "test-support",
    "SupportEmail": "test@amazon.com",
    "SupportUrl": "https://aws.amazon.com",
    "ProductType": "CLOUD_FORMATION_TEMPLATE",
    "Tags": [
        {
            "Key": "region",
            "Value": "us-east-1"
        }
    ],
    "ProvisioningArtifactParameters": {
        "Name": "test-version-name",
        "Description": "test-version-description",
        "Info": {
            "LoadTemplateFromURL": "https://s3-us-west-1.amazonaws.com/
cloudformation-templates-us-west-1/my-cfn-template.template"
        },
        "Type": "CLOUD_FORMATION_TEMPLATE"
    }
}
```

Output:

```
{
    "Tags": [
        {
            "Key": "region",
            "Value": "us-east-1"
        }
    ],
    "ProductViewDetail": {
        "CreatedTime": 1576025036.0,
        "ProductARN": "arn:aws:catalog:us-west-2:1234568542028:product/
prod-3p5abcdef3oyk",
        "Status": "CREATED",
        "ProductViewSummary": {
            "Type": "CLOUD_FORMATION_TEMPLATE",
            "Distributor": "test-distributor",
            "SupportUrl": "https://aws.amazon.com",
            "SupportEmail": "test@amazon.com",
            "Id": "prodview-abcd42wvx45um",
            "SupportDescription": "test-support",
            "ShortDescription": "test-description",
```

```
            "Owner": "test-owner",
            "Name": "test-product2",
            "HasDefaultPath": false,
            "ProductId": "prod-3p5abcdef3oyk"
        }
    },
    "ProvisioningArtifactDetail": {
        "CreatedTime": 1576025036.0,
        "Active": true,
        "Id": "pa-pq3p5lil12a34",
        "Description": "test-version-description",
        "Name": "test-version-name",
        "Type": "CLOUD_FORMATION_TEMPLATE"
    }
}
```

- Per i dettagli sull'API, consulta CreateProduct AWS CLICommand Reference.

`create-provisioning-artifact`

Il seguente esempio di codice mostra come utilizzare`create-provisioning-artifact`.

AWS CLI

Come creare un artefatto di provisioning

L'esempio `create-provisioning-artifact` seguente crea un artefatto di provisioning, utilizzando un file JSON per passare i parametri.

```
aws servicecatalog create-provisioning-artifact \
   --cli-input-json file://create-provisioning-artifact-input.json
```

Contenuto di `create-provisioning-artifact-input.json`:

```
{
    "ProductId": "prod-nfi2abcdefghi",
    "Parameters": {
        "Name": "test-provisioning-artifact",
        "Description": "test description",
        "Info": {
            "LoadTemplateFromURL": "https://s3-us-west-1.amazonaws.com/
cloudformation-templates-us-west-1/my-cfn-template.template"
```

```
        },
        "Type": "CLOUD_FORMATION_TEMPLATE"
    }
}
```

Output:

```
{
    "Info": {
        "TemplateUrl": "https://s3-us-west-1.amazonaws.com/cloudformation-templates-
us-west-1/my-cfn-template.template"
    },
    "Status": "CREATING",
    "ProvisioningArtifactDetail": {
        "Id": "pa-bb4abcdefwnaio",
        "Name": "test-provisioning-artifact",
        "Description": "test description",
        "Active": true,
        "Type": "CLOUD_FORMATION_TEMPLATE",
        "CreatedTime": 1576022545.0
    }
}
```

- Per i dettagli sull'API, consulta CreateProvisioningArtifact AWS CLICommand Reference.

`create-tag-option`

Il seguente esempio di codice mostra come utilizzare`create-tag-option`.

AWS CLI

Per creare un TagOption

L'`create-tag-option`esempio seguente crea un TagOption.

```
aws servicecatalog create-tag-option
    --key 1234
    --value name
```

Output:

```
{
```

```
    "TagOptionDetail": {
    "Id": "tag-iabcdn4fzjjms",
    "Value": "name",
    "Active": true,
    "Key": "1234"
    }
}
```

- Per i dettagli sull'API, vedere CreateTagOptionin AWS CLI Command Reference.

`delete-portfolio-share`

Il seguente esempio di codice mostra come utilizzare`delete-portfolio-share`.

AWS CLI

Come arrestare la condivisione di un portafoglio con un account

L'esempio `delete-portfolio-share` seguente interrompe la condivisione del portafoglio con l'account specificato.

```
aws servicecatalog delete-portfolio-share \
    --portfolio-id port-2s6abcdq5wdh4 \
    --account-id 123456789012
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta DeletePortfolioShare AWS CLICommand Reference.

`delete-portfolio`

Il seguente esempio di codice mostra come utilizzare`delete-portfolio`.

AWS CLI

Come eliminare un portafoglio

L'esempio `delete-portfolio` seguente elimina il portafoglio specificato.

```
aws servicecatalog delete-portfolio \
    --id port-abcdlx4gox4do
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta DeletePortfolio AWS CLICommand Reference.

`delete-product`

Il seguente esempio di codice mostra come utilizzare`delete-product`.

AWS CLI

Come eliminare un prodotto

L'esempio `delete-product` seguente elimina il prodotto specificato.

```
aws servicecatalog delete-product \
    --id prod-abcdcek6yhbxi
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta DeleteProduct AWS CLICommand Reference.

`delete-provisioning-artifact`

Il seguente esempio di codice mostra come utilizzare`delete-provisioning-artifact`.

AWS CLI

Come eliminare un artefatto di provisioning

L'esempio `delete-provisioning-artifact` seguente elimina l'artefatto di provisioning specificato.

```
aws servicecatalog delete-provisioning-artifact \
    --product-id prod-abc2uebuplcpw \
    --provisioning-artifact-id pa-pqabcddii7ouc
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta DeleteProvisioningArtifact AWS CLICommand Reference.

`delete-tag-option`

Il seguente esempio di codice mostra come utilizzare`delete-tag-option`.

AWS CLI

Per eliminare un TagOption

L'`delete-tag-option`esempio seguente elimina il valore specificato TagOption.

```
aws servicecatalog delete-tag-option \
    --id tag-iabcdn4fzjjms
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, vedere DeleteTagOptionin AWS CLI Command Reference.

`describe-copy-product-status`

Il seguente esempio di codice mostra come utilizzare`describe-copy-product-status`.

AWS CLI

Come descrivere lo stato dell'operazione di copia del prodotto

L'esempio `describe-copy-product-status` seguente mostra lo stato corrente dell'operazione di copia asincrona del prodotto specificata.

```
aws servicecatalog describe-copy-product-status \
    --copy-product-token copyproduct-znn5tf5abcd3w
```

Output:

```
{
    "CopyProductStatus": "SUCCEEDED",
    "TargetProductId": "prod-os6hog7abcdt2"
}
```

- Per i dettagli sull'API, consulta DescribeCopyProductStatus AWS CLICommand Reference.

`describe-portfolio`

Il seguente esempio di codice mostra come utilizzare`describe-portfolio`.

AWS CLI

Come descrivere un portafoglio

L'esempio `describe-portfolio` seguente mostra i dettagli del portafoglio specificato.

```
aws servicecatalog describe-portfolio \
    --id port-2s6abcdq5wdh4
```

Output:

```
{
    "TagOptions": [],
    "PortfolioDetail": {
        "ARN": "arn:aws:catalog:us-west-2:687558541234:portfolio/
port-2s6abcdq5wdh4",
        "Id": "port-2s6wuzyq5wdh4",
        "CreatedTime": 1571337221.555,
        "DisplayName": "my-portfolio",
        "ProviderName": "my-provider"
    },
    "Tags": []
}
```

- Per i dettagli sull'API, consulta DescribePortfolio AWS CLICommand Reference.

`describe-product-as-admin`

Il seguente esempio di codice mostra come utilizzare`describe-product-as-admin`.

AWS CLI

Come descrivere un prodotto come amministratore

L'esempio `describe-product-as-admin` seguente mostra i dettagli del prodotto specificato utilizzando privilegi da amministratore.

```
aws servicecatalog describe-product-as-admin \
    --id prod-abcdcek6yhbxi
```

Output:

```
{
    "TagOptions": [],
    "ProductViewDetail": {
        "ProductARN": "arn:aws:catalog:us-west-2:687558542028:product/prod-
abcdcek6yhbxi",
        "ProductViewSummary": {
            "SupportEmail": "test@amazon.com",
            "Type": "CLOUD_FORMATION_TEMPLATE",
            "Distributor": "test-distributor",
            "ShortDescription": "test-description",
            "Owner": "test-owner",
            "Id": "prodview-wi3l2j4abc6vc",
            "SupportDescription": "test-support",
            "ProductId": "prod-abcdcek6yhbxi",
            "HasDefaultPath": false,
            "Name": "test-product3",
            "SupportUrl": "https://aws.amazon.com"
        },
        "CreatedTime": 1577136715.0,
        "Status": "CREATED"
    },
    "ProvisioningArtifactSummaries": [
        {
            "CreatedTime": 1577136715.0,
            "Description": "test-version-description",
            "ProvisioningArtifactMetadata": {
                "SourceProvisioningArtifactId": "pa-abcdxkkiv5fcm"
            },
            "Name": "test-version-name-3",
            "Id": "pa-abcdxkkiv5fcm"
        }
    ],
    "Tags": [
        {
            "Value": "iad",
            "Key": "region"
        }
    ]
}
```

- Per i dettagli sull'API, consulta DescribeProductAsAdmin AWS CLICommand Reference.

`describe-provisioned-product`

Il seguente esempio di codice mostra come utilizzare`describe-provisioned-product`.

AWS CLI

Come descrivere un prodotto sottoposto a provisioning

L'esempio `describe-provisioned-product` seguente mostra i dettagli del prodotto sottoposto a provisioning specificato.

```
aws servicecatalog describe-provisioned-product \
    --id pp-dpom27bm4abcd
```

Output:

```
{
    "ProvisionedProductDetail": {
        "Status": "ERROR",
        "CreatedTime": 1577222793.358,
        "Arn": "arn:aws:servicecatalog:us-west-2:123456789012:stack/mytestppname3/
pp-dpom27bm4abcd",
        "Id": "pp-dpom27bm4abcd",
        "StatusMessage": "AmazonCloudFormationException  Parameters: [KeyName]
 must have values (Service: AmazonCloudFormation; Status Code: 400; Error Code:
 ValidationError; Request ID: 5528602a-a9ef-427c-825c-f82c31b814f5)",
        "IdempotencyToken": "527c5358-2a1a-4b9e-b1b9-7293b0ddff42",
        "LastRecordId": "rec-tfuawdjovzxge",
        "Type": "CFN_STACK",
        "Name": "mytestppname3"
    },
    "CloudWatchDashboards": []
}
```

- Per i dettagli sull'API, consulta DescribeProvisionedProduct AWS CLICommand Reference.

`describe-provisioning-artifact`

Il seguente esempio di codice mostra come utilizzare`describe-provisioning-artifact`.

AWS CLI

Come descrivere un artefatto di provisioning

L'esempio `describe-provisioning-artifact` seguente mostra i dettagli dell'artefatto sottoposto a provisioning specificato.

```
aws servicecatalog describe-provisioning-artifact \
    --provisioning-artifact-id pa-pcz347abcdcfm \
    --product-id prod-abcdfz3syn2rg
```

Output:

```
{
    "Info": {
        "TemplateUrl": "https://awsdocs.s3.amazonaws.com/servicecatalog/
myexampledevelopment-environment.template"
    },
    "ProvisioningArtifactDetail": {
        "Id": "pa-pcz347abcdcfm",
        "Active": true,
        "Type": "CLOUD_FORMATION_TEMPLATE",
        "Description": "updated description",
        "CreatedTime": 1562097906.0,
        "Name": "updated name"
    },
    "Status": "AVAILABLE"
}
```

- Per i dettagli sull'API, consulta DescribeProvisioningArtifact AWS CLICommand Reference.

`describe-tag-option`

Il seguente esempio di codice mostra come utilizzare`describe-tag-option`.

AWS CLI

Per descrivere un TagOption

L'`describe-tag-option`esempio seguente visualizza i dettagli relativi a quanto specificato TagOption.

```
aws servicecatalog describe-tag-option \
    --id tag-p3tej2abcd5qc
```

Output:

```
{
    "TagOptionDetail": {
        "Active": true,
        "Id": "tag-p3tej2abcd5qc",
        "Value": "value-3",
        "Key": "1234"
    }
}
```

- Per i dettagli sull'API, vedere DescribeTagOptionin AWS CLI Command Reference.

`disassociate-principal-from-portfolio`

Il seguente esempio di codice mostra come utilizzare`disassociate-principal-from-portfolio`.

AWS CLI

Come annullare l'associazione di un principale a un portafoglio

L'esempio `disassociate-principal-from-portfolio` seguente annulla l'associazione del principale specificato al portafoglio.

```
aws servicecatalog disassociate-principal-from-portfolio \
    --portfolio-id port-2s6abcdq5wdh4 \
    --principal-arn arn:aws:iam::123456789012:group/myendusers
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta DisassociatePrincipalFromPortfolio AWS CLICommand Reference.

`disassociate-product-from-portfolio`

Il seguente esempio di codice mostra come utilizzare`disassociate-product-from-portfolio`.

AWS CLI

Come annullare l'associazione di un prodotto a un portafoglio

L'esempio `disassociate-product-from-portfolio` seguente annulla l'associazione del prodotto specificato al portafoglio.

```
aws servicecatalog disassociate-product-from-portfolio \
    --product-id prod-3p5abcdmu3oyk \
    --portfolio-id port-2s6abcdq5wdh4
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta DisassociateProductFromPortfolio AWS CLICommand Reference.

`disassociate-tag-option-from-resource`

Il seguente esempio di codice mostra come utilizzare`disassociate-tag-option-from-resource`.

AWS CLI

Dissociare un TagOption da una risorsa

L'esempio `disassociate-tag-option-from-resource` seguente annulla l'associazione del `TagOption` specificato al portafoglio.

```
aws servicecatalog disassociate-tag-option-from-resource \
     --resource-id port-2s6abcdq5wdh4 \
     --tag-option-id tag-p3abc2pkpz5qc
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta DisassociateTagOptionFromResource AWS CLICommand Reference.

`list-accepted-portfolio-shares`

Il seguente esempio di codice mostra come utilizzare`list-accepted-portfolio-shares`.

AWS CLI

Come elencare le condivisioni di portafoglio accettate

L'esempio `list-accepted-portfolio-shares` seguente elenca tutti i portafogli per i quali la condivisione è stata accettata da questo account, includendo solo i portafogli predefiniti di Catalogo dei servizi.

```
aws servicecatalog list-accepted-portfolio-shares \
    --portfolio-share-type "AWS_SERVICECATALOG"
```

Output:

```
{
    "PortfolioDetails": [
        {
            "ARN": "arn:aws:catalog:us-west-2:123456789012:portfolio/port-
d2abcd5dpkuma",
            "Description": "AWS Service Catalog Reference blueprints for often-used
 AWS services such as EC2, S3, RDS, VPC and EMR.",
            "CreatedTime": 1574456190.687,
            "ProviderName": "AWS Service Catalog",
            "DisplayName": "Reference Architectures",
            "Id": "port-d2abcd5dpkuma"
        },
        {
            "ARN": "arn:aws:catalog:us-west-2:123456789012:portfolio/port-
abcdefaua7zpu",
            "Description": "AWS well-architected blueprints for high reliability
 applications.",
            "CreatedTime": 1574461496.092,
            "ProviderName": "AWS Service Catalog",
            "DisplayName": "High Reliability Architectures",
            "Id": "port-abcdefaua7zpu"
        }
    ]
}
```

- Per i dettagli sull'API, consulta ListAcceptedPortfolioShares AWS CLICommand Reference.

`list-portfolio-access`

Il seguente esempio di codice mostra come utilizzare`list-portfolio-access`.

AWS CLI

Come elencare gli account con accesso a un portafoglio

L'`list-portfolio-access`esempio seguente elenca gli AWS account che hanno accesso al portafoglio specificato.

```
aws servicecatalog list-portfolio-access \
    --portfolio-id port-2s6abcdq5wdh4
```

Output:

```
{
    "AccountIds": [
        "123456789012"
    ]
}
```

- Per i dettagli sull'API, vedere ListPortfolioAccessin AWS CLI Command Reference.

`list-portfolios-for-product`

Il seguente esempio di codice mostra come utilizzare`list-portfolios-for-product`.

AWS CLI

Come elencare i portafogli associati a un prodotto

L'esempio `list-portfolios-for-product` seguente elenca i portafogli associati al prodotto specificato.

```
aws servicecatalog list-portfolios-for-product \
    --product-id prod-abcdfz3syn2rg
```

Output:

```
{
    "PortfolioDetails": [
        {
            "CreatedTime": 1571337221.555,
            "Id": "port-2s6abcdq5wdh4",
            "ARN": "arn:aws:catalog:us-west-2:123456789012:portfolio/
port-2s6abcdq5wdh4",
            "DisplayName": "my-portfolio",
            "ProviderName": "my-provider"
        },
        {
```

```
            "CreatedTime": 1559665256.348,
            "Id": "port-5abcd3e5st4ei",
            "ARN": "arn:aws:catalog:us-west-2:123456789012:portfolio/
port-5abcd3e5st4ei",
            "DisplayName": "test",
            "ProviderName": "provider-name"
        }
    ]
}
```

- Per i dettagli sull'API, consulta ListPortfoliosForProduct AWS CLICommand Reference.

`list-portfolios`

Il seguente esempio di codice mostra come utilizzare`list-portfolios`.

AWS CLI

Come elencare i portafogli

L'esempio `list-portfolios` seguente elenca i portafogli di Catalogo dei servizi presenti nella Regione corrente.

```
aws servicecatalog list-portfolios
```

Output:

```
{
    "PortfolioDetails": [
        {
           "CreatedTime": 1559665256.348,
           "ARN": "arn:aws:catalog:us-east-2:123456789012:portfolio/
port-5pzcxmplst4ei",
           "DisplayName": "my-portfolio",
           "Id": "port-5pzcxmplst4ei",
           "ProviderName": "my-user"
        }
    ]
}
```

- Per i dettagli sull'API, consulta ListPortfolios AWS CLICommand Reference.

`list-principals-for-portfolio`

Il seguente esempio di codice mostra come utilizzare`list-principals-for-portfolio`.

AWS CLI

Come elencare tutti i principali per un portafoglio

L'esempio `list-principals-for-portfolio` seguente elenca tutti i principali del portafoglio specificato.

```
aws servicecatalog list-principals-for-portfolio \
    --portfolio-id port-2s6abcdq5wdh4
```

Output:

```
{
    "Principals": [
        {
            "PrincipalARN": "arn:aws:iam::123456789012:user/usertest",
            "PrincipalType": "IAM"
        }
    ]
}
```

- Per i dettagli sull'API, consulta ListPrincipalsForPortfolio AWS CLICommand Reference.

`list-provisioning-artifacts`

Il seguente esempio di codice mostra come utilizzare`list-provisioning-artifacts`.

AWS CLI

Come elencare tutti gli artefatti di provisioning per un prodotto

L'esempio `list-provisioning-artifacts` seguente elenca tutti gli artefatti di provisioning per il prodotto specificato.

```
aws servicecatalog list-provisioning-artifacts \
    --product-id prod-nfi2abcdefgcpw
```

Output:

```
{
    "ProvisioningArtifactDetails": [
        {
            "Id": "pa-abcdef54ipm6z",
            "Description": "test-version-description",
            "Type": "CLOUD_FORMATION_TEMPLATE",
            "CreatedTime": 1576021147.0,
            "Active": true,
            "Name": "test-version-name"
        },
        {
            "Id": "pa-bb4zyxwwnaio",
            "Description": "test description",
            "Type": "CLOUD_FORMATION_TEMPLATE",
            "CreatedTime": 1576022545.0,
            "Active": true,
            "Name": "test-provisioning-artifact-2"
        }
    ]
}
```

- Per i dettagli sull'API, consulta ListProvisioningArtifacts AWS CLICommand Reference.

`list-resources-for-tag-option`

Il seguente esempio di codice mostra come utilizzare`list-resources-for-tag-option`.

AWS CLI

Per elencare le risorse associate a un TagOption

L'esempio `list-resources-for-tag-option` seguente elenca le risorse associate al `TagOption` specificato.

```
aws servicecatalog list-resources-for-tag-option \
    --tag-option-id tag-p3tej2abcd5qc
```

Output:

```
{
    "ResourceDetails": [
        {
```

```
            "ARN": "arn:aws:catalog:us-west-2:123456789012:product/prod-
abcdfz3syn2rg",
            "Name": "my product",
            "Description": "description",
            "CreatedTime": 1562097906.0,
            "Id": "prod-abcdfz3syn2rg"
        }
    ]
}
```

- Per i dettagli sull'API, consulta ListResourcesForTagOption AWS CLICommand Reference.

`list-tag-options`

Il seguente esempio di codice mostra come utilizzare`list-tag-options`.

AWS CLI

L'esempio `list-tag-options` seguente elenca tutti i valori per `TagOptions`.

```
aws servicecatalog list-tag-options
```

Output:

```
{
    "TagOptionDetails": [
        {
            "Value": "newvalue",
            "Active": true,
            "Id": "tag-iabcdn4fzjjms",
            "Key": "1234"
        },
        {
            "Value": "value1",
            "Active": true,
            "Id": "tag-e3abcdvmwvrzy",
            "Key": "key"
        }
    ]
}
```

- Per i dettagli sull'API, consulta ListTagOptions AWS CLICommand Reference.

`provision-product`

Il seguente esempio di codice mostra come utilizzare`provision-product`.

AWS CLI

Come eseguire il provisioning di un prodotto

L'esempio `provision-product` seguente esegue il provisioning del prodotto specificato utilizzando l'artefatto di provisioning specificato.

```
aws servicecatalog provision-product \
    --product-id prod-abcdfz3syn2rg \
    --provisioning-artifact-id pa-abc347pcsccfm \
    --provisioned-product-name "mytestppname3"
```

Output:

```
{
    "RecordDetail": {
        "RecordId": "rec-tfuawdabcdege",
        "CreatedTime": 1577222793.362,
        "ProvisionedProductId": "pp-abcd27bm4mldq",
        "PathId": "lpv2-abcdg3jp6t5k6",
        "RecordErrors": [],
        "ProductId": "prod-abcdfz3syn2rg",
        "UpdatedTime": 1577222793.362,
        "RecordType": "PROVISION_PRODUCT",
        "ProvisionedProductName": "mytestppname3",
        "ProvisioningArtifactId": "pa-pcz347abcdcfm",
        "RecordTags": [],
        "Status": "CREATED",
        "ProvisionedProductType": "CFN_STACK"
    }
}
```

- Per i dettagli sull'API, consulta ProvisionProduct AWS CLICommand Reference.

`reject-portfolio-share`

Il seguente esempio di codice mostra come utilizzare`reject-portfolio-share`.

AWS CLI

Come rifiutare una condivisione di portafoglio

L'esempio `reject-portfolio-share` seguente rifiuta la condivisione del portafoglio indicato.

```
aws servicecatalog reject-portfolio-share \
    --portfolio-id port-2s6wuabcdefghijk
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta RejectPortfolioShare AWS CLICommand Reference.

`scan-provisioned-products`

Il seguente esempio di codice mostra come utilizzare`scan-provisioned-products`.

AWS CLI

Come elencare tutti i prodotti sottoposti a provisioning disponibili

L'esempio `scan-provisioned-products` seguente elenca i prodotti sottoposti a provisioning disponibili.

```
aws servicecatalog scan-provisioned-products
```

Output:

```
{
    "ProvisionedProducts": [
        {
            "Status": "ERROR",
            "Arn": "arn:aws:servicecatalog:us-west-2:123456789012:stack/
mytestppname3/pp-abcd27bm4mldq",
            "StatusMessage": "AmazonCloudFormationException  Parameters: [KeyName]
 must have values (Service: AmazonCloudFormation; Status Code: 400; Error Code:
 ValidationError; Request ID: 5528602a-a9ef-427c-825c-f82c31b814f5)",
            "Id": "pp-abcd27bm4mldq",
            "Type": "CFN_STACK",
            "IdempotencyToken": "527c5358-2a1a-4b9e-b1b9-7293b0ddff42",
            "CreatedTime": 1577222793.358,
```

```
            "Name": "mytestppname3",
            "LastRecordId": "rec-tfuawdabcdxge"
        }
    ]
}
```

- Per i dettagli sull'API, consulta ScanProvisionedProducts AWS CLICommand Reference.

`search-products-as-admin`

Il seguente esempio di codice mostra come utilizzare`search-products-as-admin`.

AWS CLI

Come cercare prodotti con privilegi da amministratore

L'esempio `search-products-as-admin` seguente cerca prodotti con privilegi da amministratore, utilizzando un ID portafoglio come filtro.

```
aws servicecatalog search-products-as-admin \
    --portfolio-id port-5abcd3e5st4ei
```

Output:

```
{
    "ProductViewDetails": [
        {
            "ProductViewSummary": {
                "Name": "my product",
                "Owner": "owner name",
                "Type": "CLOUD_FORMATION_TEMPLATE",
                "ProductId": "prod-abcdfz3syn2rg",
                "HasDefaultPath": false,
                "Id": "prodview-abcdmyuzv2dlu",
                "ShortDescription": "description"
            },
            "ProductARN": "arn:aws:catalog:us-west-2:123456789012:product/prod-
abcdfz3syn2rg",
            "CreatedTime": 1562097906.0,
            "Status": "CREATED"
        }
    ]
```

```
}
```

- Per i dettagli sull'API, consulta SearchProductsAsAdmin AWS CLICommand Reference.

`search-provisioned-products`

Il seguente esempio di codice mostra come utilizzare`search-provisioned-products`.

AWS CLI

Come cercare i prodotti sottoposti a provisioning

L'esempio `search-provisioned-products` seguente cerca i prodotti sottoposti a provisioning corrispondenti all'ID prodotto specificato, utilizzando un file JSON per passare i parametri.

```
aws servicecatalog search-provisioned-products \
    --cli-input-json file://search-provisioned-products-input.json
```

Contenuto di `search-provisioned-products-input.json`:

```
{
    "Filters": {
        "SearchQuery": [
            "prod-tcjvfz3syn2rg"
        ]
    }
}
```

Output:

```
{
    "ProvisionedProducts": [
        {
            "ProvisioningArtifactId": "pa-pcz347abcdcfm",
            "Name": "mytestppname3",
            "CreatedTime": 1577222793.358,
            "Id": "pp-abcd27bm4mldq",
            "Status": "ERROR",
            "UserArn": "arn:aws:iam::123456789012:user/cliuser",
            "StatusMessage": "AmazonCloudFormationException  Parameters: [KeyName]
 must have values (Service: AmazonCloudFormation; Status Code: 400; Error Code:
 ValidationError; Request ID: 5528602a-a9ef-427c-825c-f82c31b814f5)",
```

```
            "Arn": "arn:aws:servicecatalog:us-west-2:123456789012:stack/
mytestppname3/pp-abcd27bm4mldq",
            "Tags": [
                {
                    "Value": "arn:aws:catalog:us-west-2:123456789012:product/prod-
abcdfz3syn2rg",
                    "Key": "aws:servicecatalog:productArn"
                },
                {
                    "Value": "arn:aws:iam::123456789012:user/cliuser",
                    "Key": "aws:servicecatalog:provisioningPrincipalArn"
                },
                {
                    "Value": "value-3",
                    "Key": "1234"
                },
                {
                    "Value": "pa-pcz347abcdcfm",
                    "Key": "aws:servicecatalog:provisioningArtifactIdentifier"
                },
                {
                    "Value": "arn:aws:catalog:us-west-2:123456789012:portfolio/
port-2s6abcdq5wdh4",
                    "Key": "aws:servicecatalog:portfolioArn"
                },
                {
                    "Value": "arn:aws:servicecatalog:us-west-2:123456789012:stack/
mytestppname3/pp-abcd27bm4mldq",
                    "Key": "aws:servicecatalog:provisionedProductArn"
                }
            ],
            "IdempotencyToken": "527c5358-2a1a-4b9e-b1b9-7293b0ddff42",
            "UserArnSession": "arn:aws:iam::123456789012:user/cliuser",
            "Type": "CFN_STACK",
            "LastRecordId": "rec-tfuawdabcdxge",
            "ProductId": "prod-abcdfz3syn2rg"
        }
    ],
    "TotalResultsCount": 1
}
```

- Per i dettagli sull'API, consulta SearchProvisionedProducts AWS CLICommand Reference.

update-portfolio

Il seguente esempio di codice mostra come utilizzareupdate-portfolio.

AWS CLI

Come aggiornare un portafoglio

L'esempio update-portfolio seguente aggiorna il nome del portafoglio specificato.

```
aws servicecatalog update-portfolio \
    --id port-5abcd3e5st4ei \
    --display-name "New portfolio name"
```

Output:

```
{
    "PortfolioDetail": {
        "DisplayName": "New portfolio name",
        "ProviderName": "provider",
        "ARN": "arn:aws:catalog:us-west-2:123456789012:portfolio/
port-5abcd3e5st4ei",
        "Id": "port-5abcd3e5st4ei",
        "CreatedTime": 1559665256.348
    },
    "Tags": []
}
```

- Per i dettagli sull'API, consulta UpdatePortfolio AWS CLICommand Reference.

update-product

Il seguente esempio di codice mostra come utilizzareupdate-product.

AWS CLI

Come aggiornare un prodotto

L'esempio update-product seguente aggiorna il nome e il proprietario del prodotto specificato.

```
aws servicecatalog update-product \
    --id prod-os6abc7drqlt2 \
```

```
    --name "New product name" \
    --owner "Updated product owner"
```

Output:

```
{
    "Tags": [
        {
            "Value": "iad",
            "Key": "region"
        }
    ],
    "ProductViewDetail": {
        "ProductViewSummary": {
            "Owner": "Updated product owner",
            "ProductId": "prod-os6abc7drqlt2",
            "Distributor": "test-distributor",
            "SupportUrl": "https://aws.amazon.com",
            "Name": "New product name",
            "ShortDescription": "test-description",
            "HasDefaultPath": false,
            "Id": "prodview-6abcdgrfhvidy",
            "SupportDescription": "test-support",
            "SupportEmail": "test@amazon.com",
            "Type": "CLOUD_FORMATION_TEMPLATE"
        },
        "Status": "CREATED",
        "ProductARN": "arn:aws:catalog:us-west-2:123456789012:product/prod-
os6abc7drqlt2",
        "CreatedTime": 1577136255.0
    }
}
```

- Per i dettagli sull'API, consulta UpdateProduct AWS CLICommand Reference.

`update-provisioning-artifact`

Il seguente esempio di codice mostra come utilizzare`update-provisioning-artifact`.

AWS CLI

Come aggiornare un artefatto di provisioning

L'esempio `update-provisioning-artifact` seguente aggiorna il nome e la descrizione dell'artefatto di provisioning specificato, utilizzando un file JSON per passare i parametri.

```
aws servicecatalog update-provisioning-artifact \
    --cli-input-json file://update-provisioning-artifact-input.json
```

Contenuto di `update-provisioning-artifact-input.json`:

```
{
    "ProductId": "prod-abcdfz3syn2rg",
    "ProvisioningArtifactId": "pa-pcz347abcdcfm",
    "Name": "updated name",
    "Description": "updated description"
}
```

Output:

```
{
    "Info": {
        "TemplateUrl": "https://awsdocs.s3.amazonaws.com/servicecatalog/
myexampledevelopment-environment.template"
    },
    "Status": "AVAILABLE",
    "ProvisioningArtifactDetail": {
        "Active": true,
        "Description": "updated description",
        "Id": "pa-pcz347abcdcfm",
        "Name": "updated name",
        "Type": "CLOUD_FORMATION_TEMPLATE",
        "CreatedTime": 1562097906.0
    }
}
```

- Per i dettagli sull'API, consulta UpdateProvisioningArtifact AWS CLICommand Reference.

`update-tag-option`

Il seguente esempio di codice mostra come utilizzare`update-tag-option`.

AWS CLI

Per aggiornare un TagOption

L'esempio `update-tag-option` seguente aggiorna il valore di un `TagOption`, utilizzando il file JSON specificato.

```
aws servicecatalog update-tag-option --cli-input-json file://update-tag-option-input.json
```

Contenuto di `update-tag-option-input.json`:

```
{
    "Id": "tag-iabcdn4fzjjms",
    "Value": "newvalue",
    "Active": true
}
```

Output:

```
{
    "TagOptionDetail": {
        "Value": "newvalue",
        "Key": "1234",
        "Active": true,
        "Id": "tag-iabcdn4fzjjms"
    }
}
```

- Per i dettagli sull'API, consulta UpdateTagOption AWS CLICommand Reference.

Esempi di Service Quotas utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with Service Quotas.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`get-aws-default-service-quota`

Il seguente esempio di codice mostra come utilizzare`get-aws-default-service-quota`.

AWS CLI

Come descrivere una quota di servizio predefinita

L'esempio `get-aws-default-service-quota` seguente visualizza i dettagli della quota specificata.

```
aws service-quotas get-aws-default-service-quota \
    --service-code ec2 \
    --quota-code L-1216C47A
```

Output:

```
{
    "Quota": {
        "ServiceCode": "ec2",
        "ServiceName": "Amazon Elastic Compute Cloud (Amazon EC2)",
        "QuotaArn": "arn:aws:servicequotas:us-east-2::ec2/L-1216C47A",
        "QuotaCode": "L-1216C47A",
        "QuotaName": "Running On-Demand Standard (A, C, D, H, I, M, R, T, Z)
 instances",
        "Value": 5.0,
        "Unit": "None",
        "Adjustable": true,
        "GlobalQuota": false,
        "UsageMetric": {
            "MetricNamespace": "AWS/Usage",
            "MetricName": "ResourceCount",
            "MetricDimensions": {
                "Class": "Standard/OnDemand",
                "Resource": "vCPU",
                "Service": "EC2",
                "Type": "Resource"
            },
```

```
            "MetricStatisticRecommendation": "Maximum"
        }
    }
}
```

- Per i dettagli sull'API, consulta GetAwsDefaultServiceQuota AWS CLICommand Reference.

`get-requested-service-quota-change`

Il seguente esempio di codice mostra come utilizzare`get-requested-service-quota-change`.

AWS CLI

Come descrivere una richiesta di aumento della quota di servizio

L'esempio `get-requested-service-quota-change` seguente descrive la richiesta di aumento della quota specificata.

```
aws service-quotas get-requested-service-quota-change \
    --request-id d187537d15254312a9609aa51bbf7624u7W49tP0
```

Output:

```
{
    "RequestedQuota": {
        "Id": "d187537d15254312a9609aa51bbf7624u7W49tPO",
        "CaseId": "6780195351",
        "ServiceCode": "ec2",
        "ServiceName": "Amazon Elastic Compute Cloud (Amazon EC2)",
        "QuotaCode": "L-20F13EBD",
        "QuotaName": "Running Dedicated c5n Hosts",
        "DesiredValue": 2.0,
        "Status": "CASE_OPENED",
        "Created": 1580446904.067,
        "LastUpdated": 1580446953.265,
        "Requester": "{\"accountId\":\"123456789012\",\"callerArn\":
\"arn:aws:iam::123456789012:root\"}",
        "QuotaArn": "arn:aws:servicequotas:us-east-2:123456789012:ec2/L-20F13EBD",
        "GlobalQuota": false,
        "Unit": "None"
    }
}
```

- Per i dettagli sull'API, consulta GetRequestedServiceQuotaChange AWS CLICommand Reference.

`get-service-quota`

Il seguente esempio di codice mostra come utilizzare`get-service-quota`.

AWS CLI

Come descrivere una quota di servizio

L'esempio `get-service-quota` seguente visualizza i dettagli della quota specificata.

```
aws service-quotas get-service-quota \
    --service-code ec2 \
    --quota-code L-1216C47A
```

Output:

```
{
    "Quota": {
        "ServiceCode": "ec2",
        "ServiceName": "Amazon Elastic Compute Cloud (Amazon EC2)",
        "QuotaArn": "arn:aws:servicequotas:us-east-2:123456789012:ec2/L-1216C47A",
        "QuotaCode": "L-1216C47A",
        "QuotaName": "Running On-Demand Standard (A, C, D, H, I, M, R, T, Z)
 instances",
        "Value": 1920.0,
        "Unit": "None",
        "Adjustable": true,
        "GlobalQuota": false,
        "UsageMetric": {
            "MetricNamespace": "AWS/Usage",
            "MetricName": "ResourceCount",
            "MetricDimensions": {
                "Class": "Standard/OnDemand",
                "Resource": "vCPU",
                "Service": "EC2",
                "Type": "Resource"
            },
            "MetricStatisticRecommendation": "Maximum"
        }
    }
```

```
}
```

- Per i dettagli sull'API, consulta GetServiceQuota AWS CLICommand Reference.

`list-aws-default-service-quotas`

Il seguente esempio di codice mostra come utilizzare`list-aws-default-service-quotas`.

AWS CLI

Come elencare le quote predefinite per un servizio

L'esempio `list-aws-default-service-quotas` seguente elenca i valori predefiniti per le quote per il servizio specificato.

```
aws service-quotas list-aws-default-service-quotas \
    --service-code xray
```

Output:

```
{
    "Quotas": [
        {
            "ServiceCode": "xray",
            "ServiceName": "AWS X-Ray",
            "QuotaArn": "arn:aws:servicequotas:us-west-2::xray/L-C6B6F05D",
            "QuotaCode": "L-C6B6F05D",
            "QuotaName": "Indexed annotations per trace",
            "Value": 50.0,
            "Unit": "None",
            "Adjustable": false,
            "GlobalQuota": false
        },
        {
            "ServiceCode": "xray",
            "ServiceName": "AWS X-Ray",
            "QuotaArn": "arn:aws:servicequotas:us-west-2::xray/L-D781C0FD",
            "QuotaCode": "L-D781C0FD",
            "QuotaName": "Segment document size",
            "Value": 64.0,
            "Unit": "Kilobytes",
            "Adjustable": false,
```

```
            "GlobalQuota": false
        },
        {
            "ServiceCode": "xray",
            "ServiceName": "AWS X-Ray",
            "QuotaArn": "arn:aws:servicequotas:us-west-2::xray/L-998BFF16",
            "QuotaCode": "L-998BFF16",
            "QuotaName": "Trace and service graph retention in days",
            "Value": 30.0,
            "Unit": "None",
            "Adjustable": false,
            "GlobalQuota": false
        }
    ]
}
```

- Per i dettagli sull'API, consulta ListAwsDefaultServiceQuotas AWS CLICommand Reference.

`list-requested-service-quota-change-history-by-quota`

Il seguente esempio di codice mostra come utilizzare`list-requested-service-quota-change-history-by-quota`.

AWS CLI

Come elencare le richieste di aumento delle quote

L'esempio `list-requested-service-quota-change-history-by-quota` seguente elenca le richieste di aumento della quota per la quota specificata.

```
aws service-quotas list-requested-service-quota-change-history-by-quota \
    --service-code ec2 \
    --quota-code L-20F13EBD
```

Output:

```
{
    "RequestedQuotas": [
        {
            "Id": "d187537d15254312a9609aa51bbf7624u7W49tPO",
            "CaseId": "6780195351",
            "ServiceCode": "ec2",
```

```
            "ServiceName": "Amazon Elastic Compute Cloud (Amazon EC2)",
            "QuotaCode": "L-20F13EBD",
            "QuotaName": "Running Dedicated c5n Hosts",
            "DesiredValue": 2.0,
            "Status": "CASE_OPENED",
            "Created": 1580446904.067,
            "LastUpdated": 1580446953.265,
            "Requester": "{\"accountId\":\"123456789012\",\"callerArn\":
\"arn:aws:iam::123456789012:root\"}",
            "QuotaArn": "arn:aws:servicequotas:us-east-2:123456789012:ec2/
L-20F13EBD",
            "GlobalQuota": false,
            "Unit": "None"
        }
    ]
}
```

- Per i dettagli sull'API, consulta ListRequestedServiceQuotaChangeHistoryByQuota AWS CLICommand Reference.

`list-requested-service-quota-change-history`

Il seguente esempio di codice mostra come utilizzare`list-requested-service-quota-change-history`.

AWS CLI

Come elencare le richieste di aumento delle quote

L'esempio `list-requested-service-quota-change-history` seguente elenca le richieste di aumento della quota per il servizio specificato.

```
aws service-quotas list-requested-service-quota-change-history \
    --service-code ec2
```

Output:

```
{
    "RequestedQuotas": [
        {
            "Id": "d187537d15254312a9609aa51bbf7624u7W49tPO",
            "CaseId": "6780195351",
```

```
            "ServiceCode": "ec2",
            "ServiceName": "Amazon Elastic Compute Cloud (Amazon EC2)",
            "QuotaCode": "L-20F13EBD",
            "QuotaName": "Running Dedicated c5n Hosts",
            "DesiredValue": 2.0,
            "Status": "CASE_OPENED",
            "Created": 1580446904.067,
            "LastUpdated": 1580446953.265,
            "Requester": "{\"accountId\":\"123456789012\",\"callerArn\":
\"arn:aws:iam::123456789012:root\"}",
            "QuotaArn": "arn:aws:servicequotas:us-east-2:123456789012:ec2/
L-20F13EBD",
            "GlobalQuota": false,
            "Unit": "None"
        }
    ]
}
```

- Per i dettagli sull'API, consulta ListRequestedServiceQuotaChangeHistory AWS CLICommand Reference.

`list-service-quotas`

Il seguente esempio di codice mostra come utilizzare`list-service-quotas`.

AWS CLI

Come elencare le quote per un servizio

L'`list-service-quotas`esempio seguente visualizza i dettagli sulle quote per AWS CloudFormation.

```
aws service-quotas list-service-quotas \
    --service-code cloudformation
```

Output:

```
{
    "Quotas": [
        {
            "ServiceCode": "cloudformation",
            "ServiceName": "AWS CloudFormation",
```

```
            "QuotaArn": "arn:aws:servicequotas:us-
east-2:123456789012:cloudformation/L-87D14FB7",
            "QuotaCode": "L-87D14FB7",
            "QuotaName": "Output count in CloudFormation template",
            "Value": 60.0,
            "Unit": "None",
            "Adjustable": false,
            "GlobalQuota": false
        },
        {
            "ServiceCode": "cloudformation",
            "ServiceName": "AWS CloudFormation",
            "QuotaArn": "arn:aws:servicequotas:us-
east-2:123456789012:cloudformation/L-0485CB21",
            "QuotaCode": "L-0485CB21",
            "QuotaName": "Stack count",
            "Value": 200.0,
            "Unit": "None",
            "Adjustable": true,
            "GlobalQuota": false
        }
    ]
}
```

- Per i dettagli sull'API, vedere ListServiceQuotasin AWS CLI Command Reference.

`list-services`

Il seguente esempio di codice mostra come utilizzare`list-services`.

AWS CLI

Come elencare i servizi disponibili

Il comando seguente elenca i servizi disponibili in Service Quotas.

```
aws service-quotas list-services
```

Output:

```
{
    "Services": [
        {
```

```
            "ServiceCode": "AWSCloudMap",
            "ServiceName": "AWS Cloud Map"
        },
        {
            "ServiceCode": "access-analyzer",
            "ServiceName": "Access Analyzer"
        },
        {
            "ServiceCode": "acm",
            "ServiceName": "AWS Certificate Manager (ACM)"
        },

        ...truncated...

        {
            "ServiceCode": "xray",
            "ServiceName": "AWS X-Ray"
        }
    ]
}
```

Puoi aggiungere il parametro `--query` per filtrare la visualizzazione in base alle informazioni d'interesse. L'esempio seguente visualizza solo i codici di servizio.

```
aws service-quotas list-services \
    --query Services[*].ServiceCode
```

Output:

```
[
    "AWSCloudMap",
    "access-analyzer",
    "acm",
    "acm-pca",
    "amplify",
    "apigateway",
    "application-autoscaling",
        ...truncated...
    "xray"
]
```

- Per i dettagli sull'API, consulta ListServices AWS CLICommand Reference.

`request-service-quota-increase`

Il seguente esempio di codice mostra come utilizzare`request-service-quota-increase`.

AWS CLI

Come richiedere un aumento della quota di servizio

L'esempio `request-service-quota-increase` seguente richiede un aumento della quota di servizio specificata.

```
aws service-quotas request-service-quota-increase \
    --service-code ec2 \
    --quota-code L-20F13EBD \
    --desired-value 2
```

Output:

```
{
    "RequestedQuota": {
        "Id": "d187537d15254312a9609aa51bbf7624u7W49tPO",
        "ServiceCode": "ec2",
        "ServiceName": "Amazon Elastic Compute Cloud (Amazon EC2)",
        "QuotaCode": "L-20F13EBD",
        "QuotaName": "Running Dedicated c5n Hosts",
        "DesiredValue": 2.0,
        "Status": "PENDING",
        "Created": 1580446904.067,
        "Requester": "{\"accountId\":\"123456789012\",\"callerArn\":
\"arn:aws:iam::123456789012:root\"}",
        "QuotaArn": "arn:aws:servicequotas:us-east-2:123456789012:ec2/L-20F13EBD",
        "GlobalQuota": false,
        "Unit": "None"
    }
}
```

- Per i dettagli sull'API, consulta RequestServiceQuotaIncrease AWS CLICommand Reference.

Esempi di utilizzo di Amazon SES AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface con Amazon SES.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`delete-identity`

Il seguente esempio di codice mostra come usare`delete-identity`.

AWS CLI

Eliminazione di un'identità

Nell'esempio seguente viene utilizzato il comando `delete-identity` per eliminare un'identità dall'elenco delle identità verificate con Amazon SES:

```
aws ses delete-identity --identity user@example.com
```

Per ulteriori informazioni riguardo alle identità verificate, consulta Verifica degli indirizzi e-mail e dei domini in Amazon SES nella Guida per gli sviluppatori di Amazon Simple Email Service.

- Per i dettagli sull'API, consulta DeleteIdentity AWS CLICommand Reference.

`get-identity-dkim-attributes`

Il seguente esempio di codice mostra come utilizzare`get-identity-dkim-attributes`.

AWS CLI

Come ottenere gli attributi Easy DKIM di Amazon SES per un elenco di identità

L'esempio seguente utilizza il comando `get-identity-dkim-attributes` per richiamare gli attributi DKIM di Amazon SES per un elenco di identità:

```
aws ses get-identity-dkim-attributes --identities "example.com" "user@example.com"
```

Output:

```
{
   "DkimAttributes": {
       "example.com": {
           "DkimTokens": [
               "EXAMPLEjcs5xoyqytjsotsijas7236gr",
               "EXAMPLEjr76cvoc6mysspnioorxsn6ep",
               "EXAMPLEkbmkqkhlm2lyz77ppkulerm4k"
           ],
           "DkimEnabled": true,
           "DkimVerificationStatus": "Success"
       },
       "user@example.com": {
           "DkimEnabled": false,
           "DkimVerificationStatus": "NotStarted"
       }
   }
}
```

Se chiami questo comando con un'identità che non hai mai inviato per la verifica, tale identità non verrà visualizzata nell'output.

Per ulteriori informazioni su Easy DKIM, consulta Easy DKIM in Amazon SES nella Guida per gli sviluppatori di Amazon Simple Email Service.

- Per i dettagli sull'API, consulta GetIdentityDkimAttributes AWS CLICommand Reference.

`get-identity-notification-attributes`

Il seguente esempio di codice mostra come utilizzare`get-identity-notification-attributes`.

AWS CLI

Come ottenere gli attributi di notifica di Amazon SES per un elenco di identità

L'esempio seguente utilizza il comando `get-identity-notification-attributes` per recuperare gli attributi di notifica di Amazon SES per un elenco di identità:

```
aws ses get-identity-notification-attributes --
identities "user1@example.com" "user2@example.com"
```

Output:

```
{
   "NotificationAttributes": {
       "user1@example.com": {
           "ForwardingEnabled": false,
           "ComplaintTopic": "arn:aws:sns:us-east-1:EXAMPLE65304:MyTopic",
           "BounceTopic": "arn:aws:sns:us-east-1:EXAMPLE65304:MyTopic",
           "DeliveryTopic": "arn:aws:sns:us-east-1:EXAMPLE65304:MyTopic"
       },
       "user2@example.com": {
           "ForwardingEnabled": true
       }
   }
}
```

Questo comando restituisce lo stato dell'inoltro del feedback via e-mail e, se applicabile, gli Amazon Resource Names (ARNs) degli argomenti di Amazon SNS a cui vengono inviate le notifiche di rimbalzo, reclamo e consegna.

Se chiami questo comando con un'identità che non hai mai inviato per la verifica, tale identità non verrà visualizzata nell'output.

Per ulteriori informazioni sulle notifiche, consulta Utilizzo delle notifiche con Amazon SES nella Guida per gli sviluppatori di Amazon Simple Email Service.

- Per i dettagli sull'API, consulta Command Reference. GetIdentityNotificationAttributesAWS CLI

`get-identity-verification-attributes`

Il seguente esempio di codice mostra come utilizzare`get-identity-verification-attributes`.

AWS CLI

Ottenere lo stato di verifica di Amazon SES per un elenco di identità

Nell'esempio seguente viene utilizzato il comando `get-identity-verification-attributes` per richiamare lo stato di verifica di Amazon SES per un elenco di identità:

```
aws ses get-identity-verification-attributes --
identities "user1@example.com" "user2@example.com"
```

Output:

```
{
   "VerificationAttributes": {
       "user1@example.com": {
           "VerificationStatus": "Success"
       },
       "user2@example.com": {
           "VerificationStatus": "Pending"
       }
   }
}
```

Se chiami questo comando con un'identità che non hai mai inviato per la verifica, tale identità non verrà visualizzata nell'output.

Per ulteriori informazioni riguardo alle identità verificate, consulta Verifica degli indirizzi e-mail e dei domini in Amazon SES nella Guida per gli sviluppatori di Amazon Simple Email Service.

- Per i dettagli sull'API, consulta GetIdentityVerificationAttributes AWS CLICommand Reference.

`get-send-quota`

Il seguente esempio di codice mostra come utilizzare`get-send-quota`.

AWS CLI

Ottenere i limiti di invio di Amazon SES

Nell'esempio seguente viene utilizzato il comando `get-send-quota` per restituire i limiti di invio di Amazon SES:

```
aws ses get-send-quota
```

Output:

```
{
   "Max24HourSend": 200.0,
```

```
    "SentLast24Hours": 1.0,
    "MaxSendRate": 1.0
}
```

Max24 HourSend è la tua quota di invio, ovvero il numero massimo di e-mail che puoi inviare in un periodo di 24 ore. La quota di invio riflette un periodo di tempo continuo. Ogni volta che provi a inviare un messaggio e-mail, Amazon SES verifica la quantità di e-mail inviate nelle ultime 24 ore. Se il numero totale di e-mail che hai inviato è inferiore alla quota, la tua richiesta sarà accettata e l'e-mail inviata.

SentLast24Hours è il numero di email che hai inviato nelle 24 ore precedenti.

MaxSendRate è il numero massimo di e-mail che puoi inviare al secondo.

Tieni presente che i limiti di invio si basano sui destinatari e non sui messaggi. Ad esempio, un'e-mail con 10 destinatari viene conteggiata come 10 e-mail ai fini della quota sugli invii.

Per ulteriori informazioni, consulta Gestione dei limiti di invio in Amazon SES nella Guida per gli sviluppatori di Amazon Simple Email Service.

- Per i dettagli sull'API, consulta GetSendQuota AWS CLICommand Reference.

`get-send-statistics`

Il seguente esempio di codice mostra come utilizzare`get-send-statistics`.

AWS CLI

Come ottenere le statistiche di invio di Amazon SES

L'esempio seguente utilizza il comando `get-send-statistics` per restituire le statistiche di invio di Amazon SES:

```
aws ses get-send-statistics
```

Output:

```
{
    "SendDataPoints": [
        {
            "Complaints": 0,
            "Timestamp": "2013-06-12T19:32:00Z",
```

```
            "DeliveryAttempts": 2,
            "Bounces": 0,
            "Rejects": 0
        },
        {
            "Complaints": 0,
            "Timestamp": "2013-06-12T00:47:00Z",
            "DeliveryAttempts": 1,
            "Bounces": 0,
            "Rejects": 0
        }
    ]
}
```

Il risultato è un elenco di punti di dati relativi alle ultime due settimane di attività di invio. Ogni punto dati nell'elenco contiene statistiche per un intervallo di 15 minuti.

In questo esempio, sono presenti solo due punti dati perché le uniche e-mail inviate dall'utente nelle ultime due settimane rientrano in due intervalli di 15 minuti.

Per ulteriori informazioni, consulta Monitoraggio delle statistiche di utilizzo tramite Amazon SES nella Guida per gli sviluppatori di Amazon Simple Email Service.

- Per i dettagli sull'API, consulta GetSendStatistics AWS CLICommand Reference.

`list-identities`

Il seguente esempio di codice mostra come utilizzare`list-identities`.

AWS CLI

Per elencare tutte le identità (indirizzi e-mail e domini) per un account specifico AWS

Nell'esempio seguente viene utilizzato il comando `list-identities` per elencare tutte le identità che sono state inviate per la verifica con Amazon SES:

```
aws ses list-identities
```

Output:

```
{
    "Identities": [
```

```
      "user@example.com",
      "example.com"
    ]
}
```

L'elenco restituito contiene tutte le identità indipendentemente dallo stato della verifica (verified, pending verification, failure, ecc.).

In questo esempio, gli indirizzi e-mail e i domini vengono restituiti perché non è stato specificato il parametro identity-type.

Per ulteriori informazioni riguardo alla verifica, consulta Verifica degli indirizzi e-mail e dei domini in Amazon SES nella Guida per gli sviluppatori di Amazon Simple Email Service.

- Per i dettagli sull'API, consulta AWS CLI Command ListIdentitiesReference.

`send-email`

Il seguente esempio di codice mostra come utilizzare`send-email`.

AWS CLI

Invio di e-mail formattate utilizzando Amazon SES

Nell'esempio seguente viene utilizzato il comando `send-email` per inviare un messaggio e-mail formattato:

```
aws ses send-email --from sender@example.com --destination file://destination.json
 --message file://message.json
```

Output:

```
{
   "MessageId": "EXAMPLEf3a5efcd1-51adec81-d2a4-4e3f-9fe2-5d85c1b23783-000000"
}
```

La destinazione e il messaggio sono strutture di dati JSON salvate in file .json nella directory corrente. Tali file sono i seguenti:

`destination.json`:

```
{
```

```
  "ToAddresses":  ["recipient1@example.com", "recipient2@example.com"],
  "CcAddresses":  ["recipient3@example.com"],
  "BccAddresses": []
}
```

`message.json`:

```
{
   "Subject": {
       "Data": "Test email sent using the AWS CLI",
       "Charset": "UTF-8"
   },
   "Body": {
       "Text": {
           "Data": "This is the message body in text format.",
           "Charset": "UTF-8"
       },
       "Html": {
           "Data": "This message body contains HTML formatting. It can, for example,
 contain links like this one: <a class=\"ulink\" href=\"http://docs.aws.amazon.com/
ses/latest/DeveloperGuide\" target=\"_blank\">Amazon SES Developer Guide</a>.",
           "Charset": "UTF-8"
       }
   }
}
```

Sostituisci gli indirizzi e-mail del mittente e del destinatario con quelli che desideri utilizzare. Tieni presente che l'indirizzo e-mail del mittente deve essere verificato con Amazon SES. Fino a quando non ti viene concesso l'accesso alla produzione ad Amazon SES, devi verificare anche l'indirizzo e-mail di ciascun destinatario, a meno che il destinatario non sia il simulatore di mailbox Amazon SES. Per ulteriori informazioni riguardo alla verifica, consulta Verifica degli indirizzi e-mail e dei domini in Amazon SES nella Guida per gli sviluppatori di Amazon Simple Email Service.

L'ID del messaggio nell'output indica che la chiamata a send-email è stata completata correttamente.

Se non ricevi l'e-mail, controlla la casella della posta indesiderata.

Per ulteriori informazioni sull'invio di e-mail formattate, consulta Invio di e-mail formattate tramite l'API Amazon SES nella Guida per gli sviluppatori di Amazon Simple Email Service.

- Per i dettagli sull'API, consulta SendEmail AWS CLICommand Reference.

`send-raw-email`

Il seguente esempio di codice mostra come utilizzare`send-raw-email`.

AWS CLI

Invio di e-mail in formato RAW utilizzando Amazon SES

Nell'esempio seguente viene utilizzato il comando `send-raw-email` per inviare un messaggio con un allegato TXT:

```
aws ses send-raw-email --raw-message file://message.json
```

Output:

```
{
   "MessageId": "EXAMPLEf3f73d99b-c63fb06f-d263-41f8-a0fb-d0dc67d56c07-000000"
}
```

Il messaggio in formato RAW è una struttura di dati JSON salvata in un file denominato `message.json` nella directory corrente. Contiene i seguenti dati:

```
{
   "Data": "From: sender@example.com\nTo: recipient@example.com\nSubject: Test email
 sent using the AWS CLI (contains an attachment)\nMIME-Version: 1.0\nContent-type:
 Multipart/Mixed; boundary=\"NextPart\"\n\n--NextPart\nContent-Type: text/plain
\n\nThis is the message body.\n\n--NextPart\nContent-Type: text/plain;\nContent-
Disposition: attachment; filename=\"attachment.txt\"\n\nThis is the text in the
 attachment.\n\n--NextPart--"
}
```

Come si vede, quella denominata "Data" è una lunga stringa con all'interno tutto il contenuto RAW dell'e-mail in formato MIME, incluso un allegato chiamato attachment.txt.

Sostituisci sender@example.com e recipient@example.com con gli indirizzi che desideri utilizzare. Tieni presente che l'indirizzo e-mail del mittente deve essere verificato con Amazon SES. Fino a quando non ti viene concesso l'accesso alla produzione ad Amazon SES, devi verificare anche l'indirizzo e-mail del destinatario, a meno che il destinatario non sia il simulatore di mailbox Amazon SES. Per ulteriori informazioni riguardo alla verifica, consulta Verifica degli indirizzi e-mail e dei domini in Amazon SES nella Guida per gli sviluppatori di Amazon Simple Email Service.

L'ID del messaggio nell'output indica che la chiamata a ha send-raw-email avuto esito positivo.

Se non ricevi l'e-mail, controlla la casella della posta indesiderata.

Per ulteriori informazioni sull'invio di e-mail in formato RAW, consulta Invio di e-mail in formato RAW tramite l'API Amazon SES nella Guida per gli sviluppatori di Amazon Simple Email Service.

- Per i dettagli sull'API, consulta SendRawEmail AWS CLICommand Reference.

`set-identity-dkim-enabled`

Il seguente esempio di codice mostra come utilizzare`set-identity-dkim-enabled`.

AWS CLI

Come abilitare o disabilitare Easy DKIM per un'identità verificata da Amazon SES

L'esempio seguente utilizza il comando `set-identity-dkim-enabled` per disabilitare DKIM per un indirizzo e-mail verificato:

```
aws ses set-identity-dkim-enabled --identity user@example.com --no-dkim-enabled
```

Per ulteriori informazioni su Easy DKIM, consulta Easy DKIM in Amazon SES nella Guida per gli sviluppatori di Amazon Simple Email Service.

- Per i dettagli sull'API, consulta SetIdentityDkimEnabled AWS CLICommand Reference.

`set-identity-feedback-forwarding-enabled`

Il seguente esempio di codice mostra come utilizzare`set-identity-feedback-forwarding-enabled`.

AWS CLI

Come abilitare o disabilitare l'inoltro di feedback via e-mail relativo a mancato recapito e reclamo per un'identità verificata da Amazon SES

L'esempio seguente utilizza il comando `set-identity-feedback-forwarding-enabled` per abilitare un indirizzo e-mail verificato alla ricezione tramite e-mail di notifiche di mancato recapito e di reclamo:

```
aws ses set-identity-feedback-forwarding-enabled --identity user@example.com --
forwarding-enabled
```

Poiché devi ricevere notifiche di mancato recapito e reclamo tramite Amazon SNS o inoltro di feedback via e-mail, puoi disabilitare quest'ultimo solo se selezioni un argomento Amazon SNS per entrambi i tipi di notifica.

Per ulteriori informazioni sulle notifiche, consulta Utilizzo delle notifiche con Amazon SES nella Guida per gli sviluppatori di Amazon Simple Email Service.

- Per i dettagli sull'API, consulta SetIdentityFeedbackForwardingEnabled AWS CLICommand Reference.

`set-identity-notification-topic`

Il seguente esempio di codice mostra come utilizzare`set-identity-notification-topic`.

AWS CLI

Per impostare l'argomento Amazon SNS su cui Amazon SES pubblicherà notifiche di rimbalzo, reclamo e and/or consegna per un'identità verificata

L'esempio seguente utilizza il comando `set-identity-notification-topic` per specificare l'argomento Amazon SNS a cui un indirizzo e-mail verificato riceverà le notifiche di mancato recapito:

```
aws ses set-identity-notification-topic --identity user@example.com --notification-
type Bounce --sns-topic arn:aws:sns:us-east-1:EXAMPLE65304:MyTopic
```

Per ulteriori informazioni sulle notifiche, consulta Utilizzo delle notifiche con Amazon SES nella Guida per gli sviluppatori di Amazon Simple Email Service.

- Per i dettagli sull'API, consulta Command SetIdentityNotificationTopicReference AWS CLI .

`verify-domain-dkim`

Il seguente esempio di codice mostra come utilizzare`verify-domain-dkim`.

AWS CLI

Come generare i token DKIM di un dominio verificato per la firma DKIM con Amazon SES

L'esempio seguente utilizza il comando `verify-domain-dkim` per generare token DKIM per un dominio che è stato verificato con Amazon SES:

```
aws ses verify-domain-dkim --domain example.com
```

Output:

```
{
   "DkimTokens": [
       "EXAMPLEq76owjnks3lnluwg65scbemvw",
       "EXAMPLEi3dnsj67hstzaj673klariwx2",
       "EXAMPLEwfbtcukvimehexktmdtaz6naj"
   ]
}
```

Per configurare DKIM, devi utilizzare i token DKIM restituiti per aggiornare le impostazioni DNS del tuo dominio con record CNAME che puntano a chiavi pubbliche DKIM ospitate da Amazon SES. Per ulteriori informazioni, consulta Easy DKIM in Amazon SES nella Guida per gli sviluppatori di Amazon Simple Email Service.

- Per i dettagli sull'API, consulta VerifyDomainDkim AWS CLICommand Reference.

`verify-domain-identity`

Il seguente esempio di codice mostra come utilizzare`verify-domain-identity`.

AWS CLI

Verifica di un dominio con Amazon SES

Nell'esempio seguente viene utilizzato il comando `verify-domain-identity` per verificare un dominio:

```
aws ses verify-domain-identity --domain example.com
```

Output:

```
{
   "VerificationToken": "eoEmxw+YaYhb3h3iVJHuXMJXqeu1q1/wwmvjuEXAMPLE"
}
```

Per completare la verifica del dominio, devi aggiungere un record TXT con il token di verifica restituito alle impostazioni DNS del tuo dominio. Per ulteriori informazioni, consulta Verifica dei domini nella Guida per gli sviluppatori di Amazon Simple Email Service.

- Per i dettagli sull'API, consulta VerifyDomainIdentity AWS CLICommand Reference.

`verify-email-identity`

Il seguente esempio di codice mostra come utilizzare`verify-email-identity`.

AWS CLI

Aggiungere e verificare un indirizzo e-mail con Amazon SES

Nell'esempio seguente viene utilizzato il comando `verify-email-identity` per verificare un indirizzo e-mail:

```
aws ses verify-email-identity --email-address user@example.com
```

Prima di poter inviare e-mail usando Amazon SES, è necessario verificare il dominio o l'indirizzo da cui si intende inviare l'e-mail per dimostrarne la proprietà. Se non disponi ancora dell'accesso alla produzione, devi verificare anche tutti gli indirizzi e-mail a cui invii messaggi, ad eccezione degli indirizzi e-mail forniti dal Simulatore di mailbox di Amazon SES.

Dopo verify-email-identity la chiamata, l'indirizzo e-mail riceverà un'e-mail di verifica. L'utente deve fare clic sul link nell'e-mail per completare il processo di verifica.

Per ulteriori informazioni, consulta Verifica degli indirizzi e-mail in Amazon SES nella Guida per gli sviluppatori di Amazon Simple Email Service.

- Per i dettagli sull'API, consulta VerifyEmailIdentity AWS CLICommand Reference.

Esempi di utilizzo di Shield AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with Shield.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`associate-drt-log-bucket`

Il seguente esempio di codice mostra come utilizzare`associate-drt-log-bucket`.

AWS CLI

Come autorizzare il DRT ad accedere a un bucket Amazon S3

L'esempio `associate-drt-log-bucket` seguente crea un'associazione tra il DRT e il bucket S3 specificato. Ciò consente al DRT di accedere al bucket per conto dell'account:

```
aws shield associate-drt-log-bucket \
    --log-bucket flow-logs-for-website-lb
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Authorize the DDo S Response Team nella AWS Shield Advanced Developer Guide.

- Per i dettagli sull'API, consulta AssociateDrtLogBucket AWS CLICommand Reference.

`associate-drt-role`

Il seguente esempio di codice mostra come utilizzare`associate-drt-role`.

AWS CLI

Come autorizzare il DRT a mitigare gli attacchi potenziali per conto dell'utente

L'esempio `associate-drt-role` seguente crea un'associazione tra il DRT e il ruolo specificato. Il DRT può utilizzare il ruolo per accedere e gestire l'account.

```
aws shield associate-drt-role \
```

```
    --role-arn arn:aws:iam::123456789012:role/service-role/DrtRole
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Authorize the DDo S Response Team nella AWS Shield Advanced Developer Guide.

- Per i dettagli sull'API, consulta AssociateDrtRole AWS CLICommand Reference.

`create-protection`

Il seguente esempio di codice mostra come utilizzare`create-protection`.

AWS CLI

Per abilitare la protezione AWS Shield Advanced per una singola AWS risorsa

L'`create-protection`esempio seguente abilita la protezione Shield Advanced per la AWS CloudFront distribuzione specificata.

```
aws shield create-protection \
    --name "Protection for CloudFront distribution" \
    --resource-arn arn:aws:cloudfront::123456789012:distribution/E198WC25FXOWY8
```

Output:

```
{
    "ProtectionId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
}
```

Per ulteriori informazioni, consulta Specificare le risorse da proteggere nella Guida per gli sviluppatori di AWS Shield Advanced.

- Per i dettagli sull'API, vedere CreateProtectionin AWS CLI Command Reference.

`create-subscription`

Il seguente esempio di codice mostra come utilizzare`create-subscription`.

AWS CLI

Per abilitare la protezione AWS Shield Advanced per un account

L'esempio `create-subscription` seguente abilita la protezione Shield Advanced per l'account.

```
aws shield create-subscription
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Getting Started with AWS Shield Advanced nella AWS Shield Advanced Developer Guide.

- Per i dettagli sull'API, consulta CreateSubscription AWS CLICommand Reference.

`delete-protection`

Il seguente esempio di codice mostra come utilizzare`delete-protection`.

AWS CLI

Per rimuovere la protezione AWS Shield Advanced da una AWS risorsa

L'`delete-protection`esempio seguente rimuove la protezione AWS Shield Advanced specificata.

```
aws shield delete-protection \
    --protection-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Rimuovere AWS Shield Advanced da una AWS risorsa nella AWS Shield Advanced Developer Guide.

- Per i dettagli sull'API, consulta DeleteProtection AWS CLICommand Reference.

`describe-attack`

Il seguente esempio di codice mostra come utilizzare`describe-attack`.

AWS CLI

Come recuperare una descrizione dettagliata di un attacco

L'`describe-attack`esempio seguente mostra i dettagli sull'attacco DDo S con l'ID di attacco specificato. È possibile ottenere un attacco IDs eseguendo il `list-attacks` comando.

```
aws shield describe-attack --attack-id a1b2c3d4-5678-90ab-cdef-EXAMPLE22222
```

Output:

```
{
    "Attack": {
        "AttackId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
        "ResourceArn": "arn:aws:elasticloadbalancing:us-
west-2:123456789012:loadbalancer/testElb",
        "SubResources": [
            {
                "Type": "IP",
                "Id": "192.0.2.2",
                "AttackVectors": [
                    {
                        "VectorType": "SYN_FLOOD",
                        "VectorCounters": [
                            {
                                "Name": "SYN_FLOOD_BPS",
                                "Max": 982184.0,
                                "Average": 982184.0,
                                "Sum": 11786208.0,
                                "N": 12,
                                "Unit": "BPS"
                            }
                        ]
                    }
                ],
                "Counters": []
            },
            {
                "Type": "IP",
                "Id": "192.0.2.3",
                "AttackVectors": [
                    {
                        "VectorType": "SYN_FLOOD",
                        "VectorCounters": [
                            {
                                "Name": "SYN_FLOOD_BPS",
                                "Max": 982184.0,
```

```
                        "Average": 982184.0,
                        "Sum": 9821840.0,
                        "N": 10,
                        "Unit": "BPS"
                    }
                ]
            }
        ],
        "Counters": []
    },
    {
        "Type": "IP",
        "Id": "192.0.2.4",
        "AttackVectors": [
            {
                "VectorType": "SYN_FLOOD",
                "VectorCounters": [
                    {
                        "Name": "SYN_FLOOD_BPS",
                        "Max": 982184.0,
                        "Average": 982184.0,
                        "Sum": 7857472.0,
                        "N": 8,
                        "Unit": "BPS"
                    }
                ]
            }
        ],
        "Counters": []
    },
    {
        "Type": "IP",
        "Id": "192.0.2.5",
        "AttackVectors": [
            {
                "VectorType": "SYN_FLOOD",
                "VectorCounters": [
                    {
                        "Name": "SYN_FLOOD_BPS",
                        "Max": 982184.0,
                        "Average": 982184.0,
                        "Sum": 1964368.0,
                        "N": 2,
                        "Unit": "BPS"
```

```
                        }
                    ]
                }
            ],
            "Counters": []
        },
        {
            "Type": "IP",
            "Id": "2001:DB8::bcde:4321:8765:0:0",
            "AttackVectors": [
                {
                    "VectorType": "SYN_FLOOD",
                    "VectorCounters": [
                        {
                            "Name": "SYN_FLOOD_BPS",
                            "Max": 982184.0,
                            "Average": 982184.0,
                            "Sum": 1964368.0,
                            "N": 2,
                            "Unit": "BPS"
                        }
                    ]
                }
            ],
            "Counters": []
        },
        {
            "Type": "IP",
            "Id": "192.0.2.6",
            "AttackVectors": [
                {
                    "VectorType": "SYN_FLOOD",
                    "VectorCounters": [
                        {
                            "Name": "SYN_FLOOD_BPS",
                            "Max": 982184.0,
                            "Average": 982184.0,
                            "Sum": 1964368.0,
                            "N": 2,
                            "Unit": "BPS"
                        }
                    ]
                }
            ],
```

```
                "Counters": []
            }
        ],
        "StartTime": 1576024927.457,
        "EndTime": 1576025647.457,
        "AttackCounters": [],
        "AttackProperties": [
            {
                "AttackLayer": "NETWORK",
                "AttackPropertyIdentifier": "SOURCE_IP_ADDRESS",
                "TopContributors": [
                    {
                        "Name": "198.51.100.5",
                        "Value": 2024475682
                    },
                    {
                        "Name": "198.51.100.8",
                        "Value": 1311380863
                    },
                    {
                        "Name": "203.0.113.4",
                        "Value": 900599855
                    },
                    {
                        "Name": "198.51.100.4",
                        "Value": 769417366
                    },
                    {
                        "Name": "203.1.113.13",
                        "Value": 757992847
                    }
                ],
                "Unit": "BYTES",
                "Total": 92773354841
            },
            {
                "AttackLayer": "NETWORK",
                "AttackPropertyIdentifier": "SOURCE_COUNTRY",
                "TopContributors": [
                    {
                        "Name": "United States",
                        "Value": 80938161764
                    },
                    {
```

```
                        "Name": "Brazil",
                        "Value": 9929864330
                    },
                    {
                        "Name": "Netherlands",
                        "Value": 1635009446
                    },
                    {
                        "Name": "Mexico",
                        "Value": 144832971
                    },
                    {
                        "Name": "Japan",
                        "Value": 45369000
                    }
                ],
                "Unit": "BYTES",
                "Total": 92773354841
            },
            {
                "AttackLayer": "NETWORK",
                "AttackPropertyIdentifier": "SOURCE_ASN",
                "TopContributors": [
                    {
                        "Name": "12345",
                        "Value": 74953625841
                    },
                    {
                        "Name": "12346",
                        "Value": 4440087595
                    },
                    {
                        "Name": "12347",
                        "Value": 1635009446
                    },
                    {
                        "Name": "12348",
                        "Value": 1221230000
                    },
                    {
                        "Name": "12349",
                        "Value": 1199425294
                    }
                ],
```

```
                "Unit": "BYTES",
                "Total": 92755479921
            }
        ],
        "Mitigations": []
    }
}
```

Per ulteriori informazioni, consulta Review DDoS Incidents nella AWS Shield Advanced Developer Guide.

- Per i dettagli sull'API, consulta DescribeAttack AWS CLICommand Reference.

`describe-drt-access`

Il seguente esempio di codice mostra come utilizzare`describe-drt-access`.

AWS CLI

Come recuperare una descrizione delle autorizzazioni, il DRT deve mitigare gli attacchi per tuo conto

L'esempio `describe-drt-access` seguente recupera le autorizzazioni relative al ruolo e al bucket S3 di cui dispone il DRT, che gli consentono di rispondere a potenziali attacchi per tuo conto.

```
aws shield describe-drt-access
```

Output:

```
{
    "RoleArn": "arn:aws:iam::123456789012:role/service-role/DrtRole",
    "LogBucketList": [
        "flow-logs-for-website-lb"
    ]
}
```

Per ulteriori informazioni, consulta Authorize the DDo S Response Team nella AWS Shield Advanced Developer Guide.

- Per i dettagli sull'API, consulta DescribeDrtAccess AWS CLICommand Reference.

`describe-emergency-contact-settings`

Il seguente esempio di codice mostra come utilizzare`describe-emergency-contact-settings`.

AWS CLI

Come recuperare gli indirizzi e-mail di emergenza archiviati presso il DRT

L'esempio `describe-emergency-contact-settings` seguente recupera gli indirizzi e-mail archiviati nel DRT per l'account. Questi sono gli indirizzi che il DRT deve contattare quando risponde a un attacco sospetto.

```
aws shield describe-emergency-contact-settings
```

Output:

```
{
    "EmergencyContactList": [
        {
            "EmailAddress": "ops@example.com"
        },
        {
            "EmailAddress": "ddos-notifications@example.com"
       }
    ]
}
```

Per ulteriori informazioni, consulta How AWS Shield Works< https://docs.aws.amazon.com/waf/latest/developerguide/ddos -overview.html> nella Shield Advanced Developer Guide.AWS

- Per i dettagli sull'API, consulta Command Reference. DescribeEmergencyContactSettingsAWS CLI

`describe-protection`

Il seguente esempio di codice mostra come utilizzare`describe-protection`.

AWS CLI

Per recuperare i dettagli per una AWS protezione Shield Advanced

L'esempio `describe-protection` seguente visualizza i dettagli sulla protezione Shield Advanced con l'ID specificato. È possibile ottenere protezione IDs eseguendo il `list-protections` comando.

```
aws shield describe-protection \
    --protection-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Output:

```
{
    "Protection": {
        "Id": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
        "Name": "1.2.3.4",
        "ResourceArn": "arn:aws:ec2:us-west-2:123456789012:eip-allocation/
eipalloc-0ac1537af40742a6d"
    }
}
```

Per ulteriori informazioni, consulta Specificare le risorse da proteggere nella Guida per gli sviluppatori di AWS Shield Advanced.

- Per i dettagli sull'API, consulta DescribeProtection AWS CLICommand Reference.

`describe-subscription`

Il seguente esempio di codice mostra come utilizzare`describe-subscription`.

AWS CLI

Per recuperare i dettagli della protezione AWS Shield Advanced per l'account

L'esempio `describe-subscription` seguente mostra i dettagli sulla protezione Shield Advanced fornita per l'account:

```
aws shield describe-subscription
```

Output:

```
{
    "Subscription": {
        "StartTime": 1534368978.0,
        "EndTime": 1597613778.0,
```

```
        "TimeCommitmentInSeconds": 63244800,
        "AutoRenew": "ENABLED",
        "Limits": [
            {
                "Type": "GLOBAL_ACCELERATOR",
                "Max": 1000
            },
            {
                "Type": "ROUTE53_HOSTED_ZONE",
                "Max": 1000
            },
            {
                "Type": "CF_DISTRIBUTION",
                "Max": 1000
            },
            {
                "Type": "ELB_LOAD_BALANCER",
                "Max": 1000
            },
            {
                "Type": "EC2_ELASTIC_IP_ALLOCATION",
                "Max": 1000
            }
        ]
    }
}
```

Per ulteriori informazioni, consulta How AWS Shield Works nella AWS Shield Advanced Developer Guide.

- Per i dettagli sull'API, consulta DescribeSubscription AWS CLICommand Reference.

`disassociate-drt-log-bucket`

Il seguente esempio di codice mostra come utilizzare`disassociate-drt-log-bucket`.

AWS CLI

Come rimuovere l'autorizzazione a DRT ad accedere a un bucket Amazon S3 per tuo conto

L'esempio `disassociate-drt-log-bucket` seguente rimuove l'associazione tra il DRT e il bucket S3 specificato. Una volta completato questo comando, il DRT non può più accedere al bucket per conto dell'account.

```
aws shield disassociate-drt-log-bucket \
    --log-bucket flow-logs-for-website-lb
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Authorize the DDo S Response Team nella AWS Shield Advanced Developer Guide.

- Per i dettagli sull'API, consulta DisassociateDrtLogBucket AWS CLICommand Reference.

`disassociate-drt-role`

Il seguente esempio di codice mostra come utilizzare`disassociate-drt-role`.

AWS CLI

Come rimuovere l'autorizzazione al DRT per mitigare gli attacchi potenziali per tuo conto

L'esempio `disassociate-drt-role` seguente rimuove l'associazione tra il DRT e l'account. Dopo questa chiamata, il DRT non può più accedere o gestire l'account.

```
aws shield disassociate-drt-role
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Authorize the DDo S Response Team nella AWS Shield Advanced Developer Guide.

- Per i dettagli sull'API, consulta DisassociateDrtRole AWS CLICommand Reference.

`get-subscription-state`

Il seguente esempio di codice mostra come utilizzare`get-subscription-state`.

AWS CLI

Per recuperare lo stato attuale dell'abbonamento AWS Shield Advanced dell'account

L'esempio `get-subscription-state` seguente recupera lo stato della protezione Shield Advanced per l'account.

```
aws shield get-subscription-state
```

Output:

```
{
    "SubscriptionState": "ACTIVE"
}
```

Per ulteriori informazioni, consulta How AWS Shield Works nella AWS Shield Advanced Developer Guide.

- Per i dettagli sull'API, consulta GetSubscriptionState AWS CLICommand Reference.

`list-attacks`

Il seguente esempio di codice mostra come utilizzare`list-attacks`.

AWS CLI

Per recuperare i riepiloghi degli attacchi da Shield Advanced AWS

L'`list-attacks`esempio seguente recupera i riepiloghi degli attacchi per la AWS CloudFront distribuzione specificata durante il periodo di tempo specificato. La risposta include un attacco IDs che è possibile fornire al `describe-attack` comando per ottenere informazioni dettagliate su un attacco.

```
aws shield list-attacks \
    --resource-arns arn:aws:cloudfront::12345678910:distribution/E1PXMP22ZVFAOR \
    --start-time FromInclusive=1529280000,ToExclusive=1529300000
```

Output:

```
{
    "AttackSummaries": [
        {
            "AttackId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
            "ResourceArn": "arn:aws:cloudfront::123456789012:distribution/
E1PXMP22ZVFAOR",
            "StartTime": 1529280000.0,
            "EndTime": 1529449200.0,
            "AttackVectors": [
```

```
                {
                    "VectorType": "SYN_FLOOD"
                }
            ]
        }
    ]
}
```

Per ulteriori informazioni, consulta Review DDoS Incidents nella AWS Shield Advanced Developer Guide.

- Per i dettagli sull'API, consulta ListAttacks AWS CLICommand Reference.

`list-protections`

Il seguente esempio di codice mostra come utilizzare`list-protections`.

AWS CLI

Per recuperare i riepiloghi della protezione da Shield Advanced AWS

L'esempio `list-protections` seguente recupera i riepiloghi delle protezioni abilitate per l'account.

```
aws shield list-protections
```

Output:

```
{
    "Protections": [
        {
            "Id": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
            "Name": "Protection for CloudFront distribution",
            "ResourceArn": "arn:aws:cloudfront::123456789012:distribution/
E198WC25FXOWY8"
        }
    ]
}
```

Per ulteriori informazioni, consulta Specificare le risorse da proteggere nella Guida per gli sviluppatori di AWS Shield Advanced.

- Per i dettagli sull'API, consulta AWS CLI Command ListProtectionsReference.

`update-emergency-contact-settings`

Il seguente esempio di codice mostra come utilizzareu`update-emergency-contact-settings`.

AWS CLI

Come definire gli indirizzi e-mail di emergenza archiviati presso il DRT

L'esempio `update-emergency-contact-settings` seguente definisce due indirizzi e-mail che il DRT deve contattare quando risponde a un attacco sospetto.

```
aws shield update-emergency-contact-settings \
        --emergency-contact-list EmailAddress=ops@example.com EmailAddress=ddos-
notifications@example.com
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta How AWS Shield Works nella AWS Shield Advanced Developer Guide.

- Per i dettagli sull'API, consulta UpdateEmergencyContactSettings AWS CLICommand Reference.

`update-subscription`

Il seguente esempio di codice mostra come utilizzareu`update-subscription`.

AWS CLI

Per modificare l'abbonamento AWS Shield Advanced dell'account

L'`update-subscription`esempio seguente abilita il rinnovo automatico dell'abbonamento AWS Shield Advanced per l'account.

```
aws shield update-subscription \
    --auto-renew ENABLED
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta How AWS Shield Works nella AWS Shield Advanced Developer Guide.

- Per i dettagli sull'API, consulta UpdateSubscription AWS CLICommand Reference.

Esempi di firmatari che utilizzano AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with Signer.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`cancel-signing-profile`

Il seguente esempio di codice mostra come utilizzare`cancel-signing-profile`.

AWS CLI

Come eliminare un profilo di firma

L'`cancel-signing-profile`esempio seguente rimuove un profilo di firma esistente da AWS Signer.

```
aws signer cancel-signing-profile \
    --profile-name MyProfile1
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta CancelSigningProfile AWS CLICommand Reference.

`describe-signing-job`

Il seguente esempio di codice mostra come utilizzare`describe-signing-job`.

AWS CLI

Come visualizzare i dettagli su un processo di firma

L'esempio `describe-signing-job` seguente visualizza i dettagli del processo di firma specificato.

```
aws signer describe-signing-job \
    --job-id 2065c468-73e2-4385-a6c9-0123456789abc
```

Output:

```
{
    "status": "Succeeded",
    "completedAt": 1568412037,
    "platformId": "AmazonFreeRTOS-Default",
    "signingMaterial": {
        "certificateArn": "arn:aws:acm:us-
west-2:123456789012:certificate/6a55389b-306b-4e8c-a95c-0123456789abc"
    },
    "statusReason": "Signing Succeeded",
    "jobId": "2065c468-73e2-4385-a6c9-0123456789abc",
    "source": {
        "s3": {
            "version": "PNyFaUTgsQh5ZdMCcoCe6pT1gOpgB_M4",
            "bucketName": "signer-source",
            "key": "MyCode.rb"
        }
    },
    "profileName": "MyProfile2",
    "signedObject": {
        "s3": {
            "bucketName": "signer-destination",
            "key": "signed-2065c468-73e2-4385-a6c9-0123456789abc"
        }
    },
    "requestedBy": "arn:aws:iam::123456789012:user/maria",
    "createdAt": 1568412036
}
```

- Per i dettagli sull'API, consulta DescribeSigningJob AWS CLICommand Reference.

`get-signing-platform`

Il seguente esempio di codice mostra come utilizzare`get-signing-platform`.

AWS CLI

Come visualizzare i dettagli su una piattaforma di firma

L'esempio `get-signing-platform` seguente visualizza i dettagli della piattaforma di firma specificata.

```
aws signer get-signing-platform \
    --platform-id AmazonFreeRTOS-TI-CC3220SF
```

Output:

```
{
    "category": "AWS",
    "displayName": "Amazon FreeRTOS SHA1-RSA CC3220SF-Format",
    "target": "SHA1-RSA-TISHA1",
    "platformId": "AmazonFreeRTOS-TI-CC3220SF",
    "signingConfiguration": {
        "encryptionAlgorithmOptions": {
            "defaultValue": "RSA",
            "allowedValues": [
                "RSA"
            ]
        },
        "hashAlgorithmOptions": {
            "defaultValue": "SHA1",
            "allowedValues": [
                "SHA1"
            ]
        }
    },
    "maxSizeInMB": 16,
    "partner": "AmazonFreeRTOS",
    "signingImageFormat": {
        "defaultFormat": "JSONEmbedded",
        "supportedFormats": [
```

```
            "JSONEmbedded"
        ]
    }
}
```

- Per i dettagli sull'API, consulta GetSigningPlatform AWS CLICommand Reference.

`get-signing-profile`

Il seguente esempio di codice mostra come utilizzare`get-signing-profile`.

AWS CLI

Come visualizzare i dettagli su un profilo di firma

L'esempio `get-signing-profile` seguente visualizza i dettagli del profilo di firma specificato.

```
aws signer get-signing-profile \
    --profile-name MyProfile3
```

Output:

```
{
    "platformId": "AmazonFreeRTOS-TI-CC3220SF",
    "profileName": "MyProfile3",
    "status": "Active",
    "signingMaterial": {
        "certificateArn": "arn:aws:acm:us-
west-2:123456789012:certificate/6a55389b-306b-4e8c-a95c-0123456789abc"
    }
}
```

- Per i dettagli sull'API, consulta GetSigningProfile AWS CLICommand Reference.

`list-signing-jobs`

Il seguente esempio di codice mostra come utilizzare`list-signing-jobs`.

AWS CLI

Come elencare tutti i processi di firma

L'esempio `list-signing-jobs` seguente mostra i dettagli di tutti i processi di firma per l'account.

```
aws signer list-signing-jobs
```

In questo esempio, vengono restituiti due processi, uno riuscito e uno no.

```
{
    "jobs": [
        {
            "status": "Succeeded",
            "signingMaterial": {
                "certificateArn": "arn:aws:acm:us-
west-2:123456789012:certificate/6a55389b-306b-4e8c-a95c-0123456789abc"
            },
            "jobId": "2065c468-73e2-4385-a6c9-0123456789abc",
            "source": {
                "s3": {
                    "version": "PNyFaUTgsQh5ZdMCcoCe6pT1gOpgB_M4",
                    "bucketName": "signer-source",
                    "key": "MyCode.rb"
                }
            },
            "signedObject": {
                "s3": {
                    "bucketName": "signer-destination",
                    "key": "signed-2065c468-73e2-4385-a6c9-0123456789abc"
                }
            },
            "createdAt": 1568412036
        },
        {
            "status": "Failed",
            "source": {
                "s3": {
                    "version": "PNyFaUTgsQh5ZdMCcoCe6pT1gOpgB_M4",
                    "bucketName": "signer-source",
                    "key": "MyOtherCode.rb"
                }
            },
            "signingMaterial": {
                "certificateArn": "arn:aws:acm:us-
west-2:123456789012:certificate/6a55389b-306b-4e8c-a95c-0123456789abc"
```

```
            },
            "createdAt": 1568402690,
            "jobId": "74d9825e-22fc-4a0d-b962-0123456789abc"
        }
    ]
}
```

- Per i dettagli sull'API, consulta ListSigningJobs AWS CLICommand Reference.

`list-signing-platforms`

Il seguente esempio di codice mostra come utilizzare`list-signing-platforms`.

AWS CLI

Come elencare tutte le piattaforme di firma

L'esempio `list-signing-platforms` seguente mostra i dettagli di tutte le piattaforme di firma disponibili.

```
aws signer list-signing-platforms
```

Output:

```
{
    "platforms": [
        {
            "category": "AWS",
            "displayName": "AWS IoT Device Management SHA256-ECDSA ",
            "target": "SHA256-ECDSA",
            "platformId": "AWSIoTDeviceManagement-SHA256-ECDSA",
            "signingConfiguration": {
                "encryptionAlgorithmOptions": {
                    "defaultValue": "ECDSA",
                    "allowedValues": [
                        "ECDSA"
                    ]
                },
                "hashAlgorithmOptions": {
                    "defaultValue": "SHA256",
                    "allowedValues": [
                        "SHA256"
```

```
                ]
            }
        },
        "maxSizeInMB": 2048,
        "partner": "AWSIoTDeviceManagement",
        "signingImageFormat": {
            "defaultFormat": "JSONDetached",
            "supportedFormats": [
                "JSONDetached"
            ]
        }
    },
    {
        "category": "AWS",
        "displayName": "Amazon FreeRTOS SHA1-RSA CC3220SF-Format",
        "target": "SHA1-RSA-TISHA1",
        "platformId": "AmazonFreeRTOS-TI-CC3220SF",
        "signingConfiguration": {
            "encryptionAlgorithmOptions": {
                "defaultValue": "RSA",
                "allowedValues": [
                    "RSA"
                ]
            },
            "hashAlgorithmOptions": {
                "defaultValue": "SHA1",
                "allowedValues": [
                    "SHA1"
                ]
            }
        },
        "maxSizeInMB": 16,
        "partner": "AmazonFreeRTOS",
        "signingImageFormat": {
            "defaultFormat": "JSONEmbedded",
            "supportedFormats": [
                "JSONEmbedded"
            ]
        }
    },
    {
        "category": "AWS",
        "displayName": "Amazon FreeRTOS SHA256-ECDSA",
        "target": "SHA256-ECDSA",
```

```
            "platformId": "AmazonFreeRTOS-Default",
            "signingConfiguration": {
                "encryptionAlgorithmOptions": {
                    "defaultValue": "ECDSA",
                    "allowedValues": [
                        "ECDSA"
                    ]
                },
                "hashAlgorithmOptions": {
                    "defaultValue": "SHA256",
                    "allowedValues": [
                        "SHA256"
                    ]
                }
            },
            "maxSizeInMB": 16,
            "partner": "AmazonFreeRTOS",
            "signingImageFormat": {
                "defaultFormat": "JSONEmbedded",
                "supportedFormats": [
                    "JSONEmbedded"
                ]
            }
        }
    ]
}
```


- Per i dettagli sull'API, consulta ListSigningPlatforms AWS CLICommand Reference.

`list-signing-profiles`

Il seguente esempio di codice mostra come utilizzare`list-signing-profiles`.

AWS CLI

Come elencare tutti i profili di firma

L'esempio `list-signing-profiles` seguente mostra i dettagli di tutti i profili di firma per l'account.

```
aws signer list-signing-profiles
```

Output:

```
{
    "profiles": [
        {
            "platformId": "AmazonFreeRTOS-TI-CC3220SF",
            "profileName": "MyProfile4",
            "status": "Active",
            "signingMaterial": {
                "certificateArn": "arn:aws:acm:us-
west-2:123456789012:certificate/6a55389b-306b-4e8c-a95c-0123456789abc"
            }
        },
        {
            "platformId": "AWSIoTDeviceManagement-SHA256-ECDSA",
            "profileName": "MyProfile5",
            "status": "Active",
            "signingMaterial": {
                "certificateArn": "arn:aws:acm:us-
west-2:123456789012:certificate/6a55389b-306b-4e8c-a95c-0123456789abc"
            }
        }
    ]
}
```

- Per i dettagli sull'API, consulta ListSigningProfiles AWS CLICommand Reference.

`put-signing-profile`

Il seguente esempio di codice mostra come utilizzare`put-signing-profile`.

AWS CLI

Come creare un profilo di firma

L'esempio `put-signing-profile` seguente crea un profilo di firma utilizzando il certificato e la piattaforma specificati.

```
aws signer put-signing-profile \
    --profile-name MyProfile6 \
    --signing-material certificateArn=arn:aws:acm:us-
west-2:123456789012:certificate/6a55389b-306b-4e8c-a95c-0123456789abc \
    --platform AmazonFreeRTOS-TI-CC3220SF
```

Output:

```
{
    "arn": "arn:aws:signer:us-west-2:123456789012:/signing-profiles/MyProfile6"
}
```

- Per i dettagli sull'API, consulta PutSigningProfile AWS CLICommand Reference.

`start-signing-job`

Il seguente esempio di codice mostra come utilizzare`start-signing-job`.

AWS CLI

Come avviare un processo di firma

L'esempio `start-signing-job` seguente avvia un processo di firma sul codice trovato nella fonte specificata. Utilizza il profilo specificato per eseguire la firma e inserisce il codice firmato nella destinazione specificata.

```
aws signer start-signing-job \
    --source 's3={bucketName=signer-
source,key=MyCode.rb,version=PNyFaUTgsQh5ZdMCcoCe6pT1gOpgB_M4}' \
    --destination 's3={bucketName=signer-destination,prefix=signed-}' \
    --profile-name MyProfile7
```

L'output è l'ID del processo di firma.

```
{
    "jobId": "2065c468-73e2-4385-a6c9-0123456789abc"
}
```

- Per i dettagli sull'API, consulta StartSigningJob AWS CLICommand Reference.

Esempi di utilizzo di Snowball Edge AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with Snowball Edge.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`get-snowball-usage`

Il seguente esempio di codice mostra come utilizzare`get-snowball-usage`.

AWS CLI

Come ottenere informazioni sulla limitazione dei servizi Snowball per l'account

L'esempio `get-snowball-usage` seguente restituisce informazioni sulla limitazione dei servizi Snowball per l'account, nonché il numero di Snowball utilizzati dall'account.

```
aws snowball get-snowball-usage
```

Output:

```
{
    "SnowballLimit": 1,
    "SnowballsInUse": 0
}
```

PER ulteriori informazioni, consulta Limiti di AWS Snowball Edge nella Guida per gli sviluppatori di AWS Snowball.

- Per i dettagli sull'API, consulta GetSnowballUsage AWS CLICommand Reference.

`list-jobs`

Il seguente esempio di codice mostra come utilizzare`list-jobs`.

AWS CLI

Come elencare i processi attuali in Snowball presenti nel tuo account

L'esempio `list-jobs` seguente mostra un array di oggetti `JobListEntry`. In questo esempio, viene elencato un singolo processo.

```
aws snowball list-jobs
```

Output:

```
{
    "JobListEntries": [
        {
            "CreationDate": 2016-09-27T14:50Z,
             "Description": "Important Photos 2016-08-11",
             "IsMaster": TRUE,
             "JobId": "ABCd1e324fe-022f-488e-a98b-3b0566063db1",
             "JobState": "Complete",
             "JobType": "IMPORT",
             "SnowballType": "EDGE"
        }
   ]
}
```

Per ulteriori informazioni, consulta Jobs for AWS Snowball Edge nella AWS Snowball Developer Guide.

- Per i dettagli sull'API, consulta AWS CLI Command ListJobsReference.

Esempi di utilizzo di Amazon SNS AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface con Amazon SNS.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Scenari: esempi di codice che mostrano come eseguire un'attività specifica chiamando più funzioni all'interno dello stesso servizio o combinate con altri Servizi AWS.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni
- Scenari

Azioni

`add-permission`

Il seguente esempio di codice mostra come usare`add-permission`.

AWS CLI

Come aggiungere un'autorizzazione a un argomento

L'`add-permission`esempio seguente aggiunge l'autorizzazione per l' AWS account `987654321098` a utilizzare l'`Publish`azione con l'argomento specificato in AWS account`123456789012`.

```
aws sns add-permission \
    --topic-arn arn:aws:sns:us-west-2:123456789012:MyTopic \
    --label Publish-Permission \
    --aws-account-id 987654321098 \
    --action-name Publish
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, vedere AddPermissionin AWS CLI Command Reference.

`check-if-phone-number-is-opted-out`

Il seguente esempio di codice mostra come utilizzare`check-if-phone-number-is-opted-out`.

AWS CLI

Controllo delle impostazioni di opt-out dei messaggi SMS per un numero di telefono

L'`check-if-phone-number-is-opted-out`esempio seguente verifica se al numero di telefono specificato è stata disattivata la ricezione di messaggi SMS dall' AWS account corrente.

```
aws sns check-if-phone-number-is-opted-out \
    --phone-number +1555550100
```

Output:

```
{
    "isOptedOut": false
}
```

- Per i dettagli sull'API, vedere CheckIfPhoneNumberIsOptedOutin AWS CLI Command Reference.

`confirm-subscription`

Il seguente esempio di codice mostra come utilizzare`confirm-subscription`.

AWS CLI

Conferma di una sottoscrizione

Attraverso il comando `confirm-subscription` seguente viene completato il processo di conferma avviato effettuando la sottoscrizione a un argomento SNS denominato `my-topic`. Il parametro --token proviene dal messaggio di conferma inviato all'endpoint di notifica specificato nella chiamata subscribe.

```
aws sns confirm-subscription \
    --topic-arn arn:aws:sns:us-west-2:123456789012:my-topic \
    --
token 2336412f37fb687f5d51e6e241d7700ae02f7124d8268910b858cb4db727ceeb2474bb937929d3bdd7ce5a
```

Output:

```
{
    "SubscriptionArn": "arn:aws:sns:us-west-2:123456789012:my-
topic:8a21d249-4329-4871-acc6-7be709c6ea7f"
}
```

- Per i dettagli sull'API, consulta ConfirmSubscription AWS CLICommand Reference.

`create-platform-application`

Il seguente esempio di codice mostra come utilizzare`create-platform-application`.

AWS CLI

Come creare un'applicazione della piattaforma

L'esempio `create-platform-application` seguente crea un'applicazione della piattaforma Google Firebase utilizzando le credenziali di piattaforma specificate.

```
aws sns create-platform-application \
    --name MyApplication \
    --platform GCM \
    --attributes PlatformCredential=EXAMPLEabcd12345jklm67890stuv12345bcdef
```

Output:

```
{
    "PlatformApplicationArn": "arn:aws:sns:us-west-2:123456789012:app/GCM/
MyApplication"
}
```

- Per i dettagli sull'API, consulta CreatePlatformApplication AWS CLICommand Reference.

`create-topic`

Il seguente esempio di codice mostra come utilizzare`create-topic`.

AWS CLI

Creazione di un argomento SNS

Nell'esempio `create-topic` seguente viene creato un argomento SNS denominato `my-topic`.

```
aws sns create-topic \
    --name my-topic
```

Output:

```
{
```

```
    "ResponseMetadata": {
        "RequestId": "1469e8d7-1642-564e-b85d-a19b4b341f83"
    },
    "TopicArn": "arn:aws:sns:us-west-2:123456789012:my-topic"
}
```

Per ulteriori informazioni, consulta Using the AWS Command Line Interface with Amazon SQS e Amazon SNS nella Command Line Interface AWS User Guide.

- Per i dettagli sull'API, consulta AWS CLI Command CreateTopicReference.

`delete-endpoint`

Il seguente esempio di codice mostra come utilizzare`delete-endpoint`.

AWS CLI

Come eliminare un endpoint dell'applicazione della piattaforma

L'esempio `delete-endpoint` seguente elimina l'endpoint dell'applicazione della piattaforma specificato.

```
aws sns delete-endpoint \
    --endpoint-arn arn:aws:sns:us-west-2:123456789012:endpoint/GCM/
MyApplication/12345678-abcd-9012-efgh-345678901234
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta DeleteEndpoint AWS CLICommand Reference.

`delete-platform-application`

Il seguente esempio di codice mostra come utilizzare`delete-platform-application`.

AWS CLI

Come eliminare un endpoint un'applicazione della piattaforma

L'esempio `delete-platform-application` seguente elimina l'applicazione della piattaforma specificata.

```
aws sns delete-platform-application \
```

```
    --platform-application-arn arn:aws:sns:us-west-2:123456789012:app/ADM/
MyApplication
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta DeletePlatformApplication AWS CLICommand Reference.

`delete-topic`

Il seguente esempio di codice mostra come utilizzare`delete-topic`.

AWS CLI

Eliminazione di un argomento SNS

Nell'esempio `delete-topic` seguente viene eliminato l'argomento SNS specificato.

```
aws sns delete-topic \
    --topic-arn "arn:aws:sns:us-west-2:123456789012:my-topic"
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta DeleteTopic AWS CLICommand Reference.

`get-endpoint-attributes`

Il seguente esempio di codice mostra come utilizzare`get-endpoint-attributes`.

AWS CLI

Come elencare gli attributi dell'endpoint dell'applicazione della piattaforma

L'esempio `get-endpoint-attributes` seguente elenca gli attributi per l'endpoint dell'applicazione della piattaforma specificata.

```
aws sns get-endpoint-attributes \
    --endpoint-arn arn:aws:sns:us-west-2:123456789012:endpoint/GCM/
MyApplication/12345678-abcd-9012-efgh-345678901234
```

Output:

```
{
```

```
    "Attributes": {
        "Enabled": "true",
        "Token": "EXAMPLE12345..."
    }
}
```

- Per i dettagli sull'API, consulta GetEndpointAttributes AWS CLICommand Reference.

`get-platform-application-attributes`

Il seguente esempio di codice mostra come utilizzare`get-platform-application-attributes`.

AWS CLI

Come elencare gli attributi dell'applicazione della piattaforma

L'esempio `get-platform-application-attributes` seguente elenca gli attributi per l'applicazione della piattaforma specificata.

```
aws sns get-platform-application-attributes \
    --platform-application-arn arn:aws:sns:us-west-2:123456789012:app/MPNS/
MyApplication
```

Output:

```
{
    "Attributes": {
        "Enabled": "true",
        "SuccessFeedbackSampleRate": "100"
    }
}
```

- Per i dettagli sull'API, consulta GetPlatformApplicationAttributes AWS CLICommand Reference.

`get-sms-attributes`

Il seguente esempio di codice mostra come utilizzare`get-sms-attributes`.

AWS CLI

Elencare gli attributi predefiniti dei messaggi SMS

Nell'esempio `get-sms-attributes` seguente vengono elencati gli attributi predefiniti per l'invio di messaggi SMS.

```
aws sns get-sms-attributes
```

Output:

```
{
    "attributes": {
        "DefaultSenderID": "MyName"
    }
}
```

- Per informazioni dettagliate sull'API, consulta GetSMSAttributes in AWS CLI Command Reference.

`get-subscription-attributes`

Il seguente esempio di codice mostra come utilizzare`get-subscription-attributes`.

AWS CLI

Come recuperare gli attributi di abbonamento a un argomento

Il comando `get-subscription-attributes` seguente visualizza gli attributi dell'abbonamento specificato. È possibile ottenere il valore `subscription-arn` dall'output del comando `list-subscriptions`.

```
aws sns get-subscription-attributes \
    --subscription-arn "arn:aws:sns:us-west-2:123456789012:my-
topic:8a21d249-4329-4871-acc6-7be709c6ea7f"
```

Output:

```
{
    "Attributes": {
        "Endpoint": "my-email@example.com",
        "Protocol": "email",
        "RawMessageDelivery": "false",
        "ConfirmationWasAuthenticated": "false",
```

```
        "Owner": "123456789012",
        "SubscriptionArn": "arn:aws:sns:us-west-2:123456789012:my-
topic:8a21d249-4329-4871-acc6-7be709c6ea7f",
        "TopicArn": "arn:aws:sns:us-west-2:123456789012:my-topic"
    }
}
```

- Per i dettagli sull'API, consulta GetSubscriptionAttributes AWS CLICommand Reference.

`get-topic-attributes`

Il seguente esempio di codice mostra come utilizzare`get-topic-attributes`.

AWS CLI

Recupero degli attributi di un argomento

Nell'esempio `get-topic-attributes` seguente vengono visualizzati gli attributi per l'argomento specificato.

```
aws sns get-topic-attributes \
    --topic-arn "arn:aws:sns:us-west-2:123456789012:my-topic"
```

Output:

```
{
    "Attributes": {
        "SubscriptionsConfirmed": "1",
        "DisplayName": "my-topic",
        "SubscriptionsDeleted": "0",
        "EffectiveDeliveryPolicy": "{\"http\":{\"defaultHealthyRetryPolicy\":
{\"minDelayTarget\":20,\"maxDelayTarget\":20,\"numRetries\":3,\"numMaxDelayRetries
\":0,\"numNoDelayRetries\":0,\"numMinDelayRetries\":0,\"backoffFunction\":\"linear
\"},\"disableSubscriptionOverrides\":false}}",
        "Owner": "123456789012",
        "Policy": "{\"Version\":\"2008-10-17\",\"Id\":\"__default_policy_ID\",
\"Statement\":[{\"Sid\":\"__default_statement_ID\",\"Effect\":\"Allow\",\"Principal
\":{\"AWS\":\"*\"},\"Action\":[\"SNS:Subscribe\",\"SNS:ListSubscriptionsByTopic
\",\"SNS:DeleteTopic\",\"SNS:GetTopicAttributes\",\"SNS:Publish\",
\"SNS:RemovePermission\",\"SNS:AddPermission\",\"SNS:SetTopicAttributes\"],
\"Resource\":\"arn:aws:sns:us-west-2:123456789012:my-topic\",\"Condition\":
{\"StringEquals\":{\"AWS:SourceOwner\":\"0123456789012\"}}}]}",
```

```
        "TopicArn": "arn:aws:sns:us-west-2:123456789012:my-topic",
        "SubscriptionsPending": "0"
    }
}
```

- Per i dettagli sull'API, consulta GetTopicAttributes AWS CLICommand Reference.

`list-endpoints-by-platform-application`

Il seguente esempio di codice mostra come utilizzare`list-endpoints-by-platform-application`.

AWS CLI

Come elencare gli endpoint di un'applicazione di piattaforma

L'esempio `list-endpoints-by-platform-application` seguente elenca gli endpoint e gli attributi dell'endpoint per l'applicazione della piattaforma specificata.

```
aws sns list-endpoints-by-platform-application \
    --platform-application-arn arn:aws:sns:us-west-2:123456789012:app/GCM/
MyApplication
```

Output:

```
{
    "Endpoints": [
        {
            "Attributes": {
                "Token": "EXAMPLE12345...,
                "Enabled": "true"
            },
            "EndpointArn": "arn:aws:sns:us-west-2:123456789012:endpoint/GCM/
MyApplication/12345678-abcd-9012-efgh-345678901234"
        }
    ]
}
```

- Per i dettagli sull'API, consulta ListEndpointsByPlatformApplication AWS CLICommand Reference.

`list-phone-numbers-opted-out`

Il seguente esempio di codice mostra come utilizzare`list-phone-numbers-opted-out`.

AWS CLI

Elencare le opzioni di opt-out dei messaggi SMS

Nell'esempio `list-phone-numbers-opted-out` seguente sono elencati i numeri di telefono per i quali sono abilitate le opzioni di opt-out per la ricezione di SMS.

```
aws sns list-phone-numbers-opted-out
```

Output:

```
{
    "phoneNumbers": [
        "+15555550100"
    ]
}
```

- Per i dettagli sull'API, consulta ListPhoneNumbersOptedOut AWS CLICommand Reference.

`list-platform-applications`

Il seguente esempio di codice mostra come utilizzare`list-platform-applications`.

AWS CLI

Come elencare le applicazioni della piattaforma

L'esempio `list-platform-applications` seguente elenca le applicazioni della piattaforma per ADM e MPNS.

```
aws sns list-platform-applications
```

Output:

```
{
    "PlatformApplications": [
        {
```

```
            "PlatformApplicationArn": "arn:aws:sns:us-west-2:123456789012:app/ADM/
MyApplication",
            "Attributes": {
                "SuccessFeedbackSampleRate": "100",
                "Enabled": "true"
            }
        },
        {
            "PlatformApplicationArn": "arn:aws:sns:us-west-2:123456789012:app/MPNS/
MyOtherApplication",
            "Attributes": {
                "SuccessFeedbackSampleRate": "100",
                "Enabled": "true"
            }
        }
    ]
}
```

- Per i dettagli sull'API, consulta ListPlatformApplications AWS CLICommand Reference.

`list-subscriptions-by-topic`

Il seguente esempio di codice mostra come utilizzare`list-subscriptions-by-topic`.

AWS CLI

Come elencare le sottoscrizioni associate a un argomento

Il comando `list-subscriptions-by-topic` seguente recupera un elenco di abbonamenti SNS associati all'argomento specificato.

```
aws sns list-subscriptions-by-topic \
    --topic-arn "arn:aws:sns:us-west-2:123456789012:my-topic"
```

Output:

```
{
    "Subscriptions": [
        {
            "Owner": "123456789012",
            "Endpoint": "my-email@example.com",
            "Protocol": "email",
```

```
            "TopicArn": "arn:aws:sns:us-west-2:123456789012:my-topic",
            "SubscriptionArn": "arn:aws:sns:us-west-2:123456789012:my-
topic:8a21d249-4329-4871-acc6-7be709c6ea7f"
        }
    ]
}
```

- Per i dettagli sull'API, consulta ListSubscriptionsByTopic AWS CLICommand Reference.

`list-subscriptions`

Il seguente esempio di codice mostra come utilizzare`list-subscriptions`.

AWS CLI

Elencare le sottoscrizioni SNS

L'`list-subscriptions`esempio seguente visualizza un elenco degli abbonamenti SNS presenti nell'account AWS .

```
aws sns list-subscriptions
```

Output:

```
{
    "Subscriptions": [
        {
            "Owner": "123456789012",
            "Endpoint": "my-email@example.com",
            "Protocol": "email",
            "TopicArn": "arn:aws:sns:us-west-2:123456789012:my-topic",
            "SubscriptionArn": "arn:aws:sns:us-west-2:123456789012:my-
topic:8a21d249-4329-4871-acc6-7be709c6ea7f"
        }
    ]
}
```

- Per i dettagli sull'API, consulta AWS CLI Command ListSubscriptionsReference.

`list-tags-for-resource`

Il seguente esempio di codice mostra come utilizzare`list-tags-for-resource`.

AWS CLI

Come elencare i tag di un argomento

L'esempio `list-tags-for-resource` seguente elenca i tag per l'argomento Amazon SNS specificato.

```
aws sns list-tags-for-resource \
    --resource-arn arn:aws:sns:us-west-2:123456789012:MyTopic
```

Output:

```
{
    "Tags": [
        {
            "Key": "Team",
            "Value": "Alpha"
        }
    ]
}
```

- Per i dettagli sull'API, consulta ListTagsForResource AWS CLICommand Reference.

`list-topics`

Il seguente esempio di codice mostra come utilizzare`list-topics`.

AWS CLI

Elenco degli argomenti SNS

L'`list-topics`esempio seguente elenca tutti gli argomenti SNS del tuo AWS account.

```
aws sns list-topics
```

Output:

```
{
    "Topics": [
        {
            "TopicArn": "arn:aws:sns:us-west-2:123456789012:my-topic"
        }
```

```
    ]
}
```

- Per i dettagli sull'API, consulta ListTopics AWS CLICommand Reference.

`opt-in-phone-number`

Il seguente esempio di codice mostra come utilizzare`opt-in-phone-number`.

AWS CLI

Come attivare i messaggi SMS

L'esempio `opt-in-phone-number` seguente elenca i numeri di telefono per i quali sono abilitate le opzioni per la ricezione di SMS.

```
aws sns opt-in-phone-number \
    --phone-number +15555550100
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta OptInPhoneNumber AWS CLICommand Reference.

`publish`

Il seguente esempio di codice mostra come utilizzare`publish`.

AWS CLI

Esempio 1: pubblicazione di un messaggio in un argomento

Nell'esempio `publish` viene pubblicato il messaggio indicato in un argomento SNS specificato. Il messaggio proviene da un file di testo che consente di includere interruzioni di riga.

```
aws sns publish \
    --topic-arn "arn:aws:sns:us-west-2:123456789012:my-topic" \
    --message file://message.txt
```

Contenuto di `message.txt`:

```
Hello World
```

```
Second Line
```

Output:

```
{
    "MessageId": "123a45b6-7890-12c3-45d6-111122223333"
}
```

Esempio 2: pubblicare un messaggio SMS su un numero di telefono

Nell'esempio `publish` seguente viene pubblicato il messaggio `Hello world!` sul numero di telefono`+1-555-555-0100`.

```
aws sns publish \
    --message "Hello world!" \
    --phone-number +1-555-555-0100
```

Output:

```
{
    "MessageId": "123a45b6-7890-12c3-45d6-333322221111"
}
```

- Per informazioni dettagliate sull'API, consulta Publish nella documentazione di riferimento dei comandi della AWS CLI .

`put-data-protection-policy`

Il seguente esempio di codice mostra come utilizzare`put-data-protection-policy`.

AWS CLI

Come impostare la policy di protezione dei dati

Esempio 1: impedire agli editori di pubblicare messaggi con CreditCardNumber

L'`put-data-protection-policy`esempio seguente impedisce agli editori di pubblicare messaggi con. CreditCardNumber

```
aws sns put-data-protection-policy \
    --resource-arn arn:aws:sns:us-east-1:123456789012:mytopic \
```

```
    --data-protection-policy "{\"Name\":\"data_protection_policy\",\"Description
\":\"Example data protection policy\",\"Version\":\"2021-06-01\",\"Statement
\":[{\"DataDirection\":\"Inbound\",\"Principal\":[\"*\"],\"DataIdentifier\":
[\"arn:aws:dataprotection::aws:data-identifier/CreditCardNumber\"],\"Operation\":
{\"Deny\":{}}}]}"
```

Questo comando non produce alcun output.

Esempio 2: come caricare i parametri da un file

L'esempio `put-data-protection-policy` seguente carica i parametri da un file.

```
aws sns put-data-protection-policy \
    --resource-arn arn:aws:sns:us-west-2:123456789012:MyTopic \
    --data-protection-policy file://policy.json
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consultate Command PutDataProtectionPolicyReference AWS CLI .

`remove-permission`

Il seguente esempio di codice mostra come utilizzare`remove-permission`.

AWS CLI

Come rimuovere un'autorizzazione da un argomento

L'esempio `remove-permission`esempio seguente rimuove l'autorizzazione `Publish-Permission` dall'argomento specificato.

```
aws sns remove-permission \
    --topic-arn arn:aws:sns:us-west-2:123456789012:MyTopic \
    --label Publish-Permission
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta RemovePermission AWS CLICommand Reference.

`set-endpoint-attributes`

Il seguente esempio di codice mostra come utilizzare`set-endpoint-attributes`.

AWS CLI

Come configurare gli attributi degli endpoint

L'esempio `set-endpoint-attributes` seguente disabilita l'endpoint dell'applicazione della piattaforma specificato.

```
aws sns set-endpoint-attributes \
    --endpoint-arn arn:aws:sns:us-west-2:123456789012:endpoint/GCM/
MyApplication/12345678-abcd-9012-efgh-345678901234 \
    --attributes Enabled=false
```

Output:

```
{
    "Attributes": {
        "Enabled": "false",
        "Token": "EXAMPLE12345..."
    }
}
```

- Per i dettagli sull'API, consulta SetEndpointAttributes AWS CLICommand Reference.

`set-platform-application-attributes`

Il seguente esempio di codice mostra come utilizzare`set-platform-application-attributes`.

AWS CLI

Come impostare gli attributi dell'applicazione della piattaforma

L'esempio `set-platform-application-attributes` seguente imposta l'attributo `EventDeliveryFailure` per l'applicazione di piattaforma specificata sull'ARN dell'argomento Amazon SNS specificato.

```
aws sns set-platform-application-attributes \
    --platform-application-arn arn:aws:sns:us-west-2:123456789012:app/GCM/
MyApplication \
    --attributes EventDeliveryFailure=arn:aws:sns:us-
west-2:123456789012:AnotherTopic
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta SetPlatformApplicationAttributes AWS CLICommand Reference.

`set-sms-attributes`

Il seguente esempio di codice mostra come utilizzare`set-sms-attributes`.

AWS CLI

Impostazione degli attributi dei messaggi SMS

Nell'esempio `set-sms-attributes` seguente l'ID mittente predefinito per i messaggi SMS viene impostato su `MyName`.

```
aws sns set-sms-attributes \
    --attributes DefaultSenderID=MyName
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta Set SMSAttributes in AWS CLI Command Reference.

`set-subscription-attributes`

Il seguente esempio di codice mostra come utilizzare`set-subscription-attributes`.

AWS CLI

Impostazione degli attributi della sottoscrizione

Nell'esempio `set-subscription-attributes` seguente viene impostato l'attributo `RawMessageDelivery` su una sottoscrizione SQS.

```
aws sns set-subscription-attributes \
    --subscription-arn arn:aws:sns:us-
east-1:123456789012:mytopic:f248de18-2cf6-578c-8592-b6f1eaa877dc \
    --attribute-name RawMessageDelivery \
    --attribute-value true
```

Questo comando non produce alcun output.

Nell'esempio `set-subscription-attributes` seguente viene impostato un attributo `FilterPolicy` su una sottoscrizione SQS.

```
aws sns set-subscription-attributes \
    --subscription-arn arn:aws:sns:us-
east-1:123456789012:mytopic:f248de18-2cf6-578c-8592-b6f1eaa877dc \
    --attribute-name FilterPolicy \
    --attribute-value "{ \"anyMandatoryKey\": [\"any\", \"of\", \"these\"] }"
```

Questo comando non produce alcun output.

Nell'esempio `set-subscription-attributes` seguente viene rimosso l'attributo `FilterPolicy` su una sottoscrizione SQS.

```
aws sns set-subscription-attributes \
    --subscription-arn arn:aws:sns:us-
east-1:123456789012:mytopic:f248de18-2cf6-578c-8592-b6f1eaa877dc \
    --attribute-name FilterPolicy \
    --attribute-value "{}"
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta SetSubscriptionAttributes AWS CLICommand Reference.

`set-topic-attributes`

Il seguente esempio di codice mostra come utilizzare`set-topic-attributes`.

AWS CLI

Impostazione di un attributo per un argomento

Nell'esempio `set-topic-attributes` seguente vengono impostati gli attributi `DisplayName` per l'argomento specificato.

```
aws sns set-topic-attributes \
    --topic-arn arn:aws:sns:us-west-2:123456789012:MyTopic \
    --attribute-name DisplayName \
    --attribute-value MyTopicDisplayName
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta SetTopicAttributes AWS CLICommand Reference.

`subscribe`

Il seguente esempio di codice mostra come utilizzare`subscribe`.

AWS CLI

Effettuare la sottoscrizione a un argomento

Attraverso il comando `subscribe` seguente viene effettuata la sottoscrizione all'argomento specificato utilizzando un indirizzo e-mail.

```
aws sns subscribe \
    --topic-arn arn:aws:sns:us-west-2:123456789012:my-topic \
    --protocol email \
    --notification-endpoint my-email@example.com
```

Output:

```
{
    "SubscriptionArn": "pending confirmation"
}
```

- Per informazioni dettagliate sull'API, consulta Subscribe nella documentazione di riferimento dei comandi della AWS CLI .

`tag-resource`

Il seguente esempio di codice mostra come utilizzare`tag-resource`.

AWS CLI

Aggiungere un tag a un argomento

Nell'esempio `tag-resource` seguente viene aggiunto un tag di metadati all'argomento Amazon SNS specificato.

```
aws sns tag-resource \
    --resource-arn arn:aws:sns:us-west-2:123456789012:MyTopic \
    --tags Key=Team,Value=Alpha
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta TagResource AWS CLICommand Reference.

`unsubscribe`

Il seguente esempio di codice mostra come utilizzare`unsubscribe`.

AWS CLI

Annullamento della sottoscrizione a un argomento

Nell'esempio `unsubscribe` seguente viene eliminata la sottoscrizione specificata a un argomento.

```
aws sns unsubscribe \
    --subscription-arn arn:aws:sns:us-west-2:0123456789012:my-
topic:8a21d249-4329-4871-acc6-7be709c6ea7f
```

Questo comando non produce alcun output.

- Per informazioni dettagliate sull'API, consulta Unsubscribe nella documentazione di riferimento dei comandi della AWS CLI .

`untag-resource`

Il seguente esempio di codice mostra come utilizzare`untag-resource`.

AWS CLI

Come rimuovere un tag da un argomento

L'esempio `untag-resource` seguente elimina qualsiasi tag con le chiavi specificate dall'argomento Amazon SNS specificato.

```
aws sns untag-resource \
    --resource-arn arn:aws:sns:us-west-2:123456789012:MyTopic \
    --tag-keys Team
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta UntagResource AWS CLICommand Reference.

Scenari

Creazione di un endpoint di piattaforma per notifiche push

L'esempio di codice seguente mostra come creare un endpoint di piattaforma per notifiche push di Amazon SNS.

AWS CLI

Creazione di un endpoint dell'applicazione della piattaforma

Nell'esempio `create-platform-endpoint` seguente viene creato un endpoint per l'applicazione della piattaforma indicata utilizzando il token specificato.

```
aws sns create-platform-endpoint \
    --platform-application-arn arn:aws:sns:us-west-2:123456789012:app/GCM/
MyApplication \
    --token EXAMPLE12345...
```

Output:

```
{
      "EndpointArn": "arn:aws:sns:us-west-2:1234567890:endpoint/GCM/
MyApplication/12345678-abcd-9012-efgh-345678901234"
}
```

Esempi di utilizzo di Amazon SQS AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface con Amazon SQS.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`add-permission`

Il seguente esempio di codice mostra come usare`add-permission`.

AWS CLI

Come aggiungere le autorizzazioni a una coda

Questo esempio consente all' AWS account specificato di inviare messaggi alla coda specificata.

Comando:

```
aws sqs add-permission --queue-url https://sqs.us-east-1.amazonaws.com/80398EXAMPLE/
MyQueue --label SendMessagesFromMyQueue --aws-account-ids 12345EXAMPLE --
actions SendMessage
```

Output:

```
None.
```

- Per i dettagli sull'API, vedere AddPermissionin AWS CLI Command Reference.

`cancel-message-move-task`

Il seguente esempio di codice mostra come utilizzare`cancel-message-move-task`.

AWS CLI

Come annullare un'operazione di spostamento di un messaggio

L'esempio `cancel-message-move-task` seguente annulla l'attività di spostamento del messaggio specificata.

```
aws sqs cancel-message-move-task \
    --task-handle AQEB6nR4...HzlvZQ==
```

Output:

```
{
    "ApproximateNumberOfMessagesMoved": 102
```

```
}
```

Per ulteriori informazioni consulta Autorizzazioni API Amazon SQS: riferimento a operazioni e risorse nella Guida per gli sviluppatori.

- Per i dettagli sull'API, consulta CancelMessageMoveTask AWS CLICommand Reference.

`change-message-visibility-batch`

Il seguente esempio di codice mostra come utilizzare`change-message-visibility-batch`.

AWS CLI

Come modificare la visibilità del timeout di più messaggi come batch

Questo esempio modifica la visibilità del timeout dei 2 messaggi specificati impostandola su 10 ore (10 ore * 60 minuti * 60 secondi).

Comando:

```
aws sqs change-message-visibility-batch --queue-url https://sqs.us-east-1.amazonaws.com/80398EXAMPLE/MyQueue --entries file://change-message-visibility-batch.json
```

File di input (change-message-visibility-batch.json):

```
[
  {
    "Id": "FirstMessage",
        "ReceiptHandle": "AQEBhz2q...Jf3kaw==",
        "VisibilityTimeout": 36000
  },
  {
    "Id": "SecondMessage",
        "ReceiptHandle": "AQEBkTUH...HifSnw==",
        "VisibilityTimeout": 36000
  }
]
```

Output:

```
{
```

```
  "Successful": [
    {
      "Id": "SecondMessage"
    },
    {
      "Id": "FirstMessage"
    }
  ]
}
```

- Per i dettagli sull'API, consulta AWS CLI Command ChangeMessageVisibilityBatchReference.

`change-message-visibility`

Il seguente esempio di codice mostra come utilizzare`change-message-visibility`.

AWS CLI

Come modificare la visibilità del timeout di un messaggio

Questo esempio modifica la visibilità del timeout del messaggio specificato impostandola su 10 ore (10 ore * 60 minuti * 60 secondi).

Comando:

```
aws sqs change-message-visibility --queue-url https://sqs.us-
east-1.amazonaws.com/80398EXAMPLE/MyQueue --receipt-handle AQEBTpyI...t6HyQg== --
visibility-timeout 36000
```

Output:

```
None.
```

- Per i dettagli sull'API, consulta ChangeMessageVisibility AWS CLICommand Reference.

`create-queue`

Il seguente esempio di codice mostra come utilizzare`create-queue`.

AWS CLI

Come creare una coda

Questo esempio crea una coda con il nome specificato, imposta il periodo di conservazione dei messaggi su 3 giorni (3 giorni * 24 ore * 60 minuti * 60 secondi) e imposta la coda DLQ della coda sulla coda specificata con un conteggio massimo di ricezione pari a 1.000 messaggi.

Comando:

```
aws sqs create-queue --queue-name MyQueue --attributes file://create-queue.json
```

File di input (create-queue.json):

```
{
  "RedrivePolicy": "{\"deadLetterTargetArn\":\"arn:aws:sqs:us-
east-1:80398EXAMPLE:MyDeadLetterQueue\",\"maxReceiveCount\":\"1000\"}",
  "MessageRetentionPeriod": "259200"
}
```

Output:

```
{
  "QueueUrl": "https://queue.amazonaws.com/80398EXAMPLE/MyQueue"
}
```

- Per i dettagli sull'API, consulta CreateQueue AWS CLICommand Reference.

`delete-message-batch`

Il seguente esempio di codice mostra come utilizzare`delete-message-batch`.

AWS CLI

Come eliminare più messaggi in batch

Questo esempio elimina i messaggi specificati.

Comando:

```
aws sqs delete-message-batch --queue-url https://sqs.us-
east-1.amazonaws.com/80398EXAMPLE/MyQueue --entries file://delete-message-batch.json
```

File di input (delete-message-batch.json):

```
[
  {
        "Id": "FirstMessage",
        "ReceiptHandle": "AQEB1mgl...Z4GuLw=="
  },
  {
    "Id": "SecondMessage",
        "ReceiptHandle": "AQEBLsYM...VQubAA=="
  }
]
```

Output:

```
{
  "Successful": [
    {
      "Id": "FirstMessage"
    },
    {
      "Id": "SecondMessage"
    }
  ]
}
```

- Per i dettagli sull'API, consulta AWS CLI Command DeleteMessageBatchReference.

`delete-message`

Il seguente esempio di codice mostra come utilizzare`delete-message`.

AWS CLI

Come eliminare un messaggio

Questo esempio elimina il messaggio specificato.

Comando:

```
aws sqs delete-message --queue-url https://sqs.us-east-1.amazonaws.com/80398EXAMPLE/MyQueue --receipt-handle AQEBRXTo...q2doVA==
```

Output:

```
None.
```

- Per i dettagli sull'API, consulta DeleteMessage AWS CLICommand Reference.

`delete-queue`

Il seguente esempio di codice mostra come utilizzare`delete-queue`.

AWS CLI

Come eliminare una coda

Questo esempio elimina la coda specificata.

Comando:

```
aws sqs delete-queue --queue-url https://sqs.us-east-1.amazonaws.com/80398EXAMPLE/
MyNewerQueue
```

Output:

```
None.
```

- Per i dettagli sull'API, consulta DeleteQueue AWS CLICommand Reference.

`get-queue-attributes`

Il seguente esempio di codice mostra come utilizzare`get-queue-attributes`.

AWS CLI

Come ottenere gli attributi di una coda

Questo esempio ottiene tutti gli attributi della coda specificata.

Comando:

```
aws sqs get-queue-attributes --queue-url https://sqs.us-
east-1.amazonaws.com/80398EXAMPLE/MyQueue --attribute-names All
```

Output:

```
{
  "Attributes": {
    "ApproximateNumberOfMessagesNotVisible": "0",
    "RedrivePolicy": "{\"deadLetterTargetArn\":\"arn:aws:sqs:us-
east-1:80398EXAMPLE:MyDeadLetterQueue\",\"maxReceiveCount\":1000}",
    "MessageRetentionPeriod": "345600",
    "ApproximateNumberOfMessagesDelayed": "0",
    "MaximumMessageSize": "262144",
    "CreatedTimestamp": "1442426968",
    "ApproximateNumberOfMessages": "0",
    "ReceiveMessageWaitTimeSeconds": "0",
    "DelaySeconds": "0",
    "VisibilityTimeout": "30",
    "LastModifiedTimestamp": "1442426968",
    "QueueArn": "arn:aws:sqs:us-east-1:80398EXAMPLE:MyNewQueue"
  }
}
```

Questo esempio ottiene solo la dimensione massima dei messaggi e gli attributi di timeout della visibilità per la coda specificata.

Comando:

```
aws sqs get-queue-attributes --queue-url https://sqs.us-
east-1.amazonaws.com/80398EXAMPLE/MyNewQueue --attribute-
names MaximumMessageSize VisibilityTimeout
```

Output:

```
{
  "Attributes": {
    "VisibilityTimeout": "30",
    "MaximumMessageSize": "262144"
  }
}
```

- Per i dettagli sull'API, consulta GetQueueAttributes AWS CLICommand Reference.

`get-queue-url`

Il seguente esempio di codice mostra come utilizzare`get-queue-url`.

AWS CLI

Come ottenere l'URL di una coda

Questo esempio ottiene l'URL della coda specificata.

Comando:

```
aws sqs get-queue-url --queue-name MyQueue
```

Output:

```
{
  "QueueUrl": "https://queue.amazonaws.com/80398EXAMPLE/MyQueue"
}
```

- Per i dettagli sull'API, consulta GetQueueUrl AWS CLICommand Reference.

`list-dead-letter-source-queues`

Il seguente esempio di codice mostra come utilizzare`list-dead-letter-source-queues`.

AWS CLI

Come elencare le code DLQ di origine

Questo esempio elenca le code associate alla coda DLQ di origine specificata.

Comando:

```
aws sqs list-dead-letter-source-queues --queue-url https://sqs.us-
east-1.amazonaws.com/80398EXAMPLE/MyDeadLetterQueue
```

Output:

```
{
  "queueUrls": [
    "https://queue.amazonaws.com/80398EXAMPLE/MyQueue",
    "https://queue.amazonaws.com/80398EXAMPLE/MyOtherQueue"
  ]
}
```

- Per i dettagli sull'API, consulta ListDeadLetterSourceQueues AWS CLICommand Reference.

`list-message-move-tasks`

Il seguente esempio di codice mostra come utilizzare`list-message-move-tasks`.

AWS CLI

Come elencare le attività di spostamento dei messaggi

L'esempio `list-message-move-tasks` seguente elenca le 2 attività di spostamento dei messaggi più recenti nella coda specificata.

```
aws sqs list-message-move-tasks \
    --source-arn arn:aws:sqs:us-west-2:80398EXAMPLE:MyQueue \
    --max-results 2
```

Output:

```
{
    "Results": [
        {
            "TaskHandle": "AQEB6nR4...HzlvZQ==",
            "Status": "RUNNING",
            "SourceArn": "arn:aws:sqs:us-west-2:80398EXAMPLE:MyQueue1",
            "DestinationArn": "arn:aws:sqs:us-west-2:80398EXAMPLE:MyQueue2",
            "MaxNumberOfMessagesPerSecond": 50,
            "ApproximateNumberOfMessagesMoved": 203,
            "ApproximateNumberOfMessagesToMove": 30,
            "StartedTimestamp": 1442428276921
        },

        {
            "Status": "COMPLETED",
            "SourceArn": "arn:aws:sqs:us-west-2:80398EXAMPLE:MyQueue1",
            "DestinationArn": "arn:aws:sqs:us-west-2:80398EXAMPLE:MyQueue2",
            "ApproximateNumberOfMessagesMoved": 29,
            "ApproximateNumberOfMessagesToMove": 0,
            "StartedTimestamp": 1342428272093
        }
    ]
}
```

Per ulteriori informazioni consulta Autorizzazioni API Amazon SQS: riferimento a operazioni e risorse nella Guida per gli sviluppatori.

- Per i dettagli sull'API, consulta ListMessageMoveTasks AWS CLICommand Reference.

`list-queue-tags`

Il seguente esempio di codice mostra come utilizzare`list-queue-tags`.

AWS CLI

Come elencare tutti i tag di allocazione dei costi per una coda

L'esempio `list-queue-tags` seguente visualizza tutti i tag di allocazione dei costi associati alla coda specificata.

```
aws sqs list-queue-tags \
    --queue-url https://sqs.us-west-2.amazonaws.com/123456789012/MyQueue
```

Output:

```
{
    "Tags": {
        "Team": "Alpha"
    }
}
```

Per ulteriori informazioni, consulta Elenco dei tag di allocazione dei costi nella Guida per sviluppatori di Amazon Simple Queue Service.

- Per i dettagli sull'API, consulta ListQueueTags AWS CLICommand Reference.

`list-queues`

Il seguente esempio di codice mostra come utilizzare`list-queues`.

AWS CLI

Come elencare le code

Questo esempio elenca tutte le code.

Comando:

```
aws sqs list-queues
```

Output:

```
{
  "QueueUrls": [
    "https://queue.amazonaws.com/80398EXAMPLE/MyDeadLetterQueue",
    "https://queue.amazonaws.com/80398EXAMPLE/MyQueue",
    "https://queue.amazonaws.com/80398EXAMPLE/MyOtherQueue",
    "https://queue.amazonaws.com/80398EXAMPLE/TestQueue1",
        "https://queue.amazonaws.com/80398EXAMPLE/TestQueue2"
  ]
}
```

Questo esempio elenca solo le code che iniziano con "My".

Comando:

```
aws sqs list-queues --queue-name-prefix My
```

Output:

```
{
  "QueueUrls": [
    "https://queue.amazonaws.com/80398EXAMPLE/MyDeadLetterQueue",
    "https://queue.amazonaws.com/80398EXAMPLE/MyQueue",
    "https://queue.amazonaws.com/80398EXAMPLE/MyOtherQueue"
  ]
}
```

- Per i dettagli sull'API, consulta ListQueues AWS CLICommand Reference.

`purge-queue`

Il seguente esempio di codice mostra come utilizzare`purge-queue`.

AWS CLI

Come rimuovere una coda

Questo esempio elimina tutti i messaggi nella coda specificata.

Comando:

```
aws sqs purge-queue --queue-url https://sqs.us-east-1.amazonaws.com/80398EXAMPLE/MyNewQueue
```

Output:

```
None.
```

- Per i dettagli sull'API, consulta PurgeQueue AWS CLICommand Reference.

`receive-message`

Il seguente esempio di codice mostra come utilizzare`receive-message`.

AWS CLI

Come ricevere un messaggio

Questo esempio riceve fino a 10 messaggi disponibili e restituisce tutti gli attributi disponibili.

Comando:

```
aws sqs receive-message --queue-url https://sqs.us-east-1.amazonaws.com/80398EXAMPLE/MyQueue --attribute-names All --message-attribute-names All --max-number-of-messages 10
```

Output:

```
{
  "Messages": [
    {
      "Body": "My first message.",
      "ReceiptHandle": "AQEBzbVv...fqNzFw==",
      "MD5OfBody": "1000f835...a35411fa",
      "MD5OfMessageAttributes": "9424c491...26bc3ae7",
      "MessageId": "d6790f8d-d575-4f01-bc51-40122EXAMPLE",
      "Attributes": {
        "ApproximateFirstReceiveTimestamp": "1442428276921",
        "SenderId": "AIDAIAZKMSNQ7TEXAMPLE",
```

```
        "ApproximateReceiveCount": "5",
        "SentTimestamp": "1442428276921"
      },
      "MessageAttributes": {
        "PostalCode": {
          "DataType": "String",
          "StringValue": "ABC123"
        },
        "City": {
          "DataType": "String",
          "StringValue": "Any City"
        }
      }
    }
  ]
}
```

Questo esempio riceve il prossimo messaggio disponibile, restituendo solo gli SentTimestamp attributi SenderId and e l'attributo PostalCode message.

Comando:

```
aws sqs receive-message --queue-url https://sqs.us-
east-1.amazonaws.com/80398EXAMPLE/MyQueue --attribute-names SenderId SentTimestamp
 --message-attribute-names PostalCode
```

Output:

```
{
  "Messages": [
    {
      "Body": "My first message.",
      "ReceiptHandle": "AQEB6nR4...HzlvZQ==",
      "MD5OfBody": "1000f835...a35411fa",
      "MD5OfMessageAttributes": "b8e89563...e088e74f",
      "MessageId": "d6790f8d-d575-4f01-bc51-40122EXAMPLE",
      "Attributes": {
        "SenderId": "AIDAIAZKMSNQ7TEXAMPLE",
        "SentTimestamp": "1442428276921"
      },
      "MessageAttributes": {
        "PostalCode": {
          "DataType": "String",
```

```
            "StringValue": "ABC123"
          }
        }
      }
    ]
}
```

- Per i dettagli sull'API, consulta ReceiveMessage AWS CLICommand Reference.

`remove-permission`

Il seguente esempio di codice mostra come utilizzare`remove-permission`.

AWS CLI

Come rimuovere un'autorizzazione

Questo esempio rimuove l'autorizzazione con l'etichetta specificata dalla coda specificata.

Comando:

```
aws sqs remove-permission --queue-url https://sqs.us-
east-1.amazonaws.com/80398EXAMPLE/MyQueue --label SendMessagesFromMyQueue
```

Output:

```
None.
```

- Per i dettagli sull'API, consulta RemovePermission AWS CLICommand Reference.

`send-message-batch`

Il seguente esempio di codice mostra come utilizzare`send-message-batch`.

AWS CLI

Come inviare più messaggi in batch

Questo esempio invia alla coda specificata 2 messaggi con il corpo del messaggio, il periodo di ritardo e gli attributi del messaggio specificati.

Comando:

```
aws sqs send-message-batch --queue-url https://sqs.us-
east-1.amazonaws.com/80398EXAMPLE/MyQueue --entries file://send-message-batch.json
```

File di input (send-message-batch.json):

```
[
  {
    "Id": "FuelReport-0001-2015-09-16T140731Z",
        "MessageBody": "Fuel report for account 0001 on 2015-09-16 at 02:07:31 PM.",
        "DelaySeconds": 10,
        "MessageAttributes": {
          "SellerName": {
            "DataType": "String",
                "StringValue": "Example Store"
      },
          "City": {
        "DataType": "String",
        "StringValue": "Any City"
      },
          "Region": {
            "DataType": "String",
                "StringValue": "WA"
      },
          "PostalCode": {
            "DataType": "String",
                "StringValue": "99065"
          },
          "PricePerGallon": {
            "DataType": "Number",
                "StringValue": "1.99"
      }
        }
  },
  {
    "Id": "FuelReport-0002-2015-09-16T140930Z",
        "MessageBody": "Fuel report for account 0002 on 2015-09-16 at 02:09:30 PM.",
        "DelaySeconds": 10,
        "MessageAttributes": {
          "SellerName": {
            "DataType": "String",
                "StringValue": "Example Fuels"
      },
          "City": {
```

```
        "DataType": "String",
        "StringValue": "North Town"
      },
          "Region": {
            "DataType": "String",
                "StringValue": "WA"
      },
          "PostalCode": {
            "DataType": "String",
                "StringValue": "99123"
          },
          "PricePerGallon": {
            "DataType": "Number",
                "StringValue": "1.87"
      }
        }
  }
]
```

Output:

```
{
  "Successful": [
    {
      "MD5OfMessageBody": "203c4a38...7943237e",
      "MD5OfMessageAttributes": "10809b55...baf283ef",
      "Id": "FuelReport-0001-2015-09-16T140731Z",
      "MessageId": "d175070c-d6b8-4101-861d-adeb3EXAMPLE"
    },
    {
      "MD5OfMessageBody": "2cf0159a...c1980595",
      "MD5OfMessageAttributes": "55623928...ae354a25",
      "Id": "FuelReport-0002-2015-09-16T140930Z",
      "MessageId": "f9b7d55d-0570-413e-b9c5-a9264EXAMPLE"
    }
  ]
}
```

- Per i dettagli sull'API, consulta AWS CLI Command SendMessageBatchReference.

`send-message`

Il seguente esempio di codice mostra come utilizzare`send-message`.

AWS CLI

Come inviare un messaggio

Questo esempio invia alla coda specificata un messaggio con il corpo del messaggio, il periodo di ritardo e gli attributi del messaggio specificato.

Comando:

```
aws sqs send-message --queue-url https://sqs.us-east-1.amazonaws.com/80398EXAMPLE/MyQueue --message-body "Information about the largest city in Any Region." --delay-seconds 10 --message-attributes file://send-message.json
```

File di input (send-message.json):

```
{
  "City": {
    "DataType": "String",
    "StringValue": "Any City"
  },
  "Greeting": {
    "DataType": "Binary",
    "BinaryValue": "Hello, World!"
  },
  "Population": {
    "DataType": "Number",
    "StringValue": "1250800"
  }
}
```

Output:

```
{
  "MD5OfMessageBody": "51b0a325...39163aa0",
  "MD5OfMessageAttributes": "00484c68...59e48f06",
  "MessageId": "da68f62c-0c07-4bee-bf5f-7e856EXAMPLE"
}
```

- Per i dettagli sull'API, consulta SendMessage AWS CLICommand Reference.

`set-queue-attributes`

Il seguente esempio di codice mostra come utilizzare`set-queue-attributes`.

AWS CLI

Come impostare gli attributi della coda

Questo esempio imposta la coda specificata su un ritardo di distribuzione di 10 secondi, una dimensione massima del messaggio di 128 KB (128 KB * 1.024 byte), un periodo di conservazione dei messaggi di 3 giorni (3 giorni * 24 ore * 60 minuti * 60 secondi), un tempo di attesa per la ricezione dei messaggi di 20 secondi e un timeout di visibilità predefinito di 60 secondi. Questo esempio associa inoltre la coda DLQ specificata a un numero massimo di ricezione di 1.000 messaggi.

Comando:

```
aws sqs set-queue-attributes --queue-url https://sqs.us-east-1.amazonaws.com/80398EXAMPLE/MyNewQueue --attributes file://set-queue-attributes.json
```

File di input (set-queue-attributes.json):

```
{
  "DelaySeconds": "10",
  "MaximumMessageSize": "131072",
  "MessageRetentionPeriod": "259200",
  "ReceiveMessageWaitTimeSeconds": "20",
  "RedrivePolicy": "{\"deadLetterTargetArn\":\"arn:aws:sqs:us-east-1:80398EXAMPLE:MyDeadLetterQueue\",\"maxReceiveCount\":\"1000\"}",
  "VisibilityTimeout": "60"
}
```

Output:

```
None.
```

- Per i dettagli sull'API, consulta AWS CLI Command SetQueueAttributesReference.

`start-message-move-task`

Il seguente esempio di codice mostra come utilizzare`start-message-move-task`.

AWS CLI

Esempio 1: *come avviare un'attività di spostamento di messaggi*

L'esempio `start-message-move-task` seguente avvia un'operazione di spostamento dei messaggi per reindirizzare i messaggi dalla coda DLQ specificata alla coda di origine.

```
aws sqs start-message-move-task \
    --source-arn arn:aws:sqs:us-west-2:80398EXAMPLE:MyQueue
```

Output:

```
{
    "TaskHandle": "AQEB6nR4...HzlvZQ=="
}
```

Per ulteriori informazioni, consulta il seguente argomento in questa guida.

Esempio 2: *come avviare un'attività di spostamento di messaggi con una frequenza massima*

L'esempio `start-message-move-task` seguente avvia un'attività di spostamento dei messaggi per reindirizzare i messaggi dalla coda DLQ specificata alla coda di destinazione specificata a una velocità massima di 50 messaggi al secondo.

```
aws sqs start-message-move-task \
    --source-arn arn:aws:sqs:us-west-2:80398EXAMPLE:MyQueue1 \
    --destination-arn arn:aws:sqs:us-west-2:80398EXAMPLE:MyQueue2 \
    --max-number-of-messages-per-second 50
```

Output:

```
{
    "TaskHandle": "AQEB6nR4...HzlvZQ=="
}
```

Per ulteriori informazioni consulta Autorizzazioni API Amazon SQS: riferimento a operazioni e risorse nella Guida per gli sviluppatori.

- Per i dettagli sull'API, consulta StartMessageMoveTask AWS CLICommand Reference.

`tag-queue`

Il seguente esempio di codice mostra come utilizzare`tag-queue`.

AWS CLI

Come aggiungere tag di allocazione dei costi a una coda

L'esempio `tag-queue` seguente aggiunge un tag di allocazione dei costi alla coda Amazon SQS specificata.

```
aws sqs tag-queue \
    --queue-url https://sqs.us-west-2.amazonaws.com/123456789012/MyQueue \
    --tags Priority=Highest
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Aggiunta di tag per l'allocazione dei costi nella Guida per gli sviluppatori di Amazon Simple Queue Service.

- Per i dettagli sull'API, consulta TagQueue AWS CLICommand Reference.

`untag-queue`

Il seguente esempio di codice mostra come utilizzare`untag-queue`.

AWS CLI

Come rimuovere i tag di allocazione dei costi da una coda

L'esempio `untag-queue` seguente rimuove un tag di allocazione dei costi dalla coda Amazon SQS specificata.

```
aws sqs untag-queue \
    --queue-url https://sqs.us-west-2.amazonaws.com/123456789012/MyQueue \
    --tag-keys "Priority"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Aggiunta di tag per l'allocazione dei costi nella Guida per gli sviluppatori di Amazon Simple Queue Service.

- Per i dettagli sull'API, consulta UntagQueue AWS CLICommand Reference.

Esempi di Storage Gateway utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with Storage Gateway.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`describe-gateway-information`

Il seguente esempio di codice mostra come utilizzare`describe-gateway-information`.

AWS CLI

Come descrivere un gateway

Il comando `describe-gateway-information` seguente restituisce i metadati relativi al gateway specificato. Per specificare il gateway da descrivere, utilizza il nome della risorsa Amazon (ARN) del gateway nel comando.

Questo esempio specifica un gateway con l'id `sgw-12A3456B` nell'account `123456789012`:

```
aws storagegateway describe-gateway-information --gateway-
arn "arn:aws:storagegateway:us-west-2:123456789012:gateway/sgw-12A3456B"
```

Questo comando restituisce un blocco JSON contenente i metadati sul gateway, ad esempio il nome, le interfacce di rete, il fuso orario configurato e lo stato (se il gateway è in esecuzione o meno).

- Per i dettagli sull'API, consulta DescribeGatewayInformation AWS CLICommand Reference.

`list-file-shares`

Il seguente esempio di codice mostra come utilizzare`list-file-shares`.

AWS CLI

Come elencare le condivisioni di file

L'`command-name`esempio seguente elenca i widget disponibili nel tuo AWS account.

```
aws storagegateway list-file-shares \
    --gateway-arn arn:aws:storagegateway:us-east-1:209870788375:gateway/sgw-FB02E292
```

Output:

```
{
    "FileShareInfoList": [
        {
            "FileShareType": "NFS",
            "FileShareARN": "arn:aws:storagegateway:us-east-1:111122223333:share/
share-2FA12345",
            "FileShareId": "share-2FA12345",
            "FileShareStatus": "AVAILABLE",
            "GatewayARN": "arn:aws:storagegateway:us-east-1:111122223333:gateway/
sgw-FB0AAAAA"
        }
    ],
    "Marker": null
}
```

Per ulteriori informazioni, vedere ListFileShares AWSlo Storage Gateway Service API Reference.

- Per i dettagli sull'API, vedere ListFileSharesin AWS CLI Command Reference.

`list-gateways`

Il seguente esempio di codice mostra come utilizzare`list-gateways`.

AWS CLI

Come elencare i gateway per un account

Il comando `list-gateways` seguente elenca tutti i gateway definiti per un account:

```
aws storagegateway list-gateways
```

Questo comando genera un blocco JSON che contiene un elenco di gateway Amazon Resource Names ()ARNs.

- Per i dettagli sull'API, consulta AWS CLI Command ListGatewaysReference.

`list-volumes`

Il seguente esempio di codice mostra come utilizzare`list-volumes`.

AWS CLI

Come elencare i volumi configurati per un gateway

Il comando `list-volumes` seguente restituisce un elenco di volumi configurati per il gateway specificato. Per specificare il gateway da descrivere, utilizza il nome della risorsa Amazon (ARN) del gateway nel comando.

Questo esempio specifica un gateway con l'id `sgw-12A3456B` nell'account `123456789012`:

```
aws storagegateway list-volumes --gateway-arn "arn:aws:storagegateway:us-
west-2:123456789012:gateway/sgw-12A3456B"
```

Questo comando restituisce un blocco JSON contenente un elenco di volumi che include il tipo e l'ARN di ogni volume.

- Per i dettagli sull'API, consulta ListVolumes AWS CLICommand Reference.

`refresh-cache`

Il seguente esempio di codice mostra come utilizzare`refresh-cache`.

AWS CLI

Come aggiornare la cache di condivisione dei file

L'esempio `refresh-cache` seguente aggiorna la cache per la condivisione dei file specificata.

```
aws storagegateway refresh-cache \
    --file-share-arn arn:aws:storagegateway:us-east-1:111122223333:share/
share-2FA12345
```

Output:

```
{
    "FileShareARN": "arn:aws:storagegateway:us-east-1:111122223333:share/
share-2FA12345",
    "NotificationId": "4954d4b1-abcd-ef01-1234-97950a7d3483"
}
```

Per ulteriori informazioni, vedere ListFileShares AWSlo Storage Gateway Service API Reference.

- Per i dettagli sull'API, vedere RefreshCachein AWS CLI Command Reference.

AWS STS esempi utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with AWS STS.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`assume-role-with-saml`

Il seguente esempio di codice mostra come utilizzare`assume-role-with-saml`.

AWS CLI

Come ottenere credenziali a breve termine per un ruolo autenticato con SAML

Il comando `assume-role-with-saml` seguente recupera un set di credenziali a breve termine per il ruolo IAM `TestSaml`. La richiesta in questo esempio viene autenticata utilizzando l'asserzione SAML fornita dal gestore dell'identità al momento dell'autenticazione.

```
aws sts assume-role-with-saml \
    --role-arn arn:aws:iam::123456789012:role/TestSaml \
    --principal-arn arn:aws:iam::123456789012:saml-provider/SAML-test \
    --saml-
assertion "VERYLONGENCODEDASSERTIONEXAMPLExzYW1sOkF1ZGllbmNlPmJsYW5rPC9zYW1sOkF1ZGllbmNlPjwv
+PHNhbWw6TmFtZUlEIEZvcm1hdD0idXJuOm9hc2lzOm5hbWVzOnRjOlNBTUw6Mi4wOm5hbWVpZC1mb3JtYXQ6dHJhbnN
+PHNhbWw6U3ViamVjdENvbmZpcm1hdGlvbiBNZXRob2Q9InVybjpvYXNpczpuYW1lczp0YzpTQU1MOjIuMDpjbTpiZWF
```

Output:

```
{
    "Issuer": "https://integ.example.com/idp/shibboleth</Issuer",
    "AssumedRoleUser": {
        "Arn": "arn:aws:sts::123456789012:assumed-role/TestSaml",
        "AssumedRoleId": "ARO456EXAMPLE789:TestSaml"
    },
    "Credentials": {
        "AccessKeyId": "ASIAV3ZUEFP6EXAMPLE",
        "SecretAccessKey": "8P+SQvWIuLnKhh8d++jpw0nNmQRBZvNEXAMPLEKEY",
        "SessionToken": "IQoJb3JpZ2luX2VjEOz////////////////////
wEXAMPLEtMSJHMEUCIDoKK3JH9uGQE1z0sINr5M4jk
+Na8KHDcCYRVjJCZEvOAiEA3OvJGtw1EcViOleS2vhs8VdCKFJQWPQrmGdeehM4IC1NtBmUpp2wUE8phUZampKsburED
+xo0rKwT38xVqr7ZD0u0iPPkUL64lIZbqBAz
+scqKmlzm8FDrypNC9Yjc8fPOLn9FX9KSYvKTr4rvx3iSIlTJabIQwj2ICCR/oLxBA==",
        "Expiration": "2019-11-01T20:26:47Z"
    },
    "Audience": "https://signin.aws.amazon.com/saml",
    "SubjectType": "transient",
    "PackedPolicySize": "6",
```

```
    "NameQualifier": "SbdGOnUkh1i4+EXAMPLExL/jEvs=",
    "Subject": "SamlExample"
}
```

Per ulteriori informazioni, consulta Richiesta di credenziali di sicurezza temporanee nella AWS Guida per l'utente di IAM.

- Per i dettagli sull'API, consulta AssumeRoleWithSaml AWS CLICommand Reference.

`assume-role-with-web-identity`

Il seguente esempio di codice mostra come utilizzare`assume-role-with-web-identity`.

AWS CLI

Per ottenere credenziali a breve termine per un ruolo autenticato con Web Identity (OAuth 2."0)

Il comando `assume-role-with-web-identity` seguente recupera un set di credenziali a breve termine per il ruolo IAM `app1`. La richiesta viene autenticata utilizzando il token di identità Web fornito dal provider di identità Web specificato. Alla sessione vengono applicate due policy aggiuntive per limitare ulteriormente le azioni concesse all'utente. Le credenziali scadono un'ora dopo la loro generazione.

```
aws sts assume-role-with-web-identity \
    --duration-seconds 3600 \
    --role-session-name "app1" \
    --provider-id "www.amazon.com" \
    --policy-arns "arn:aws:iam::123456789012:policy/
q=webidentitydemopolicy1","arn:aws:iam::123456789012:policy/webidentitydemopolicy2"
 \
    --role-arn arn:aws:iam::123456789012:role/FederatedWebIdentityRole \
    --web-identity-token "Atza
%7CIQEBLjAsAhRFiXuWpUXuRvQ9PZL3GMFcYevydwIUFAHZwXZXXXXXXXXJnrulxKDHwy87oGKPznh0D6bEQZTSCzyoC
CrKqjG7nPBjNIL016GGvuS5gSvPRUxWES3VYfm1wl7WTI7jn-Pcb6M-
buCgHhFOzTQxod27L9CqnOLio7N3gZAGpsp6n1-
AJBOCJckcyXe2c6uD0srOJeZlKUm2eTDVMf8IehDVI0r1QOnTV6KzzAI3OY87Vd_cVMQ"
```

Output:

```
{
    "SubjectFromWebIdentityToken": "amzn1.account.AF6RHO7KZU5XRVQJGXK6HB56KR2A",
```

```
    "Audience": "client.5498841531868486423.1548@apps.example.com",
    "AssumedRoleUser": {
        "Arn": "arn:aws:sts::123456789012:assumed-role/FederatedWebIdentityRole/
app1",
        "AssumedRoleId": "AROACLKWSDQRAOEXAMPLE:app1"
    },
    "Credentials": {
        "AccessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "SecretAccessKey": "wJalrXUtnFEMI/K7MDENG/bPxRfiCYzEXAMPLEKEY",
        "SessionToken": "AQoEXAMPLEH4aoAH0gNCAPyJxz4BlCFFxWNE1OPTgk5TthT
+FvwqnKwRcOIfrRh3c/LTo6UDdyJwOOvEVPvLXCrrrUtdnniCEXAMPLE/
IvU1dYUg2RVAJBanLiHb4IgRmpRV3zrkuWJOgQs8IZZaIv2BXIa2R4OlgkBN9bkUDNCJiBeb/
AXlzBBko7b15fjrBs2+cTQtpZ3CYWFXG8C5zqx37wnOE49mRl/+OtkIKGO7fAE",
        "Expiration": "2020-05-19T18:06:10+00:00"
    },
    "Provider": "www.amazon.com"
}
```

Per ulteriori informazioni, consulta Richiesta di credenziali di sicurezza temporanee nella AWS Guida per l'utente di IAM.

- Per i dettagli sull'API, consulta AssumeRoleWithWebIdentityCommand Reference.AWS CLI

`assume-role`

Il seguente esempio di codice mostra come utilizzare`assume-role`.

AWS CLI

Come assumere un ruolo

Il comando `assume-role` seguente recupera un set di credenziali a breve termine per il ruolo IAM `s3-access-example`.

```
aws sts assume-role \
    --role-arn arn:aws:iam::123456789012:role/xaccounts3access \
    --role-session-name s3-access-example
```

Output:

```
{
    "AssumedRoleUser": {
```

```
        "AssumedRoleId": "AROA3XFRBF535PLBIFPI4:s3-access-example",
        "Arn": "arn:aws:sts::123456789012:assumed-role/xaccounts3access/s3-access-
example"
    },
    "Credentials": {
        "SecretAccessKey": "9drTJvcXLB89EXAMPLELB8923FB892xMFI",
        "SessionToken": "AQoXdzELDDY//////////
wEaoAK1wvxJY12r2IrDFT2IvAzTCn3zHoZ7YNtpiQLF0MqZye/qwjzP2iEXAMPLEbw/
m3hsj8VBTkPORGvr9jM5sgP+w9IZWZnU+LWhmg
+a5fDi2oTGUYcdg9uexQ4mtCHIHfi4citgqZTgco40Yqr4lIlo4V2b2Dyauk0eYFNebHtYlFVgAUj
+7Indz3LU0aTWk1WKIjHmmMCIoTkyYp/k7kUG7moeEYKSitwQIi6Gjn+nyzM
+PtoA3685ixzv0R7i5rjQi0YE0lf1oeie3bDiNHncmzosRM6SFiPzSvp6h/32xQuZsjcypmwsPSDtTPYcs0+YN/8BRi2
IcrxSpnWEXAMPLEXSDFTAQAM6Dl9zR0tXoybnlrZIwMLlMi1Kcgo5OytwU=",
        "Expiration": "2016-03-15T00:05:07Z",
        "AccessKeyId": "ASIAJEXAMPLEXEG2JICEA"
    }
}
```

L'output del comando contiene una chiave di accesso, una chiave segreta e un token di sessione che puoi utilizzare per l'autenticazione in AWS.

Per l'utilizzo della AWS CLI, è possibile impostare un profilo denominato associato a un ruolo. Quando utilizzi il profilo, la AWS CLI chiamerà assume-role e gestirà le credenziali per te. Per ulteriori informazioni, consulta Utilizzare un ruolo IAM nella CLI nella AWS CLI User Guide AWS .

- Per i dettagli sull'API, consulta AWS CLI Command AssumeRoleReference.

`assume-root`

Il seguente esempio di codice mostra come utilizzare`assume-root`.

AWS CLI

Come avviare una sessione privilegiata

Il comando `assume-root` seguente recupera una serie di credenziali a breve termine che puoi utilizzare per rimuovere una policy di bucket Amazon S3 non configurata correttamente per un account membro della tua organizzazione.

```
aws sts assume-root \
    --duration-seconds 900 \
    --target-principal 111122223333 \
```

```
    --task-policy-arn arn=arn:aws:iam::aws:policy/root-task/S3UnlockBucketPolicy
```

Output:

```
{
    "Credentials": {
        "SecretAccessKey": "9drTJvcXLB89EXAMPLELB8923FB892xMFI",
        "SessionToken": "AQoXdzELDDY//////////
wEaoAK1wvxJY12r2IrDFT2IvAzTCn3zHoZ7YNtpiQLF0MqZye/qwjzP2iEXAMPLEbw/
m3hsj8VBTkPORGvr9jM5sgP+w9IZWZnU+LWhmg
+a5fDi2oTGUYcdg9uexQ4mtCHIHfi4citgqZTgco40Yqr4lIlo4V2b2Dyauk0eYFNebHtYlFVgAUj
+7Indz3LU0aTWk1WKIjHmmMCIoTkyYp/k7kUG7moeEYKSitwQIi6Gjn+nyzM
+PtoA3685ixzv0R7i5rjQi0YE0lf1oeie3bDiNHncmzosRM6SFiPzSvp6h/32xQuZsjcypmwsPSDtTPYcs0+YN/8BRi2
IcrxSpnWEXAMPLEXSDFTAQAM6Dl9zR0tXoybnlrZIwMLlMi1Kcgo5OytwU=",
        "Expiration": "2024-11-15T00:05:07Z",
        "AccessKeyId": "ASIAJEXAMPLEXEG2JICEA"
    },
    "SourceIdentity": "Alice",
}
```

L'output del comando contiene una chiave di accesso, una chiave segreta e un token di sessione che puoi utilizzare per eseguire azioni privilegiate nell'account membro. Per ulteriori informazioni, consulta Esegui un'attività privilegiata su un account membro di AWS Organizations nella AWS IAM User Guide.

- Per i dettagli sull'API, consulta AssumeRoot AWS CLICommand Reference.

`decode-authorization-message`

Il seguente esempio di codice mostra come utilizzare`decode-authorization-message`.

AWS CLI

Come decodificare un messaggio di autorizzazione codificato restituito in risposta a una richiesta

Il seguente esempio `decode-authorization-message` decodifica informazioni aggiuntive sullo stato di autorizzazione di una richiesta da un messaggio codificato restituito in risposta a una richiesta Amazon Web Services.

```
aws sts decode-authorization-message \
    --encoded-message EXAMPLEWodyRNrtlQARDip-
eTA6i6DrlUhHhPQrLWB_lAbl5pAKxl9mPDLexYcGBreyIKQC1BGBIpBKr3dFDkwqeO7e2NMk5j_hmzAiChJN-8oy3Ewi
```

```
0jau7BMjOTWwOtHPhV_Zaz87yENdipr745EjQwRd5LaoL3vN8_5ZfA9UiBMKDgVh1gjqZJFUiQoubv78V1RbHNYnK44E
pOu3FZjwYStfvTb3GHs3-6rLribGO9jZOktkfE6vqxlFzLyeDr4P2ihC1wty9tArCvvGzIAUNmARQJ2VVWPxioqgoqCz
JWP5pwe_mAyqh0NLw-r1S56YC_90onj9A80sNrHlI-
tIiNd7tgNTYzDuPQYD2FMDBnp82V9eVmYGtPp5NIeSpuf3fOHanFuBZgENxZQZ2dlH3xJGMTtYayzZrRXjiq_SfX9zeB
FaoPIb8LmmKVBLpIB0iFhU9sEHPqKHVPi6jdxXqKaZaFGvYVmVOiuQdNQKuyk0p067POFrZECLjjOtNPBOZCcuEKEXAM
```

Output:

```
{
    "DecodedMessage": "{\"allowed\":false,\"explicitDeny\":true,\"matchedStatements
\":{\"items\":[{\"statementId\":\"VisualEditor0\",\"effect\":\"DENY\",\"principals
\":{\"items\":[{\"value\":\"AROA123456789EXAMPLE\"}]},\"principalGroups
\":{\"items\":[]},\"actions\":{\"items\":[{\"value\":\"ec2:RunInstances
\"}]},\"resources\":{\"items\":[{\"value\":\"*\"}]},\"conditions\":{\"items
\":[]}}]},\"failures\":{\"items\":[]},\"context\":{\"principal\":{\"id\":
\"AROA123456789EXAMPLE:Ana\",\"arn\":\"arn:aws:sts::111122223333:assumed-role/
Developer/Ana\"},\"action\":\"RunInstances\",\"resource\":\"arn:aws:ec2:us-
east-1:111122223333:instance/*\",\"conditions\":{\"items\":[{\"key\":
\"ec2:MetadataHttpPutResponseHopLimit\",\"values\":{\"items\":[{\"value\":
\"2\"}]}},{\"key\":\"ec2:InstanceMarketType\",\"values\":{\"items\":[{\"value
\":\"on-demand\"}]}},{\"key\":\"aws:Resource\",\"values\":{\"items\":[{\"value
\":\"instance/*\"}]}},{\"key\":\"aws:Account\",\"values\":{\"items\":[{\"value
\":\"111122223333\"}]}},{\"key\":\"ec2:AvailabilityZone\",\"values\":{\"items\":
[{\"value\":\"us-east-1f\"}]}},{\"key\":\"ec2:ebsOptimized\",\"values\":{\"items
\":[{\"value\":\"false\"}]}},{\"key\":\"ec2:IsLaunchTemplateResource\",\"values
\":{\"items\":[{\"value\":\"false\"}]}},{\"key\":\"ec2:InstanceType\",\"values\":
{\"items\":[{\"value\":\"t2.micro\"}]}},{\"key\":\"ec2:RootDeviceType\",\"values
\":{\"items\":[{\"value\":\"ebs\"}]}},{\"key\":\"aws:Region\",\"values\":{\"items
\":[{\"value\":\"us-east-1\"}]}},{\"key\":\"ec2:MetadataHttpEndpoint\",\"values
\":{\"items\":[{\"value\":\"enabled\"}]}},{\"key\":\"aws:Service\",\"values\":
{\"items\":[{\"value\":\"ec2\"}]}},{\"key\":\"ec2:InstanceID\",\"values\":{\"items
\":[{\"value\":\"*\"}]}},{\"key\":\"ec2:MetadataHttpTokens\",\"values\":{\"items
\":[{\"value\":\"required\"}]}},{\"key\":\"aws:Type\",\"values\":{\"items\":
[{\"value\":\"instance\"}]}},{\"key\":\"ec2:Tenancy\",\"values\":{\"items\":
[{\"value\":\"default\"}]}},{\"key\":\"ec2:Region\",\"values\":{\"items\":[{\"value
\":\"us-east-1\"}]}},{\"key\":\"aws:ARN\",\"values\":{\"items\":[{\"value\":
\"arn:aws:ec2:us-east-1:111122223333:instance/*\"}]}}]}}}"
}
```

Per ulteriori informazioni, consulta Logica di valutazione delle policy nella Guida per l'utente di AWS IAM.

- Per i dettagli sull'API, consulta DecodeAuthorizationMessage AWS CLICommand Reference.

`get-caller-identity`

Il seguente esempio di codice mostra come utilizzare`get-caller-identity`.

AWS CLI

Come ottenere dettagli sull'identità IAM corrente

Il comando `get-caller-identity` seguente visualizza informazioni sull'identità IAM utilizzata per autenticare la richiesta. Il chiamante è un utente IAM.

```
aws sts get-caller-identity
```

Output:

```
{
    "UserId": "AIDASAMPLEUSERID",
    "Account": "123456789012",
    "Arn": "arn:aws:iam::123456789012:user/DevAdmin"
}
```

- Per i dettagli sull'API, consulta GetCallerIdentity AWS CLICommand Reference.

`get-federation-token`

Il seguente esempio di codice mostra come utilizzare`get-federation-token`.

AWS CLI

Come restituire un set di credenziali di sicurezza temporanee utilizzando le credenziali della chiave di accesso utente IAM

Il seguente esempio `get-federation-token` restituisce un set di credenziali di sicurezza temporanee (ovvero l'ID chiave di accesso, una chiave di accesso segreta e un token di sicurezza) per un utente. Devi chiamare l'operazione `GetFederationToken` tramite le credenziali di sicurezza a lungo termine di un utente IAM.

```
aws sts get-federation-token \
    --name Bob \
    --policy file://myfile.json \
    --policy-arns arn=arn:aws:iam::aws:policy/AmazonS3ReadOnlyAccess \
```

```
    --duration-seconds 900
```

Contenuto di `myfile.json`:

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "ec2:Describe*",
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": "elasticloadbalancing:Describe*",
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "cloudwatch:ListMetrics",
                "cloudwatch:GetMetricStatistics",
                "cloudwatch:Describe*"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": "autoscaling:Describe*",
            "Resource": "*"
        }
    ]
}
```

Output:

```
{
    "Credentials": {
        "AccessKeyId": "ASIAIOSFODNN7EXAMPLE",
        "SecretAccessKey": "wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY",
        "SessionToken": "EXAMPLEpZ2luX2VjEGoaCXVzLXdlc3QtMiJIMEYCIQC/
W9pL5ArQyDD5JwFL3/h5+WGopQ24GEXweNctwhi9sgIhAMkg
+MZE35iWM8s4r5Lr25f9rSTVPFH98G42QQunWMTfKq0DCOP/////////
```

```
wEQAxoMNDUyOTI1MTcwNTA3Igxuy3AOpuuoLsk3MJwqgQPg8QOd9HuoClUxq26wnc/nm
+eZLjHDyGf2KUAHK2DuaS/nrGSEXAMPLE",
        "Expiration": "2023-12-20T02:06:07+00:00"
    },
    "FederatedUser": {
        "FederatedUserId": "111122223333:Bob",
        "Arn": "arn:aws:sts::111122223333:federated-user/Bob"
    },
    "PackedPolicySize": 36
}
```

Per ulteriori informazioni, consulta Richiesta di credenziali di sicurezza temporanee nella AWS Guida per l'utente di IAM.

- Per i dettagli sull'API, consulta GetFederationToken AWS CLICommand Reference.

`get-session-token`

Il seguente esempio di codice mostra come utilizzare`get-session-token`.

AWS CLI

Come ottenere un set di credenziali a breve termine per un'identità IAM

il comando `get-session-token` seguente recupera un set di credenziali a breve termine per l'identità IAM che esegue la chiamata. Le credenziali risultanti possono essere utilizzate per richieste in cui l'autenticazione a più fattori (MFA) è richiesta dalla policy. Le credenziali scadono 15 minuti dopo la loro generazione.

```
aws sts get-session-token \
    --duration-seconds 900 \
    --serial-number "YourMFADeviceSerialNumber" \
    --token-code 123456
```

Output:

```
{
    "Credentials": {
        "AccessKeyId": "ASIAIOSFODNN7EXAMPLE",
        "SecretAccessKey": "wJalrXUtnFEMI/K7MDENG/bPxRfiCYzEXAMPLEKEY",
        "SessionToken": "AQoEXAMPLEH4aoAH0gNCAPyJxz4BlCFFxWNE1OPTgk5TthT
+FvwqnKwRcOIfrRh3c/LTo6UDdyJwOOvEVPvLXCrrrUtdnniCEXAMPLE/
```

```
IvU1dYUg2RVAJBanLiHb4IgRmpRV3zrkuWJOgQs8IZZaIv2BXIa2R4OlgkBN9bkUDNCJiBeb/
AXlzBBko7b15fjrBs2+cTQtpZ3CYWFXG8C5zqx37wnOE49mRl/+OtkIKGO7fAE",
        "Expiration": "2020-05-19T18:06:10+00:00"
    }
}
```

Per ulteriori informazioni, consulta Richiesta di credenziali di sicurezza temporanee nella AWS Guida per l'utente di IAM.

- Per i dettagli sull'API, consulta GetSessionToken AWS CLICommand Reference.

Supporto esempi utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with Supporto.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`add-attachments-to-set`

Il seguente esempio di codice mostra come utilizzare`add-attachments-to-set`.

AWS CLI

Come aggiungere un allegato a un set

L'`add-attachments-to-set`esempio seguente aggiunge un'immagine a un set che potete quindi specificare per un caso di supporto nel vostro AWS account.

```
aws support add-attachments-to-set \
```

```
    --attachment-set-id "as-2f5a6faa2a4a1e600-mu-nk5xQlBr70-
G1cUos5LZkd38KOAHZa9BMDVzNEXAMPLE" \
    --attachments fileName=troubleshoot-screenshot.png,data=base64-encoded-string
```

Output:

```
{
    "attachmentSetId": "as-2f5a6faa2a4a1e600-mu-nk5xQlBr70-
G1cUos5LZkd38KOAHZa9BMDVzNEXAMPLE",
    "expiryTime": "2020-05-14T17:04:40.790+0000"
}
```

Per ulteriori informazioni, consulta Gestione dei casi nella Guida per l'utente del Supporto AWS .

- Per i dettagli sull'API, consulta AddAttachmentsToSet AWS CLICommand Reference.

`add-communication-to-case`

Il seguente esempio di codice mostra come utilizzare`add-communication-to-case`.

AWS CLI

Come aggiungere una comunicazione a un caso

L'`add-communication-to-case`esempio seguente aggiunge comunicazioni a una richiesta di supporto nel tuo AWS account.

```
aws support add-communication-to-case \
    --case-id "case-12345678910-2013-c4c1d2bf33c5cf47" \
    --communication-body "I'm attaching a set of images to this case." \
    --cc-email-addresses "myemail@example.com" \
    --attachment-set-id "as-2f5a6faa2a4a1e600-mu-nk5xQlBr70-
G1cUos5LZkd38KOAHZa9BMDVzNEXAMPLE"
```

Output:

```
{
    "result": true
}
```

Per ulteriori informazioni, consulta Gestione dei casi nella Guida per l'utente del Supporto AWS .

- Per i dettagli sull'API, consulta AddCommunicationToCase AWS CLICommand Reference.

`create-case`

Il seguente esempio di codice mostra come utilizzare`create-case`.

AWS CLI

Come creare un caso

L'`create-case`esempio seguente crea una richiesta di supporto per il tuo AWS account.

```
aws support create-case \
    --category-code "using-aws" \
    --cc-email-addresses "myemail@example.com" \
    --communication-body "I want to learn more about an AWS service." \
    --issue-type "technical" \
    --language "en" \
    --service-code "general-info" \
    --severity-code "low" \
    --subject "Question about my account"
```

Output:

```
{
    "caseId": "case-12345678910-2013-c4c1d2bf33c5cf47"
}
```

Per ulteriori informazioni, consulta Gestione dei casi nella Guida per l'utente del Supporto AWS .

- Per i dettagli sull'API, consulta CreateCase AWS CLICommand Reference.

`describe-attachment`

Il seguente esempio di codice mostra come utilizzare`describe-attachment`.

AWS CLI

Come descrivere un allegato

L'esempio `describe-attachment` seguente restituisce le informazioni sull'allegato con l'ID specificato.

```
aws support describe-attachment \
    --attachment-id "attachment-KBnjRNrePd9D6Jx0-Mm00xZuDEaL2JAj_0-
gJv9qqDooTipsz3V1Nb19rCfkZneeQeDPgp8X1iVJyHH7UuhZDdNeqGoduZsPrAhyMakqlc60-
iJjL5HqyYGiT1FG8EXAMPLE"
```

Output:

```
{
    "attachment": {
        "fileName": "troubleshoot-screenshot.png",
        "data": "base64-blob"
    }
}
```

Per ulteriori informazioni, consulta Gestione dei casi nella Guida per l'utente del Supporto AWS .

- Per i dettagli sull'API, consulta DescribeAttachment AWS CLICommand Reference.

`describe-cases`

Il seguente esempio di codice mostra come utilizzare`describe-cases`.

AWS CLI

Come descrivere un caso

L'`describe-cases`esempio seguente restituisce informazioni sulla richiesta di supporto specificata nell' AWS account.

```
aws support describe-cases \
    --display-id "1234567890" \
    --after-time "2020-03-23T21:31:47.774Z" \
    --include-resolved-cases \
    --language "en" \
    --no-include-communications \
    --max-item 1
```

Output:

```
{
    "cases": [
```

```
        {
            "status": "resolved",
            "ccEmailAddresses": [],
            "timeCreated": "2020-03-23T21:31:47.774Z",
            "caseId": "case-12345678910-2013-c4c1d2bf33c5cf47",
            "severityCode": "low",
            "language": "en",
            "categoryCode": "using-aws",
            "serviceCode": "general-info",
            "submittedBy": "myemail@example.com",
            "displayId": "1234567890",
            "subject": "Question about my account"
        }
    ]
}
```

Per ulteriori informazioni, consulta Gestione dei casi nella Guida per l'utente del Supporto AWS .

- Per i dettagli sull'API, consulta DescribeCases AWS CLICommand Reference.

`describe-communications`

Il seguente esempio di codice mostra come utilizzare`describe-communications`.

AWS CLI

Come descrivere la comunicazione più recente relativa a un caso

L'`describe-communications`esempio seguente restituisce la comunicazione più recente per il caso di supporto specificato nel tuo AWS account.

```
aws support describe-communications \
    --case-id "case-12345678910-2013-c4c1d2bf33c5cf47" \
    --after-time "2020-03-23T21:31:47.774Z" \
    --max-item 1
```

Output:

```
{
    "communications": [
        {
            "body": "I want to learn more about an AWS service.",
            "attachmentSet": [],
```

```
            "caseId": "case-12345678910-2013-c4c1d2bf33c5cf47",
            "timeCreated": "2020-05-12T23:12:35.000Z",
            "submittedBy": "Amazon Web Services"
        }
    ],
    "NextToken": "eyJuZXh0VG9rZW4iOiBudWxsLCAiYm90b190cnVuY2F0ZV9hbW91bnQEXAMPLE=="
}
```

Per ulteriori informazioni, consulta Gestione dei casi nella Guida per l'utente del Supporto AWS .

- Per i dettagli sull'API, consulta DescribeCommunications AWS CLICommand Reference.

`describe-services`

Il seguente esempio di codice mostra come utilizzare`describe-services`.

AWS CLI

Per elencare AWS i servizi e le categorie di servizi

L'esempio `describe-services` seguente elenca le categorie dei servizi disponibili per la richiesta di informazioni generali.

```
aws support describe-services \
    --service-code-list "general-info"
```

Output:

```
{
    "services": [
        {
            "code": "general-info",
            "name": "General Info and Getting Started",
            "categories": [
                {
                    "code": "charges",
                    "name": "How Will I Be Charged?"
                },
                {
                    "code": "gdpr-queries",
                    "name": "Data Privacy Query"
                },
```

```
                {
                    "code": "reserved-instances",
                    "name": "Reserved Instances"
                },
                {
                    "code": "resource",
                    "name": "Where is my Resource?"
                },
                {
                    "code": "using-aws",
                    "name": "Using AWS & Services"
                },
                {
                    "code": "free-tier",
                    "name": "Free Tier"
                },
                {
                    "code": "security-and-compliance",
                    "name": "Security & Compliance"
                },
                {
                    "code": "account-structure",
                    "name": "Account Structure"
                }
            ]
        }
    ]
}
```

Per ulteriori informazioni, consulta Gestione dei casi nella Guida per l'utente del Supporto AWS .

- Per i dettagli sull'API, consulta DescribeServices AWS CLICommand Reference.

`describe-severity-levels`

Il seguente esempio di codice mostra come utilizzare`describe-severity-levels`.

AWS CLI

Come elencare i livelli di gravità disponibili

L'esempio `describe-severity-levels` seguente elenca i livelli di gravità disponibili per un caso di supporto.

```
aws support describe-severity-levels
```

Output:

```
{
    "severityLevels": [
        {
            "code": "low",
            "name": "Low"
        },
        {
            "code": "normal",
            "name": "Normal"
        },
        {
            "code": "high",
            "name": "High"
        },
        {
            "code": "urgent",
            "name": "Urgent"
        },
        {
            "code": "critical",
            "name": "Critical"
        }
    ]
}
```

Per ulteriori informazioni, consulta Selezione di una gravità nella Guida per l'utente del Supporto AWS .

- Per i dettagli sull'API, consulta DescribeSeverityLevels AWS CLICommand Reference.

`describe-trusted-advisor-check-refresh-statuses`

Il seguente esempio di codice mostra come utilizzare`describe-trusted-advisor-check-refresh-statuses`.

AWS CLI

Per elencare gli stati di aggiornamento dei controlli di AWS Trusted Advisor

L'esempio `describe-trusted-advisor-check-refresh-statuses` seguente elenca gli stati di aggiornamento per due controlli di Trusted Advisor, ovvero Autorizzazioni bucket di Amazon S3 e Uso IAM.

```
aws support describe-trusted-advisor-check-refresh-statuses \
    --check-id "Pfx0RwqBli" "zXCkfM1nI3"
```

Output:

```
{
    "statuses": [
        {
            "checkId": "Pfx0RwqBli",
            "status": "none",
            "millisUntilNextRefreshable": 0
        },
        {
            "checkId": "zXCkfM1nI3",
            "status": "none",
            "millisUntilNextRefreshable": 0
        }
    ]
}
```

Per ulteriori informazioni, consulta AWS Trusted Advisor nella Guida per l'utente del Supporto AWS .

- Per i dettagli sull'API, consulta AWS CLI Command DescribeTrustedAdvisorCheckRefreshStatusesReference.

`describe-trusted-advisor-check-result`

Il seguente esempio di codice mostra come utilizzare`describe-trusted-advisor-check-result`.

AWS CLI

Per elencare i risultati di un controllo di AWS Trusted Advisor

L'esempio `describe-trusted-advisor-check-result` seguente elenca i risultati del controllo Uso IAM.

```
aws support describe-trusted-advisor-check-result \
    --check-id "zXCkfM1nI3"
```

Output:

```
{
    "result": {
        "checkId": "zXCkfM1nI3",
        "timestamp": "2020-05-13T21:38:05Z",
        "status": "ok",
        "resourcesSummary": {
            "resourcesProcessed": 1,
            "resourcesFlagged": 0,
            "resourcesIgnored": 0,
            "resourcesSuppressed": 0
        },
        "categorySpecificSummary": {
            "costOptimizing": {
                "estimatedMonthlySavings": 0.0,
                "estimatedPercentMonthlySavings": 0.0
            }
        },
        "flaggedResources": [
            {
                "status": "ok",
                "resourceId": "47DEQpj8HBSa-_TImW-5JCeuQeRkm5NMpJWZEXAMPLE",
                "isSuppressed": false
            }
        ]
    }
}
```

Per ulteriori informazioni, consulta AWS Trusted Advisor nella Guida per l'utente del Supporto AWS .

- Per i dettagli sull'API, consulta DescribeTrustedAdvisorCheckResult AWS CLICommand Reference.

`describe-trusted-advisor-check-summaries`

Il seguente esempio di codice mostra come utilizzare`describe-trusted-advisor-check-summaries`.

AWS CLI

Per elencare i riepiloghi dei controlli di AWS Trusted Advisor

L'esempio `describe-trusted-advisor-check-summaries` seguente elenca i risultati di due controlli di Trusted Advisor, ovvero Autorizzazioni bucket di Amazon S3 e Uso IAM.

```
aws support describe-trusted-advisor-check-summaries \
    --check-ids "Pfx0RwqBli" "zXCkfM1nI3"
```

Output:

```
{
    "summaries": [
        {
            "checkId": "Pfx0RwqBli",
            "timestamp": "2020-05-13T21:38:12Z",
            "status": "ok",
            "hasFlaggedResources": true,
            "resourcesSummary": {
                "resourcesProcessed": 44,
                "resourcesFlagged": 0,
                "resourcesIgnored": 0,
                "resourcesSuppressed": 0
            },
            "categorySpecificSummary": {
                "costOptimizing": {
                    "estimatedMonthlySavings": 0.0,
                    "estimatedPercentMonthlySavings": 0.0
                }
            }
        },
        {
            "checkId": "zXCkfM1nI3",
            "timestamp": "2020-05-13T21:38:05Z",
            "status": "ok",
            "hasFlaggedResources": true,
            "resourcesSummary": {
                "resourcesProcessed": 1,
                "resourcesFlagged": 0,
                "resourcesIgnored": 0,
                "resourcesSuppressed": 0
            },
```

```
            "categorySpecificSummary": {
                "costOptimizing": {
                    "estimatedMonthlySavings": 0.0,
                    "estimatedPercentMonthlySavings": 0.0
                }
            }
        }
    ]
}
```

Per ulteriori informazioni, consulta AWS Trusted Advisor nella Guida per l'utente del Supporto AWS .

- Per i dettagli sull'API, consulta DescribeTrustedAdvisorCheckSummaries AWS CLICommand Reference.

`describe-trusted-advisor-checks`

Il seguente esempio di codice mostra come utilizzare`describe-trusted-advisor-checks`.

AWS CLI

Per elencare i controlli AWS Trusted Advisor disponibili

L'`describe-trusted-advisor-checks`esempio seguente elenca gli assegni Trusted Advisor disponibili nel tuo AWS account. Queste informazioni includono il nome, l'ID, la descrizione, la categoria e i metadati del controllo. Nota che l'output è abbreviato per motivi di leggibilità.

```
aws support describe-trusted-advisor-checks \
    --language "en"
```

Output:

```
{
    "checks": [
        {
            "id": "zXCkfM1nI3",
            "name": "IAM Use",
            "description": "Checks for your use of AWS Identity and Access
 Management (IAM). You can use IAM to create users, groups, and roles in AWS, and
 you can use permissions to control access to AWS resources. \n<br>\n<br>\n<b>Alert
 Criteria</b><br>\nYellow: No IAM users have been created for this account.\n<br>
```

```
\n<br>\n<b>Recommended Action</b><br>\nCreate one or more IAM users and groups in
 your account. You can then create additional users whose permissions are limited
 to perform specific tasks in your AWS environment. For more information, see <a
 href=\"https://docs.aws.amazon.com/IAM/latest/UserGuide/IAMGettingStarted.html\"
 target=\"_blank\">Getting Started</a>. \n<br><br>\n<b>Additional Resources</b><br>
\n<a href=\"https://docs.aws.amazon.com/IAM/latest/UserGuide/IAM_Introduction.html\"
 target=\"_blank\">What Is IAM?</a>",
            "category": "security",
            "metadata": []
        }
    ]
}
```

Per ulteriori informazioni, consulta AWS Trusted Advisor nella Guida per l'utente del Supporto AWS .

- Per i dettagli sull'API, consulta DescribeTrustedAdvisorChecks AWS CLICommand Reference.

`refresh-trusted-advisor-check`

Il seguente esempio di codice mostra come utilizzare`refresh-trusted-advisor-check`.

AWS CLI

Per aggiornare un controllo di AWS Trusted Advisor

L'`refresh-trusted-advisor-check`esempio seguente aggiorna il check di Amazon S3 Bucket Permissions Trusted Advisor nel tuo account. AWS

```
aws support refresh-trusted-advisor-check \
    --check-id "Pfx0RwqBli"
```

Output:

```
{
    "status": {
        "checkId": "Pfx0RwqBli",
        "status": "enqueued",
        "millisUntilNextRefreshable": 3599992
    }
}
```

Per ulteriori informazioni, consulta AWS Trusted Advisor nella Guida per l'utente del Supporto AWS .

- Per i dettagli sull'API, consulta RefreshTrustedAdvisorCheckCommand Reference.AWS CLI

`resolve-case`

Il seguente esempio di codice mostra come utilizzare`resolve-case`.

AWS CLI

Come risolvere un caso di supporto

L'`resolve-case`esempio seguente risolve un caso di assistenza nel tuo AWS account.

```
aws support resolve-case \
    --case-id "case-12345678910-2013-c4c1d2bf33c5cf47"
```

Output:

```
{
    "finalCaseStatus": "resolved",
    "initialCaseStatus": "work-in-progress"
}
```

Per ulteriori informazioni, consulta Gestione dei casi nella Guida per l'utente del Supporto AWS .

- Per i dettagli sull'API, consulta AWS CLI Command ResolveCaseReference.

Esempi di Amazon SWF con AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando Amazon SWF. AWS Command Line Interface

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`count-closed-workflow-executions`

Il seguente esempio di codice mostra come usare`count-closed-workflow-executions`.

AWS CLI

Conteggio delle esecuzioni del flusso di lavoro chiuso

È possibile utilizzare `swf count-closed-workflow-executions` per recuperare il numero di flussi di lavoro chiusi per un determinato dominio. È possibile specificare dei filtri per contare classi specifiche di esecuzioni.

Gli argomenti `--domain` e sia `--close-time-filter` che `--start-time-filter` sono obbligatori. Tutti gli altri argomenti sono facoltativi.

```
aws swf count-closed-workflow-executions \
    --domain DataFrobtzz \
    --close-time-filter "{ \"latestDate\" : 1377129600, \"oldestDate\" :
 1370044800 }"
```

Output:

```
{
    "count": 2,
    "truncated": false
}
```

Se "truncated" è `true`, allora "count" rappresenta il numero massimo che può essere restituito da Amazon SWF. Eventuali ulteriori risultati vengono troncati.

Per ridurre il numero di risultati restituiti, puoi:

modificare i valori `--start-time-filter` o `--close-time-filter` per restringere l'intervallo di tempo in cui viene effettuata la ricerca. Ciascuno di questi si esclude a vicenda: puoi specificarne solo uno in una richiesta. Utilizza gli argomenti `--close-status-filter`, `--execution-filter`, `--tag-filter` o `--type-filter` per filtrare ulteriormente i risultati. Tuttavia, anche questi argomenti si escludono a vicenda.

Vedi anche CountClosedWorkflowExecutionsnel riferimento alle API di Amazon Simple Workflow Service

- Per i dettagli sull'API, consulta CountClosedWorkflowExecutions AWS CLICommand Reference.

`count-open-workflow-executions`

Il seguente esempio di codice mostra come utilizzare`count-open-workflow-executions`.

AWS CLI

Conteggio delle esecuzioni del flusso di lavoro aperto

È possibile utilizzare `swf count-open-workflow-executions` per recuperare il numero di flussi di lavoro aperti per un determinato dominio. È possibile specificare dei filtri per contare classi specifiche di esecuzioni.

Gli argomenti `--domain` e `--start-time-filter` sono obbligatori. Tutti gli altri argomenti sono facoltativi.

```
aws swf count-open-workflow-executions \
    --domain DataFrobtzz \
    --start-time-filter "{ \"latestDate\" : 1377129600, \"oldestDate\" :
 1370044800 }"
```

Output:

```
{
    "count": 4,
    "truncated": false
}
```

Se “truncated” è `true`, allora “count” rappresenta il numero massimo che può essere restituito da Amazon SWF. Eventuali ulteriori risultati vengono troncati.

Per ridurre il numero di risultati restituiti, puoi:

modificare i valori `--start-time-filter` per restringere l'intervallo di tempo in cui viene effettuata la ricerca. Utilizza gli argomenti `--close-status-filter`, `--execution-filter`,

`--tag-filter` o `--type-filter` per filtrare ulteriormente i risultati. Ciascuno di questi si esclude a vicenda: è possibile specificare solo uno di questi in una richiesta.

Per ulteriori informazioni, consulta il riferimento CountOpenWorkflowExecutions all'API di Amazon Simple Workflow Service

- Per i dettagli sull'API, consulta CountOpenWorkflowExecutions AWS CLICommand Reference.

`deprecate-domain`

Il seguente esempio di codice mostra come utilizzare`deprecate-domain`.

AWS CLI

Deprecare un dominio

Per impostare come obsoleto un dominio (che resta comunque visibile, ma non può essere utilizzato per creare nuove esecuzioni di flussi di lavoro o registrare tipi), utilizza `swf deprecate-domain`. Esiste un solo parametro obbligatorio, `--name`, che assume il nome del dominio da dichiarare obsoleto.

```
aws swf deprecate-domain \
    --name MyNeatNewDomain ""
```

Come per `register-domain`, non viene restituito alcun output. Se tuttavia utilizzi `list-domains` per visualizzare i domini registrati, vedrai che il dominio è stato dichiarato obsoleto e non compare più tra i dati restituiti.

```
aws swf list-domains \
    --registration-status REGISTERED
        {
            "domainInfos": [
                {
                    "status": "REGISTERED",
                    "name": "DataFrobotz"
                },
                {
                    "status": "REGISTERED",
                    "name": "erontest"
                }
            ]
```

```
        }
```

Se utilizzi `--registration-status DEPRECATED` con `list-domains` vedrai il tuo dominio obsoleto.

```
aws swf list-domains \
    --registration-status DEPRECATED
        {
            "domainInfos": [
                {
                    "status": "DEPRECATED",
                    "name": "MyNeatNewDomain"
                }
            ]
        }
```

Puoi ancora utilizzare `describe-domain` per ottenere informazioni su un dominio obsoleto.

```
aws swf describe-domain \
    --name MyNeatNewDomain
        {
            "domainInfo": {
                "status": "DEPRECATED",
                "name": "MyNeatNewDomain"
            },
            "configuration": {
                "workflowExecutionRetentionPeriodInDays": "0"
            }
        }
```

Vedi anche DeprecateDomainnel riferimento alle API di Amazon Simple Workflow Service

- Per i dettagli sull'API, consulta DeprecateDomain AWS CLICommand Reference.

`describe-domain`

Il seguente esempio di codice mostra come utilizzare`describe-domain`.

AWS CLI

Ottenere informazioni su un dominio

Per ottenere informazioni dettagliate su un dominio specifico, utilizza il comando `swf describe-domain`. Esiste un solo parametro obbligatorio: `--name`, che assume il nome del dominio su cui si desidera ottenere informazioni.

```
aws swf describe-domain \
    --name DataFrobotz
        {
            "domainInfo": {
                "status": "REGISTERED",
                "name": "DataFrobotz"
            },
            "configuration": {
                "workflowExecutionRetentionPeriodInDays": "1"
            }
        }
```

Puoi anche utilizzare `describe-domain` per ottenere informazioni sui domini obsoleti.

```
aws swf describe-domain \
    --name MyNeatNewDomain
        {
            "domainInfo": {
                "status": "DEPRECATED",
                "name": "MyNeatNewDomain"
            },
            "configuration": {
                "workflowExecutionRetentionPeriodInDays": "0"
            }
        }
```

Vedi anche DescribeDomainnel riferimento alle API di Amazon Simple Workflow Service

- Per i dettagli sull'API, consulta DescribeDomain AWS CLICommand Reference.

`list-activity-types`

Il seguente esempio di codice mostra come utilizzare`list-activity-types`.

AWS CLI

Elenco dei tipi di attività

Per ottenere un elenco dei tipi di attività per un dominio, utilizza `swf list-activity-types`. Gli argomenti `--domain` e `--registration-status` sono obbligatori.

```
aws swf list-activity-types \
    --domain DataFrobtzz \
    --registration-status REGISTERED
```

Output:

```
{
    "typeInfos": [
        {
            "status": "REGISTERED",
            "creationDate": 1371454150.451,
            "activityType": {
                "version": "1",
                "name": "confirm-user-email"
            },
            "description": "subscribe confirm-user-email activity"
        },
        {
            "status": "REGISTERED",
            "creationDate": 1371454150.709,
            "activityType": {
                "version": "1",
                "name": "confirm-user-phone"
            },
            "description": "subscribe confirm-user-phone activity"
        },
        {
            "status": "REGISTERED",
            "creationDate": 1371454149.871,
            "activityType": {
                "version": "1",
                "name": "get-subscription-info"
            },
            "description": "subscribe get-subscription-info activity"
        },
        {
            "status": "REGISTERED",
            "creationDate": 1371454150.909,
            "activityType": {
                "version": "1",
```

```
                "name": "send-subscription-success"
            },
            "description": "subscribe send-subscription-success activity"
        },
        {
            "status": "REGISTERED",
            "creationDate": 1371454150.085,
            "activityType": {
                "version": "1",
                "name": "subscribe-user-sns"
            },
            "description": "subscribe subscribe-user-sns activity"
        }
    ]
}
```

Puoi utilizzare l'argomento `--name` per selezionare solo i tipi di attività con un nome particolare:

```
aws swf list-activity-types \
    --domain DataFrobtzz \
    --registration-status REGISTERED \
    --name "send-subscription-success"
```

Output:

```
{
    "typeInfos": [
        {
            "status": "REGISTERED",
            "creationDate": 1371454150.909,
            "activityType": {
                "version": "1",
                "name": "send-subscription-success"
            },
            "description": "subscribe send-subscription-success activity"
        }
    ]
}
```

Per recuperare i risultati nelle pagine, puoi impostare l'argomento `--maximum-page-size`. Se vengono restituiti più risultati di quelli contenuti in una pagina di risultati, verrà restituito un nextPageToken "" nel set di risultati:

```
aws swf list-activity-types \
    --domain DataFrobtzz \
    --registration-status REGISTERED \
    --maximum-page-size 2
```

Output:

```
{
    "nextPageToken": "AAAAKgAAAAEAAAAAAAAAA1Gp1BelJq
+PmHvAnDxJYbup8+0R4LVtbXLDl7QNY7C3OpHo9Sz06D/GuFz1OyC73umBQ1tOPJ/gC/
aYpzDMqUIWIA1T9W0s2DryyZX4OC/6Lhk9/
o5kdsuWMSBkHhgaZjgwp3WJINIFJFdaSMxY2vYAX7AtRtpcqJuBDDRE9RaRqDGYqIYUMltarkiqpSY1ZVveBasBvlvyU
WGAaqehiDz7/JzLT/wWNNUMOd+Nhe",
    "typeInfos": [
        {
            "status": "REGISTERED",
            "creationDate": 1371454150.451,
            "activityType": {
                "version": "1",
                "name": "confirm-user-email"
            },
            "description": "subscribe confirm-user-email activity"
        },
        {
            "status": "REGISTERED",
            "creationDate": 1371454150.709,
            "activityType": {
                "version": "1",
                "name": "confirm-user-phone"
            },
            "description": "subscribe confirm-user-phone activity"
        }
    ]
}
```

È possibile passare il nextPageToken valore alla chiamata successiva `list-activity-types` all'`--next-page-token`argomento, recuperando la pagina successiva di risultati:

```
aws swf list-activity-types \
    --domain DataFrobtzz \
    --registration-status REGISTERED \
    --maximum-page-size 2 \
```

```
    --next-page-token "AAAAKgAAAAEAAAAAAAAAA1Gp1BelJq
+PmHvAnDxJYbup8+0R4LVtbXLDl7QNY7C3OpHo9Sz06D/GuFz1OyC73umBQ1tOPJ/gC/
aYpzDMqUIWIA1T9W0s2DryyZX4OC/6Lhk9/
o5kdsuWMSBkHhgaZjgwp3WJINIFJFdaSMxY2vYAX7AtRtpcqJuBDDRE9RaRqDGYqIYUMltarkiqpSY1ZVveBasBvlvyU
WGAaqehiDz7/JzLT/wWNNUMOd+Nhe"
```

Output:

```
{
    "nextPageToken": "AAAAKgAAAAEAAAAAAAAAAw+7LZ4GRZPzTqBHsp2wBxWB8m1sgLCclgCuq3J+h/
m3+vOfFqtkcjLwV5cc4OjNAzTCuq/
XcylPumGwkjbajtqpZpbqOcVNfjFxGoi0LB2Olbvv0krbUISBvlpFPmSWpDSZJsxg5UxCcweteSlFn1PNSZ/
MoinBZo8OTkjMuzcsTuKOzH9wCaR8ITcALJ3SaqHU3pyIRS5hPmFA3OLIc8zaAepjlaujo6hntNSCruB4"
    "typeInfos": [
        {
            "status": "REGISTERED",
            "creationDate": 1371454149.871,
            "activityType": {
                "version": "1",
                "name": "get-subscription-info"
            },
            "description": "subscribe get-subscription-info activity"
        },
        {
            "status": "REGISTERED",
            "creationDate": 1371454150.909,
            "activityType": {
                "version": "1",
                "name": "send-subscription-success"
            },
            "description": "subscribe send-subscription-success activity"
        }
    ]
}
```

Se ci sono ancora altri risultati da restituire, verrà restituito nextPageToken "" con i risultati. Quando non ci sono più pagine di risultati da restituire, "nextPageToken" non verrà restituito nel set di risultati.

È possibile utilizzare l'argomento `--reverse-order` per invertire l'ordine dei risultati restituiti. Ciò influisce anche sui risultati paginati.

```
aws swf list-activity-types \
    --domain DataFrobtzz \
    --registration-status REGISTERED \
    --maximum-page-size 2 \
    --reverse-order
```

Output:

```
{
    "nextPageToken": "AAAAKgAAAAEAAAAAAAAAAwXcpu5ePSyQkrC
+8WMbmSrenuZC2ZkIXQYBPB/b9xIOVkj+bMEFhGj0KmmJ4rF7iddhjf7UMYCsfGkEn7mk
+yMCgVc1JxDWmB0EH46bhcmcLmYNQihMDmUWocpr7To6/R7CLu0St1gkFayxOidJXErQW0zdNfQaIWAnF/
cwioBbXlkz1fQzmDeU3M5oYGMPQIrUqkPq7pMEW0q0lK5eDN97NzFYdZZ/rlcLDWPZhUjY",
    "typeInfos": [
        {
            "status": "REGISTERED",
            "creationDate": 1371454150.085,
            "activityType": {
                "version": "1",
                "name": "subscribe-user-sns"
            },
            "description": "subscribe subscribe-user-sns activity"
        },
        {
            "status": "REGISTERED",
            "creationDate": 1371454150.909,
            "activityType": {
                "version": "1",
                "name": "send-subscription-success"
            },
            "description": "subscribe send-subscription-success activity"
        }
    ]
}
```

Vedi anche ListActivityTypesnel riferimento alle API di Amazon Simple Workflow Service

- Per i dettagli sull'API, consulta ListActivityTypes AWS CLICommand Reference.

`list-domains`

Il seguente esempio di codice mostra come utilizzare`list-domains`.

AWS CLI

Esempio 1: come elencare i domini registrati

Il comando `list-domains` seguente elenca i domini `REGISTERED` SWF che hai registrato per il tuo account.

```
aws swf list-domains \
    --registration-status REGISTERED
```

Output:

```
{
  "domainInfos": [
    {
      "status": "REGISTERED",
      "name": "DataFrobotz"
    },
    {
      "status": "REGISTERED",
      "name": "erontest"
    }
  ]
}
```

Per ulteriori informazioni, consulta il riferimento ListDomainsall'API di Amazon Simple Workflow Service

Esempio 2: come elencare i domini obsoleti

Il comando `list-domains` seguente elenca i domini `DEPRECATED` SWF che hai registrato per il tuo account. I domini obsoleti sono domini che non possono registrare nuovi flussi di lavoro o attività, ma che possono comunque essere interrogati.

```
aws swf list-domains \
    --registration-status DEPRECATED
```

Output:

```
{
  "domainInfos": [
    {
```

```
      "status": "DEPRECATED",
      "name": "MyNeatNewDomain"
    }
  ]
}
```

Per ulteriori informazioni, consulta il riferimento ListDomainsall'API di Amazon Simple Workflow Service

Esempio 3: come elencare la prima pagina dei domini registrati

Il comando `list-domains` seguente elenca i domini `REGISTERED` SWF della prima pagina che hai registrato per il tuo account utilizzando l'opzione `--maximum-page-size`.

```
aws swf list-domains \
    --registration-status REGISTERED \
    --maximum-page-size 1
```

Output:

```
{
    "domainInfos": [
        {
            "status": "REGISTERED",
            "name": "DataFrobotz"
        }
    ],
"nextPageToken": "AAAAKgAAAAEAAAAAAAAAA2QJKNtidVgd49TTeNwYcpD
+QKT2ynuEbibcQWe2QKrslMGe63gpS0MgZGpcpoKttL4OCXRFn98Xif557it
+wSZUsvUDtImjDLvguyuyyFdIZtvIxIKEOPm3k2r4OjAGaFsGOuVbrKljvla7wdU7FYH3OlkNCP8b7PBj9SBkUyGoiAg
}
```

Per ulteriori informazioni, consulta il riferimento ListDomainsall'API di Amazon Simple Workflow Service

Esempio 4: come elencare la pagina singola specificata dei domini registrati

Il comando `list-domains` seguente elenca i domini `REGISTERED` SWF della prima pagina che hai registrato per il tuo account utilizzando l'opzione `--maximum-page-size`.

Quando effettuerai nuovamente la chiamata, fornendo il valore di `nextPageToken` nell'argomento `--next-page-token`, riceverai un'altra pagina di risultati.

```
aws swf list-domains \
    --registration-status REGISTERED \
    --maximum-page-size 1 \
    --next-page-token "AAAAKgAAAAEAAAAAAAAAA2QJKNtidVgd49TTeNwYcpD
+QKT2ynuEbibcQWe2QKrslMGe63gpS0MgZGpcpoKttL4OCXRFn98Xif557it
+wSZUsvUDtImjDLvguyuyyFdIZtvIxIKEOPm3k2r4OjAGaFsGOuVbrKljvla7wdU7FYH3OlkNCP8b7PBj9SBkUyGoiAg
```

Output:

```
{
    "domainInfos": [
        {
            "status": "REGISTERED",
            "name": "erontest"
        }
    ]
}
```

Quando le pagine dei risultati da recuperare saranno terminate, `nextPageToken` non verrà restituito nei risultati.

Per ulteriori informazioni, consulta il riferimento ListDomainsall'API di Amazon Simple Workflow Service

- Per i dettagli sull'API, consulta ListDomains AWS CLICommand Reference.

`list-workflow-types`

Il seguente esempio di codice mostra come utilizzare`list-workflow-types`.

AWS CLI

Elenco dei tipi di flusso di lavoro

Per ottenere un elenco dei tipi di flusso di lavoro per un dominio, utilizza `swf list-workflow-types`. Gli argomenti `--domain` e `--registration-status` sono obbligatori. Ecco un semplice esempio.

```
aws swf list-workflow-types \
    --domain DataFrobtzz \
    --registration-status REGISTERED
```

Output:

```
{
    "typeInfos": [
        {
            "status": "REGISTERED",
            "creationDate": 1371454149.598,
            "description": "DataFrobtzz subscribe workflow",
            "workflowType": {
                "version": "v3",
                "name": "subscribe"
            }
        }
    ]
}
```

Come con `list-activity-types`, è possibile utilizzare l'argomento `--name` per selezionare solo i tipi di flusso di lavoro con un nome particolare e utilizzare l'argomento `--maximum-page-size` in coordinamento con `--next-page-token` per i risultati della pagina. Per invertire l'ordine in cui vengono restituiti i risultati, utilizza `--reverse-order`.

Vedi anche ListWorkflowTypesnel riferimento alle API di Amazon Simple Workflow Service

- Per i dettagli sull'API, consulta ListWorkflowTypes AWS CLICommand Reference.

`register-domain`

Il seguente esempio di codice mostra come utilizzare`register-domain`.

AWS CLI

Registrazione di un dominio

Puoi usare la AWS CLI per registrare nuovi domini. Utilizza il comando `swf register-domain`. Sono necessari due parametri`--name`, che richiedono il nome di dominio e `--workflow-execution-retention-period-in-days` un numero intero per specificare il numero di giorni necessari per conservare i dati di esecuzione del flusso di lavoro su questo dominio, fino a un periodo massimo di 90 giorni (per ulteriori informazioni, consultate le domande frequenti su SWF < faqs/ #retain_limit >). https://aws.amazon.com/swf/ I dati di esecuzione dei flussi di lavoro non saranno conservati una volta superato il numero di giorni specificato.

```
aws swf register-domain \
```

```
    --name MyNeatNewDomain \
    --workflow-execution-retention-period-in-days 0
    ""
```

Quando si registra un dominio, non viene restituito alcun valore (""), ma è possibile utilizzare `swf list-domains` o `swf describe-domain` per visualizzare il nuovo dominio.

```
aws swf list-domains \
    --registration-status REGISTERED
        {
            "domainInfos": [
                {
                    "status": "REGISTERED",
                    "name": "DataFrobotz"
                },
                {
                    "status": "REGISTERED",
                    "name": "MyNeatNewDomain"
                },
                {
                    "status": "REGISTERED",
                    "name": "erontest"
                }
            ]
        }
```

Con `swf describe-domain`:

```
aws swf describe-domain --
name MyNeatNewDomain
{
    "domainInfo": {
        "status": "REGISTERED",
        "name": "MyNeatNewDomain"
    },
    "configuration": {
        "workflowExecutionRetentionPeriodInDays": "0"

  }
}
```

Vedi anche RegisterDomainnel riferimento alle API di Amazon Simple Workflow Service

- Per i dettagli sull'API, consulta RegisterDomain AWS CLICommand Reference.

`register-workflow-type`

Il seguente esempio di codice mostra come utilizzare`register-workflow-type`.

AWS CLI

Registrazione di un tipo di workflow

Per registrare un tipo di workflow con la AWS CLI, utilizzate il `swf register-workflow-type` comando.

```
aws swf register-workflow-type \
    --domain DataFrobtzz \
    --name "MySimpleWorkflow" \
    --workflow-version "v1"
```

Se il comando ha esito positivo, non produce alcun output.

In caso di errore (ad esempio, se provi a registrare due volte lo stesso tipo di flusso di lavoro o specifichi un dominio che non esiste) riceverai una risposta in JSON.

```
{
    "message": "WorkflowType=[name=MySimpleWorkflow, version=v1]",
    "__type": "com.amazonaws.swf.base.model#TypeAlreadyExistsFault"
}
```

`--domain`, `--name` e `--workflow-version` sono obbligatori. Puoi anche impostare la descrizione del flusso di lavoro, i timeout e la policy del flusso di lavoro secondario.

Per ulteriori informazioni, consulta il riferimento RegisterWorkflowTypeall'API di Amazon Simple Workflow Service

- Per i dettagli sull'API, consulta RegisterWorkflowType AWS CLICommand Reference.

Esempi di Systems Manager che utilizzano AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with Systems Manager.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`add-tags-to-resource`

Il seguente esempio di codice mostra come utilizzare`add-tags-to-resource`.

AWS CLI

Esempio 1: come aggiungere tag a una finestra di manutenzione

L'esempio seguente `add-tags-to-resource` aggiunge un tag alla finestra di manutenzione specificata.

```
aws ssm add-tags-to-resource \
    --resource-type "MaintenanceWindow" \
    --resource-id "mw-03eb9db428EXAMPLE" \
    --tags "Key=Stack,Value=Production"
```

Questo comando non produce alcun output.

Esempio 2: come aggiungere tag a un parametro

L'esempio seguente `add-tags-to-resource` aggiunge due tag al parametro specificato.

```
aws ssm add-tags-to-resource \
    --resource-type "Parameter" \
    --resource-id "My-Parameter" \
    --tags '[{"Key":"Region","Value":"East"},{"Key":"Environment",
 "Value":"Production"}]'
```

Questo comando non produce alcun output.

Esempio 3: come aggiungere tag a un documento SSM

L'esempio seguente `add-tags-to-resource` aggiunge un tag al documento specificato.

```
aws ssm add-tags-to-resource \
    --resource-type "Document" \
    --resource-id "My-Document" \
    --tags "Key=Quarter,Value=Q322"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Applicazione di tag alle risorse di Systems Manager nella Guida per l'utente di AWS Systems Manager.

- Per i dettagli sull'API, consulta AddTagsToResource AWS CLICommand Reference.

`associate-ops-item-related-item`

Il seguente esempio di codice mostra come utilizzare`associate-ops-item-related-item`.

AWS CLI

Come associare un elemento correlato

L'`associate-ops-item-related-item`esempio seguente associa un elemento correlato a. OpsItem

```
aws ssm associate-ops-item-related-item \
    --ops-item-id "oi-649fExample" \
    --association-type "RelatesTo" \
    --resource-type "AWS::SSMIncidents::IncidentRecord" \
    --resource-uri "arn:aws:ssm-incidents::111122223333:incident-record/Example-
Response-Plan/c2bde883-f7d5-343a-b13a-bf5fe9ea689f"
```

Output:

```
{
    "AssociationId": "61d7178d-a30d-4bc5-9b4e-a9e74EXAMPLE"
}
```

Per ulteriori informazioni, vedere Utilizzo degli incidenti di Incident Manager OpsCenter nella Guida per l'utente di AWS Systems Manager.

- Per i dettagli sull'API, vedere AssociateOpsItemRelatedItemin AWS CLI Command Reference.

`cancel-command`

Il seguente esempio di codice mostra come utilizzare`cancel-command`.

AWS CLI

Esempio 1: come annullare un comando per tutte le istanze

L'esempio seguente `cancel-command` tenta di annullare il comando specificato che è già in esecuzione per tutte le istanze.

```
aws ssm cancel-command \
    --command-id "662add3d-5831-4a10-b64a-f2ff3EXAMPLE"
```

Questo comando non produce alcun output.

Esempio 2: come annullare un comando per istanze specifiche

L'esempio seguente `cancel-command` tenta di annullare un comando solo per l'istanza specificata.

```
aws ssm cancel-command \
    --command-id "662add3d-5831-4a10-b64a-f2ff3EXAMPLE"
    --instance-ids "i-02573cafcfEXAMPLE"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Applicazione di tag a parametri di Systems Manager nella Guida per l'utente di AWS Systems Manager.

- Per i dettagli sull'API, consulta CancelCommand AWS CLICommand Reference.

`cancel-maintenance-window-execution`

Il seguente esempio di codice mostra come utilizzare`cancel-maintenance-window-execution`.

AWS CLI

Come annullare l'esecuzione di una finestra di manutenzione

L'esempio `cancel-maintenance-window-execution` seguente arresta l'esecuzione della finestra di manutenzione specificata già in corso.

```
aws ssm cancel-maintenance-window-execution \
    --window-execution-id j2l8d5b5c-mw66-tk4d-r3g9-1d4d1EXAMPLE
```

Output:

```
{
    "WindowExecutionId": "j2l8d5b5c-mw66-tk4d-r3g9-1d4d1EXAMPLE"
}
```

Per ulteriori informazioni, vedere i tutorial di Windows (AWS CLI) sulla manutenzione di Systems Manager nella Guida per l'utente di AWS Systems Manager.

- Per i dettagli sulle API, vedere CancelMaintenanceWindowExecutionin AWS CLI Command Reference.

`create-activation`

Il seguente esempio di codice mostra come utilizzare`create-activation`.

AWS CLI

Come creare un'attivazione di un'istanza gestita

L'esempio seguente `create-activation` crea un'attivazione gestita dell'istanza.

```
aws ssm create-activation \
    --default-instance-name "HybridWebServers" \
    --iam-role "HybridWebServersRole" \
    --registration-limit 5
```

Output:

```
{
    "ActivationId": "5743558d-563b-4457-8682-d16c3EXAMPLE",
```

```
    "ActivationCode": "dRmgnYaFv567vEXAMPLE"
}
```

Per ulteriori informazioni, consulta Passaggio 4: creare un'attivazione di istanze gestite per un ambiente ibrido nella Guida per l'utente di AWS Systems Manager.

- Per i dettagli sull'API, consulta CreateActivation AWS CLICommand Reference.

`create-association-batch`

Il seguente esempio di codice mostra come utilizzare`create-association-batch`.

AWS CLI

Come creare più associazioni

Questo esempio associa un documento di configurazione a più istanze. L'output restituisce un elenco di operazioni riuscite e non riuscite, se applicabile.

Comando:

```
aws ssm create-association-batch --entries "Name=AWS-
UpdateSSMAgent,InstanceId=i-1234567890abcdef0" "Name=AWS-
UpdateSSMAgent,InstanceId=i-9876543210abcdef0"
```

Output:

```
{
  "Successful": [
      {
          "Name": "AWS-UpdateSSMAgent",
          "InstanceId": "i-1234567890abcdef0",
          "AssociationVersion": "1",
          "Date": 1550504725.007,
          "LastUpdateAssociationDate": 1550504725.007,
          "Status": {
              "Date": 1550504725.007,
              "Name": "Associated",
              "Message": "Associated with AWS-UpdateSSMAgent"
          },
          "Overview": {
              "Status": "Pending",
```

```
                "DetailedStatus": "Creating"
            },
            "DocumentVersion": "$DEFAULT",
            "AssociationId": "8dfe3659-4309-493a-8755-0123456789ab",
            "Targets": [
                {
                    "Key": "InstanceIds",
                    "Values": [
                        "i-1234567890abcdef0"
                    ]
                }
            ]
        },
        {
            "Name": "AWS-UpdateSSMAgent",
            "InstanceId": "i-9876543210abcdef0",
            "AssociationVersion": "1",
            "Date": 1550504725.057,
            "LastUpdateAssociationDate": 1550504725.057,
            "Status": {
                "Date": 1550504725.057,
                "Name": "Associated",
                "Message": "Associated with AWS-UpdateSSMAgent"
            },
            "Overview": {
                "Status": "Pending",
                "DetailedStatus": "Creating"
            },
            "DocumentVersion": "$DEFAULT",
            "AssociationId": "9c9f7f20-5154-4fed-a83e-0123456789ab",
            "Targets": [
                {
                    "Key": "InstanceIds",
                    "Values": [
                        "i-9876543210abcdef0"
                    ]
                }
            ]
        }
    ],
    "Failed": []
}
```

- Per i dettagli sull'API, consulta CreateAssociationBatch AWS CLICommand Reference.

create-association

Il seguente esempio di codice mostra come utilizzarecreate-association.

AWS CLI

Esempio 1: associare un documento utilizzando un'istanza IDs

Questo esempio associa un documento di configurazione a un'istanza, utilizzando instance IDs.

```
aws ssm create-association \
    --instance-id "i-0cb2b964d3e14fd9f" \
    --name "AWS-UpdateSSMAgent"
```

Output:

```
{
    "AssociationDescription": {
        "Status": {
            "Date": 1487875500.33,
            "Message": "Associated with AWS-UpdateSSMAgent",
            "Name": "Associated"
        },
        "Name": "AWS-UpdateSSMAgent",
        "InstanceId": "i-0cb2b964d3e14fd9f",
        "Overview": {
            "Status": "Pending",
            "DetailedStatus": "Creating"
        },
        "AssociationId": "b7c3266e-a544-44db-877e-b20d3a108189",
        "DocumentVersion": "$DEFAULT",
        "LastUpdateAssociationDate": 1487875500.33,
        "Date": 1487875500.33,
        "Targets": [
            {
                "Values": [
                    "i-0cb2b964d3e14fd9f"
                ],
                "Key": "InstanceIds"
            }
        ]
    }
}
```

Per ulteriori informazioni, vedere il riferimento CreateAssociationall'API AWS Systems Manager.

Esempio 2: come associare un documento utilizzando destinazioni

Questo esempio associa un documento di configurazione a un'istanza, utilizzando destinazioni.

```
aws ssm create-association \
    --name "AWS-UpdateSSMAgent" \
    --targets "Key=instanceids,Values=i-0cb2b964d3e14fd9f"
```

Output:

```
{
    "AssociationDescription": {
        "Status": {
            "Date": 1487875500.33,
            "Message": "Associated with AWS-UpdateSSMAgent",
            "Name": "Associated"
        },
        "Name": "AWS-UpdateSSMAgent",
        "InstanceId": "i-0cb2b964d3e14fd9f",
        "Overview": {
            "Status": "Pending",
            "DetailedStatus": "Creating"
        },
        "AssociationId": "b7c3266e-a544-44db-877e-b20d3a108189",
        "DocumentVersion": "$DEFAULT",
        "LastUpdateAssociationDate": 1487875500.33,
        "Date": 1487875500.33,
        "Targets": [
            {
                "Values": [
                    "i-0cb2b964d3e14fd9f"
                ],
                "Key": "InstanceIds"
            }
        ]
    }
}
```

Per ulteriori informazioni, vedere il riferimento CreateAssociationall'API AWS Systems Manager.

Esempio 3: come creare un'associazione che viene eseguita una sola volta

Questo esempio crea una nuova associazione che viene eseguita solo una volta nella data e ora specificate. Le associazioni create con una data passata o presente (nel momento in cui viene elaborata la data è passata) vengono eseguite immediatamente.

```
aws ssm create-association \
    --name "AWS-UpdateSSMAgent" \
    --targets "Key=instanceids,Values=i-0cb2b964d3e14fd9f" \
    --schedule-expression "at(2020-05-14T15:55:00)"  \
    --apply-only-at-cron-interval
```

Output:

```
{
    "AssociationDescription": {
        "Status": {
            "Date": 1487875500.33,
            "Message": "Associated with AWS-UpdateSSMAgent",
            "Name": "Associated"
        },
        "Name": "AWS-UpdateSSMAgent",
        "InstanceId": "i-0cb2b964d3e14fd9f",
        "Overview": {
            "Status": "Pending",
            "DetailedStatus": "Creating"
        },
        "AssociationId": "b7c3266e-a544-44db-877e-b20d3a108189",
        "DocumentVersion": "$DEFAULT",
        "LastUpdateAssociationDate": 1487875500.33,
        "Date": 1487875500.33,
        "Targets": [
            {
                "Values": [
                    "i-0cb2b964d3e14fd9f"
                ],
                "Key": "InstanceIds"
            }
        ]
    }
}
```

Per ulteriori informazioni, vedere il riferimento CreateAssociationall'API AWS Systems Manager o Reference: Cron and rate expressions for Systems Manager nella AWS Systems Manager User Guide.

- Per i dettagli sull'API, vedere CreateAssociationin AWS CLI Command Reference.

`create-document`

Il seguente esempio di codice mostra come utilizzare`create-document`.

AWS CLI

Come creare un documento

L'esempio seguente `create-document` crea un documento di Systems Manager.

```
aws ssm create-document \
    --content file://exampleDocument.yml \
    --name "Example" \
    --document-type "Automation" \
    --document-format YAML
```

Output:

```
{
    "DocumentDescription": {
        "Hash": "fc2410281f40779e694a8b95975d0f9f316da8a153daa94e3d9921102EXAMPLE",
        "HashType": "Sha256",
        "Name": "Example",
        "Owner": "29884EXAMPLE",
        "CreatedDate": 1583256349.452,
        "Status": "Creating",
        "DocumentVersion": "1",
        "Description": "Document Example",
        "Parameters": [
            {
                "Name": "AutomationAssumeRole",
                "Type": "String",
                "Description": "(Required) The ARN of the role that allows
 Automation to perform the actions on your behalf. If no role is specified, Systems
 Manager Automation uses your IAM permissions to execute this document.",
                "DefaultValue": ""
            },
```

```
            {
                "Name": "InstanceId",
                "Type": "String",
                "Description": "(Required) The ID of the Amazon EC2 instance.",
                "DefaultValue": ""
            }
        ],
        "PlatformTypes": [
            "Windows",
            "Linux"
        ],
        "DocumentType": "Automation",
        "SchemaVersion": "0.3",
        "LatestVersion": "1",
        "DefaultVersion": "1",
        "DocumentFormat": "YAML",
        "Tags": []
    }
}
```

Per ulteriori informazioni, consulta Creazione di documenti di Systems Manager nella Guida per l'utente di AWS Systems Manager.

- Per i dettagli sull'API, consulta CreateDocument AWS CLICommand Reference.

`create-maintenance-window`

Il seguente esempio di codice mostra come utilizzare`create-maintenance-window`.

AWS CLI

Esempio 1: come creare una finestra di manutenzione

L'esempio seguente `create-maintenance-window` crea una nuova finestra di manutenzione che ogni cinque minuti per un massimo di due ore (se necessario), impedisce l'avvio di nuove attività entro un'ora dalla fine dell'esecuzione della finestra di manutenzione, consente destinazioni non associate (istanze che non sono state registrate nella finestra di manutenzione) e indica, tramite l'uso di tag personalizzati, che il suo creatore intende utilizzarla in un tutorial.

```
aws ssm create-maintenance-window \
    --name "My-Tutorial-Maintenance-Window" \
    --schedule "rate(5 minutes)" \
```

```
    --duration 2 --cutoff 1 \
    --allow-unassociated-targets \
    --tags "Key=Purpose,Value=Tutorial"
```

Output:

```
{
    "WindowId": "mw-0c50858d01EXAMPLE"
}
```

Esempio 2: come creare una finestra di manutenzione che viene eseguita solo una volta

L'esempio seguente `create-maintenance-window` crea una nuova finestra di manutenzione che viene eseguita una sola volta nella data e all'ora specificate.

```
aws ssm create-maintenance-window \
    --name My-One-Time-Maintenance-Window \
    --schedule "at(2020-05-14T15:55:00)" \
    --duration 5 \
    --cutoff 2 \
    --allow-unassociated-targets \
    --tags "Key=Environment,Value=Production"
```

Output:

```
{
    "WindowId": "mw-01234567890abcdef"
}
```

Per ulteriori informazioni, consulta Finestre di manutenzione nella Guida per l'utente di AWS Systems Manager.

- Per i dettagli sull'API, consulta CreateMaintenanceWindow AWS CLICommand Reference.

`create-ops-item`

Il seguente esempio di codice mostra come utilizzare`create-ops-item`.

AWS CLI

Per creare un OpsItems

L'`create-ops-item`esempio seguente utilizza la chiave /aws/resources OperationalData per creare una OpsItem risorsa correlata ad Amazon DynamoDB.

```
aws ssm create-ops-item \
    --title "EC2 instance disk full" \
    --description "Log clean up may have failed which caused the disk to be full" \
    --priority 2 \
    --source ec2 \
    --operational-data '{"/aws/resources":{"Value":"[{\"arn
\": \"arn:aws:dynamodb:us-west-2:12345678:table/OpsItems
\"}]","Type":"SearchableString"}}' \
    --notifications Arn="arn:aws:sns:us-west-2:12345678:TestUser"
```

Output:

```
{
    "OpsItemId": "oi-1a2b3c4d5e6f"
}
```

Per ulteriori informazioni, vedere Creating OpsItems in the AWS Systems Manager User Guide.

- Per i dettagli sull'API, vedere CreateOpsItemin AWS CLI Command Reference.

`create-patch-baseline`

Il seguente esempio di codice mostra come utilizzare`create-patch-baseline`.

AWS CLI

Esempio 1: come creare una baseline delle patch con approvazione automatica

L'esempio seguente `create-patch-baseline` crea una baseline delle patch per Windows Server che approva le patch per un ambiente di produzione sette giorni dopo che sono state rilasciate da Microsoft.

```
aws ssm create-patch-baseline \
    --name "Windows-Production-Baseline-AutoApproval" \
    --operating-system "WINDOWS" \
    --approval-
rules "PatchRules=[{PatchFilterGroup={PatchFilters=[{Key=MSRC_SEVERITY,Values=[Critical,Impo
{Key=CLASSIFICATION,Values=[SecurityUpdates,Updates,UpdateRollups,CriticalUpdates]}]},Approv
 \
```

```
    --description "Baseline containing all updates approved for Windows Server
 production systems"
```

Output:

```
{
    "BaselineId": "pb-045f10b4f3EXAMPLE"
}
```

Esempio 2: come creare una baseline delle patch con una data limite di approvazione

L'esempio seguente `create-patch-baseline` crea una baseline delle patch per Windows Server che approva tutte le patch per un ambiente di produzione rilasciate entro il 7 luglio 2020.

```
aws ssm create-patch-baseline \
    --name "Windows-Production-Baseline-AutoApproval" \
    --operating-system "WINDOWS" \
    --approval-
rules "PatchRules=[{PatchFilterGroup={PatchFilters=[{Key=MSRC_SEVERITY,Values=[Critical,Impo
{Key=CLASSIFICATION,Values=[SecurityUpdates,Updates,UpdateRollups,CriticalUpdates]}]},Approv
 \
    --description "Baseline containing all updates approved for Windows Server
 production systems"
```

Output:

```
{
    "BaselineId": "pb-045f10b4f3EXAMPLE"
}
```

Esempio 3: come creare una baseline delle patch con regole di approvazione archiviate in un file JSON

L'esempio seguente `create-patch-baseline` crea una baseline delle patch per Amazon Linux 2017.09 che approva le patch per un ambiente di produzione sette giorni dopo il loro rilascio, specifica le regole di approvazione per la baseline della patch e specifica un repository personalizzato per le patch.

```
aws ssm create-patch-baseline \
    --cli-input-json file://my-amazon-linux-approval-rules-and-repo.json
```

Contenuto di `my-amazon-linux-approval-rules-and-repo.json`:

```
{
    "Name": "Amazon-Linux-2017.09-Production-Baseline",
    "Description": "My approval rules patch baseline for Amazon Linux 2017.09
 instances",
    "OperatingSystem": "AMAZON_LINUX",
    "Tags": [
        {
            "Key": "Environment",
            "Value": "Production"
        }
    ],
    "ApprovalRules": {
        "PatchRules": [
            {
                "ApproveAfterDays": 7,
                "EnableNonSecurity": true,
                "PatchFilterGroup": {
                    "PatchFilters": [
                        {
                            "Key": "SEVERITY",
                            "Values": [
                                "Important",
                                "Critical"
                            ]
                        },
                        {
                            "Key": "CLASSIFICATION",
                            "Values": [
                                "Security",
                                "Bugfix"
                            ]
                        },
                        {
                            "Key": "PRODUCT",
                            "Values": [
                                "AmazonLinux2017.09"
                            ]
                        }
                    ]
                }
            }
        ]
```

```
    },
    "Sources": [
        {
            "Name": "My-AL2017.09",
            "Products": [
                "AmazonLinux2017.09"
            ],
            "Configuration": "[amzn-main] \nname=amzn-main-Base
\nmirrorlist=http://repo./$awsregion./$awsdomain//$releasever/main/mirror.list //
nmirrorlist_expire=300//nmetadata_expire=300 \npriority=10 \nfailovermethod=priority
 \nfastestmirror_enabled=0 \ngpgcheck=1 \ngpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-
KEY-amazon-ga \nenabled=1 \nretries=3 \ntimeout=5\nreport_instanceid=yes"
        }
    ]
}
```

Esempio 4: come creare una baseline delle patch che specifichi le patch approvate e rifiutate

L'esempio seguente `create-patch-baseline` specifica esplicitamente le patch da approvare e rifiutare come eccezione alle regole di approvazione predefinite.

```
aws ssm create-patch-baseline \
    --name "Amazon-Linux-2017.09-Alpha-Baseline" \
    --description "My custom approve/reject patch baseline for Amazon Linux 2017.09
 instances" \
    --operating-system "AMAZON_LINUX" \
    --approved-patches "CVE-2018-1234567,example-pkg-EE-2018*.amzn1.noarch" \
    --approved-patches-compliance-level "HIGH" \
    --approved-patches-enable-non-security \
    --tags "Key=Environment,Value=Alpha"
```

Per ulteriori informazioni, consulta Creare una baseline delle patch personalizzata nella Guida per l'utente di AWS Systems Manager.

- Per i dettagli sull'API, consulta CreatePatchBaseline AWS CLICommand Reference.

`create-resource-data-sync`

Il seguente esempio di codice mostra come utilizzare`create-resource-data-sync`.

AWS CLI

Come creare una sincronizzazione dei dati delle risorse

In questo esempio viene creata una sincronizzazione dei dati delle risorse. Se il comando va a buon fine, non viene generato output.

Comando:

```
aws ssm create-resource-data-sync --sync-name "ssm-resource-data-sync" --s3-destination "BucketName=ssm-bucket,Prefix=inventory,SyncFormat=JsonSerDe,Region=us-east-1"
```

- Per i dettagli sull'API, consulta CreateResourceDataSync AWS CLICommand Reference.

`delete-activation`

Il seguente esempio di codice mostra come utilizzare`delete-activation`.

AWS CLI

Come eliminare un'attivazione di un'istanza gestita

L'esempio seguente `delete-activation` elimina l'attivazione di un'istanza gestita.

```
aws ssm delete-activation \
    --activation-id "aa673477-d926-42c1-8757-1358cEXAMPLE"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, vedere Configurazione di AWS Systems Manager per ambienti ibridi nella Guida per l'utente di AWS Systems Manager.

- Per i dettagli sull'API, vedere DeleteActivationin AWS CLI Command Reference.

`delete-association`

Il seguente esempio di codice mostra come utilizzare`delete-association`.

AWS CLI

Esempio 1: come eliminare un'associazione utilizzando l'ID dell'associazione

L'esempio seguente `delete-association` elimina l'associazione per l'ID di associazione specificato. Se il comando va a buon fine, non viene generato output.

```
aws ssm delete-association \
    --association-id "8dfe3659-4309-493a-8755-0123456789ab"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Modifica e creazione di una nuova versione di un'associazione nella Guida per l'utente di AWS Systems Manager.

Esempio 2: come eliminare un'associazione

L'esempio seguente `delete-association` elimina l'associazione tra un'istanza e un documento. Se il comando va a buon fine, non viene generato output.

```
aws ssm delete-association \
    --instance-id "i-1234567890abcdef0" \
    --name "AWS-UpdateSSMAgent"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Utilizzo delle associazioni in Systems Manager nella Guida per l'utente di AWS Systems Manager.

- Per i dettagli sull'API, consulta DeleteAssociation AWS CLICommand Reference.

`delete-document`

Il seguente esempio di codice mostra come utilizzare`delete-document`.

AWS CLI

Come eliminare un documento

L'esempio seguente `delete-document` elimina un documento Systems Manager.

```
aws ssm delete-document \
    --name "Example"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Creazione di documenti di Systems Manager nella Guida per l'utente di AWS Systems Manager.

- Per i dettagli sull'API, consulta DeleteDocument AWS CLICommand Reference.

`delete-inventory`

Il seguente esempio di codice mostra come utilizzare`delete-inventory`.

AWS CLI

Come eliminare un tipo di inventario personalizzato

Questo esempio elimina uno schema di inventario personalizzato.

Comando:

```
aws ssm delete-inventory --type-name "Custom:RackInfo" --schema-delete-
option "DeleteSchema"
```

Output:

```
{
  "DeletionId": "d72ac9e8-1f60-4d40-b1c6-bf8c78c68c4d",
  "TypeName": "Custom:RackInfo",
  "DeletionSummary": {
      "TotalCount": 1,
      "RemainingCount": 1,
      "SummaryItems": [
          {
              "Version": "1.0",
              "Count": 1,
              "RemainingCount": 1
          }
      ]
  }
}
```

Come disabilitare un tipo di inventario personalizzato

Questo esempio disabilita uno schema di inventario personalizzato.

Comando:

```
aws ssm delete-inventory --type-name "Custom:RackInfo" --schema-delete-
option "DisableSchema"
```

Output:

```
{
  "DeletionId": "6961492a-8163-44ec-aa1e-923364dd0850",
  "TypeName": "Custom:RackInformation",
  "DeletionSummary": {
      "TotalCount": 0,
      "RemainingCount": 0,
      "SummaryItems": []
  }
}
```

- Per i dettagli sull'API, consulta DeleteInventory AWS CLICommand Reference.

`delete-maintenance-window`

Il seguente esempio di codice mostra come utilizzare`delete-maintenance-window`.

AWS CLI

Come eliminare una finestra di manutenzione

Questo esempio `delete-maintenance-window` rimuove la finestra di manutenzione specificata.

```
aws ssm delete-maintenance-window \
    --window-id "mw-1a2b3c4d5e6f7g8h9"
```

Output:

```
{
    "WindowId":"mw-1a2b3c4d5e6f7g8h9"
}
```

Per ulteriori informazioni, vedere Delete a Maintenance Window (AWS CLI) nella AWS Systems Manager User Guide.

- Per i dettagli sull'API, vedere DeleteMaintenanceWindowin AWS CLI Command Reference.

`delete-parameter`

Il seguente esempio di codice mostra come utilizzare`delete-parameter`.

AWS CLI

Come eliminare un parametro

L'esempio seguente `delete-parameter` elimina il singolo parametro specificato.

```
aws ssm delete-parameter \
    --name "MyParameter"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Utilizzo dell'Archivio parametri nella Guida per l'utente di AWS Systems Manager.

- Per i dettagli sull'API, consulta DeleteParameter AWS CLICommand Reference.

`delete-parameters`

Il seguente esempio di codice mostra come utilizzare`delete-parameters`.

AWS CLI

Come eliminare un elenco di parametri

L'esempio `delete-parameters` seguente elimina il singolo parametro specificato.

```
aws ssm delete-parameters \
    --names "MyFirstParameter" "MySecondParameter" "MyInvalidParameterName"
```

Output:

```
{
    "DeletedParameters": [
        "MyFirstParameter",
        "MySecondParameter"
    ],
    "InvalidParameters": [
        "MyInvalidParameterName"
    ]
}
```

Per ulteriori informazioni, consulta Utilizzo dell'Archivio parametri nella Guida per l'utente di AWS Systems Manager.

- Per i dettagli sull'API, consulta DeleteParameters AWS CLICommand Reference.

`delete-patch-baseline`

Il seguente esempio di codice mostra come utilizzare`delete-patch-baseline`.

AWS CLI

Come eliminare una baseline delle patch

L'esempio seguente `delete-patch-baseline` elimina la baseline delle patch specificata.

```
aws ssm delete-patch-baseline \
    --baseline-id "pb-045f10b4f382baeda"
```

Output:

```
{
    "BaselineId": "pb-045f10b4f382baeda"
}
```

Per ulteriori informazioni, consulta Aggiornare o eliminare una baseline delle patch (console) nella Guida per l'utente di AWS Systems Manager.

- Per i dettagli sull'API, consulta DeletePatchBaseline AWS CLICommand Reference.

`delete-resource-data-sync`

Il seguente esempio di codice mostra come utilizzare`delete-resource-data-sync`.

AWS CLI

Come eliminare una sincronizzazione dei dati delle risorse

In questo esempio viene eliminata una sincronizzazione dei dati delle risorse. Se il comando va a buon fine, non viene generato output.

Comando:

```
aws ssm delete-resource-data-sync --sync-name "ssm-resource-data-sync"
```

- Per i dettagli sull'API, consulta DeleteResourceDataSync AWS CLICommand Reference.

`deregister-managed-instance`

Il seguente esempio di codice mostra come utilizzare`deregister-managed-instance`.

AWS CLI

Come annullare la registrazione di un'istanza gestita

L'esempio seguente `deregister-managed-instance` annulla la registrazione dell'istanza gestita specificata.

```
aws ssm deregister-managed-instance \
    --instance-id 'mi-08ab247cdfEXAMPLE'
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Annullare la registrazione di nodi gestiti in ambienti ibridi e multi-cloud nella Guida per l'utente di AWS Systems Manager.

- Per i dettagli sull'API, consulta DeregisterManagedInstance AWS CLICommand Reference.

`deregister-patch-baseline-for-patch-group`

Il seguente esempio di codice mostra come utilizzare`deregister-patch-baseline-for-patch-group`.

AWS CLI

Annullamento della registrazione di un gruppo di patch da una baseline delle patch

L'esempio seguente `deregister-patch-baseline-for-patch-group` annulla la registrazione del gruppo di patch specificato dalle baseline della patch specificata.

```
aws ssm deregister-patch-baseline-for-patch-group \
    --patch-group "Production" \
    --baseline-id "pb-0ca44a362fEXAMPLE"
```

Output:

```
{
  "PatchGroup":"Production",
  "BaselineId":"pb-0ca44a362fEXAMPLE"
```

```
}
```

Per ulteriori informazioni, consulta Aggiungere un gruppo di patch a una baseline delle patch nella Guida per l'utente di AWS Systems Manager.

- Per i dettagli sull'API, consulta DeregisterPatchBaselineForPatchGroup AWS CLICommand Reference.

`deregister-target-from-maintenance-window`

Il seguente esempio di codice mostra come utilizzare`deregister-target-from-maintenance-window`.

AWS CLI

Come rimuovere una destinazione da una finestra di manutenzione

L'esempio seguente `deregister-target-from-maintenance-window` rimuove la destinazione specificata dalla finestra di manutenzione specificata.

```
aws ssm deregister-target-from-maintenance-window \
    --window-id "mw-ab12cd34ef56gh78" \
    --window-target-id "1a2b3c4d-1a2b-1a2b-1a2b-1a2b3c4d-1a2"
```

Output:

```
{
    "WindowId":"mw-ab12cd34ef56gh78",
    "WindowTargetId":"1a2b3c4d-1a2b-1a2b-1a2b-1a2b3c4d-1a2"
}
```

Per ulteriori informazioni, vedere Update a Maintenance Window (AWS CLI) nella AWS Systems Manager User Guide.

- Per i dettagli sull'API, vedere DeregisterTargetFromMaintenanceWindowin AWS CLI Command Reference.

`deregister-task-from-maintenance-window`

Il seguente esempio di codice mostra come utilizzare`deregister-task-from-maintenance-window`.

AWS CLI

Come rimuovere un'attività da una finestra di manutenzione

L'esempio seguente `deregister-task-from-maintenance-window` rimuove l'attività specificata dalla finestra di manutenzione specificata.

```
aws ssm deregister-task-from-maintenance-window \
    --window-id "mw-ab12cd34ef56gh78" \
    --window-task-id "1a2b3c4d-1a2b-1a2b-1a2b-1a2b3c4d5e6c"
```

Output:

```
{
    "WindowTaskId":"1a2b3c4d-1a2b-1a2b-1a2b-1a2b3c4d5e6c",
    "WindowId":"mw-ab12cd34ef56gh78"
}
```

Per ulteriori informazioni, vedere i tutorial di Windows (AWS CLI) sulla manutenzione di Systems Manager nella Guida per l'utente di AWS Systems Manager.

- Per i dettagli sulle API, vedere DeregisterTaskFromMaintenanceWindowin AWS CLI Command Reference.

`describe-activations`

Il seguente esempio di codice mostra come utilizzare`describe-activations`.

AWS CLI

Come descrivere le attivazioni

L'`describe-activations`esempio seguente elenca i dettagli sulle attivazioni nel tuo AWS account.

```
aws ssm describe-activations
```

Output:

```
{
```

```
    "ActivationList": [
        {
            "ActivationId": "5743558d-563b-4457-8682-d16c3EXAMPLE",
            "Description": "Example1",
            "IamRole": "HybridWebServersRole,
            "RegistrationLimit": 5,
            "RegistrationsCount": 5,
            "ExpirationDate": 1584316800.0,
            "Expired": false,
            "CreatedDate": 1581954699.792
        },
        {
            "ActivationId": "3ee0322b-f62d-40eb-b672-13ebfEXAMPLE",
            "Description": "Example2",
            "IamRole": "HybridDatabaseServersRole",
            "RegistrationLimit": 5,
            "RegistrationsCount": 5,
            "ExpirationDate": 1580515200.0,
            "Expired": true,
            "CreatedDate": 1578064132.002
        },
    ]
}
```

Per ulteriori informazioni, consulta Passaggio 4: creare un'attivazione di istanze gestite per un ambiente ibrido nella Guida per l'utente di AWS Systems Manager.

- Per i dettagli sull'API, consulta DescribeActivations AWS CLICommand Reference.

`describe-association-execution-targets`

Il seguente esempio di codice mostra come utilizzare`describe-association-execution-targets`.

AWS CLI

Come ottenere dettagli sull'esecuzione di un'associazione

L'esempio seguente `describe-association-execution-targets` descrive l'esecuzione dell'associazione specificata.

```
aws ssm describe-association-execution-targets \
```

```
    --association-id "8dfe3659-4309-493a-8755-0123456789ab" \
    --execution-id "7abb6378-a4a5-4f10-8312-0123456789ab"
```

Output:

```
{
    "AssociationExecutionTargets": [
        {
            "AssociationId": "8dfe3659-4309-493a-8755-0123456789ab",
            "AssociationVersion": "1",
            "ExecutionId": "7abb6378-a4a5-4f10-8312-0123456789ab",
            "ResourceId": "i-1234567890abcdef0",
            "ResourceType": "ManagedInstance",
            "Status": "Success",
            "DetailedStatus": "Success",
            "LastExecutionDate": 1550505538.497,
            "OutputSource": {
                "OutputSourceId": "97fff367-fc5a-4299-aed8-0123456789ab",
                "OutputSourceType": "RunCommand"
            }
        }
    ]
}
```

Per ulteriori informazioni, consulta Visualizzazione della cronologia delle associazioni nella Guida per l'utente di AWS Systems Manager.

- Per i dettagli sull'API, consulta DescribeAssociationExecutionTargets AWS CLICommand Reference.

`describe-association-executions`

Il seguente esempio di codice mostra come utilizzare`describe-association-executions`.

AWS CLI

Esempio 1: come ottenere i dettagli di tutte le esecuzioni di un'associazione

L'esempio seguente `describe-association-executions` descrive tutte le esecuzioni dell'associazione specificata.

```
aws ssm describe-association-executions \
```

```
    --association-id "8dfe3659-4309-493a-8755-0123456789ab"
```

Output:

```
{
    "AssociationExecutions": [
        {
            "AssociationId": "8dfe3659-4309-493a-8755-0123456789ab",
            "AssociationVersion": "1",
            "ExecutionId": "474925ef-1249-45a2-b93d-0123456789ab",
            "Status": "Success",
            "DetailedStatus": "Success",
            "CreatedTime": 1550505827.119,
            "ResourceCountByStatus": "{Success=1}"
        },
        {
            "AssociationId": "8dfe3659-4309-493a-8755-0123456789ab",
            "AssociationVersion": "1",
            "ExecutionId": "7abb6378-a4a5-4f10-8312-0123456789ab",
            "Status": "Success",
            "DetailedStatus": "Success",
            "CreatedTime": 1550505536.843,
            "ResourceCountByStatus": "{Success=1}"
        },
        ...
    ]
}
```

Per ulteriori informazioni, consulta Visualizzazione della cronologia delle associazioni nella Guida per l'utente di AWS Systems Manager.

Esempio 2: come ottenere i dettagli di tutte le esecuzioni di un'associazione dopo una data e un'ora specifiche

L'esempio seguente `describe-association-executions` descrive tutte le esecuzioni di un'associazione dopo la data e l'ora specificate.

```
aws ssm describe-association-executions \
    --association-id "8dfe3659-4309-493a-8755-0123456789ab" \
    --filters "Key=CreatedTime,Value=2019-02-18T16:00:00Z,Type=GREATER_THAN"
```

Output:

```
{
    "AssociationExecutions": [
        {
            "AssociationId": "8dfe3659-4309-493a-8755-0123456789ab",
            "AssociationVersion": "1",
            "ExecutionId": "474925ef-1249-45a2-b93d-0123456789ab",
            "Status": "Success",
            "DetailedStatus": "Success",
            "CreatedTime": 1550505827.119,
            "ResourceCountByStatus": "{Success=1}"
        },
        {
            "AssociationId": "8dfe3659-4309-493a-8755-0123456789ab",
            "AssociationVersion": "1",
            "ExecutionId": "7abb6378-a4a5-4f10-8312-0123456789ab",
            "Status": "Success",
            "DetailedStatus": "Success",
            "CreatedTime": 1550505536.843,
            "ResourceCountByStatus": "{Success=1}"
        },
        ...
    ]
}
```

Per ulteriori informazioni, consulta Visualizzazione della cronologia delle associazioni nella Guida per l'utente di AWS Systems Manager.

- Per i dettagli sull'API, consulta DescribeAssociationExecutions AWS CLICommand Reference.

`describe-association`

Il seguente esempio di codice mostra come utilizzare`describe-association`.

AWS CLI

Esempio 1: come ottenere i dettagli di un'associazione

L'esempio seguente `describe-association` descrive l'associazione per l'ID di associazione specificato.

```
aws ssm describe-association \
    --association-id "8dfe3659-4309-493a-8755-0123456789ab"
```

Output:


```
{
    "AssociationDescription": {
        "Name": "AWS-GatherSoftwareInventory",
        "AssociationVersion": "1",
        "Date": 1534864780.995,
        "LastUpdateAssociationDate": 1543235759.81,
        "Overview": {
            "Status": "Success",
            "AssociationStatusAggregatedCount": {
                "Success": 2
            }
        },
        "DocumentVersion": "$DEFAULT",
        "Parameters": {
            "applications": [
                "Enabled"
            ],
            "awsComponents": [
                "Enabled"
            ],
            "customInventory": [
                "Enabled"
            ],
            "files": [
                ""
            ],
            "instanceDetailedInformation": [
                "Enabled"
            ],
            "networkConfig": [
                "Enabled"
            ],
            "services": [
                "Enabled"
            ],
            "windowsRegistry": [
                ""
            ],
            "windowsRoles": [
                "Enabled"
            ],
            "windowsUpdates": [
```

```
                "Enabled"
            ]
        },
        "AssociationId": "8dfe3659-4309-493a-8755-0123456789ab",
        "Targets": [
            {
                "Key": "InstanceIds",
                "Values": [
                    "*"
                ]
            }
        ],
        "ScheduleExpression": "rate(24 hours)",
        "LastExecutionDate": 1550501886.0,
        "LastSuccessfulExecutionDate": 1550501886.0,
        "AssociationName": "Inventory-Association"
    }
}
```

Per ulteriori informazioni, consulta Modifica e creazione di una nuova versione di un'associazione nella Guida per l'utente di AWS Systems Manager.

Esempio 2: come ottenere i dettagli di un'associazione per un'istanza e un documento specifici

L'esempio seguente `describe-association` descrive l'associazione tra un'istanza e un documento.

```
aws ssm describe-association \
    --instance-id "i-1234567890abcdef0" \
    --name "AWS-UpdateSSMAgent"
```

Output:

```
{
    "AssociationDescription": {
        "Status": {
            "Date": 1487876122.564,
            "Message": "Associated with AWS-UpdateSSMAgent",
            "Name": "Associated"
        },
        "Name": "AWS-UpdateSSMAgent",
        "InstanceId": "i-1234567890abcdef0",
        "Overview": {
```

```
            "Status": "Pending",
            "DetailedStatus": "Associated",
            "AssociationStatusAggregatedCount": {
                "Pending": 1
            }
        },
        "AssociationId": "d8617c07-2079-4c18-9847-1234567890ab",
        "DocumentVersion": "$DEFAULT",
        "LastUpdateAssociationDate": 1487876122.564,
        "Date": 1487876122.564,
        "Targets": [
            {
                "Values": [
                    "i-1234567890abcdef0"
                ],
                "Key": "InstanceIds"
            }
        ]
    }
}
```

Per ulteriori informazioni, consulta Modifica e creazione di una nuova versione di un'associazione nella Guida per l'utente di AWS Systems Manager.

- Per i dettagli sull'API, consulta DescribeAssociation AWS CLICommand Reference.

`describe-automation-executions`

Il seguente esempio di codice mostra come utilizzare`describe-automation-executions`.

AWS CLI

Come descrivere un'esecuzione di automazione

L'esempio seguente `describe-automation-executions` mostra i dettagli sull'esecuzione di un'automazione.

```
aws ssm describe-automation-executions \
    --filters Key=ExecutionId,Values=73c8eef8-f4ee-4a05-820c-e354fEXAMPLE
```

Output:

```
{
```

```
    "AutomationExecutionMetadataList": [
        {
            "AutomationExecutionId": "73c8eef8-f4ee-4a05-820c-e354fEXAMPLE",
            "DocumentName": "AWS-StartEC2Instance",
            "DocumentVersion": "1",
            "AutomationExecutionStatus": "Success",
            "ExecutionStartTime": 1583737233.748,
            "ExecutionEndTime": 1583737234.719,
            "ExecutedBy": "arn:aws:sts::29884EXAMPLE:assumed-role/mw_service_role/
OrchestrationService",
            "LogFile": "",
            "Outputs": {},
            "Mode": "Auto",
            "Targets": [],
            "ResolvedTargets": {
                "ParameterValues": [],
                "Truncated": false
            },
            "AutomationType": "Local"
        }
    ]
}
```

Per ulteriori informazioni, consulta Esecuzione di un flusso di lavoro di automazione Simple (Semplice) nella Guida per l'utente di AWS Systems Manager.

- Per i dettagli sull'API, consulta DescribeAutomationExecutions AWS CLICommand Reference.

`describe-automation-step-executions`

Il seguente esempio di codice mostra come utilizzare`describe-automation-step-executions`.

AWS CLI

Esempio 1: come descrivere tutti i passaggi per l'esecuzione di un'automazione

L'esempio seguente `describe-automation-step-executions` mostra i dettagli sui passaggi di un'esecuzione di automazione.

```
aws ssm describe-automation-step-executions \
    --automation-execution-id 73c8eef8-f4ee-4a05-820c-e354fEXAMPLE
```

Output:

```
{
    "StepExecutions": [
        {
            "StepName": "startInstances",
            "Action": "aws:changeInstanceState",
            "ExecutionStartTime": 1583737234.134,
            "ExecutionEndTime": 1583737234.672,
            "StepStatus": "Success",
            "Inputs": {
                "DesiredState": "\"running\"",
                "InstanceIds": "[\"i-0cb99161f6EXAMPLE\"]"
            },
            "Outputs": {
                "InstanceStates": [
                    "running"
                ]
            },
            "StepExecutionId": "95e70479-cf20-4d80-8018-7e4e2EXAMPLE",
            "OverriddenParameters": {}
        }
    ]
}
```

Esempio 2: come descrivere un passaggio specifico per l'esecuzione di un'automazione

L'esempio seguente `describe-automation-step-executions` mostra i dettagli su un passaggio specifico di un'esecuzione di automazione.

```
aws ssm describe-automation-step-executions \
    --automation-execution-id 73c8eef8-f4ee-4a05-820c-e354fEXAMPLE \
    --filters Key=StepExecutionId,Values=95e70479-cf20-4d80-8018-7e4e2EXAMPLE
```

Per ulteriori informazioni, consulta Procedura dettagliata per l'esecuzione di un flusso di lavoro di automazione (riga di comando) nella Guida per l'utente di AWS Systems Manager.

- Per i dettagli sull'API, consulta DescribeAutomationStepExecutions AWS CLICommand Reference.

`describe-available-patches`

Il seguente esempio di codice mostra come utilizzare`describe-available-patches`.

AWS CLI

Come ottenere le patch disponibili

L'esempio seguente `describe-available-patches` recupera i dettagli su tutte le patch disponibili per Windows Server 2019 che hanno una gravità MSRC pari a Critical.

```
aws ssm describe-available-patches \
    --
filters "Key=PRODUCT,Values=WindowsServer2019" "Key=MSRC_SEVERITY,Values=Critical"
```

Output:

```
{
    "Patches": [
        {
            "Id": "fe6bd8c2-3752-4c8b-ab3e-1a7ed08767ba",
            "ReleaseDate": 1544047205.0,
            "Title": "2018-11 Update for Windows Server 2019 for x64-based Systems
 (KB4470788)",
            "Description": "Install this update to resolve issues in Windows. For a
 complete listing of the issues that are included in this update, see the associated
 Microsoft Knowledge Base article for more information. After you install this item,
 you may have to restart your computer.",
            "ContentUrl": "https://support.microsoft.com/en-us/kb/4470788",
            "Vendor": "Microsoft",
            "ProductFamily": "Windows",
            "Product": "WindowsServer2019",
            "Classification": "SecurityUpdates",
            "MsrcSeverity": "Critical",
            "KbNumber": "KB4470788",
            "MsrcNumber": "",
            "Language": "All"
        },
        {
            "Id": "c96115e1-5587-4115-b851-22baa46a3f11",
            "ReleaseDate": 1549994410.0,
            "Title": "2019-02 Security Update for Adobe Flash Player for Windows
 Server 2019 for x64-based Systems (KB4487038)",
            "Description": "A security issue has been identified in a Microsoft
 software product that could affect your system. You can help protect your system
 by installing this update from Microsoft. For a complete listing of the issues that
```

```
 are included in this update, see the associated Microsoft Knowledge Base article.
 After you install this update, you may have to restart your system.",
            "ContentUrl": "https://support.microsoft.com/en-us/kb/4487038",
            "Vendor": "Microsoft",
            "ProductFamily": "Windows",
            "Product": "WindowsServer2019",
            "Classification": "SecurityUpdates",
            "MsrcSeverity": "Critical",
            "KbNumber": "KB4487038",
            "MsrcNumber": "",
            "Language": "All"
        },
        ...
    ]
}
```

Come ottenere i dettagli di una patch specifica

L'esempio seguente `describe-available-patches` recupera i dettagli per la patch specificata.

```
aws ssm describe-available-patches \
    --filters "Key=PATCH_ID,Values=KB4480979"
```

Output:

```
{
    "Patches": [
        {
            "Id": "680861e3-fb75-432e-818e-d72e5f2be719",
            "ReleaseDate": 1546970408.0,
            "Title": "2019-01 Security Update for Adobe Flash Player for Windows
 Server 2016 for x64-based Systems (KB4480979)",
            "Description": "A security issue has been identified in a Microsoft
 software product that could affect your system. You can help protect your system
 by installing this update from Microsoft. For a complete listing of the issues that
 are included in this update, see the associated Microsoft Knowledge Base article.
 After you install this update, you may have to restart your system.",
            "ContentUrl": "https://support.microsoft.com/en-us/kb/4480979",
            "Vendor": "Microsoft",
            "ProductFamily": "Windows",
            "Product": "WindowsServer2016",
            "Classification": "SecurityUpdates",
```

```
            "MsrcSeverity": "Critical",
            "KbNumber": "KB4480979",
            "MsrcNumber": "",
            "Language": "All"
        }
    ]
}
```

Per ulteriori informazioni, consulta Come funzionano le operazioni di Patch Manager nella Guida per l'utente di AWS Systems Manager.

- Per i dettagli sull'API, consulta DescribeAvailablePatches AWS CLICommand Reference.

`describe-document-permission`

Il seguente esempio di codice mostra come utilizzare`describe-document-permission`.

AWS CLI

Come descrivere le autorizzazioni dei documenti

L'esempio seguente `describe-document-permission` visualizza i dettagli delle autorizzazioni relative a un documento Systems Manager condiviso pubblicamente.

```
aws ssm describe-document-permission \
    --name "Example" \
    --permission-type "Share"
```

Output:

```
{
    "AccountIds": [
        "all"
    ],
    "AccountSharingInfoList": [
        {
            "AccountId": "all",
            "SharedDocumentVersion": "$DEFAULT"
        }
    ]
}
```

Per ulteriori informazioni, consulta Condividere un documento Systems Manager nella Guida per l'utente di AWS Systems Manager.

- Per i dettagli sull'API, consulta DescribeDocumentPermission AWS CLICommand Reference.

`describe-document`

Il seguente esempio di codice mostra come utilizzare`describe-document`.

AWS CLI

Come visualizzare i dettagli di un documento

L'`describe-document`esempio seguente mostra i dettagli su un documento Systems Manager nel tuo AWS account.

```
aws ssm describe-document \
    --name "Example"
```

Output:

```
{
    "Document": {
        "Hash": "fc2410281f40779e694a8b95975d0f9f316da8a153daa94e3d9921102EXAMPLE",
        "HashType": "Sha256",
        "Name": "Example",
        "Owner": "29884EXAMPLE",
        "CreatedDate": 1583257938.266,
        "Status": "Active",
        "DocumentVersion": "1",
        "Description": "Document Example",
        "Parameters": [
            {
                "Name": "AutomationAssumeRole",
                "Type": "String",
                "Description": "(Required) The ARN of the role that allows
 Automation to perform the actions on your behalf. If no role is specified, Systems
 Manager Automation uses your IAM permissions to execute this document.",
                "DefaultValue": ""
            },
            {
                "Name": "InstanceId",
                "Type": "String",
```

```
                "Description": "(Required) The ID of the Amazon EC2 instance.",
                "DefaultValue": ""
            }
        ],
        "PlatformTypes": [
            "Windows",
            "Linux"
        ],
        "DocumentType": "Automation",
        "SchemaVersion": "0.3",
        "LatestVersion": "1",
        "DefaultVersion": "1",
        "DocumentFormat": "YAML",
        "Tags": []
    }
}
```

Per ulteriori informazioni, consulta Creazione di documenti di Systems Manager nella Guida per l'utente di AWS Systems Manager.

- Per i dettagli sull'API, vedere DescribeDocumentin AWS CLI Command Reference.

`describe-effective-instance-associations`

Il seguente esempio di codice mostra come utilizzare`describe-effective-instance-associations`.

AWS CLI

Come ottenere dettagli sulle associazioni efficaci per un'istanza

L'esempio seguente `describe-effective-instance-associations` recupera i dettagli sulle associazioni efficaci per un'istanza.

Comando:

```
aws ssm describe-effective-instance-associations --instance-id "i-1234567890abcdef0"
```

Output:

```
{
    "Associations": [
        {
```

```
            "AssociationId": "8dfe3659-4309-493a-8755-0123456789ab",
            "InstanceId": "i-1234567890abcdef0",
            "Content": "{\n    \"schemaVersion\": \"1.2\",\n    \"description\":
 \"Update the Amazon SSM Agent to the latest version or specified version.\",\n
   \"parameters\": {\n        \"version\": {\n            \"default\": \"\",\n
         \"description\": \"(Optional) A specific version of the Amazon SSM Agent
 to install. If not specified, the agent will be updated to the latest version.\",
\n            \"type\": \"String\"\n        },\n        \"allowDowngrade\": {\n
            \"default\": \"false\",\n            \"description\": \"(Optional)
 Allow the Amazon SSM Agent service to be downgraded to an earlier version. If
 set to false, the service can be upgraded to newer versions only (default). If
 set to true, specify the earlier version.\",\n            \"type\": \"String\",
\n            \"allowedValues\": [\n                \"true\",\n
 \"false\"\n            ]\n        }\n    },\n    \"runtimeConfig\": {\n
 \"aws:updateSsmAgent\": {\n            \"properties\": [\n                {\n
              \"agentName\": \"amazon-ssm-agent\",\n                \"source\":
 \"https://s3.{Region}.amazonaws.com/amazon-ssm-{Region}/ssm-agent-manifest.json\",
\n                \"allowDowngrade\": \"{{ allowDowngrade }}\",\n
 \"targetVersion\": \"{{ version }}\"\n                }\n            ]\n        }\n
    }\n}\n",
            "AssociationVersion": "1"
        }
    ]
}
```

- Per i dettagli sull'API, consulta DescribeEffectiveInstanceAssociations AWS CLICommand Reference.

`describe-effective-patches-for-patch-baseline`

Il seguente esempio di codice mostra come utilizzare`describe-effective-patches-for-patch-baseline`.

AWS CLI

Esempio 1: ottenimento di tutte le patch definite da una baseline delle patch

L'`describe-effective-patches-for-patch-baseline`esempio seguente restituisce le patch definite da una linea di base di patch personalizzata nell'account corrente AWS . Si noti che per una baseline personalizzata, è richiesto solo l'ID per `--baseline-id`.

```
aws ssm describe-effective-patches-for-patch-baseline \
```

```
    --baseline-id "pb-08b654cf9b9681f04"
```

Output:

```
{
    "EffectivePatches": [
        {
            "Patch": {
                "Id": "fe6bd8c2-3752-4c8b-ab3e-1a7ed08767ba",
                "ReleaseDate": 1544047205.0,
                "Title": "2018-11 Update for Windows Server 2019 for x64-based
 Systems (KB4470788)",
                "Description": "Install this update to resolve issues in Windows.
 For a complete listing of the issues that are included in this update, see the
 associated Microsoft Knowledge Base article for more information. After you install
 this item, you may have to restart your computer.",
                "ContentUrl": "https://support.microsoft.com/en-us/kb/4470788",
                "Vendor": "Microsoft",
                "ProductFamily": "Windows",
                "Product": "WindowsServer2019",
                "Classification": "SecurityUpdates",
                "MsrcSeverity": "Critical",
                "KbNumber": "KB4470788",
                "MsrcNumber": "",
                "Language": "All"
            },
            "PatchStatus": {
                "DeploymentStatus": "APPROVED",
                "ComplianceLevel": "CRITICAL",
                "ApprovalDate": 1544047205.0
            }
        },
        {
            "Patch": {
                "Id": "915a6b1a-f556-4d83-8f50-b2e75a9a7e58",
                "ReleaseDate": 1549994400.0,
                "Title": "2019-02 Cumulative Update for .NET Framework 3.5 and 4.7.2
 for Windows Server 2019 for x64 (KB4483452)",
                "Description": "A security issue has been identified in a Microsoft
 software product that could affect your system. You can help protect your system by
 installing this update from Microsoft. For a complete listing of the issues that
 are included in this update, see the associated Microsoft Knowledge Base article.
 After you install this update, you may have to restart your system.",
```

```
                "ContentUrl": "https://support.microsoft.com/en-us/kb/4483452",
                "Vendor": "Microsoft",
                "ProductFamily": "Windows",
                "Product": "WindowsServer2019",
                "Classification": "SecurityUpdates",
                "MsrcSeverity": "Important",
                "KbNumber": "KB4483452",
                "MsrcNumber": "",
                "Language": "All"
            },
            "PatchStatus": {
                "DeploymentStatus": "APPROVED",
                "ComplianceLevel": "CRITICAL",
                "ApprovalDate": 1549994400.0
            }
        },
        ...
    ],
    "NextToken": "--token string truncated--"
}
```

Esempio 2: per ottenere tutte le patch definite da una AWS baseline di patch gestite

L'`describe-effective-patches-for-patch-baseline`esempio seguente restituisce le patch definite da una baseline di patch AWS gestite. Si noti che per una baseline AWS gestita, è necessario l'ARN di base completo per `--baseline-id`

```
aws ssm describe-effective-patches-for-patch-baseline \
    --baseline-id "arn:aws:ssm:us-east-2:733109147000:patchbaseline/
pb-020d361a05defe4ed"
```

Vedi l'esempio 1 per un output di esempio.

Per ulteriori informazioni, consulta Come vengono selezionate le patch di sicurezza nella Guida per l'utente di AWS Systems Manager.

- Per i dettagli sull'API, consulta Command DescribeEffectivePatchesForPatchBaselineReference AWS CLI .

`describe-instance-associations-status`

Il seguente esempio di codice mostra come utilizzare`describe-instance-associations-status`.

AWS CLI

Come descrivere lo stato delle associazioni delle istanze

Questo esempio mostra i dettagli delle associazioni per un'istanza.

Comando:

```
aws ssm describe-instance-associations-status --instance-id "i-1234567890abcdef0"
```

Output:

```
{
  "InstanceAssociationStatusInfos": [
      {
          "AssociationId": "8dfe3659-4309-493a-8755-0123456789ab",
          "Name": "AWS-GatherSoftwareInventory",
          "DocumentVersion": "1",
          "AssociationVersion": "1",
          "InstanceId": "i-1234567890abcdef0",
          "ExecutionDate": 1550501886.0,
          "Status": "Success",
          "ExecutionSummary": "1 out of 1 plugin processed, 1 success, 0 failed, 0
 timedout, 0 skipped. ",
          "AssociationName": "Inventory-Association"
      },
      {
          "AssociationId": "5c5a31f6-6dae-46f9-944c-0123456789ab",
          "Name": "AWS-UpdateSSMAgent",
          "DocumentVersion": "1",
          "AssociationVersion": "1",
          "InstanceId": "i-1234567890abcdef0",
          "ExecutionDate": 1550505828.548,
          "Status": "Success",
          "DetailedStatus": "Success",
          "AssociationName": "UpdateSSMAgent"
      }
  ]
```

```
}
```

- Per i dettagli sull'API, consulta DescribeInstanceAssociationsStatus AWS CLICommand Reference.

`describe-instance-information`

Il seguente esempio di codice mostra come utilizzare`describe-instance-information`.

AWS CLI

Esempio 1: come descrivere le informazioni sulle istanze gestite

L'esempio seguente `describe-instance-information` recupera i dettagli di ciascuna delle istanze gestite.

```
aws ssm describe-instance-information
```

Esempio 2: come descrivere le informazioni su una specifica istanza gestita

L'esempio seguente `describe-instance-information` mostra i dettagli dell'istanza gestita `i-028ea792daEXAMPLE`.

```
aws ssm describe-instance-information \
    --filters "Key=InstanceIds,Values=i-028ea792daEXAMPLE"
```

Esempio 3: come descrivere le informazioni sulle istanze gestite con una chiave tag specifica

L'esempio seguente `describe-instance-information` mostra i dettagli delle istanze gestite con la chiave tag `DEV`.

```
aws ssm describe-instance-information \
    --filters "Key=tag-key,Values=DEV"
```

Output:

```
{
    "InstanceInformationList": [
        {
            "InstanceId": "i-028ea792daEXAMPLE",
```

```
            "PingStatus": "Online",
            "LastPingDateTime": 1582221233.421,
            "AgentVersion": "2.3.842.0",
            "IsLatestVersion": true,
            "PlatformType": "Linux",
            "PlatformName": "SLES",
            "PlatformVersion": "15.1",
            "ResourceType": "EC2Instance",
            "IPAddress": "192.0.2.0",
            "ComputerName": "ip-198.51.100.0.us-east-2.compute.internal",
            "AssociationStatus": "Success",
            "LastAssociationExecutionDate": 1582220806.0,
            "LastSuccessfulAssociationExecutionDate": 1582220806.0,
            "AssociationOverview": {
                "DetailedStatus": "Success",
                "InstanceAssociationStatusAggregatedCount": {
                    "Success": 2
                }
            }
        }
    ]
}
```

Per ulteriori informazioni, consulta Istanze gestite nella Guida per l'utente di AWS Systems Manager.

- Per i dettagli sull'API, consulta DescribeInstanceInformation AWS CLICommand Reference.

`describe-instance-patch-states-for-patch-group`

Il seguente esempio di codice mostra come utilizzare`describe-instance-patch-states-for-patch-group`.

AWS CLI

Esempio 1: come ottenere gli stati dell'istanza per un gruppo di patch

L'esempio seguente `describe-instance-patch-states-for-patch-group` recupera i dettagli sugli stati di riepilogo delle patch per istanza per il gruppo di patch specificato.

```
aws ssm describe-instance-patch-states-for-patch-group \
    --patch-group "Production"
```

Output:


```
{
    "InstancePatchStates": [
        {
            "InstanceId": "i-02573cafcfEXAMPLE",
            "PatchGroup": "Production",
            "BaselineId": "pb-0c10e65780EXAMPLE",
            "SnapshotId": "a3f5ff34-9bc4-4d2c-a665-4d1c1EXAMPLE",
            "OwnerInformation": "",
            "InstalledCount": 32,
            "InstalledOtherCount": 1,
            "InstalledPendingRebootCount": 0,
            "InstalledRejectedCount": 0,
            "MissingCount": 2,
            "FailedCount": 0,
            "UnreportedNotApplicableCount": 2671,
            "NotApplicableCount": 400,
            "OperationStartTime": "2021-08-04T11:03:50.590000-07:00",
            "OperationEndTime": "2021-08-04T11:04:21.555000-07:00",
            "Operation": "Scan",
            "RebootOption": "NoReboot",
            "CriticalNonCompliantCount": 0,
            "SecurityNonCompliantCount": 1,
            "OtherNonCompliantCount": 0
        },
        {
            "InstanceId": "i-0471e04240EXAMPLE",
            "PatchGroup": "Production",
            "BaselineId": "pb-09ca3fb51fEXAMPLE",
            "SnapshotId": "05d8ffb0-1bbe-4812-ba2d-d9b7bEXAMPLE",
            "OwnerInformation": "",
            "InstalledCount": 32,
            "InstalledOtherCount": 1,
            "InstalledPendingRebootCount": 0,
            "InstalledRejectedCount": 0,
            "MissingCount": 2,
            "FailedCount": 0,
            "UnreportedNotApplicableCount": 2671,
            "NotApplicableCount": 400,
            "OperationStartTime": "2021-08-04T22:06:20.340000-07:00",
            "OperationEndTime": "2021-08-04T22:07:11.220000-07:00",
            "Operation": "Scan",
            "RebootOption": "NoReboot",
```

```
            "CriticalNonCompliantCount": 0,
            "SecurityNonCompliantCount": 1,
            "OtherNonCompliantCount": 0
        }
    ]
}
```

Esempio 2: come ottenere gli stati delle istanze per un gruppo di patch con più di cinque patch mancanti

L'esempio seguente `describe-instance-patch-states-for-patch-group` recupera i dettagli sugli stati di riepilogo delle patch per il gruppo di patch specificato per le istanze con più di cinque patch mancanti.

```
aws ssm describe-instance-patch-states-for-patch-group \
    --filters Key=MissingCount,Type=GreaterThan,Values=5 \
    --patch-group "Production"
```

Output:

```
{
    "InstancePatchStates": [
        {
            "InstanceId": "i-02573cafcfEXAMPLE",
            "PatchGroup": "Production",
            "BaselineId": "pb-0c10e65780EXAMPLE",
            "SnapshotId": "a3f5ff34-9bc4-4d2c-a665-4d1c1EXAMPLE",
            "OwnerInformation": "",
            "InstalledCount": 46,
            "InstalledOtherCount": 4,
            "InstalledPendingRebootCount": 1,
            "InstalledRejectedCount": 1,
            "MissingCount": 7,
            "FailedCount": 0,
            "UnreportedNotApplicableCount": 232,
            "NotApplicableCount": 654,
            "OperationStartTime": "2021-08-04T11:03:50.590000-07:00",
            "OperationEndTime": "2021-08-04T11:04:21.555000-07:00",
            "Operation": "Scan",
            "RebootOption": "NoReboot",
            "CriticalNonCompliantCount": 0,
            "SecurityNonCompliantCount": 1,
```

```
            "OtherNonCompliantCount": 1
        }
    ]
}
```

Esempio 3: come ottenere gli stati delle istanze per un gruppo di patch con meno di dieci istanze che richiedono un riavvio

L'esempio seguente `describe-instance-patch-states-for-patch-group` recupera i dettagli sugli stati di riepilogo delle patch per il gruppo di patch specificato per le istanze con meno di dieci istanze che richiedono un riavvio.

```
aws ssm describe-instance-patch-states-for-patch-group \
    --filters Key=InstalledPendingRebootCount,Type=LessThan,Values=10 \
    --patch-group "Production"
```

Output:

```
{
    "InstancePatchStates": [
        {
            "InstanceId": "i-02573cafcfEXAMPLE",
            "BaselineId": "pb-0c10e65780EXAMPLE",
            "SnapshotId": "a3f5ff34-9bc4-4d2c-a665-4d1c1EXAMPLE",
            "PatchGroup": "Production",
            "OwnerInformation": "",
            "InstalledCount": 32,
            "InstalledOtherCount": 1,
            "InstalledPendingRebootCount": 4,
            "InstalledRejectedCount": 0,
            "MissingCount": 2,
            "FailedCount": 0,
            "UnreportedNotApplicableCount": 846,
            "NotApplicableCount": 212,
            "OperationStartTime": "2021-08-046T11:03:50.590000-07:00",
            "OperationEndTime": "2021-08-06T11:04:21.555000-07:00",
            "Operation": "Scan",
            "RebootOption": "NoReboot",
            "CriticalNonCompliantCount": 0,
            "SecurityNonCompliantCount": 1,
            "OtherNonCompliantCount": 0
        }
```

```
    ]
}
```

Per ulteriori informazioni, consulta Comprensione dei valori dello stato di conformità delle patch nella Guida per l'utente di AWS Systems Manager.

- Per i dettagli sull'API, consulta DescribeInstancePatchStatesForPatchGroup AWS CLICommand Reference.

`describe-instance-patch-states`

Il seguente esempio di codice mostra come utilizzare`describe-instance-patch-states`.

AWS CLI

Come ottenere gli stati di riepilogo delle patch per le istanze

Questo esempio `describe-instance-patch-states` viene recuperato lo stato del riepilogo della patch per un'istanza.

```
aws ssm describe-instance-patch-states \
    --instance-ids "i-1234567890abcdef0"
```

Output:

```
{
    "InstancePatchStates": [
        {
            "InstanceId": "i-1234567890abcdef0",
            "PatchGroup": "my-patch-group",
            "BaselineId": "pb-0713accee01234567",
            "SnapshotId": "521c3536-930c-4aa9-950e-01234567abcd",
            "CriticalNonCompliantCount": 2,
            "SecurityNonCompliantCount": 2,
            "OtherNonCompliantCount": 1,
            "InstalledCount": 123,
            "InstalledOtherCount": 334,
            "InstalledPendingRebootCount": 0,
            "InstalledRejectedCount": 0,
            "MissingCount": 1,
            "FailedCount": 2,
            "UnreportedNotApplicableCount": 11,
            "NotApplicableCount": 2063,
```

```
            "OperationStartTime": "2021-05-03T11:00:56-07:00",
            "OperationEndTime": "2021-05-03T11:01:09-07:00",
            "Operation": "Scan",
            "LastNoRebootInstallOperationTime": "2020-06-14T12:17:41-07:00",
            "RebootOption": "RebootIfNeeded"
        }
    ]
}
```

Per ulteriori informazioni, consulta Informazioni sulla conformità delle patch nella Guida per l'utente di AWS Systems Manager.

- Per i dettagli sull'API, consulta DescribeInstancePatchStates AWS CLICommand Reference.

`describe-instance-patches`

Il seguente esempio di codice mostra come utilizzare`describe-instance-patches`.

AWS CLI

Esempio 1: come ottenere i dettagli sullo stato della patch per un'istanza

L'esempio seguente `describe-instance-patches` recupera i dettagli delle patch per l'istanza specificata.

```
aws ssm describe-instance-patches \
    --instance-id "i-1234567890abcdef0"
```

Output:

```
{
    "Patches": [
        {
            "Title": "2019-01 Security Update for Adobe Flash Player for Windows
 Server 2016 for x64-based Systems (KB4480979)",
            "KBId": "KB4480979",
            "Classification": "SecurityUpdates",
            "Severity": "Critical",
            "State": "Installed",
            "InstalledTime": "2019-01-09T00:00:00+00:00"
        },
        {
            "Title": "",
```

```
            "KBId": "KB4481031",
            "Classification": "",
            "Severity": "",
            "State": "InstalledOther",
            "InstalledTime": "2019-02-08T00:00:00+00:00"
        },
        ...
    ],
    "NextToken": "--token string truncated--"
}
```

Esempio 2: come ottenere un elenco di patch nello Stato mancante per un'istanza

L'esempio seguente `describe-instance-patches` recupera informazioni sulle patch che si trovano nello stato Missing per l'istanza specificata.

```
aws ssm describe-instance-patches \
    --instance-id "i-1234567890abcdef0" \
    --filters Key=State,Values=Missing
```

Output:

```
{
    "Patches": [
        {
            "Title": "Windows Malicious Software Removal Tool x64 - February 2019
 (KB890830)",
            "KBId": "KB890830",
            "Classification": "UpdateRollups",
            "Severity": "Unspecified",
            "State": "Missing",
            "InstalledTime": "1970-01-01T00:00:00+00:00"
        },
        ...
    ],
    "NextToken": "--token string truncated--"
}
```

Per ulteriori informazioni, consulta Informazioni sugli stati di conformità delle patch nella Guida per l'utente di AWS Systems Manager.

Esempio 3: Per ottenere un elenco di patch installate dopo un' InstalledTime istanza specificata

L'esempio seguente `describe-instance-patches` recupera le informazioni sulle patch installate a partire da un determinato periodo di tempo per l'istanza specificata combinando l'uso di `--filters` e `--query`.

```
aws ssm describe-instance-patches \
    --instance-id "i-1234567890abcdef0" \
    --filters Key=State,Values=Installed \
    --query "Patches[?InstalledTime >= `2023-01-01T16:00:00`]"
```

Output:

```
{
    "Patches": [
        {
            "Title": "2023-03 Cumulative Update for Windows Server 2019 (1809) for
 x64-based Systems (KB5023702)",
            "KBId": "KB5023702",
            "Classification": "SecurityUpdates",
            "Severity": "Critical",
            "State": "Installed",
            "InstalledTime": "2023-03-16T11:00:00+00:00"
        },
        ...
    ],
    "NextToken": "--token string truncated--"
}
```

- Per i dettagli sull'API, consulta DescribeInstancePatches AWS CLICommand Reference.

`describe-inventory-deletions`

Il seguente esempio di codice mostra come utilizzare`describe-inventory-deletions`.

AWS CLI

Come ottenere le eliminazioni dell'inventario

Questo esempio recupera i dettagli delle operazioni di eliminazione dell'inventario.

Comando:

```
aws ssm describe-inventory-deletions
```

Output:

```
{
  "InventoryDeletions": [
      {
          "DeletionId": "6961492a-8163-44ec-aa1e-01234567850",
          "TypeName": "Custom:RackInformation",
          "DeletionStartTime": 1550254911.0,
          "LastStatus": "InProgress",
          "LastStatusMessage": "The Delete is in progress",
          "DeletionSummary": {
              "TotalCount": 0,
              "RemainingCount": 0,
              "SummaryItems": []
          },
          "LastStatusUpdateTime": 1550254911.0
      },
      {
          "DeletionId": "d72ac9e8-1f60-4d40-b1c6-987654321c4d",
          "TypeName": "Custom:RackInfo",
          "DeletionStartTime": 1550254859.0,
          "LastStatus": "InProgress",
          "LastStatusMessage": "The Delete is in progress",
          "DeletionSummary": {
              "TotalCount": 1,
              "RemainingCount": 1,
              "SummaryItems": [
                  {
                      "Version": "1.0",
                      "Count": 1,
                      "RemainingCount": 1
                  }
              ]
          },
          "LastStatusUpdateTime": 1550254859.0
      }
  ]
}
```

Come ottenere i dettagli di una specifica eliminazione dell'inventario

Questo esempio recupera i dettagli di un'operazione di eliminazione dell'inventario specifica.

Comando:

```
aws ssm describe-inventory-deletions --deletion-id "d72ac9e8-1f60-4d40-
b1c6-987654321c4d"
```

Output:

```
{
  "InventoryDeletions": [
      {
          "DeletionId": "d72ac9e8-1f60-4d40-b1c6-987654321c4d",
          "TypeName": "Custom:RackInfo",
          "DeletionStartTime": 1550254859.0,
          "LastStatus": "InProgress",
          "LastStatusMessage": "The Delete is in progress",
          "DeletionSummary": {
              "TotalCount": 1,
              "RemainingCount": 1,
              "SummaryItems": [
                  {
                      "Version": "1.0",
                      "Count": 1,
                      "RemainingCount": 1
                  }
              ]
          },
          "LastStatusUpdateTime": 1550254859.0
      }
  ]
}
```

- Per i dettagli sull'API, consulta DescribeInventoryDeletions AWS CLICommand Reference.

`describe-maintenance-window-execution-task-invocations`

Il seguente esempio di codice mostra come utilizzare`describe-maintenance-window-execution-task-invocations`.

AWS CLI

Come eseguire le invocazioni di attività specifiche per l'esecuzione di un'operazione in una finestra di manutenzione

L'esempio seguente `describe-maintenance-window-execution-task-invocations` elenca le invocazioni per l'attività specificata eseguite come parte dell'esecuzione della finestra di manutenzione specificata.

```
aws ssm describe-maintenance-window-execution-task-invocations \
    --window-execution-id "518d5565-5969-4cca-8f0e-da3b2a638355" \
    --task-id "ac0c6ae1-daa3-4a89-832e-d384503b6586"
```

Output:

```
{
    "WindowExecutionTaskInvocationIdentities": [
        {
            "Status": "SUCCESS",
            "Parameters": "{\"documentName\":\"AWS-RunShellScript\",\"instanceIds\":
[\"i-0000293ffd8c57862\"],\"parameters\":{\"commands\":[\"df\"]},\"maxConcurrency\":
\"1\",\"maxErrors\":\"1\"}",
            "InvocationId": "e274b6e1-fe56-4e32-bd2a-8073c6381d8b",
            "StartTime": 1487692834.723,
            "EndTime": 1487692834.871,
            "WindowExecutionId": "518d5565-5969-4cca-8f0e-da3b2a638355",
            "TaskExecutionId": "ac0c6ae1-daa3-4a89-832e-d384503b6586"
        }
    ]
}
```

Per ulteriori informazioni, vedere Visualizza informazioni sulle attività e sulle esecuzioni delle attività (AWS CLI) nella Guida per l'utente di Systems AWS Manager.

- Per i dettagli sull'API, vedere DescribeMaintenanceWindowExecutionTaskInvocationsin AWS CLI Command Reference.

`describe-maintenance-window-execution-tasks`

Il seguente esempio di codice mostra come utilizzare`describe-maintenance-window-execution-tasks`.

AWS CLI

Come elencare tutte le attività associate all'esecuzione di una finestra di manutenzione

L'esempio seguente `ssm describe-maintenance-window-execution-tasks` elenca le attività associate all'esecuzione della finestra di manutenzione specificata.

```
aws ssm describe-maintenance-window-execution-tasks \
    --window-execution-id "518d5565-5969-4cca-8f0e-da3b2EXAMPLE"
```

Output:

```
{
    "WindowExecutionTaskIdentities": [
        {
            "Status": "SUCCESS",
            "TaskArn": "AWS-RunShellScript",
            "StartTime": 1487692834.684,
            "TaskType": "RUN_COMMAND",
            "EndTime": 1487692835.005,
            "WindowExecutionId": "518d5565-5969-4cca-8f0e-da3b2EXAMPLE",
            "TaskExecutionId": "ac0c6ae1-daa3-4a89-832e-d3845EXAMPLE"
        }
    ]
}
```

Per ulteriori informazioni, vedere Visualizza informazioni sulle attività e sulle esecuzioni delle attività (AWS CLI) nella Guida per l'utente di Systems AWS Manager.

- Per i dettagli sull'API, vedere DescribeMaintenanceWindowExecutionTasksin AWS CLI Command Reference.

`describe-maintenance-window-executions`

Il seguente esempio di codice mostra come utilizzare`describe-maintenance-window-executions`.

AWS CLI

Esempio 1: come elencare tutte le esecuzioni per una finestra di manutenzione

L'esempio seguente `describe-maintenance-window-executions` elenca tutte le esecuzioni per la finestra di manutenzione specificata.

```
aws ssm describe-maintenance-window-executions \
```

```
    --window-id "mw-ab12cd34eEXAMPLE"
```

Output:

```
{
    "WindowExecutions": [
        {
            "WindowId": "mw-ab12cd34eEXAMPLE",
            "WindowExecutionId": "6027b513-64fe-4cf0-be7d-1191aEXAMPLE",
            "Status": "IN_PROGRESS",
            "StartTime": "2021-08-04T11:00:00.000000-07:00"

        },
        {
            "WindowId": "mw-ab12cd34eEXAMPLE",
            "WindowExecutionId": "ff75b750-4834-4377-8f61-b3cadEXAMPLE",
            "Status": "SUCCESS",
            "StartTime": "2021-08-03T11:00:00.000000-07:00",
            "EndTime": "2021-08-03T11:37:21.450000-07:00"
        },
        {
            "WindowId": "mw-ab12cd34eEXAMPLE",
            "WindowExecutionId": "9fac7dd9-ff21-42a5-96ad-bbc4bEXAMPLE",
            "Status": "FAILED",
            "StatusDetails": "One or more tasks in the orchestration failed.",
            "StartTime": "2021-08-02T11:00:00.000000-07:00",
            "EndTime": "2021-08-02T11:22:36.190000-07:00"
        }
    ]
}
```

Esempio 2: come elencare tutte le esecuzioni relative a una finestra di manutenzione prima di una data specificata

L'esempio seguente `describe-maintenance-window-executions` elenca tutte le esecuzioni per la finestra di manutenzione specificata prima della data specificata.

```
aws ssm describe-maintenance-window-executions \
    --window-id "mw-ab12cd34eEXAMPLE" \
    --filters "Key=ExecutedBefore,Values=2021-08-03T00:00:00Z"
```

Output:

```
{
    "WindowExecutions": [
        {
        "WindowId": "mw-ab12cd34eEXAMPLE",
        "WindowExecutionId": "9fac7dd9-ff21-42a5-96ad-bbc4bEXAMPLE",
        "Status": "FAILED",
        "StatusDetails": "One or more tasks in the orchestration failed.",
        "StartTime": "2021-08-02T11:00:00.000000-07:00",
        "EndTime": "2021-08-02T11:22:36.190000-07:00"
    }
    ]
}
```

Esempio 3: come elencare tutte le esecuzioni per una finestra di manutenzione dopo una data specificata

L'esempio seguente `describe-maintenance-window-executions` elenca tutte le esecuzioni per la finestra di manutenzione specificata dopo la data specificata.

```
aws ssm describe-maintenance-window-executions \
    --window-id "mw-ab12cd34eEXAMPLE" \
    --filters "Key=ExecutedAfter,Values=2021-08-04T00:00:00Z"
```

Output:

```
{
    "WindowExecutions": [
        {
        "WindowId": "mw-ab12cd34eEXAMPLE",
        "WindowExecutionId": "6027b513-64fe-4cf0-be7d-1191aEXAMPLE",
        "Status": "IN_PROGRESS",
        "StartTime": "2021-08-04T11:00:00.000000-07:00"
        }
    ]
}
```

Per ulteriori informazioni, vedere Visualizza informazioni sulle attività e sulle esecuzioni delle attività (AWS CLI) nella Systems AWS Manager User Guide.

- Per i dettagli sull'API, vedere DescribeMaintenanceWindowExecutionsin AWS CLI Command Reference.

`describe-maintenance-window-schedule`

Il seguente esempio di codice mostra come utilizzare`describe-maintenance-window-schedule`.

AWS CLI

Esempio 1: come elencare tutte le esecuzioni in arrivo per una finestra di manutenzione

L'esempio `describe-maintenance-window-schedule` seguente elenca tutte le prossime esecuzioni per la finestra di manutenzione specificata.

```
aws ssm describe-maintenance-window-schedule \
    --window-id mw-ab12cd34eEXAMPLE
```

Output:

```
{
    "ScheduledWindowExecutions": [
        {
            "WindowId": "mw-ab12cd34eEXAMPLE",
            "Name": "My-First-Maintenance-Window",
            "ExecutionTime": "2020-02-19T16:00Z"
        },
        {
            "WindowId": "mw-ab12cd34eEXAMPLE",
            "Name": "My-First-Maintenance-Window",
            "ExecutionTime": "2020-02-26T16:00Z"
        },
        ...
    ]
}
```

Esempio 2: come elencare tutte le esecuzioni in arrivo relative a una finestra di manutenzione prima di una data specificata

L'esempio `describe-maintenance-window-schedule` seguente elenca tutte le prossime esecuzioni per la finestra di manutenzione specificata verificatesi prima della data specificata.

```
aws ssm describe-maintenance-window-schedule \
    --window-id mw-0ecb1226dd7b2e9a6 \
    --filters "Key=ScheduledBefore,Values=2020-02-15T06:00:00Z"
```

Esempio 3: come elencare tutte le esecuzioni in arrivo per una finestra di manutenzione dopo una data specificata

L'esempio `describe-maintenance-window-schedule` seguente elenca tutte le prossime esecuzioni per la finestra di manutenzione specificata verificatesi dopo la data specificata.

```
aws ssm describe-maintenance-window-schedule \
    --window-id mw-0ecb1226dd7b2e9a6 \
    --filters "Key=ScheduledAfter,Values=2020-02-15T06:00:00Z"
```

Per ulteriori informazioni, vedere View Information About Maintenance Windows (AWS CLI) nella AWS Systems Manager User Guide.

- Per i dettagli sull'API, vedere DescribeMaintenanceWindowSchedulein AWS CLI Command Reference.

`describe-maintenance-window-targets`

Il seguente esempio di codice mostra come utilizzare`describe-maintenance-window-targets`.

AWS CLI

Esempio 1: elencare tutti gli obiettivi per una finestra di manutenzione

L'esempio seguente `describe-maintenance-window-targets` elenca tutte le destinazioni per una finestra di manutenzione.

```
aws ssm describe-maintenance-window-targets \
    --window-id "mw-06cf17cbefEXAMPLE"
```

Output:

```
{
    "Targets": [
        {
            "ResourceType": "INSTANCE",
            "OwnerInformation": "Single instance",
            "WindowId": "mw-06cf17cbefEXAMPLE",
            "Targets": [
                {
                    "Values": [
                        "i-0000293ffdEXAMPLE"
```

```
                    ],
                    "Key": "InstanceIds"
                }
            ],
            "WindowTargetId": "350d44e6-28cc-44e2-951f-4b2c9EXAMPLE"
        },
        {
            "ResourceType": "INSTANCE",
            "OwnerInformation": "Two instances in a list",
            "WindowId": "mw-06cf17cbefEXAMPLE",
            "Targets": [
                {
                    "Values": [
                        "i-0000293ffdEXAMPLE",
                        "i-0cb2b964d3EXAMPLE"
                    ],
                    "Key": "InstanceIds"
                }
            ],
            "WindowTargetId": "e078a987-2866-47be-bedd-d9cf4EXAMPLE"
        }
    ]
}
```

Esempio 2: come elencare tutte le destinazioni di una finestra di manutenzione che corrispondono a un determinato valore delle informazioni dello specifico proprietario

Questo esempio `describe-maintenance-window-targets` elenca tutti gli obiettivi di una finestra di manutenzione con un valore specifico.

```
aws ssm describe-maintenance-window-targets \
    --window-id "mw-0ecb1226ddEXAMPLE" \
    --filters "Key=OwnerInformation,Values=CostCenter1"
```

Output:

```
{
    "Targets": [
        {
            "WindowId": "mw-0ecb1226ddEXAMPLE",
            "WindowTargetId": "da89dcc3-7f9c-481d-ba2b-edcb7d0057f9",
            "ResourceType": "INSTANCE",
            "Targets": [
```

```
                {
                    "Key": "tag:Environment",
                    "Values": [
                        "Prod"
                    ]
                }
            ],
            "OwnerInformation": "CostCenter1",
            "Name": "ProdTarget1"
        }
    ]
}
```

Per ulteriori informazioni, vedere View Information About Maintenance Windows (AWS CLI) nella AWS Systems Manager User Guide.

- Per i dettagli sull'API, vedere DescribeMaintenanceWindowTargetsin AWS CLI Command Reference.

`describe-maintenance-window-tasks`

Il seguente esempio di codice mostra come utilizzare`describe-maintenance-window-tasks`.

AWS CLI

Esempio 1: come elencare tutte le attività per una finestra di manutenzione

L'esempio seguente `describe-maintenance-window-tasks` elenca tutte le attività per la finestra di manutenzione specificata.

```
aws ssm describe-maintenance-window-tasks \
    --window-id "mw-06cf17cbefEXAMPLE"
```

Output:

```
{
    "Tasks": [
        {
            "WindowId": "mw-06cf17cbefEXAMPLE",
            "WindowTaskId": "018b31c3-2d77-4b9e-bd48-c91edEXAMPLE",
            "TaskArn": "AWS-RestartEC2Instance",
            "TaskParameters": {},
            "Type": "AUTOMATION",
```

```
            "Description": "Restarting EC2 Instance for maintenance",
            "MaxConcurrency": "1",
            "MaxErrors": "1",
            "Name": "My-Automation-Example-Task",
            "Priority": 0,
            "ServiceRoleArn": "arn:aws:iam::111222333444:role/aws-service-role/
ssm.amazonaws.com/AWSServiceRoleForAmazonSSM",
            "Targets": [
                {
                    "Key": "WindowTargetIds",
                    "Values": [
                        "da89dcc3-7f9c-481d-ba2b-edcb7EXAMPLE"
                    ]
                }
            ]
        },
        {
            "WindowId": "mw-06cf17cbefEXAMPLE",
            "WindowTaskId": "1943dee0-0a17-4978-9bf4-3cc2fEXAMPLE",
            "TaskArn": "AWS-DisableS3BucketPublicReadWrite",
            "TaskParameters": {},
            "Type": "AUTOMATION",
            "Description": "Automation task to disable read/write access on public
 S3 buckets",
            "MaxConcurrency": "10",
            "MaxErrors": "5",
            "Name": "My-Disable-S3-Public-Read-Write-Access-Automation-Task",
            "Priority": 0,
            "ServiceRoleArn": "arn:aws:iam::111222333444:role/aws-service-role/
ssm.amazonaws.com/AWSServiceRoleForAmazonSSM",
            "Targets": [
                {
                    "Key": "WindowTargetIds",
                    "Values": [
                        "da89dcc3-7f9c-481d-ba2b-edcb7EXAMPLE"
                    ]
                }
            ]
        }
    ]
}
```

Esempio 2: per elencare tutte le attività per una finestra di manutenzione che richiama il documento di comando AWS- RunPowerShellScript

L'esempio seguente `describe-maintenance-window-tasks` elenca tutte le attività per la finestra di manutenzione specificata che invoca il documento di comando `AWS-RunPowerShellScript`.

```
aws ssm describe-maintenance-window-tasks \
    --window-id "mw-ab12cd34eEXAMPLE" \
    --filters "Key=TaskArn,Values=AWS-RunPowerShellScript"
```

Output:

```
{
    "Tasks": [
        {
            "WindowId": "mw-ab12cd34eEXAMPLE",
            "WindowTaskId": "0d36e6b4-3a4f-411e-adcb-3558eEXAMPLE",
            "TaskArn": "AWS-RunPowerShellScript",
            "Type": "RUN_COMMAND",
            "Targets": [
                {
                    "Key": "WindowTargetIds",
                    "Values": [
                        "da89dcc3-7f9c-481d-ba2b-edcb7EXAMPLE"
                    ]
                }
            ],
            "TaskParameters": {},
            "Priority": 1,
            "ServiceRoleArn": "arn:aws:iam::111222333444:role/aws-service-role/
ssm.amazonaws.com/AWSServiceRoleForAmazonSSM",
            "MaxConcurrency": "1",
            "MaxErrors": "1",
            "Name": "MyTask"
        }
    ]
}
```

Esempio 3: come elencare tutte le attività per una finestra di manutenzione con una priorità di 3

L'esempio seguente `describe-maintenance-window-tasks` elenca tutte le attività per la finestra di manutenzione specificata che hanno un valore `Priority` di 3.

```
aws ssm describe-maintenance-window-tasks \
    --window-id "mw-ab12cd34eEXAMPLE" \
    --filters "Key=Priority,Values=3"
```

Output:

```
{
    "Tasks": [
        {
            "WindowId": "mw-ab12cd34eEXAMPLE",
            "WindowTaskId": "0d36e6b4-3a4f-411e-adcb-3558eEXAMPLE",
            "TaskArn": "AWS-RunPowerShellScript",
            "Type": "RUN_COMMAND",
            "Targets": [
                {
                    "Key": "WindowTargetIds",
                    "Values": [
                        "da89dcc3-7f9c-481d-ba2b-edcb7EXAMPLE"
                    ]
                }
            ],
            "TaskParameters": {},
            "Priority": 3,
            "ServiceRoleArn": "arn:aws:iam::111222333444:role/aws-service-role/
ssm.amazonaws.com/AWSServiceRoleForAmazonSSM",
            "MaxConcurrency": "1",
            "MaxErrors": "1",
            "Name": "MyRunCommandTask"
        },
        {
            "WindowId": "mw-ab12cd34eEXAMPLE",
            "WindowTaskId": "ee45feff-ad65-4a6c-b478-5cab8EXAMPLE",
            "TaskArn": "AWS-RestartEC2Instance",
            "Type": "AUTOMATION",
            "Targets": [
                {
                    "Key": "WindowTargetIds",
                    "Values": [
                        "da89dcc3-7f9c-481d-ba2b-edcb7EXAMPLE"
                    ]
```

```
                }
            ],
            "TaskParameters": {},
            "Priority": 3,
            "ServiceRoleArn": "arn:aws:iam::111222333444:role/aws-service-role/
ssm.amazonaws.com/AWSServiceRoleForAmazonSSM",
            "MaxConcurrency": "10",
            "MaxErrors": "5",
            "Name": "My-Automation-Task",
            "Description": "A description for my Automation task"
        }
    ]
}
```

Esempio 4: come elencare tutte le attività per una finestra di manutenzione con priorità pari a 1 e utilizzare Run Command

Questo esempio `describe-maintenance-window-tasks` elenca tutte le attività per la finestra di manutenzione specificata che hanno un valore `Priority` di 1 e un utilizzo di `Run Command`.

```
aws ssm describe-maintenance-window-tasks \
    --window-id "mw-ab12cd34eEXAMPLE" \
    --filters "Key=Priority,Values=1" "Key=TaskType,Values=RUN_COMMAND"
```

Output:

```
{
    "Tasks": [
        {
            "WindowId": "mw-ab12cd34eEXAMPLE",
            "WindowTaskId": "0d36e6b4-3a4f-411e-adcb-3558eEXAMPLE",
            "TaskArn": "AWS-RunPowerShellScript",
            "Type": "RUN_COMMAND",
            "Targets": [
                {
                    "Key": "WindowTargetIds",
                    "Values": [
                        "da89dcc3-7f9c-481d-ba2b-edcb7EXAMPLE"
                    ]
                }
            ],
            "TaskParameters": {},
```

```
            "Priority": 1,
            "ServiceRoleArn": "arn:aws:iam::111222333444:role/aws-service-role/
ssm.amazonaws.com/AWSServiceRoleForAmazonSSM",
            "MaxConcurrency": "1",
            "MaxErrors": "1",
            "Name": "MyRunCommandTask"
        }
    ]
}
```

Per ulteriori informazioni, vedere Visualizza informazioni sulle finestre di manutenzione (AWS CLI) nella AWS Systems Manager User Guide.

- Per i dettagli sull'API, vedere DescribeMaintenanceWindowTasksin AWS CLI Command Reference.

`describe-maintenance-windows-for-target`

Il seguente esempio di codice mostra come utilizzare`describe-maintenance-windows-for-target`.

AWS CLI

Come elencare tutte le finestre di manutenzione associate a un'istanza specifica

L'esempio `describe-maintenance-windows-for-target` seguente elenca le finestre di manutenzione con destinazioni o attività associate all'istanza specificata.

```
aws ssm describe-maintenance-windows-for-target \
    --targets Key=InstanceIds,Values=i-1234567890EXAMPLE \
    --resource-type INSTANCE
```

Output:

```
{
    "WindowIdentities": [
        {
            "WindowId": "mw-0c5ed765acEXAMPLE",
            "Name": "My-First-Maintenance-Window"
        }
    ]
```

```
}
```

Per ulteriori informazioni, vedere View Information About Maintenance Windows (AWS CLI) nella AWS Systems Manager User Guide.

- Per i dettagli sull'API, vedere DescribeMaintenanceWindowsForTargetin AWS CLI Command Reference.

`describe-maintenance-windows`

Il seguente esempio di codice mostra come utilizzare`describe-maintenance-windows`.

AWS CLI

Esempio 1: come elencare tutte le finestre di manutenzione

L'`describe-maintenance-windows`esempio seguente elenca tutte le finestre di manutenzione AWS dell'account nella regione corrente.

```
aws ssm describe-maintenance-windows
```

Output:

```
{
    "WindowIdentities": [
        {
            "WindowId": "mw-0ecb1226ddEXAMPLE",
            "Name": "MyMaintenanceWindow-1",
            "Enabled": true,
            "Duration": 2,
            "Cutoff": 1,
            "Schedule": "rate(180 minutes)",
            "NextExecutionTime": "2020-02-12T23:19:20.596Z"
        },
        {
            "WindowId": "mw-03eb9db428EXAMPLE",
            "Name": "MyMaintenanceWindow-2",
            "Enabled": true,
            "Duration": 3,
            "Cutoff": 1,
            "Schedule": "rate(7 days)",
```

```
            "NextExecutionTime": "2020-02-17T23:22:00.956Z"
        },
    ]
}
```

Esempio 2: come elencare tutte le finestre di manutenzione abilitate

L'esempio seguente `describe-maintenance-windows` elenca tutte le finestre di manutenzione abilitate.

```
aws ssm describe-maintenance-windows \
    --filters "Key=Enabled,Values=true"
```

Esempio 3: come elencare le finestre di manutenzione che corrispondono a un nome specifico

Questo esempio `describe-maintenance-windows` elenca tutte le finestre di manutenzione con il nome specificato.

```
aws ssm describe-maintenance-windows \
    --filters "Key=Name,Values=MyMaintenanceWindow"
```

Per ulteriori informazioni, vedere View Information About Maintenance Windows (AWS CLI) nella AWS Systems Manager User Guide.

- Per i dettagli sull'API, vedere DescribeMaintenanceWindowsin AWS CLI Command Reference.

describe-ops-items

Il seguente esempio di codice mostra come utilizzare`describe-ops-items`.

AWS CLI

Per elencare un set di OpsItems

L'`describe-ops-items`esempio seguente mostra un elenco di tutti i dati aperti OpsItems nel tuo AWS account.

```
aws ssm describe-ops-items \
    --ops-item-filters "Key=Status,Values=Open,Operator=Equal"
```

Output:

```
{
    "OpsItemSummaries": [
        {
            "CreatedBy": "arn:aws:sts::111222333444:assumed-role/OpsItem-CWE-Role/
fbf77cbe264a33509569f23e4EXAMPLE",
            "CreatedTime": "2020-03-14T17:02:46.375000-07:00",
            "LastModifiedBy": "arn:aws:sts::111222333444:assumed-role/OpsItem-CWE-
Role/fbf77cbe264a33509569f23e4EXAMPLE",
            "LastModifiedTime": "2020-03-14T17:02:46.375000-07:00",
            "Source": "SSM",
            "Status": "Open",
            "OpsItemId": "oi-7cfc5EXAMPLE",
            "Title": "SSM Maintenance Window execution failed",
            "OperationalData": {
                "/aws/dedup": {
                    "Value": "{\"dedupString\":\"SSMOpsItems-SSM-maintenance-window-
execution-failed\"}",
                    "Type": "SearchableString"
                },
                "/aws/resources": {
                    "Value": "[{\"arn\":\"arn:aws:ssm:us-
east-2:111222333444:maintenancewindow/mw-034093d322EXAMPLE\"}]",
                    "Type": "SearchableString"
                }
            },
            "Category": "Availability",
            "Severity": "3"
        },
        {
            "CreatedBy": "arn:aws:sts::1112223233444:assumed-role/OpsItem-CWE-Role/
fbf77cbe264a33509569f23e4EXAMPLE",
            "CreatedTime": "2020-02-26T11:43:15.426000-08:00",
            "LastModifiedBy": "arn:aws:sts::111222333444:assumed-role/OpsItem-CWE-
Role/fbf77cbe264a33509569f23e4EXAMPLE",
            "LastModifiedTime": "2020-02-26T11:43:15.426000-08:00",
            "Source": "EC2",
            "Status": "Open",
            "OpsItemId": "oi-6f966EXAMPLE",
            "Title": "EC2 instance stopped",
            "OperationalData": {
                "/aws/automations": {
```

```
                    "Value": "[ { \"automationType\": \"AWS:SSM:Automation\",
 \"automationId\": \"AWS-RestartEC2Instance\" } ]",
                    "Type": "SearchableString"
                },
                "/aws/dedup": {
                    "Value": "{\"dedupString\":\"SSMOpsItems-EC2-instance-stopped
\"}",
                    "Type": "SearchableString"
                },
                "/aws/resources": {
                    "Value": "[{\"arn\":\"arn:aws:ec2:us-
east-2:111222333444:instance/i-0beccfbc02EXAMPLE\"}]",
                    "Type": "SearchableString"
                }
            },
            "Category": "Availability",
            "Severity": "3"
        }
    ]
}
```

Per ulteriori informazioni, vedere Working with OpsItems nella AWS Systems Manager User Guide.

- Per i dettagli sull'API, vedere DescribeOpsItemsin AWS CLI Command Reference.

`describe-parameters`

Il seguente esempio di codice mostra come utilizzare`describe-parameters`.

AWS CLI

Esempio 1: come elencare tutti i parametri

L'`describe-parameters`esempio seguente elenca tutti i parametri dell' AWS account corrente e della regione.

```
aws ssm describe-parameters
```

Output:

```
{
    "Parameters": [
```

```
        {
            "Name": "MySecureStringParameter",
            "Type": "SecureString",
            "KeyId": "alias/aws/ssm",
            "LastModifiedDate": 1582155479.205,
            "LastModifiedUser": "arn:aws:sts::111222333444:assumed-role/Admin/
Richard-Roe-Managed",
            "Description": "This is a SecureString parameter",
            "Version": 2,
            "Tier": "Advanced",
            "Policies": [
                {
                    "PolicyText": "{\"Type\":\"Expiration\",\"Version\":\"1.0\",
\"Attributes\":{\"Timestamp\":\"2020-07-07T22:30:00Z\"}}",
                    "PolicyType": "Expiration",
                    "PolicyStatus": "Pending"
                },
                {
                    "PolicyText": "{\"Type\":\"ExpirationNotification\",\"Version\":
\"1.0\",\"Attributes\":{\"Before\":\"12\",\"Unit\":\"Hours\"}}",
                    "PolicyType": "ExpirationNotification",
                    "PolicyStatus": "Pending"
                }
            ]
        },
        {
            "Name": "MyStringListParameter",
            "Type": "StringList",
            "LastModifiedDate": 1582154764.222,
            "LastModifiedUser": "arn:aws:iam::111222333444:user/Mary-Major",
            "Description": "This is a StringList parameter",
            "Version": 1,
            "Tier": "Standard",
            "Policies": []
        },
        {
            "Name": "MyStringParameter",
            "Type": "String",
            "LastModifiedDate": 1582154711.976,
            "LastModifiedUser": "arn:aws:iam::111222333444:user/Alejandro-Rosalez",
            "Description": "This is a String parameter",
            "Version": 1,
            "Tier": "Standard",
            "Policies": []
```

```
        },
        {
            "Name": "latestAmi",
            "Type": "String",
            "LastModifiedDate": 1580862415.521,
            "LastModifiedUser": "arn:aws:sts::111222333444:assumed-role/lambda-ssm-
role/Automation-UpdateSSM-Param",
            "Version": 3,
            "Tier": "Standard",
            "Policies": []
        }
    ]
}
```

Esempio 2: come elencare tutti i parametri che corrispondono a metadati specifici

Questo esempio `describe-parameters` elenca tutti i parametri che corrispondono a un filtro.

aws ssm describe-parameters --filters «Key=Type, Values=» StringList

Output:

```
{
    "Parameters": [
        {
            "Name": "MyStringListParameter",
            "Type": "StringList",
            "LastModifiedDate": 1582154764.222,
            "LastModifiedUser": "arn:aws:iam::111222333444:user/Mary-Major",
            "Description": "This is a StringList parameter",
            "Version": 1,
            "Tier": "Standard",
            "Policies": []
        }
    ]
}
```

Per ulteriori informazioni, consulta Ricerca dei parametri di Systems Manager nella Guida per l'utente di AWS Systems Manager.

- Per i dettagli sull'API, consulta Command Reference. DescribeParametersAWS CLI

`describe-patch-baselines`

Il seguente esempio di codice mostra come utilizzare`describe-patch-baselines`.

AWS CLI

Esempio 1: come elencare tutte le baseline delle patch

L'esempio seguente `describe-patch-baselines` recupera i dettagli di tutte le baseline delle patch di un account nella regione corrente.

```
aws ssm describe-patch-baselines
```

Output:

```
{
    "BaselineIdentities": [
        {
            "BaselineName": "AWS-SuseDefaultPatchBaseline",
            "DefaultBaseline": true,
            "BaselineDescription": "Default Patch Baseline for Suse Provided by
 AWS.",
            "BaselineId": "arn:aws:ssm:us-east-2:733109147000:patchbaseline/
pb-0123fdb36e334a3b2",
            "OperatingSystem": "SUSE"
        },
        {
            "BaselineName": "AWS-DefaultPatchBaseline",
            "DefaultBaseline": false,
            "BaselineDescription": "Default Patch Baseline Provided by AWS.",
            "BaselineId": "arn:aws:ssm:us-east-2:733109147000:patchbaseline/
pb-020d361a05defe4ed",
            "OperatingSystem": "WINDOWS"
        },
        ...
        {
            "BaselineName": "MyWindowsPatchBaseline",
            "DefaultBaseline": true,
            "BaselineDescription": "My patch baseline for EC2 instances for Windows
 Server",
            "BaselineId": "pb-0ad00e0dd7EXAMPLE",
            "OperatingSystem": "WINDOWS"
        }
```

```
    ]
}
```

Esempio 2: Per elencare tutte le linee di base delle patch fornite da AWS

L'`describe-patch-baselines`esempio seguente elenca tutte le linee di base delle patch fornite da. AWS

```
aws ssm describe-patch-baselines \
    --filters "Key=OWNER,Values=[AWS]"
```

Esempio 3: come elencare tutte le baseline delle patch possedute

L'esempio seguente `describe-patch-baselines` elenca tutte le baseline delle patch personalizzate create nell'account nella regione corrente.

```
aws ssm describe-patch-baselines \
    --filters "Key=OWNER,Values=[Self]"
```

Per ulteriori informazioni, consulta Informazioni sulle baseline delle patch predefinite e personalizzate nella Guida per l'utente di AWS Systems Manager.

- Per i dettagli sull'API, vedere DescribePatchBaselinesin AWS CLI Command Reference.

`describe-patch-group-state`

Il seguente esempio di codice mostra come utilizzare`describe-patch-group-state`.

AWS CLI

Come ottenere lo stato di un gruppo di patch

L'esempio seguente `describe-patch-group-state` recupera il riepilogo di alto livello sulla conformità delle patch per un gruppo di patch.

```
aws ssm describe-patch-group-state \
    --patch-group "Production"
```

Output:

```
{
    "Instances": 21,
```

```
    "InstancesWithCriticalNonCompliantPatches": 1,
    "InstancesWithFailedPatches": 2,
    "InstancesWithInstalledOtherPatches": 3,
    "InstancesWithInstalledPatches": 21,
    "InstancesWithInstalledPendingRebootPatches": 2,
    "InstancesWithInstalledRejectedPatches": 1,
    "InstancesWithMissingPatches": 3,
    "InstancesWithNotApplicablePatches": 4,
    "InstancesWithOtherNonCompliantPatches": 1,
    "InstancesWithSecurityNonCompliantPatches": 1,
    "InstancesWithUnreportedNotApplicablePatches": 2
}
```

Per ulteriori informazioni, vedere Informazioni sui gruppi di patch < https://docs.aws.amazon.com/systems-manager/ latest/userguide/sysman -patch-patchgroups.html>__ e Informazioni sui valori dello stato di conformità delle patch nella Systems Manager User Guide.AWS

- Per i AWS CLI dettagli DescribePatchGroupStatesull'API, vedere in Command Reference.

`describe-patch-groups`

Il seguente esempio di codice mostra come utilizzare`describe-patch-groups`.

AWS CLI

Come visualizzare le registrazioni dei gruppi di patch

L'esempio seguente `describe-patch-groups` elenca le registrazioni dei gruppi di patch.

```
aws ssm describe-patch-groups
```

Output:

```
{
    "Mappings": [
        {
            "PatchGroup": "Production",
            "BaselineIdentity": {
                "BaselineId": "pb-0123456789abcdef0",
                "BaselineName": "ProdPatching",
                "OperatingSystem": "WINDOWS",
                "BaselineDescription": "Patches for Production",
                "DefaultBaseline": false
```

```
            }
        },
        {
            "PatchGroup": "Development",
            "BaselineIdentity": {
                "BaselineId": "pb-0713accee01234567",
                "BaselineName": "DevPatching",
                "OperatingSystem": "WINDOWS",
                "BaselineDescription": "Patches for Development",
                "DefaultBaseline": true
            }
        },
        ...
    ]
}
```

Per ulteriori informazioni, vedere Create a Patch Group < https://docs.aws.amazon.com/systems-manager/ latest/userguide/sysman - patch-group-tagging .html>__ e Aggiungere un gruppo di patch a una Patch Baseline nella Systems Manager AWS User Guide.

- Per i dettagli sull'API, vedere in Command Reference. DescribePatchGroupsAWS CLI

`describe-patch-properties`

Il seguente esempio di codice mostra come utilizzare`describe-patch-properties`.

AWS CLI

Come elencare la disponibilità delle patch di Amazon Linux

L'`describe-patch-properties`esempio seguente mostra un elenco dei prodotti Amazon Linux per i quali le patch sono disponibili nel tuo AWS account.

```
aws ssm describe-patch-properties \
    --operating-system AMAZON_LINUX \
    --property PRODUCT
```

Output:

```
{
    "Properties": [
        {
            "Name": "AmazonLinux2012.03"
```

```
        },
        {
            "Name": "AmazonLinux2012.09"
        },
        {
            "Name": "AmazonLinux2013.03"
        },
        {
            "Name": "AmazonLinux2013.09"
        },
        {
            "Name": "AmazonLinux2014.03"
        },
        {
            "Name": "AmazonLinux2014.09"
        },
        {
            "Name": "AmazonLinux2015.03"
        },
        {
            "Name": "AmazonLinux2015.09"
        },
        {
            "Name": "AmazonLinux2016.03"
        },
        {
            "Name": "AmazonLinux2016.09"
        },
        {
            "Name": "AmazonLinux2017.03"
        },
        {
            "Name": "AmazonLinux2017.09"
        },
        {
            "Name": "AmazonLinux2018.03"
        }
    ]
}
```

Per ulteriori informazioni, consulta informazioni baseline delle patch nella Guida per l'utente di AWS Systems Manager.

- Per i dettagli sull'API, consulta DescribePatchProperties AWS CLICommand Reference.

`describe-sessions`

Il seguente esempio di codice mostra come utilizzare`describe-sessions`.

AWS CLI

Esempio 1: come avviare una sessione di Session Manager

L'esempio `describe-sessions` seguente recupera un elenco delle sessioni attive create più di recente (sia connesse che disconnesse) nel corso degli ultimi 30 giorni e avviate dall'utente specificato. Questo comando restituisce solo i risultati per le connessioni alle destinazioni avviate utilizzando Session Manager. Non elenca le connessioni effettuate tramite altri mezzi, ad esempio Remote Desktop Connections o SSH.

```
aws ssm describe-sessions \
    --state "Active" \
    --filters "key=Owner,value=arn:aws:sts::123456789012:assumed-role/Administrator/
Shirley-Rodriguez"
```

Output:

```
{
    "Sessions": [
        {
            "SessionId": "John-07a16060613c408b5",
            "Target": "i-1234567890abcdef0",
            "Status": "Connected",
            "StartDate": 1550676938.352,
            "Owner": "arn:aws:sts::123456789012:assumed-role/Administrator/Shirley-
Rodriguez",
            "OutputUrl": {}
        },
        {
            "SessionId": "John-01edf534b8b56e8eb",
            "Target": "i-9876543210abcdef0",
            "Status": "Connected",
            "StartDate": 1550676842.194,
            "Owner": "arn:aws:sts::123456789012:assumed-role/Administrator/Shirley-
Rodriguez",
            "OutputUrl": {}
        }
    ]
```

```
}
```

Esempio 2: come avviare una sessione di Session Manager

L'esempio `describe-sessions` seguente recupera un elenco delle sessioni terminate più di recente negli ultimi 30 giorni per tutti gli utenti.

```
aws ssm describe-sessions \
    --state "History"
```

Output:

```
{
    "Sessions": [
        {
            "SessionId": "Mary-Major-0022b1eb2b0d9e3bd",
            "Target": "i-1234567890abcdef0",
            "Status": "Terminated",
            "StartDate": 1550520701.256,
            "EndDate": 1550521931.563,
            "Owner": "arn:aws:sts::123456789012:assumed-role/Administrator/Mary-
Major"
        },
        {
            "SessionId": "Jane-Roe-0db53f487931ed9d4",
            "Target": "i-9876543210abcdef0",
            "Status": "Terminated",
            "StartDate": 1550161369.149,
            "EndDate": 1550162580.329,
            "Owner": "arn:aws:sts::123456789012:assumed-role/Administrator/Jane-Roe"
        },
        ...
    ],
    "NextToken": "--token string truncated--"
}
```

Per ulteriori informazioni, consulta Visualizzazione della cronologia delle sessioni nella Guida per l'utente di AWS Systems Manager.

- Per i dettagli sull'API, consulta DescribeSessions AWS CLICommand Reference.

`disassociate-ops-item-related-item`

Il seguente esempio di codice mostra come utilizzare`disassociate-ops-item-related-item`.

AWS CLI

Come eliminare un'associazione di elementi correlati

L'`disassociate-ops-item-related-item`esempio seguente elimina l'associazione tra OpsItem e un elemento correlato.

```
aws ssm disassociate-ops-item-related-item \
    --ops-item-id "oi-f99f2EXAMPLE" \
    --association-id "e2036148-cccb-490e-ac2a-390e5EXAMPLE"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, vedere Utilizzo degli incidenti di Incident Manager OpsCenter nella Guida per l'utente di AWS Systems Manager.

- Per i dettagli sull'API, vedere DisassociateOpsItemRelatedItemin AWS CLI Command Reference.

`get-automation-execution`

Il seguente esempio di codice mostra come utilizzare`get-automation-execution`.

AWS CLI

Come visualizzare i dettagli sull'esecuzione di un'automazione

L'esempio seguente `get-automation-execution` visualizza informazioni dettagliate sull'esecuzione di un'automazione.

```
aws ssm get-automation-execution \
    --automation-execution-id 73c8eef8-f4ee-4a05-820c-e354fEXAMPLE
```

Output:

```
{
    "AutomationExecution": {
        "AutomationExecutionId": "73c8eef8-f4ee-4a05-820c-e354fEXAMPLE",
        "DocumentName": "AWS-StartEC2Instance",
```

```
        "DocumentVersion": "1",
        "ExecutionStartTime": 1583737233.748,
        "ExecutionEndTime": 1583737234.719,
        "AutomationExecutionStatus": "Success",
        "StepExecutions": [
            {
                "StepName": "startInstances",
                "Action": "aws:changeInstanceState",
                "ExecutionStartTime": 1583737234.134,
                "ExecutionEndTime": 1583737234.672,
                "StepStatus": "Success",
                "Inputs": {
                    "DesiredState": "\"running\"",
                    "InstanceIds": "[\"i-0cb99161f6EXAMPLE\"]"
                },
                "Outputs": {
                    "InstanceStates": [
                        "running"
                    ]
                },
                "StepExecutionId": "95e70479-cf20-4d80-8018-7e4e2EXAMPLE",
                "OverriddenParameters": {}
            }
        ],
        "StepExecutionsTruncated": false,
        "Parameters": {
            "AutomationAssumeRole": [
                ""
            ],
            "InstanceId": [
                "i-0cb99161f6EXAMPLE"
            ]
        },
        "Outputs": {},
        "Mode": "Auto",
        "ExecutedBy": "arn:aws:sts::29884EXAMPLE:assumed-role/mw_service_role/
OrchestrationService",
        "Targets": [],
        "ResolvedTargets": {
            "ParameterValues": [],
            "Truncated": false
        }
    }
```

```
}
```

Per ulteriori informazioni, vedere Procedura dettagliata: applicare una patch a un'AMI Linux (AWS CLI) nella AWS Systems Manager User Guide.

- Per i dettagli sull'API, vedere GetAutomationExecutionin AWS CLI Command Reference.

`get-calendar-state`

Il seguente esempio di codice mostra come utilizzare`get-calendar-state`.

AWS CLI

Esempio 1: come ottenere lo stato di un calendario delle modifiche

L'esempio `get-calendar-state` seguente restituisce lo stato di un calendario all'ora corrente. Poiché nell'esempio non è specificata un'ora, viene riportato lo stato corrente del calendario.

```
aws ssm get-calendar-state \
    --calendar-names "MyCalendar"
```

Output:

```
{
    "State": "OPEN",
    "AtTime": "2020-02-19T22:28:51Z",
    "NextTransitionTime": "2020-02-24T21:15:19Z"
}
```

Esempio 2: come ottenere lo stato di un calendario delle modifiche a un'ora specificata

L'esempio `get-calendar-state` seguente restituisce lo stato di un calendario all'ora specificata.

```
aws ssm get-calendar-state \
    --calendar-names "MyCalendar" \
    --at-time "2020-07-19T21:15:19Z"
```

Output:

```
{
```

```
    "State": "CLOSED",
    "AtTime": "2020-07-19T21:15:19Z"
}
```

Per ulteriori informazioni, consulta Ottenere lo stato di un calendario delle modifiche nella Guida per l'utente di AWS Systems Manager.

- Per i dettagli sull'API, consulta GetCalendarState AWS CLICommand Reference.

`get-command-invocation`

Il seguente esempio di codice mostra come utilizzare`get-command-invocation`.

AWS CLI

Come visualizzare i dettagli di una invocazione di comando

L'esempio seguente `get-command-invocation` elenca tutte le invocazioni del comando specificato sull'istanza specificata.

```
aws ssm get-command-invocation \
    --command-id "ef7fdfd8-9b57-4151-a15c-db9a12345678" \
    --instance-id "i-1234567890abcdef0"
```

Output:

```
{
    "CommandId": "ef7fdfd8-9b57-4151-a15c-db9a12345678",
    "InstanceId": "i-1234567890abcdef0",
    "Comment": "b48291dd-ba76-43e0-b9df-13e11ddaac26:6960febb-2907-4b59-8e1a-
d6ce8EXAMPLE",
    "DocumentName": "AWS-UpdateSSMAgent",
    "DocumentVersion": "",
    "PluginName": "aws:updateSsmAgent",
    "ResponseCode": 0,
    "ExecutionStartDateTime": "2020-02-19T18:18:03.419Z",
    "ExecutionElapsedTime": "PT0.091S",
    "ExecutionEndDateTime": "2020-02-19T18:18:03.419Z",
    "Status": "Success",
    "StatusDetails": "Success",
    "StandardOutputContent": "Updating amazon-ssm-agent from 2.3.842.0 to latest
\nSuccessfully downloaded https://s3.us-east-2.amazonaws.com/amazon-ssm-us-east-2/
```

```
ssm-agent-manifest.json\namazon-ssm-agent 2.3.842.0 has already been installed,
 update skipped\n",
    "StandardOutputUrl": "",
    "StandardErrorContent": "",
    "StandardErrorUrl": "",
    "CloudWatchOutputConfig": {
        "CloudWatchLogGroupName": "",
        "CloudWatchOutputEnabled": false
    }
}
```

Per ulteriori informazioni, consulta Informazioni sugli stati dei comandi nella Guida per l'utente di AWS Systems Manager.

- Per i dettagli sull'API, consulta GetCommandInvocation AWS CLICommand Reference.

`get-connection-status`

Il seguente esempio di codice mostra come utilizzare`get-connection-status`.

AWS CLI

Come visualizzare lo stato della connessione di un'istanza gestita

Questo esempio `get-connection-status` restituisce lo stato della connessione dell'istanza gestita specificata.

```
aws ssm get-connection-status \
    --target i-1234567890abcdef0
```

Output:

```
{
    "Target": "i-1234567890abcdef0",
    "Status": "connected"
}
```

- Per i dettagli sull'API, consulta GetConnectionStatus AWS CLICommand Reference.

`get-default-patch-baseline`

Il seguente esempio di codice mostra come utilizzare`get-default-patch-baseline`.

AWS CLI

Esempio 1: come visualizzare la baseline delle patch di Windows predefinita

L'esempio seguente `get-default-patch-baseline` recupera i dettagli per la baseline delle patch predefinita per Windows Server.

```
aws ssm get-default-patch-baseline
```

Output:

```
{
  "BaselineId": "pb-0713accee01612345",
  "OperatingSystem": "WINDOWS"
}
```

Esempio 2: come visualizzare la baseline delle patch predefinita per Amazon Linux

L'esempio seguente `get-default-patch-baseline` recupera i dettagli per la baseline delle patch predefinita per Amazon Linux.

```
aws ssm get-default-patch-baseline \
    --operating-system AMAZON_LINUX
```

Output:

```
{
    "BaselineId": "pb-047c6eb9c8fc12345",
    "OperatingSystem": "AMAZON_LINUX"
}
```

Per ulteriori informazioni, vedere Informazioni sulle patch di base predefinite e personalizzate < https://docs.aws.amazon.com/systems-manager/ latest/userguide/sysman -patch-baselines.html>__ e Impostazione di una baseline di patch esistente come predefinita nella Systems Manager User Guide.AWS

- Per i AWS CLI dettagli GetDefaultPatchBaselinesull'API, vedere in Command Reference.

`get-deployable-patch-snapshot-for-instance`

Il seguente esempio di codice mostra come utilizzare`get-deployable-patch-snapshot-for-instance`.

AWS CLI

Come recuperare l'istantanea corrente per la baseline delle patch, un'istanza utilizza

L'esempio seguente `get-deployable-patch-snapshot-for-instance` recupera i dettagli dello snapshot corrente per la baseline delle patch specificata utilizzata da un'istanza. Questo comando deve essere eseguito dall'istanza utilizzando le credenziali dell'istanza. Per assicurarti che utilizzi le credenziali dell'istanza, esegui `aws configure` e specifica solo la Regione dell'istanza. Lascia i campi `Access Key` e `Secret Key` vuoti.

Suggerimento: Utilizza `uuidgen` per generare un `snapshot-id`.

```
aws ssm get-deployable-patch-snapshot-for-instance \
    --instance-id "i-1234567890abcdef0" \
    --snapshot-id "521c3536-930c-4aa9-950e-01234567abcd"
```

Output:

```
{
    "InstanceId": "i-1234567890abcdef0",
    "SnapshotId": "521c3536-930c-4aa9-950e-01234567abcd",
    "Product": "AmazonLinux2018.03",
    "SnapshotDownloadUrl": "https://patch-baseline-snapshot-us-
east-1.s3.amazonaws.com/
ed85194ef27214f5984f28b4d664d14f7313568fea7d4b6ac6c10ad1f729d7e7-773304212436/
AMAZON_LINUX-521c3536-930c-4aa9-950e-01234567abcd?X-Amz-Algorithm=AWS4-HMAC-
SHA256&X-Amz-Date=20190215T164031Z&X-Amz-SignedHeaders=host&X-Amz-Expires=86400&X-
Amz-Credential=AKIAJ5C56P35AEBRX2QQ%2F20190215%2Fus-east-1%2Fs3%2Faws4_request&X-
Amz-Signature=efaaaf6e3878e77f48a6697e015efdbda9c426b09c5822055075c062f6ad2149"
}
```

Per ulteriori informazioni, consulta Nome parametro: Snapshot ID nella Guida per l'utente di AWS Systems Manager.

- Per i dettagli sull'API, consulta GetDeployablePatchSnapshotForInstance AWS CLICommand Reference.

`get-document`

Il seguente esempio di codice mostra come utilizzare`get-document`.

AWS CLI

Come ottenere il contenuto del documento

L'esempio seguente `get-document` visualizza il contenuto di un documento Systems Manager.

```
aws ssm get-document \
    --name "AWS-RunShellScript"
```

Output:

```
{
    "Name": "AWS-RunShellScript",
    "DocumentVersion": "1",
    "Status": "Active",
    "Content": "{\n    \"schemaVersion\":\"1.2\",\n    \"description\":\"Run a
 shell script or specify the commands to run.\",\n    \"parameters\":{\n
 \"commands\":{\n            \"type\":\"StringList\",\n            \"description
\":\"(Required) Specify a shell script or a command to run.\",\n
 \"minItems\":1,\n            \"displayType\":\"textarea\"\n        },\n
 \"workingDirectory\":{\n            \"type\":\"String\",\n            \"default
\":\"\",\n            \"description\":\"(Optional) The path to the working
 directory on your instance.\",\n            \"maxChars\":4096\n        },\n
   \"executionTimeout\":{\n            \"type\":\"String\",\n            \"default
\":\"3600\",\n            \"description\":\"(Optional) The time in seconds for a
 command to complete before it is considered to have failed. Default is 3600 (1
 hour). Maximum is 172800 (48 hours).\",\n            \"allowedPattern\":\"([1-9]
[0-9]{0,4})|(1[0-6][0-9]{4})|(17[0-1][0-9]{3})|(172[0-7][0-9]{2})|(172800)\"\n
        }\n    },\n    \"runtimeConfig\":{\n        \"aws:runShellScript\":{\n
            \"properties\":[\n                {\n                    \"id\":
\"0.aws:runShellScript\",\n                    \"runCommand\":\"{{ commands }}\",\n
                    \"workingDirectory\":\"{{ workingDirectory }}\",\n
       \"timeoutSeconds\":\"{{ executionTimeout }}\"\n                }\n
  ]\n        }\n    }\n}\n",
    "DocumentType": "Command",
    "DocumentFormat": "JSON"
}
```

Per ulteriori informazioni, consulta Documenti di AWS Systems Manager nella Guida per l'utente di AWS Systems Manager.

- Per i dettagli sull'API, consulta GetDocument AWS CLICommand Reference.

`get-inventory-schema`

Il seguente esempio di codice mostra come utilizzare`get-inventory-schema`.

AWS CLI

Come visualizzare lo schema dell'inventario

Questo esempio restituisce un elenco di nomi di tipi di inventario per l'account.

Comando:

```
aws ssm get-inventory-schema
```

Output:

```
{
  "Schemas": [
      {
          "TypeName": "AWS:AWSComponent",
          "Version": "1.0",
          "Attributes": [
              {
                  "Name": "Name",
                  "DataType": "STRING"
              },
              {
                  "Name": "ApplicationType",
                  "DataType": "STRING"
              },
              {
                  "Name": "Publisher",
                  "DataType": "STRING"
              },
              {
                  "Name": "Version",
                  "DataType": "STRING"
```

```
                },
                {
                    "Name": "InstalledTime",
                    "DataType": "STRING"
                },
                {
                    "Name": "Architecture",
                    "DataType": "STRING"
                },
                {
                    "Name": "URL",
                    "DataType": "STRING"
                }
            ]
        },
        ...
    ],
    "NextToken": "--token string truncated--"
}
```

Come visualizzare lo schema di inventario per un tipo di inventario specifico

Questo esempio restituisce lo schema di inventario per un tipo di inventario di AWS:AWS Component.

Comando:

```
aws ssm get-inventory-schema --type-name "AWS:AWSComponent"
```

- Per i dettagli sull'API, consulta GetInventorySchema AWS CLICommand Reference.

`get-inventory`

Il seguente esempio di codice mostra come utilizzare`get-inventory`.

AWS CLI

Come visualizzare l'inventario

Questo esempio ottiene i metadati personalizzati per l'inventario.

Comando:

```
aws ssm get-inventory
```

Output:

```
{
  "Entities": [
      {
          "Data": {
              "AWS:InstanceInformation": {
                  "Content": [
                      {
                          "ComputerName": "ip-172-31-44-222.us-
west-2.compute.internal",
                          "InstanceId": "i-0cb2b964d3e14fd9f",
                          "IpAddress": "172.31.44.222",
                          "AgentType": "amazon-ssm-agent",
                          "ResourceType": "EC2Instance",
                          "AgentVersion": "2.0.672.0",
                          "PlatformVersion": "2016.09",
                          "PlatformName": "Amazon Linux AMI",
                          "PlatformType": "Linux"
                      }
                  ],
                  "TypeName": "AWS:InstanceInformation",
                  "SchemaVersion": "1.0",
                  "CaptureTime": "2017-02-20T18:03:58Z"
              }
          },
          "Id": "i-0cb2b964d3e14fd9f"
      }
  ]
}
```

- Per i dettagli sull'API, consulta GetInventory AWS CLICommand Reference.

`get-maintenance-window-execution-task-invocation`

Il seguente esempio di codice mostra come utilizzare`get-maintenance-window-execution-task-invocation`.

AWS CLI

Come ottenere informazioni sull'invocazione di un'attività in una finestra di manutenzione

L'esempio `get-maintenance-window-execution-task-invocation` seguente elenca le informazioni sull'invocazione di un'attività specificata che fa parte dell'esecuzione della finestra di manutenzione specificata.

```
aws ssm get-maintenance-window-execution-task-invocation \
    --window-execution-id "bc494bfa-e63b-49f6-8ad1-aa9f2EXAMPLE" \
    --task-id "96f2ad59-97e3-461d-a63d-40c8aEXAMPLE" \
    --invocation-id "a5273e2c-d2c6-4880-b3e1-5e550EXAMPLE"
```

Output:

```
{
    "Status": "SUCCESS",
    "Parameters": "{\"comment\":\"\",\"documentName\":\"AWS-RunPowerShellScript\",
\"instanceIds\":[\"i-1234567890EXAMPLE\"],\"maxConcurrency\":\"1\",\"maxErrors\":
\"1\",\"parameters\":{\"executionTimeout\":[\"3600\"],\"workingDirectory\":[\"\"],
\"commands\":[\"echo Hello\"]},\"timeoutSeconds\":600}",
    "ExecutionId": "03b6baa0-5460-4e15-83f2-ea685EXAMPLE",
    "InvocationId": "a5273e2c-d2c6-4880-b3e1-5e550EXAMPLE",
    "StartTime": 1549998326.421,
    "TaskType": "RUN_COMMAND",
    "EndTime": 1550001931.784,
    "WindowExecutionId": "bc494bfa-e63b-49f6-8ad1-aa9f2EXAMPLE",
    "StatusDetails": "Failed",
    "TaskExecutionId": "96f2ad59-97e3-461d-a63d-40c8aEXAMPLE"
}
```

Per ulteriori informazioni, vedere Visualizza informazioni sulle attività e sulle esecuzioni delle attività (AWS CLI) nella Guida per l'utente di Systems AWS Manager.

- Per i dettagli sull'API, vedere GetMaintenanceWindowExecutionTaskInvocationin AWS CLI Command Reference.

`get-maintenance-window-execution-task`

Il seguente esempio di codice mostra come utilizzare`get-maintenance-window-execution-task`.

AWS CLI

Come ottenere informazioni sull'esecuzione di un'attività in una finestra di manutenzione

L'esempio seguente `get-maintenance-window-execution-task` elenca le informazioni su un'attività che fa parte dell'esecuzione della finestra di manutenzione specificata.

```
aws ssm get-maintenance-window-execution-task \
    --window-execution-id "518d5565-5969-4cca-8f0e-da3b2EXAMPLE" \
    --task-id "ac0c6ae1-daa3-4a89-832e-d3845EXAMPLE"
```

Output:

```
{
    "WindowExecutionId": "518d5565-5969-4cca-8f0e-da3b2EXAMPLE",
    "TaskExecutionId": "ac0c6ae1-daa3-4a89-832e-d3845EXAMPLE",
    "TaskArn": "AWS-RunPatchBaseline",
    "ServiceRole": "arn:aws:iam::111222333444:role/aws-service-role/
ssm.amazonaws.com/AWSServiceRoleForAmazonSSM",
    "Type": "RUN_COMMAND",
    "TaskParameters": [
        {
            "BaselineOverride": {
                "Values": [
                    ""
                ]
            },
            "InstallOverrideList": {
                "Values": [
                    ""
                ]
            },
            "Operation": {
                "Values": [
                    "Scan"
                ]
            },
            "RebootOption": {
                "Values": [
                    "RebootIfNeeded"
                ]
            },
            "SnapshotId": {
```

```
                "Values": [
                    "{{ aws:ORCHESTRATION_ID }}"
                ]
            },
            "aws:InstanceId": {
                "Values": [
                    "i-02573cafcfEXAMPLE",
                    "i-0471e04240EXAMPLE",
                    "i-07782c72faEXAMPLE"
                ]
            }
        }
    ],
    "Priority": 1,
    "MaxConcurrency": "1",
    "MaxErrors": "3",
    "Status": "SUCCESS",
    "StartTime": "2021-08-04T11:45:35.088000-07:00",
    "EndTime": "2021-08-04T11:53:09.079000-07:00"
}
```

Per ulteriori informazioni, vedere Visualizza informazioni sulle attività e sulle esecuzioni delle attività (AWS CLI) nella Systems AWS Manager User Guide.

- Per i dettagli sull'API, vedere GetMaintenanceWindowExecutionTaskin AWS CLI Command Reference.

`get-maintenance-window-execution`

Il seguente esempio di codice mostra come utilizzare`get-maintenance-window-execution`.

AWS CLI

Come ottenere informazioni sull'esecuzione di un'attività in una finestra di manutenzione

L'esempio seguente `get-maintenance-window-execution` elenca le informazioni su un'attività eseguita come parte dell'esecuzione della finestra di manutenzione specificata.

```
aws ssm get-maintenance-window-execution \
    --window-execution-id "518d5565-5969-4cca-8f0e-da3b2EXAMPLE"
```

Output:

```
{
    "Status": "SUCCESS",
    "TaskIds": [
        "ac0c6ae1-daa3-4a89-832e-d3845EXAMPLE"
    ],
    "StartTime": 1487692834.595,
    "EndTime": 1487692835.051,
    "WindowExecutionId": "518d5565-5969-4cca-8f0e-da3b2EXAMPLE",
}
```

Per ulteriori informazioni, vedere Visualizza informazioni sulle attività e sulle esecuzioni delle attività (AWS CLI) nella Guida per l'utente di Systems AWS Manager.

- Per i dettagli sull'API, vedere GetMaintenanceWindowExecutionin AWS CLI Command Reference.

`get-maintenance-window-task`

Il seguente esempio di codice mostra come utilizzare`get-maintenance-window-task`.

AWS CLI

Come ottenere informazioni su un'attività in una finestra di manutenzione

L'esempio `get-maintenance-window-task` seguente recupera i dettagli sull'attività della finestra di manutenzione specificata.

```
aws ssm get-maintenance-window-task \
    --window-id mw-0c5ed765acEXAMPLE \
    --window-task-id 0e842a8d-2d44-4886-bb62-af8dcEXAMPLE
```

Output:

```
{
    "ServiceRoleArn": "arn:aws:iam::111222333444:role/aws-service-role/
ssm.amazonaws.com/AWSServiceRoleForAmazonSSM",
    "MaxErrors": "1",
    "TaskArn": "AWS-RunPowerShellScript",
    "MaxConcurrency": "1",
    "WindowTaskId": "0e842a8d-2d44-4886-bb62-af8dcEXAMPLE",
    "TaskParameters": {},
```

```
    "Priority": 1,
    "TaskInvocationParameters": {
        "RunCommand": {
            "Comment": "",
            "TimeoutSeconds": 600,
            "Parameters": {
                "commands": [
                    "echo Hello"
                ],
                "executionTimeout": [
                    "3600"
                ],
                "workingDirectory": [
                    ""
                ]
            }
        }
    },
    "WindowId": "mw-0c5ed765acEXAMPLE",
    "TaskType": "RUN_COMMAND",
    "Targets": [
        {
            "Values": [
                "84c818da-b619-4d3d-9651-946f3EXAMPLE"
            ],
            "Key": "WindowTargetIds"
        }
    ],
    "Name": "ExampleTask"
}
```

Per ulteriori informazioni, vedere View Information About Maintenance Windows (AWS CLI) nella AWS Systems Manager User Guide.

- Per i dettagli sull'API, vedere GetMaintenanceWindowTaskin AWS CLI Command Reference.

`get-maintenance-window`

Il seguente esempio di codice mostra come utilizzare`get-maintenance-window`.

AWS CLI

Come ottenere le informazioni su una finestra di manutenzione

L'esempio seguente `get-maintenance-window` recupera i dettagli sulla finestra di manutenzione specificata.

```
aws ssm get-maintenance-window \
    --window-id "mw-03eb9db428EXAMPLE"
```

Output:

```
{
    "AllowUnassociatedTargets": true,
    "CreatedDate": 1515006912.957,
    "Cutoff": 1,
    "Duration": 6,
    "Enabled": true,
    "ModifiedDate": 2020-01-01T10:04:04.099Z,
    "Name": "My-Maintenance-Window",
    "Schedule": "rate(3 days)",
    "WindowId": "mw-03eb9db428EXAMPLE",
    "NextExecutionTime": "2020-02-25T00:08:15.099Z"
}
```

Per ulteriori informazioni, vedere Visualizza informazioni sulle finestre di manutenzione (AWS CLI) nella AWS Systems Manager User Guide.

- Per i dettagli sull'API, vedere GetMaintenanceWindowin AWS CLI Command Reference.

`get-ops-item`

Il seguente esempio di codice mostra come utilizzare`get-ops-item`.

AWS CLI

Per visualizzare informazioni su un OpsItem

L'`get-ops-item`esempio seguente visualizza i dettagli relativi a quanto specificato OpsItem.

```
aws ssm get-ops-item \
    --ops-item-id oi-0b725EXAMPLE
```

Output:

```
{
```

```
    "OpsItem": {
        "CreatedBy": "arn:aws:sts::111222333444:assumed-role/OpsItem-CWE-Role/
fbf77cbe264a33509569f23e4EXAMPLE",
        "CreatedTime": "2019-12-04T15:52:16.793000-08:00",
        "Description": "CloudWatch Event Rule SSMOpsItems-EC2-instance-terminated
 was triggered. Your EC2 instance has terminated. See below for more details.",
        "LastModifiedBy": "arn:aws:sts::111222333444:assumed-role/OpsItem-CWE-Role/
fbf77cbe264a33509569f23e4EXAMPLE",
        "LastModifiedTime": "2019-12-04T15:52:16.793000-08:00",
        "Notifications": [],
        "RelatedOpsItems": [],
        "Status": "Open",
        "OpsItemId": "oi-0b725EXAMPLE",
        "Title": "EC2 instance terminated",
        "Source": "EC2",
        "OperationalData": {
            "/aws/automations": {
                "Value": "[ { \"automationType\": \"AWS:SSM:Automation\",
 \"automationId\": \"AWS-CreateManagedWindowsInstance\" }, { \"automationType\":
 \"AWS:SSM:Automation\", \"automationId\": \"AWS-CreateManagedLinuxInstance\" } ]",
                "Type": "SearchableString"
            },
            "/aws/dedup": {
                "Value": "{\"dedupString\":\"SSMOpsItems-EC2-instance-terminated
\"}",
                "Type": "SearchableString"
            },
            "/aws/resources": {
                "Value": "[{\"arn\":\"arn:aws:ec2:us-east-2:111222333444:instance/
i-05adec7e97EXAMPLE\"}]",
                "Type": "SearchableString"
            },
            "event-time": {
                "Value": "2019-12-04T23:52:16Z",
                "Type": "String"
            },
            "instance-state": {
                "Value": "terminated",
                "Type": "String"
            }
        },
        "Category": "Availability",
        "Severity": "4"
    }
```

```
}
```

Per ulteriori informazioni, vedere Working with OpsItems nella AWS Systems Manager User Guide.

- Per i dettagli sull'API, vedere GetOpsItemin AWS CLI Command Reference.

`get-ops-summary`

Il seguente esempio di codice mostra come utilizzare`get-ops-summary`.

AWS CLI

Per visualizzare un riepilogo di tutti OpsItems

L'`get-ops-summary`esempio seguente mostra un riepilogo di tutti i dati OpsItems presenti nel tuo AWS account.

```
aws ssm get-ops-summary
```

Output:

```
{
    "Entities": [
        {
            "Id": "oi-4309fEXAMPLE",
            "Data": {
                "AWS:OpsItem": {
                    "CaptureTime": "2020-02-26T18:58:32.918Z",
                    "Content": [
                        {
                            "AccountId": "111222333444",
                            "Category": "Availability",
                            "CreatedBy": "arn:aws:sts::111222333444:assumed-role/
OpsItem-CWE-Role/fbf77cbe264a33509569f23e4EXAMPLE",
                            "CreatedTime": "2020-02-26T19:10:44.149Z",
                            "Description": "CloudWatch Event Rule SSMOpsItems-EC2-
instance-terminated was triggered. Your EC2 instance has terminated. See below for
 more details.",
                            "LastModifiedBy": "arn:aws:sts::111222333444:assumed-
role/OpsItem-CWE-Role/fbf77cbe264a33509569f23e4EXAMPLE",
                            "LastModifiedTime": "2020-02-26T19:10:44.149Z",
                            "Notifications": "",
```

```
                            "OperationalData": "{\"/aws/automations\":
{\"type\":\"SearchableString\",\"value\":\"[ { \\\"automationType\\\": \\
\"AWS:SSM:Automation\\\", \\\"automationId\\\": \\\"AWS-CreateManagedWindowsInstance
\\\" }, { \\\"automationType\\\": \\\"AWS:SSM:Automation\\\", \\\"automationId
\\\": \\\"AWS-CreateManagedLinuxInstance\\\" } ]\"},\"/aws/resources\":
{\"type\":\"SearchableString\",\"value\":\"[{\\\"arn\\\":\\\"arn:aws:ec2:us-
east-2:111222333444:instance/i-0acbd0800fEXAMPLE\\\"}]\"},\"/aws/dedup\":{\"type\":
\"SearchableString\",\"value\":\"{\\\"dedupString\\\":\\\"SSMOpsItems-EC2-instance-
terminated\\\"}\"}}",
                            "OpsItemId": "oi-4309fEXAMPLE",
                            "RelatedItems": "",
                            "Severity": "3",
                            "Source": "EC2",
                            "Status": "Open",
                            "Title": "EC2 instance terminated"
                        }
                    ]
                }
            }
        },
        {
            "Id": "oi-bb2a0e6a4541",
            "Data": {
                "AWS:OpsItem": {
                    "CaptureTime": "2019-11-26T19:20:06.161Z",
                    "Content": [
                        {
                            "AccountId": "111222333444",
                            "Category": "Availability",
                            "CreatedBy": "arn:aws:sts::111222333444:assumed-role/
OpsItem-CWE-Role/fbf77cbe264a33509569f23e4EXAMPLE",
                            "CreatedTime": "2019-11-26T20:00:07.237Z",
                            "Description": "CloudWatch Event Rule SSMOpsItems-SSM-
maintenance-window-execution-failed was triggered. Your SSM Maintenance Window
 execution has failed. See below for more details.",
                            "LastModifiedBy": "arn:aws:sts::111222333444:assumed-
role/OpsItem-CWE-Role/fbf77cbe264a33509569f23e4EXAMPLE",
                            "LastModifiedTime": "2019-11-26T20:00:07.237Z",
                            "Notifications": "",
                            "OperationalData": "{\"/aws/resources\":{\"type
\":\"SearchableString\",\"value\":\"[{\\\"arn\\\":\\\"arn:aws:ssm:us-
east-2:111222333444:maintenancewindow/mw-0e83ba440dEXAMPLE\\\"}]\"},\"/aws/dedup\":
{\"type\":\"SearchableString\",\"value\":\"{\\\"dedupString\\\":\\\"SSMOpsItems-SSM-
maintenance-window-execution-failed\\\"}\"}}",
```

```
                            "OpsItemId": "oi-bb2a0EXAMPLE",
                            "RelatedItems": "",
                            "Severity": "3",
                            "Source": "SSM",
                            "Status": "Open",
                            "Title": "SSM Maintenance Window execution failed"
                        }
                    ]
                }
            }
        }
    ]
}
```

Per ulteriori informazioni, vedere Working with OpsItems nella AWS Systems Manager User Guide.

- Per i dettagli sull'API, vedere GetOpsSummaryin AWS CLI Command Reference.

`get-parameter-history`

Il seguente esempio di codice mostra come utilizzare`get-parameter-history`.

AWS CLI

Come ottenere una cronologia dei valori per un parametro

L'esempio seguente `get-parameter-history` elenca la cronologia delle modifiche per il parametro specificato, incluso il relativo valore.

```
aws ssm get-parameter-history \
    --name "MyStringParameter"
```

Output:

```
{
    "Parameters": [
        {
            "Name": "MyStringParameter",
            "Type": "String",
            "LastModifiedDate": 1582154711.976,
```

```
            "LastModifiedUser": "arn:aws:iam::111222333444:user/Mary-Major",
            "Description": "This is the first version of my String parameter",
            "Value": "Veni",
            "Version": 1,
            "Labels": [],
            "Tier": "Standard",
            "Policies": []
        },
        {
            "Name": "MyStringParameter",
            "Type": "String",
            "LastModifiedDate": 1582156093.471,
            "LastModifiedUser": "arn:aws:iam::111222333444:user/Mary-Major",
            "Description": "This is the second version of my String parameter",
            "Value": "Vidi",
            "Version": 2,
            "Labels": [],
            "Tier": "Standard",
            "Policies": []
        },
        {
            "Name": "MyStringParameter",
            "Type": "String",
            "LastModifiedDate": 1582156117.545,
            "LastModifiedUser": "arn:aws:iam::111222333444:user/Mary-Major",
            "Description": "This is the third version of my String parameter",
            "Value": "Vici",
            "Version": 3,
            "Labels": [],
            "Tier": "Standard",
            "Policies": []
        }
    ]
}
```

Per ulteriori informazioni, consulta Utilizzo delle versioni dei parametri nella Guida per l'utente di AWS Systems Manager.

- Per i dettagli sull'API, consulta GetParameterHistory AWS CLICommand Reference.

`get-parameter`

Il seguente esempio di codice mostra come utilizzare`get-parameter`.

AWS CLI

Esempio 1: come visualizzare il valore di un parametro

L'esempio seguente `get-parameter` elenca il valore per il singolo parametro specificato.

```
aws ssm get-parameter \
    --name "MyStringParameter"
```

Output:

```
{
    "Parameter": {
        "Name": "MyStringParameter",
        "Type": "String",
        "Value": "Veni",
        "Version": 1,
        "LastModifiedDate": 1530018761.888,
        "ARN": "arn:aws:ssm:us-east-2:111222333444:parameter/MyStringParameter"
        "DataType": "text"
    }
}
```

Per ulteriori informazioni, consulta Utilizzo dell'Archivio parametri nella Guida per l'utente di AWS Systems Manager.

Esempio 2: per decrittografare il valore di un parametro SecureString

L'esempio seguente `get-parameter` decrittografa il valore del parametro specificato `SecureString`.

```
aws ssm get-parameter \
    --name "MySecureStringParameter" \
    --with-decryption
```

Output:

```
{
    "Parameter": {
        "Name": "MySecureStringParameter",
        "Type": "SecureString",
```

```
        "Value": "16679b88-310b-4895-a943-e0764EXAMPLE",
        "Version": 2,
        "LastModifiedDate": 1582155479.205,
        "ARN": "arn:aws:ssm:us-east-2:111222333444:parameter/
MySecureStringParameter"
        "DataType": "text"
    }
}
```

Per ulteriori informazioni, consulta Utilizzo dell'Archivio parametri nella Guida per l'utente di AWS Systems Manager.

Esempio 3: come visualizzare il valore di un parametro utilizzando le etichette

L'esempio seguente `get-parameter` elenca il valore per il singolo parametro specificato con un'etichetta specificata.

```
aws ssm get-parameter \
    --name "MyParameter:label"
```

Output:

```
{
    "Parameter": {
        "Name": "MyParameter",
        "Type": "String",
        "Value": "parameter version 2",
        "Version": 2,
        "Selector": ":label",
        "LastModifiedDate": "2021-07-12T09:49:15.865000-07:00",
        "ARN": "arn:aws:ssm:us-west-2:786973925828:parameter/MyParameter",
        "DataType": "text"
    }
}
```

Per ulteriori informazioni, consulta Utilizzo delle versioni dei parametri nella Guida per l'utente di AWS Systems Manager.

Esempio 4: come visualizzare il valore di un parametro utilizzando le versioni

L'esempio seguente `get-parameter` elenca il valore per la versione a parametro singolo specificata.

```
aws ssm get-parameter \
    --name "MyParameter:2"
```

Output:

```
{
    "Parameter": {
        "Name": "MyParameter",
        "Type": "String",
        "Value": "parameter version 2",
        "Version": 2,
        "Selector": ":2",
        "LastModifiedDate": "2021-07-12T09:49:15.865000-07:00",
        "ARN": "arn:aws:ssm:us-west-2:786973925828:parameter/MyParameter",
        "DataType": "text"
    }
}
```

Per ulteriori informazioni, consulta Utilizzo delle versioni dei parametri nella Guida per l'utente di AWS Systems Manager.

- Per i dettagli sull'API, vedere GetParameterin AWS CLI Command Reference.

`get-parameters-by-path`

Il seguente esempio di codice mostra come utilizzare`get-parameters-by-path`.

AWS CLI

Come elencare i parametri in un percorso specifico

L'esempio `get-parameters-by-path` seguente elenca i valori dei tre parametri specificati.

```
aws ssm get-parameters-by-path \
    --path "/site/newyork/department/"
```

Output:

```
{
    "Parameters": [
        {
```

```
            "Name": "/site/newyork/department/marketing",
            "Type": "String",
            "Value": "Floor 2",
            "Version": 1,
            "LastModifiedDate": 1530018761.888,
            "ARN": "arn:aws:ssm:us-east-1:111222333444:parameter/site/newyork/
department/marketing"
        },
        {
            "Name": "/site/newyork/department/infotech",
            "Type": "String",
            "Value": "Floor 3",
            "Version": 1,
            "LastModifiedDate": 1530018823.429,
            "ARN": "arn:aws:ssm:us-east-1:111222333444:parameter/site/newyork/
department/infotech"
        },
        ...
    ]
}
```

Per ulteriori informazioni, consulta Utilizzo delle gerarchie dei parametri nella Guida per l'utente di AWS Systems Manager.

- Per i dettagli sull'API, consulta GetParametersByPath AWS CLICommand Reference.

`get-parameters`

Il seguente esempio di codice mostra come utilizzare`get-parameters`.

AWS CLI

Esempio 1: come elencare i valori di un parametro

L'esempio seguente `get-parameters` elenca i valori dei tre parametri specificati.

```
aws ssm get-parameters \
    --names "MyStringParameter" "MyStringListParameter" "MyInvalidParameterName"
```

Output:

```
{
```

```
    "Parameters": [
        {
            "Name": "MyStringListParameter",
            "Type": "StringList",
            "Value": "alpha,beta,gamma",
            "Version": 1,
            "LastModifiedDate": 1582154764.222,
            "ARN": "arn:aws:ssm:us-east-2:111222333444:parameter/
MyStringListParameter"
            "DataType": "text"
        },
        {
            "Name": "MyStringParameter",
            "Type": "String",
            "Value": "Vici",
            "Version": 3,
            "LastModifiedDate": 1582156117.545,
            "ARN": "arn:aws:ssm:us-east-2:111222333444:parameter/MyStringParameter"
            "DataType": "text"
        }
    ],
    "InvalidParameters": [
        "MyInvalidParameterName"
    ]
}
```

Per ulteriori informazioni, consulta Utilizzo dell'Archivio parametri nella Guida per l'utente di AWS Systems Manager.

Esempio 2: come elencare i nomi e i valori di più parametri utilizzando l'opzione ``--query``

L'esempio seguente `get-parameters` elenca i nomi e i valori dei parametri specificati.

```
aws ssm get-parameters \
    --names MyStringParameter MyStringListParameter \
    --query "Parameters[*].{Name:Name,Value:Value}"
```

Output:

```
[
    {
        "Name": "MyStringListParameter",
        "Value": "alpha,beta,gamma"
```

```
    },
    {
        "Name": "MyStringParameter",
        "Value": "Vidi"
    }
]
```

Per ulteriori informazioni, consulta Utilizzo dell'Archivio parametri nella Guida per l'utente di AWS Systems Manager.

Esempio 3: come visualizzare il valore di un parametro utilizzando le etichette

L'esempio seguente `get-parameter` elenca il valore per il singolo parametro specificato con un'etichetta specificata.

```
aws ssm get-parameter \
    --name "MyParameter:label"
```

Output:

```
{
    "Parameters": [
        {
            "Name": "MyLabelParameter",
            "Type": "String",
            "Value": "parameter by label",
            "Version": 1,
            "Selector": ":label",
            "LastModifiedDate": "2021-07-12T09:49:15.865000-07:00",
            "ARN": "arn:aws:ssm:us-west-2:786973925828:parameter/MyParameter",
            "DataType": "text"
        },
        {
            "Name": "MyVersionParameter",
            "Type": "String",
            "Value": "parameter by version",
            "Version": 2,
            "Selector": ":2",
            "LastModifiedDate": "2021-03-24T16:20:28.236000-07:00",
            "ARN": "arn:aws:ssm:us-west-2:786973925828:parameter/unlabel-param",
            "DataType": "text"
        }
```

```
    ],
    "InvalidParameters": []
}
```

Per ulteriori informazioni, consulta Utilizzo delle versioni dei parametri nella Guida per l'utente di AWS Systems Manager.

- Per i dettagli sull'API, consulta GetParameters AWS CLICommand Reference.

`get-patch-baseline-for-patch-group`

Il seguente esempio di codice mostra come utilizzare`get-patch-baseline-for-patch-group`.

AWS CLI

Come visualizzare la baseline delle patch per un gruppo di patch

L'esempio seguente `get-patch-baseline-for-patch-group` recupera i dettagli relativi alla baseline delle patch per il gruppo di patch specificato.

```
aws ssm get-patch-baseline-for-patch-group \
    --patch-group "DEV"
```

Output:

```
{
    "PatchGroup": "DEV",
    "BaselineId": "pb-0123456789abcdef0",
    "OperatingSystem": "WINDOWS"
}
```

Per ulteriori informazioni, vedere Create a Patch Group < https://docs.aws.amazon.com/systems-manager/ latest/userguide/sysman - patch-group-tagging .html>__ e Aggiungere un gruppo di patch a una Patch Baseline nella Systems Manager AWS User Guide.

- Per i dettagli sull'API, vedere in Command Reference. GetPatchBaselineForPatchGroupAWS CLI

`get-patch-baseline`

Il seguente esempio di codice mostra come utilizzare`get-patch-baseline`.

AWS CLI

Visualizzazione di una baseline delle patch

Nell'esempio seguente `get-patch-baseline` vengono recuperati i dettagli per la baseline delle patch specificata.

```
aws ssm get-patch-baseline \
    --baseline-id "pb-0123456789abcdef0"
```

Output:

```
{
    "BaselineId": "pb-0123456789abcdef0",
    "Name": "WindowsPatching",
    "OperatingSystem": "WINDOWS",
    "GlobalFilters": {
        "PatchFilters": []
    },
    "ApprovalRules": {
        "PatchRules": [
            {
                "PatchFilterGroup": {
                    "PatchFilters": [
                        {
                            "Key": "PRODUCT",
                            "Values": [
                                "WindowsServer2016"
                            ]
                        }
                    ]
                },
                "ComplianceLevel": "CRITICAL",
                "ApproveAfterDays": 0,
                "EnableNonSecurity": false
            }
        ]
    },
    "ApprovedPatches": [],
    "ApprovedPatchesComplianceLevel": "UNSPECIFIED",
    "ApprovedPatchesEnableNonSecurity": false,
    "RejectedPatches": [],
    "RejectedPatchesAction": "ALLOW_AS_DEPENDENCY",
```

```
    "PatchGroups": [
        "QA",
        "DEV"
    ],
    "CreatedDate": 1550244180.465,
    "ModifiedDate": 1550244180.465,
    "Description": "Patches for Windows Servers",
    "Sources": []
}
```

Per ulteriori informazioni, consulta informazioni baseline delle patch nella Guida per l'utente di AWS Systems Manager.

- Per i dettagli sull'API, consulta GetPatchBaseline AWS CLICommand Reference.

`get-service-setting`

Il seguente esempio di codice mostra come utilizzare`get-service-setting`.

AWS CLI

Come recuperare l'impostazione del servizio per il throughput dell'archivio parametri

L'esempio `get-service-setting` seguente recupera l'impostazione del servizio corrente per il throughput dell'archivio parametri nella Regione specificata.

```
aws ssm get-service-setting \
    --setting-id arn:aws:ssm:us-east-1:123456789012:servicesetting/ssm/parameter-store/high-throughput-enabled
```

Output:

```
{
    "ServiceSetting": {
        "SettingId": "/ssm/parameter-store/high-throughput-enabled",
        "SettingValue": "false",
        "LastModifiedDate": 1555532818.578,
        "LastModifiedUser": "System",
        "ARN": "arn:aws:ssm:us-east-1:123456789012:servicesetting/ssm/parameter-store/high-throughput-enabled",
        "Status": "Default"
    }
```

```
}
```

Per ulteriori informazioni, consulta Incremento della capacità di throughput in Parameter Store nella Guida per l'utente di AWS Systems Manager.

- Per i dettagli sull'API, consulta GetServiceSetting AWS CLICommand Reference.

`label-parameter-version`

Il seguente esempio di codice mostra come utilizzare`label-parameter-version`.

AWS CLI

Esempio 1: come aggiungere un'etichetta alla versione più recente di un parametro

L'esempio `label-parameter-version` seguente aggiunge un'etichetta alla versione più recente del parametro specificato.

```
aws ssm label-parameter-version \
    --name "MyStringParameter" \
    --labels "ProductionReady"
```

Output:

```
{
    "InvalidLabels": [],
    "ParameterVersion": 3
}
```

Per ulteriori informazioni, consulta Utilizzo delle versioni dei parametri nella Guida per l'utente di AWS Systems Manager.

Esempio 2: come aggiungere un'etichetta a una versione specifica di un parametro

L'esempio `label-parameter-version` seguente aggiunge un'etichetta alla versione specificata di un parametro.

```
aws ssm label-parameter-version \
    --name "MyStringParameter" \
    --labels "ProductionReady" \
    --parameter-version "2" --labels "DevelopmentReady"
```

Per ulteriori informazioni, consulta Utilizzo delle versioni dei parametri nella Guida per l'utente di AWS Systems Manager.

- Per i dettagli sull'API, consulta LabelParameterVersion AWS CLICommand Reference.

`list-association-versions`

Il seguente esempio di codice mostra come utilizzare`list-association-versions`.

AWS CLI

Come recuperare tutte le versioni di un'associazione per un determinato ID di associazione

L'esempio seguente `list-association-versions` elenca tutte le versioni delle associazioni specificate.

```
aws ssm list-association-versions \
    --association-id "8dfe3659-4309-493a-8755-0123456789ab"
```

Output:

```
{
"AssociationVersions": [
        {
            "AssociationId": "8dfe3659-4309-493a-8755-0123456789ab",
            "AssociationVersion": "1",
            "CreatedDate": 1550505536.726,
            "Name": "AWS-UpdateSSMAgent",
            "Parameters": {
                "allowDowngrade": [
                    "false"
                ],
                "version": [
                    ""
                ]
            },
            "Targets": [
                {
                    "Key": "InstanceIds",
                    "Values": [
                        "i-1234567890abcdef0"
                    ]
                }
```

```
            ],
            "ScheduleExpression": "cron(0 00 12 ? * SUN *)",
            "AssociationName": "UpdateSSMAgent"
        }
    ]
}
```

Per ulteriori informazioni, consulta Utilizzo delle associazioni in Systems Manager nella AWS Guida per l'utente di Systems Manager.

- Per i dettagli sull'API, consulta ListAssociationVersions AWS CLICommand Reference.

`list-associations`

Il seguente esempio di codice mostra come utilizzare`list-associations`.

AWS CLI

Esempio 1: come elencare le associazioni per un'istanza specifica

Il seguente esempio di associazione di elenchi elenca tutte le associazioni con Update AssociationName. SSMAgent

```
aws ssm list-associations /
    --association-filter-list "key=AssociationName,value=UpdateSSMAgent"
```

Output:

```
{
    "Associations": [
        {
            "Name": "AWS-UpdateSSMAgent",
            "InstanceId": "i-1234567890abcdef0",
            "AssociationId": "8dfe3659-4309-493a-8755-0123456789ab",
            "AssociationVersion": "1",
            "Targets": [
                {
                    "Key": "InstanceIds",
                    "Values": [
                        "i-016648b75dd622dab"
                    ]
                }
```

```
            ],
            "Overview": {
                "Status": "Pending",
                "DetailedStatus": "Associated",
                "AssociationStatusAggregatedCount": {
                    "Pending": 1
                }
            },
            "ScheduleExpression": "cron(0 00 12 ? * SUN *)",
            "AssociationName": "UpdateSSMAgent"
        }
    ]
}
```

Per ulteriori informazioni, consulta Utilizzo delle associazioni in Systems Manager nella Guida per l'utente di Systems Manager.

Esempio 2: come elencare le associazioni per un documento specifico

Il seguente esempio di associazioni di elenchi elenca tutte le associazioni per il documento specificato.

```
aws ssm list-associations /
    --association-filter-list "key=Name,value=AWS-UpdateSSMAgent"
```

Output:

```
{
    "Associations": [
        {
            "Name": "AWS-UpdateSSMAgent",
            "InstanceId": "i-1234567890abcdef0",
            "AssociationId": "8dfe3659-4309-493a-8755-0123456789ab",
            "AssociationVersion": "1",
            "Targets": [
                {
                    "Key": "InstanceIds",
                    "Values": [
                        "i-1234567890abcdef0"
                    ]
                }
            ],
```

```
            "LastExecutionDate": 1550505828.548,
            "Overview": {
                "Status": "Success",
                "DetailedStatus": "Success",
                "AssociationStatusAggregatedCount": {
                    "Success": 1
                }
            },
            "ScheduleExpression": "cron(0 00 12 ? * SUN *)",
            "AssociationName": "UpdateSSMAgent"
        },
    {
            "Name": "AWS-UpdateSSMAgent",
            "InstanceId": "i-9876543210abcdef0",
            "AssociationId": "fbc07ef7-b985-4684-b82b-0123456789ab",
            "AssociationVersion": "1",
            "Targets": [
                {
                    "Key": "InstanceIds",
                    "Values": [
                        "i-9876543210abcdef0"
                    ]
                }
            ],
            "LastExecutionDate": 1550507531.0,
            "Overview": {
                "Status": "Success",
                "AssociationStatusAggregatedCount": {
                    "Success": 1
                }
            }
        }
    ]
}
```

Per ulteriori informazioni, consulta Utilizzo delle associazioni in Systems Manager nella Guida per l'utente di Systems Manager.

- Per i dettagli sull'API, vedere ListAssociationsin AWS CLI Command Reference.

`list-command-invocations`

Il seguente esempio di codice mostra come utilizzare`list-command-invocations`.

AWS CLI

Come elencare le invocazioni di un comando specifico

L'esempio seguente `list-command-invocations` elenca tutte le invocazioni di un comando.

```
aws ssm list-command-invocations \
    --command-id "ef7fdfd8-9b57-4151-a15c-db9a12345678" \
    --details
```

Output:

```
{
    "CommandInvocations": [
        {
            "CommandId": "ef7fdfd8-9b57-4151-a15c-db9a12345678",
            "InstanceId": "i-02573cafcfEXAMPLE",
            "InstanceName": "",
            "Comment": "b48291dd-ba76-43e0-
b9df-13e11ddaac26:6960febb-2907-4b59-8e1a-d6ce8EXAMPLE",
            "DocumentName": "AWS-UpdateSSMAgent",
            "DocumentVersion": "",
            "RequestedDateTime": 1582136283.089,
            "Status": "Success",
            "StatusDetails": "Success",
            "StandardOutputUrl": "",
            "StandardErrorUrl": "",
            "CommandPlugins": [
                {
                    "Name": "aws:updateSsmAgent",
                    "Status": "Success",
                    "StatusDetails": "Success",
                    "ResponseCode": 0,
                    "ResponseStartDateTime": 1582136283.419,
                    "ResponseFinishDateTime": 1582136283.51,
                    "Output": "Updating amazon-ssm-agent from 2.3.842.0 to latest
\nSuccessfully downloaded https://s3.us-east-2.amazonaws.com/amazon-ssm-us-east-2/
ssm-agent-manifest.json\namazon-ssm-agent 2.3.842.0 has already been installed,
 update skipped\n",
                    "StandardOutputUrl": "",
                    "StandardErrorUrl": "",
                    "OutputS3Region": "us-east-2",
                    "OutputS3BucketName": "",
```

```
                    "OutputS3KeyPrefix": ""
                }
            ],
            "ServiceRole": "",
            "NotificationConfig": {
                "NotificationArn": "",
                "NotificationEvents": [],
                "NotificationType": ""
            },
            "CloudWatchOutputConfig": {
                "CloudWatchLogGroupName": "",
                "CloudWatchOutputEnabled": false
            }
        },
        {
            "CommandId": "ef7fdfd8-9b57-4151-a15c-db9a12345678",
            "InstanceId": "i-0471e04240EXAMPLE",
            "InstanceName": "",
            "Comment": "b48291dd-ba76-43e0-
b9df-13e11ddaac26:6960febb-2907-4b59-8e1a-d6ce8EXAMPLE",
            "DocumentName": "AWS-UpdateSSMAgent",
            "DocumentVersion": "",
            "RequestedDateTime": 1582136283.02,
            "Status": "Success",
            "StatusDetails": "Success",
            "StandardOutputUrl": "",
            "StandardErrorUrl": "",
            "CommandPlugins": [
                {
                    "Name": "aws:updateSsmAgent",
                    "Status": "Success",
                    "StatusDetails": "Success",
                    "ResponseCode": 0,
                    "ResponseStartDateTime": 1582136283.812,
                    "ResponseFinishDateTime": 1582136295.031,
                    "Output": "Updating amazon-ssm-agent from 2.3.672.0 to latest
\nSuccessfully downloaded https://s3.us-east-2.amazonaws.com/amazon-ssm-us-east-2/
ssm-agent-manifest.json\nSuccessfully downloaded https://s3.us-east-2.amazonaws.com/
amazon-ssm-us-east-2/amazon-ssm-agent-updater/2.3.842.0/amazon-ssm-agent-updater-
snap-amd64.tar.gz\nSuccessfully downloaded https://s3.us-east-2.amazonaws.com/
amazon-ssm-us-east-2/amazon-ssm-agent/2.3.672.0/amazon-ssm-agent-snap-amd64.tar.gz
\nSuccessfully downloaded https://s3.us-east-2.amazonaws.com/amazon-ssm-us-east-2/
amazon-ssm-agent/2.3.842.0/amazon-ssm-agent-snap-amd64.tar.gz\nInitiating amazon-
ssm-agent update to 2.3.842.0\namazon-ssm-agent updated successfully to 2.3.842.0",
```

```
                    "StandardOutputUrl": "",
                    "StandardErrorUrl": "",
                    "OutputS3Region": "us-east-2",
                    "OutputS3BucketName": "",
                    "OutputS3KeyPrefix": "8bee3135-398c-4d31-99b6-e42d2EXAMPLE/
i-0471e04240EXAMPLE/awsupdateSsmAgent"
                }
            ],
            "ServiceRole": "",
            "NotificationConfig": {
                "NotificationArn": "",
                "NotificationEvents": [],
                "NotificationType": ""
            },
            "CloudWatchOutputConfig": {
                "CloudWatchLogGroupName": "",
                "CloudWatchOutputEnabled": false
            }
        }
    ]
}
```

Per ulteriori informazioni, consulta Informazioni sugli stati dei comandi nella Guida per l'utente di AWS Systems Manager.

- Per i dettagli sull'API, consulta ListCommandInvocations AWS CLICommand Reference.

`list-commands`

Il seguente esempio di codice mostra come utilizzare`list-commands`.

AWS CLI

Esempio 1: come ottenere lo stato di un comando specifico

L'esempio seguente `list-commands` recupera e visualizza lo stato del comando specificato.

```
aws ssm list-commands \
    --command-id "0831e1a8-a1ac-4257-a1fd-c831bEXAMPLE"
```

Esempio 2: come ottenere lo stato dei comandi richiesti dopo una data specifica

L'esempio seguente `list-commands` recupera i dettagli dei comandi richiesti dopo la data specificata.

```
aws ssm list-commands \
    --filter "key=InvokedAfter,value=2020-02-01T00:00:00Z"
```

Esempio 3: Per elencare tutti i comandi richiesti in un AWS account

L'`list-commands`esempio seguente elenca tutti i comandi richiesti dagli utenti nell' AWS account e nella regione correnti.

```
aws ssm list-commands
```

Output:

```
{
    "Commands": [
        {
            "CommandId": "8bee3135-398c-4d31-99b6-e42d2EXAMPLE",
            "DocumentName": "AWS-UpdateSSMAgent",
            "DocumentVersion": "",
            "Comment": "b48291dd-ba76-43e0-
b9df-13e11ddaac26:6960febb-2907-4b59-8e1a-d6ce8EXAMPLE",
            "ExpiresAfter": "2020-02-19T11:28:02.500000-08:00",
            "Parameters": {},
            "InstanceIds": [
                "i-028ea792daEXAMPLE",
                "i-02feef8c46EXAMPLE",
                "i-038613f3f0EXAMPLE",
                "i-03a530a2d4EXAMPLE",
                "i-083b678d37EXAMPLE",
                "i-0dee81debaEXAMPLE"
            ],
            "Targets": [],
            "RequestedDateTime": "2020-02-19T10:18:02.500000-08:00",
            "Status": "Success",
            "StatusDetails": "Success",
            "OutputS3BucketName": "",
            "OutputS3KeyPrefix": "",
            "MaxConcurrency": "50",
            "MaxErrors": "100%",
            "TargetCount": 6,
```

```
            "CompletedCount": 6,
            "ErrorCount": 0,
            "DeliveryTimedOutCount": 0,
            "ServiceRole": "",
            "NotificationConfig": {
                "NotificationArn": "",
                "NotificationEvents": [],
                "NotificationType": ""
            },
            "CloudWatchOutputConfig": {
                "CloudWatchLogGroupName": "",
                "CloudWatchOutputEnabled": false
            }
        }
        {
            "CommandId": "e9ade581-c03d-476b-9b07-26667EXAMPLE",
            "DocumentName": "AWS-FindWindowsUpdates",
            "DocumentVersion": "1",
            "Comment": "",
            "ExpiresAfter": "2020-01-24T12:37:31.874000-08:00",
            "Parameters": {
                "KbArticleIds": [
                    ""
                ],
                "UpdateLevel": [
                    "All"
                ]
            },
            "InstanceIds": [],
            "Targets": [
                {
                    "Key": "InstanceIds",
                    "Values": [
                        "i-00ec29b21eEXAMPLE",
                        "i-09911ddd90EXAMPLE"
                    ]
                }
            ],
            "RequestedDateTime": "2020-01-24T11:27:31.874000-08:00",
            "Status": "Success",
            "StatusDetails": "Success",
            "OutputS3BucketName": "my-us-east-2-bucket",
            "OutputS3KeyPrefix": "my-rc-output",
            "MaxConcurrency": "50",
```

```
            "MaxErrors": "0",
            "TargetCount": 2,
            "CompletedCount": 2,
            "ErrorCount": 0,
            "DeliveryTimedOutCount": 0,
            "ServiceRole": "arn:aws:iam::111222333444:role/aws-service-role/
ssm.amazonaws.com/AWSServiceRoleForAmazonSSM",
            "NotificationConfig": {
                "NotificationArn": "arn:aws:sns:us-east-2:111222333444:my-us-east-2-
notification-arn",
                "NotificationEvents": [
                    "All"
                ],
                "NotificationType": "Invocation"
            },
            "CloudWatchOutputConfig": {
                "CloudWatchLogGroupName": "",
                "CloudWatchOutputEnabled": false
            }
        }
        {
            "CommandId": "d539b6c3-70e8-4853-80e5-0ce4fEXAMPLE",
            "DocumentName": "AWS-RunPatchBaseline",
            "DocumentVersion": "1",
            "Comment": "",
            "ExpiresAfter": "2020-01-24T12:21:04.350000-08:00",
            "Parameters": {
                "InstallOverrideList": [
                    ""
                ],
                "Operation": [
                    "Install"
                ],
                "RebootOption": [
                    "RebootIfNeeded"
                ],
                "SnapshotId": [
                    ""
                ]
            },
            "InstanceIds": [],
            "Targets": [
                {
                    "Key": "InstanceIds",
```

```
                    "Values": [
                        "i-00ec29b21eEXAMPLE",
                        "i-09911ddd90EXAMPLE"
                    ]
                }
            ],
            "RequestedDateTime": "2020-01-24T11:11:04.350000-08:00",
            "Status": "Success",
            "StatusDetails": "Success",
            "OutputS3BucketName": "my-us-east-2-bucket",
            "OutputS3KeyPrefix": "my-rc-output",
            "MaxConcurrency": "50",
            "MaxErrors": "0",
            "TargetCount": 2,
            "CompletedCount": 2,
            "ErrorCount": 0,
            "DeliveryTimedOutCount": 0,
            "ServiceRole": "arn:aws:iam::111222333444:role/aws-service-role/
ssm.amazonaws.com/AWSServiceRoleForAmazonSSM",
            "NotificationConfig": {
                "NotificationArn": "arn:aws:sns:us-east-2:111222333444:my-us-east-2-
notification-arn",
                "NotificationEvents": [
                    "All"
                ],
                "NotificationType": "Invocation"
            },
            "CloudWatchOutputConfig": {
                "CloudWatchLogGroupName": "",
                "CloudWatchOutputEnabled": false
            }
        }
    ]
}
```

Per ulteriori informazioni, consulta Eseguire comandi utilizzando Systems Manager Run Command nella Guida per l'utente di AWS Systems Manager.

- Per i dettagli sull'API, vedere ListCommandsin AWS CLI Command Reference.

`list-compliance-items`

Il seguente esempio di codice mostra come utilizzare`list-compliance-items`.

AWS CLI

Come elencare gli elementi di conformità per un'istanza specifica

Questo esempio elenca tutti gli elementi di conformità per l'istanza specificata.

Comando:

```
aws ssm list-compliance-items --resource-ids "i-1234567890abcdef0" --resource-
types "ManagedInstance"
```

Output:

```
{
  "ComplianceItems": [
      {
          "ComplianceType": "Association",
          "ResourceType": "ManagedInstance",
          "ResourceId": "i-1234567890abcdef0",
          "Id": "8dfe3659-4309-493a-8755-0123456789ab",
          "Title": "",
          "Status": "COMPLIANT",
          "Severity": "UNSPECIFIED",
          "ExecutionSummary": {
              "ExecutionTime": 1550408470.0
          },
          "Details": {
              "DocumentName": "AWS-GatherSoftwareInventory",
              "DocumentVersion": "1"
          }
      },
      {
          "ComplianceType": "Association",
          "ResourceType": "ManagedInstance",
          "ResourceId": "i-1234567890abcdef0",
          "Id": "e4c2ed6d-516f-41aa-aa2a-0123456789ab",
          "Title": "",
          "Status": "COMPLIANT",
          "Severity": "UNSPECIFIED",
          "ExecutionSummary": {
              "ExecutionTime": 1550508475.0
          },
          "Details": {
              "DocumentName": "AWS-UpdateSSMAgent",
```

```
                "DocumentVersion": "1"
            }
        },
                ...
    ],
    "NextToken": "--token string truncated--"
}
```

Come elencare gli elementi di conformità per un'istanza e un ID di associazione specifici

Questo esempio elenca tutti gli elementi di conformità per l'ID di istanza e associazione specificati.

Comando:

```
aws ssm list-compliance-items --resource-ids "i-1234567890abcdef0" --resource-
types "ManagedInstance" --
filters "Key=ComplianceType,Values=Association,Type=EQUAL" "Key=Id,Values=e4c2ed6d-516f-41aa
aa2a-0123456789ab,Type=EQUAL"
```

Come elencare gli elementi di conformità per un'istanza dopo una data e un'ora specifiche

Questo esempio elenca tutti gli elementi di conformità per un'istanza dopo la data e l'ora specificate.

Comando:

```
aws ssm list-compliance-items --resource-ids "i-1234567890abcdef0" --resource-
types "ManagedInstance" --
filters "Key=ExecutionTime,Values=2019-02-18T16:00:00Z,Type=GREATER_THAN"
```

- Per i dettagli sull'API, consulta ListComplianceItems AWS CLICommand Reference.

`list-compliance-summaries`

Il seguente esempio di codice mostra come utilizzare`list-compliance-summaries`.

AWS CLI

Come elencare i riepiloghi di conformità per tutti i tipi di conformità

Questo esempio elenca i riepiloghi di conformità per tutti i tipi di conformità presenti nell'account.

Comando:

```
aws ssm list-compliance-summaries
```

Output:

```
{
  "ComplianceSummaryItems": [
      {
          "ComplianceType": "Association",
          "CompliantSummary": {
              "CompliantCount": 2,
              "SeveritySummary": {
                  "CriticalCount": 0,
                  "HighCount": 0,
                  "MediumCount": 0,
                  "LowCount": 0,
                  "InformationalCount": 0,
                  "UnspecifiedCount": 2
              }
          },
          "NonCompliantSummary": {
              "NonCompliantCount": 0,
              "SeveritySummary": {
                  "CriticalCount": 0,
                  "HighCount": 0,
                  "MediumCount": 0,
                  "LowCount": 0,
                  "InformationalCount": 0,
                  "UnspecifiedCount": 0
              }
          }
      },
      {
          "ComplianceType": "Patch",
          "CompliantSummary": {
              "CompliantCount": 1,
              "SeveritySummary": {
                  "CriticalCount": 0,
                  "HighCount": 0,
                  "MediumCount": 0,
                  "LowCount": 0,
                  "InformationalCount": 0,
                  "UnspecifiedCount": 1
              }
```

```
            },
            "NonCompliantSummary": {
                "NonCompliantCount": 1,
                "SeveritySummary": {
                    "CriticalCount": 1,
                    "HighCount": 0,
                    "MediumCount": 0,
                    "LowCount": 0,
                    "InformationalCount": 0,
                    "UnspecifiedCount": 0
                }
            }
        },
                ...
    ],
    "NextToken": "eyJOZXh0VG9rZW4iOiBudWxsLCAiYm90b190cnVuY2F0ZV9hbW91bnQiOiAyfQ=="
}
```

Come elencare i riepiloghi di conformità per un tipo di conformità specifico

Questo esempio elenca il riepilogo della conformità per il tipo di conformità Patch.

Comando:

```
aws ssm list-compliance-summaries --
filters "Key=ComplianceType,Values=Patch,Type=EQUAL"
```

- Per i dettagli sull'API, consulta ListComplianceSummaries AWS CLICommand Reference.

`list-document-metadata-history`

Il seguente esempio di codice mostra come utilizzare`list-document-metadata-history`.

AWS CLI

Esempio: come visualizzare la cronologia e lo stato delle approvazioni di un modello di modifica

L'esempio `list-document-metadata-history` seguente restituisce la cronologia delle approvazioni per il modello di modifica di Change Manager specificato.

```
aws ssm list-document-metadata-history \
    --name MyChangeManageTemplate \
```

```
    --metadata DocumentReviews
```

Output:

```
{
    "Name": "MyChangeManagerTemplate",
    "DocumentVersion": "1",
    "Author": "arn:aws:iam::111222333444;:user/JohnDoe",
    "Metadata": {
        "ReviewerResponse": [
            {
                "CreateTime": "2021-07-30T11:58:28.025000-07:00",
                "UpdatedTime": "2021-07-30T12:01:19.274000-07:00",
                "ReviewStatus": "APPROVED",
                "Comment": [
                    {
                        "Type": "COMMENT",
                        "Content": "I approve this template version"
                    }
                ],
                "Reviewer": "arn:aws:iam::111222333444;:user/ShirleyRodriguez"
            },
            {
                "CreateTime": "2021-07-30T11:58:28.025000-07:00",
                "UpdatedTime": "2021-07-30T11:58:28.025000-07:00",
                "ReviewStatus": "PENDING"
            }
        ]
    }
}
```

Per ulteriori informazioni, consulta Revisione e approvazione o rifiuto dei modelli di modifica nella Guida per l'utente di AWS Systems Manager.

- Per i dettagli sull'API, consulta ListDocumentMetadataHistory AWS CLICommand Reference.

`list-document-versions`

Il seguente esempio di codice mostra come utilizzare`list-document-versions`.

AWS CLI

Come elencare le versioni del documento

L'esempio seguente `list-document-versions` elenca tutte le versioni di un documento Systems Manager.

```
aws ssm list-document-versions \
    --name "Example"
```

Output:

```
{
    "DocumentVersions": [
        {
            "Name": "Example",
            "DocumentVersion": "1",
            "CreatedDate": 1583257938.266,
            "IsDefaultVersion": true,
            "DocumentFormat": "YAML",
            "Status": "Active"
        }
    ]
}
```

Per ulteriori informazioni, consulta Invio di comandi che utilizzano il parametro della versione del documento nella Guida per l'utente di AWS Systems Manager.

- Per i dettagli sull'API, consulta ListDocumentVersions AWS CLICommand Reference.

`list-documents`

Il seguente esempio di codice mostra come utilizzare`list-documents`.

AWS CLI

Esempio 1: come elencare i documenti

L'esempio seguente `list-documents` elenca i documenti di proprietà dell'account richiedente contrassegnati con il tag personalizzato.

```
aws ssm list-documents \
    --filters Key=Owner,Values=Self Key=tag:DocUse,Values=Testing
```

Output:

```
{
    "DocumentIdentifiers": [
        {
            "Name": "Example",
            "Owner": "29884EXAMPLE",
            "PlatformTypes": [
                "Windows",
                "Linux"
            ],
            "DocumentVersion": "1",
            "DocumentType": "Automation",
            "SchemaVersion": "0.3",
            "DocumentFormat": "YAML",
            "Tags": [
                {
                    "Key": "DocUse",
                    "Value": "Testing"
                }
            ]
        }
    ]
}
```

Per ulteriori informazioni, consulta Documenti AWS Systems Manager nella Guida per l'utente di AWS Systems Manager.

Esempio 2: come elencare i documenti condivisi

L'`list-documents`esempio seguente elenca i documenti condivisi, inclusi i documenti privati condivisi non di proprietà di AWS.

```
aws ssm list-documents \
    --filters Key=Name,Values=sharedDocNamePrefix  Key=Owner,Values=Private
```

Output:

```
{
    "DocumentIdentifiers": [
        {
            "Name": "Example",
            "Owner": "12345EXAMPLE",
            "PlatformTypes": [
```

```
                "Windows",
                "Linux"
            ],
            "DocumentVersion": "1",
            "DocumentType": "Command",
            "SchemaVersion": "0.3",
            "DocumentFormat": "YAML",
            "Tags": []
        }
    ]
}
```

Per ulteriori informazioni, consulta Documenti di AWS Systems Manager nella Guida per l'utente di AWS Systems Manager.

- Per i dettagli sull'API, consulta ListDocuments AWS CLICommand Reference.

`list-inventory-entries`

Il seguente esempio di codice mostra come utilizzare`list-inventory-entries`.

AWS CLI

Esempio 1: come visualizzare le voci relative al tipo di inventario specifiche per un'istanza

L'`list-inventory-entries`esempio seguente elenca le voci di inventario per il tipo di AWS inventario:Application su un'istanza specifica.

```
aws ssm list-inventory-entries \
    --instance-id "i-1234567890abcdef0" \
    --type-name "AWS:Application"
```

Output:

```
{
  "TypeName": "AWS:Application",
  "InstanceId": "i-1234567890abcdef0",
  "SchemaVersion": "1.1",
  "CaptureTime": "2019-02-15T12:17:55Z",
  "Entries": [
    {
      "Architecture": "i386",
      "Name": "Amazon SSM Agent",
```

```
      "PackageId": "{88a60be2-89a1-4df8-812a-80863c2a2b68}",
      "Publisher": "Amazon Web Services",
      "Version": "2.3.274.0"
    },
    {
      "Architecture": "x86_64",
      "InstalledTime": "2018-05-03T13:42:34Z",
      "Name": "AmazonCloudWatchAgent",
      "Publisher": "",
      "Version": "1.200442.0"
    }
  ]
}
```

Esempio 2: come visualizzare le voci di inventario personalizzate assegnate a un'istanza

L'esempio seguente `list-inventory-entries` elenca una voce di inventario personalizzata assegnata a un'istanza.

```
aws ssm list-inventory-entries \
    --instance-id "i-1234567890abcdef0" \
    --type-name "Custom:RackInfo"
```

Output:

```
{
  "TypeName": "Custom:RackInfo",
  "InstanceId": "i-1234567890abcdef0",
  "SchemaVersion": "1.0",
  "CaptureTime": "2021-05-22T10:01:01Z",
  "Entries": [
    {
      "RackLocation": "Bay B/Row C/Rack D/Shelf E"
    }
  ]
}
```

- Per i dettagli sull'API, consulta AWS CLI Command ListInventoryEntriesReference.

`list-ops-item-related-items`

Il seguente esempio di codice mostra come utilizzare`list-ops-item-related-items`.

AWS CLI

Per elencare le risorse relative agli elementi correlati di un OpsItem

L'`list-ops-item-related-items`esempio seguente elenca le risorse relative agli elementi correlati di un. OpsItem

```
aws ssm list-ops-item-related-items \
    --ops-item-id "oi-f99f2EXAMPLE"
```

Output:

```
{
    "Summaries": [
        {
            "OpsItemId": "oi-f99f2EXAMPLE",
            "AssociationId": "e2036148-cccb-490e-ac2a-390e5EXAMPLE",
            "ResourceType": "AWS::SSMIncidents::IncidentRecord",
            "AssociationType": "IsParentOf",
            "ResourceUri": "arn:aws:ssm-incidents::111122223333:incident-record/
example-response/64bd9b45-1d0e-2622-840d-03a87a1451fa",
            "CreatedBy": {
                "Arn": "arn:aws:sts::111122223333:assumed-role/
AWSServiceRoleForIncidentManager/IncidentResponse"
            },
            "CreatedTime": "2021-08-11T18:47:14.994000+00:00",
            "LastModifiedBy": {
                "Arn": "arn:aws:sts::111122223333:assumed-role/
AWSServiceRoleForIncidentManager/IncidentResponse"
            },
            "LastModifiedTime": "2021-08-11T18:47:14.994000+00:00"
        }
    ]
}
```

Per ulteriori informazioni, vedere Utilizzo degli incidenti di Incident Manager OpsCenter nella Guida per l'utente di AWS Systems Manager.

- Per i dettagli sull'API, vedere ListOpsItemRelatedItemsin AWS CLI Command Reference.

`list-resource-compliance-summaries`

Il seguente esempio di codice mostra come utilizzare`list-resource-compliance-summaries`.

AWS CLI

Come elencare i conteggi riepilogativi della conformità a livello di risorsa

Questo esempio elenca i conteggi riepilogativi della conformità a livello di risorsa.

Comando:

```
aws ssm list-resource-compliance-summaries
```

Output:

```
{
  "ResourceComplianceSummaryItems": [
      {
          "ComplianceType": "Association",
          "ResourceType": "ManagedInstance",
          "ResourceId": "i-1234567890abcdef0",
          "Status": "COMPLIANT",
          "OverallSeverity": "UNSPECIFIED",
          "ExecutionSummary": {
              "ExecutionTime": 1550509273.0
          },
          "CompliantSummary": {
              "CompliantCount": 2,
              "SeveritySummary": {
                  "CriticalCount": 0,
                  "HighCount": 0,
                  "MediumCount": 0,
                  "LowCount": 0,
                  "InformationalCount": 0,
                  "UnspecifiedCount": 2
              }
          },
          "NonCompliantSummary": {
              "NonCompliantCount": 0,
              "SeveritySummary": {
                  "CriticalCount": 0,
                  "HighCount": 0,
                  "MediumCount": 0,
                  "LowCount": 0,
                  "InformationalCount": 0,
                  "UnspecifiedCount": 0
```

```
            }
        }
    },
    {
        "ComplianceType": "Patch",
        "ResourceType": "ManagedInstance",
        "ResourceId": "i-9876543210abcdef0",
        "Status": "COMPLIANT",
        "OverallSeverity": "UNSPECIFIED",
        "ExecutionSummary": {
            "ExecutionTime": 1550248550.0,
            "ExecutionId": "7abb6378-a4a5-4f10-8312-0123456789ab",
            "ExecutionType": "Command"
        },
        "CompliantSummary": {
            "CompliantCount": 397,
            "SeveritySummary": {
                "CriticalCount": 0,
                "HighCount": 0,
                "MediumCount": 0,
                "LowCount": 0,
                "InformationalCount": 0,
                "UnspecifiedCount": 397
            }
        },
        "NonCompliantSummary": {
            "NonCompliantCount": 0,
            "SeveritySummary": {
                "CriticalCount": 0,
                "HighCount": 0,
                "MediumCount": 0,
                "LowCount": 0,
                "InformationalCount": 0,
                "UnspecifiedCount": 0
            }
        }
    }
  ],
  "NextToken": "--token string truncated--"
}
```

Come elencare i riepiloghi di conformità a livello di risorsa per un tipo di conformità specifico

Questo esempio elenca i riepiloghi di conformità a livello di risorsa per il tipo di conformità Patch.

Comando:

```
aws ssm list-resource-compliance-summaries --
filters "Key=ComplianceType,Values=Patch,Type=EQUAL"
```

- Per i dettagli sull'API, consulta ListResourceComplianceSummaries AWS CLICommand Reference.

`list-resource-data-sync`

Il seguente esempio di codice mostra come utilizzare`list-resource-data-sync`.

AWS CLI

Come elencare le configurazioni di sincronizzazione dei dati delle risorse

L'esempio seguente recupera informazioni sulle configurazioni di sincronizzazione dei dati delle risorse.

```
aws ssm list-resource-data-sync
```

Output:

```
{
    "ResourceDataSyncItems": [
        {
            "SyncName": "MyResourceDataSync",
            "S3Destination": {
                "BucketName": "ssm-resource-data-sync",
                "SyncFormat": "JsonSerDe",
                "Region": "us-east-1"
            },
            "LastSyncTime": 1550261472.003,
            "LastSuccessfulSyncTime": 1550261472.003,
            "LastStatus": "Successful",
            "SyncCreatedTime": 1543235736.72,
            "LastSyncStatusMessage": "The sync was successfully completed"
        }
    ]
}
```

- Per i dettagli sull'API, consulta ListResourceDataSync AWS CLICommand Reference.

`list-tags-for-resource`

Il seguente esempio di codice mostra come utilizzare`list-tags-for-resource`.

AWS CLI

Come elencare i tag applicati a una baseline delle patch

L'esempio seguente `list-tags-for-resource` elenca i tag per una baseline delle patch.

```
aws ssm list-tags-for-resource \
    --resource-type "PatchBaseline" \
    --resource-id "pb-0123456789abcdef0"
```

Output:

```
{
    "TagList": [
        {
            "Key": "Environment",
            "Value": "Production"
        },
        {
            "Key": "Region",
            "Value": "EMEA"
        }
    ]
}
```

Per ulteriori informazioni, vedete Tagging AWS Resources nella Guida AWS generale.

- Per i dettagli sull'API, consulta ListTagsForResource AWS CLICommand Reference.

`modify-document-permission`

Il seguente esempio di codice mostra come utilizzare`modify-document-permission`.

AWS CLI

Come modificare le autorizzazioni del documento

L'esempio seguente `modify-document-permission` condivide pubblicamente un documento Systems Manager.

```
aws ssm modify-document-permission \
    --name "Example" \
    --permission-type "Share" \
    --account-ids-to-add "All"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Condividi un documento Systems Manager nella Guida per l'utente di AWS Systems Manager.

- Per i dettagli sull'API, consulta ModifyDocumentPermission AWS CLICommand Reference.

`put-compliance-items`

Il seguente esempio di codice mostra come utilizzare`put-compliance-items`.

AWS CLI

Come registrare un tipo di conformità e i dettagli di conformità su un'istanza designata

Questo esempio registra il tipo di conformità `Custom:AVCheck` nell'istanza gestita specificata. Se il comando va a buon fine, non viene generato output.

Comando:

```
aws ssm put-compliance-items --resource-id "i-1234567890abcdef0" --
resource-type "ManagedInstance" --compliance-type "Custom:AVCheck"
 --execution-summary "ExecutionTime=2019-02-18T16:00:00Z" --
items "Id=Version2.0,Title=ScanHost,Severity=CRITICAL,Status=COMPLIANT"
```

- Per i dettagli sull'API, consulta PutComplianceItems AWS CLICommand Reference.

`put-inventory`

Il seguente esempio di codice mostra come utilizzare`put-inventory`.

AWS CLI

Come assegnare metadati di un cliente a un'istanza

Questo esempio assegna le informazioni sulla posizione del rack a un'istanza. Se il comando va a buon fine, non viene generato output.

Comando (Linux):

```
aws ssm put-inventory --instance-id "i-016648b75dd622dab" --items
 '[{"TypeName": "Custom:RackInfo","SchemaVersion": "1.0","CaptureTime":
 "2019-01-22T10:01:01Z","Content":[{"RackLocation": "Bay B/Row C/Rack D/Shelf
 E"}]}]'
```

Comando (Windows):

```
aws ssm put-inventory --instance-id "i-016648b75dd622dab" --
items "TypeName=Custom:RackInfo,SchemaVersion=1.0,CaptureTime=2019-01-22T10:01:01Z,Content=[
 B/Row C/Rack D/Shelf F'}]"
```

- Per i dettagli sull'API, consulta PutInventory AWS CLICommand Reference.

`put-parameter`

Il seguente esempio di codice mostra come utilizzare`put-parameter`.

AWS CLI

Esempio 1: come modificare il valore di un parametro

L'esempio seguente `put-parameter` modifica il valore del parametro specificato.

```
aws ssm put-parameter \
    --name "MyStringParameter" \
    --type "String" \
    --value "Vici" \
    --overwrite
```

Output:

```
{
    "Version": 2,
    "Tier": "Standard"
}
```

Per ulteriori informazioni, vedere Creazione di un parametro Systems Manager (AWS CLI), Gestione dei livelli dei parametri e Utilizzo delle politiche dei parametri nella Guida per l'utente di AWS Systems Manager.

Esempio 2: come creare un parametro avanzato

L'esempio seguente `put-parameter` crea un parametro avanzato.

```
aws ssm put-parameter \
    --name "MyAdvancedParameter" \
    --description "This is an advanced parameter" \
    --value "Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod
 tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam,
 quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat
 [truncated]" \
    --type "String" \
    --tier Advanced
```

Output:

```
{
    "Version": 1,
    "Tier": "Advanced"
}
```

Per ulteriori informazioni, vedere Creazione di un parametro Systems Manager (AWS CLI), Gestione dei livelli dei parametri e Utilizzo delle politiche dei parametri nella Guida per l'utente di AWS Systems Manager.

Esempio 3: come convertire un parametro standard in un parametro avanzato

L'esempio `put-parameter` seguente converte un parametro standard esistente in un parametro avanzato.

```
aws ssm put-parameter \
    --name "MyConvertedParameter" \
    --value "abc123" \
    --type "String" \
    --tier Advanced \
    --overwrite
```

Output:

```
{
    "Version": 2,
```

```
    "Tier": "Advanced"
}
```

Per ulteriori informazioni, vedere Creazione di un parametro Systems Manager (AWS CLI), Gestione dei livelli dei parametri e Utilizzo delle politiche dei parametri nella Guida per l'utente di AWS Systems Manager.

Esempio 4: come creare un parametro con una policy collegata

L'esempio seguente `put-parameter` crea un parametro avanzato con una policy dei parametri allegata.

```
aws ssm put-parameter \
    --name "/Finance/Payroll/q2accesskey" \
    --value "P@sSwW)rd" \
    --type "SecureString" \
    --tier Advanced \
    --policies "[{\"Type\":\"Expiration\",\"Version\":\"1.0\",\"Attributes\":
{\"Timestamp\":\"2020-06-30T00:00:00.000Z\"}},{\"Type\":\"ExpirationNotification\",
\"Version\":\"1.0\",\"Attributes\":{\"Before\":\"5\",\"Unit\":\"Days\"}},{\"Type\":
\"NoChangeNotification\",\"Version\":\"1.0\",\"Attributes\":{\"After\":\"60\",\"Unit
\":\"Days\"}}]"
```

Output:

```
{
    "Version": 1,
    "Tier": "Advanced"
}
```

Per ulteriori informazioni, vedere Creazione di un parametro Systems Manager (AWS CLI), Gestione dei livelli dei parametri e Utilizzo delle politiche dei parametri nella Guida per l'utente di AWS Systems Manager.

Esempio 5: come aggiungere una policy a un parametro esistente

L'esempio seguente `put-parameter` collega una policy a un parametro avanzato esistente.

```
aws ssm put-parameter \
    --name "/Finance/Payroll/q2accesskey" \
    --value "N3wP@sSwW)rd" \
```

```
    --type "SecureString" \
    --tier Advanced \
    --policies "[{\"Type\":\"Expiration\",\"Version\":\"1.0\",\"Attributes\":
{\"Timestamp\":\"2020-06-30T00:00:00.000Z\"}},{\"Type\":\"ExpirationNotification\",
\"Version\":\"1.0\",\"Attributes\":{\"Before\":\"5\",\"Unit\":\"Days\"}},{\"Type\":
\"NoChangeNotification\",\"Version\":\"1.0\",\"Attributes\":{\"After\":\"60\",\"Unit
\":\"Days\"}}]"
    --overwrite
```

Output:

```
{
    "Version": 2,
    "Tier": "Advanced"
}
```

Per ulteriori informazioni, vedere Creazione di un parametro Systems Manager (AWS CLI), Gestione dei livelli dei parametri e Utilizzo delle politiche dei parametri nella Guida per l'utente di AWS Systems Manager.

- Per i dettagli sull'API, vedere PutParameterin AWS CLI Command Reference.

`register-default-patch-baseline`

Il seguente esempio di codice mostra come utilizzare`register-default-patch-baseline`.

AWS CLI

Come impostare la baseline delle patch predefinita

L'esempio seguente `register-default-patch-baseline` registra la baseline delle patch personalizzata specificata come baseline di patch predefinita per il tipo di sistema operativo che supporta.

```
aws ssm register-default-patch-baseline \
    --baseline-id "pb-abc123cf9bEXAMPLE"
```

Output:

```
{
    "BaselineId":"pb-abc123cf9bEXAMPLE"
```

```
}
```

L'`register-default-patch-baseline`esempio seguente registra la linea di base delle patch predefinita fornita da per AWS CentOS come linea di base delle patch predefinita.

```
aws ssm register-default-patch-baseline \
    --baseline-id "arn:aws:ssm:us-east-2:733109147000:patchbaseline/
pb-0574b43a65ea646ed"
```

Output:

```
{
    "BaselineId":"pb-abc123cf9bEXAMPLE"
}
```

Per ulteriori informazioni, consulta Informazioni sulle baseline delle patch predefinite e personalizzate nella Guida per l'utente di AWS Systems Manager.

- Per i dettagli sull'API, vedere RegisterDefaultPatchBaselinein Command Reference.AWS CLI

`register-patch-baseline-for-patch-group`

Il seguente esempio di codice mostra come utilizzare`register-patch-baseline-for-patch-group`.

AWS CLI

Come registrare una baseline delle patch per un gruppo di patch

L'esempio seguente `register-patch-baseline-for-patch-group` registra una baseline delle patch per un gruppo di patch.

```
aws ssm register-patch-baseline-for-patch-group \
    --baseline-id "pb-045f10b4f382baeda" \
    --patch-group "Production"
```

Output:

```
{
    "BaselineId": "pb-045f10b4f382baeda",
```

```
    "PatchGroup": "Production"
}
```

Per ulteriori informazioni, vedere Create a Patch Group < https://docs.aws.amazon.com/systems-manager/ latest/userguide/sysman - patch-group-tagging .html>__ e Aggiungere un gruppo di patch a una Patch Baseline nella Systems Manager AWS User Guide.

- Per i dettagli sull'API, vedere in Command Reference. RegisterPatchBaselineForPatchGroupAWS CLI

`register-target-with-maintenance-window`

Il seguente esempio di codice mostra come utilizzare`register-target-with-maintenance-window`.

AWS CLI

Esempio 1: come registrare una singola destinazione con una finestra di manutenzione

L'esempio seguente `register-target-with-maintenance-window` registra un'istanza con una finestra di manutenzione.

```
aws ssm register-target-with-maintenance-window \
    --window-id "mw-ab12cd34ef56gh78" \
    --target "Key=InstanceIds,Values=i-0000293ffd8c57862" \
    --owner-information "Single instance" \
    --resource-type "INSTANCE"
```

Output:

```
{
    "WindowTargetId":"1a2b3c4d-1a2b-1a2b-1a2b-1a2b3c4d-1a2"
}
```

Esempio 2: per registrare più destinazioni con una finestra di manutenzione utilizzando l'istanza IDs

L'`register-target-with-maintenance-window`esempio seguente registra due istanze con una finestra di manutenzione specificando la relativa istanza. IDs

```
aws ssm register-target-with-maintenance-window \
```

```
    --window-id "mw-ab12cd34ef56gh78" \
    --target "Key=InstanceIds,Values=i-0000293ffd8c57862,i-0cb2b964d3e14fd9f" \
    --owner-information "Two instances in a list" \
    --resource-type "INSTANCE"
```

Output:

```
{
    "WindowTargetId":"1a2b3c4d-1a2b-1a2b-1a2b-1a2b3c4d-1a2"
}
```

Esempio 3: come registrare le destinazioni in una finestra di manutenzione utilizzando i tag delle risorse

L'esempio seguente `register-target-with-maintenance-window` registra le istanze con una finestra di manutenzione specificando i tag di risorsa che sono stati applicati alle istanze.

```
aws ssm register-target-with-maintenance-window \
    --window-id "mw-06cf17cbefcb4bf4f" \
    --targets "Key=tag:Environment,Values=Prod" "Key=Role,Values=Web" \
    --owner-information "Production Web Servers" \
    --resource-type "INSTANCE"
```

Output:

```
{
    "WindowTargetId":"1a2b3c4d-1a2b-1a2b-1a2b-1a2b3c4d-1a2"
}
```

Esempio 4: come registrare le destinazioni utilizzando un gruppo di chiavi dei tag

L'esempio seguente `register-target-with-maintenance-window` registra tutte le istanze che hanno una o più chiavi dei tag assegnate, indipendentemente dai valori delle chiavi.

```
aws ssm register-target-with-maintenance-window \
    --window-id "mw-0c50858d01EXAMPLE" \
    --resource-type "INSTANCE" \
    --target "Key=tag-key,Values=Name,Instance-Type,CostCenter"
```

Output:

```
{
    "WindowTargetId":"1a2b3c4d-1a2b-1a2b-1a2b-1a2b3c4d-1a2"
}
```

Esempio 5: come registrare le destinazioni utilizzando un nome del gruppo di risorse

L'esempio seguente `register-target-with-maintenance-window` registra un gruppo di risorse specificato, indipendentemente dal tipo di risorse che contiene.

```
aws ssm register-target-with-maintenance-window \
    --window-id "mw-0c50858d01EXAMPLE" \
    --resource-type "RESOURCE_GROUP" \
    --target "Key=resource-groups:Name,Values=MyResourceGroup"
```

Output:

```
{
    "WindowTargetId":"1a2b3c4d-1a2b-1a2b-1a2b-1a2b3c4d-1a2"
}
```

Per ulteriori informazioni, vedere Registrare un'istanza di destinazione con la finestra di manutenzione (AWS CLI) nella AWS Systems Manager User Guide.

- Per i dettagli sull'API, consulta RegisterTargetWithMaintenanceWindow AWS CLICommand Reference.

`register-task-with-maintenance-window`

Il seguente esempio di codice mostra come utilizzare`register-task-with-maintenance-window`.

AWS CLI

Esempio 1: come registrare un'attività di automazione con una finestra di manutenzione

L'esempio seguente `register-task-with-maintenance-window` registra un'attività di automazione con una finestra di manutenzione destinata a un'istanza.

```
aws ssm register-task-with-maintenance-window \
    --window-id "mw-082dcd7649EXAMPLE" \
    --targets Key=InstanceIds,Values=i-1234520122EXAMPLE \
```

```
    --task-arn AWS-RestartEC2Instance \
    --service-role-arn arn:aws:iam::111222333444:role/SSM --task-type AUTOMATION \
    --task-invocation-parameters "{\"Automation\":{\"DocumentVersion\":\"\$LATEST\",
\"Parameters\":{\"InstanceId\":[\"{{RESOURCE_ID}}\"]}}}" \
    --priority 0 \
    --max-concurrency 1 \
    --max-errors 1 \
    --name "AutomationExample" \
    --description "Restarting EC2 Instance for maintenance"
```

Output:

```
{
    "WindowTaskId":"11144444-5555-6666-7777-88888888"
}
```

Per ulteriori informazioni, vedere Registrare un'attività con la finestra di manutenzione (AWS CLI) nella Guida per l'utente di AWS Systems Manager.

Esempio 2: come registrare un'attività con una finestra di manutenzione

L'esempio seguente `register-task-with-maintenance-window` registra un'attività Lambda con una finestra di manutenzione destinata a un'istanza.

```
aws ssm register-task-with-maintenance-window \
    --window-id "mw-082dcd7649dee04e4" \
    --targets Key=InstanceIds,Values=i-12344d305eEXAMPLE \
    --task-arn arn:aws:lambda:us-east-1:111222333444:function:SSMTestLAMBDA \
    --service-role-arn arn:aws:iam::111222333444:role/SSM \
    --task-type LAMBDA \
    --task-invocation-parameters '{"Lambda":{"Payload":"{\"InstanceId\":
\"{{RESOURCE_ID}}\",\"targetType\":\"{{TARGET_TYPE}}\"}","Qualifier":"$LATEST"}}' \
    --priority 0 \
    --max-concurrency 10 \
    --max-errors 5 \
    --name "Lambda_Example" \
    --description "My Lambda Example"
```

Output:

```
{
    "WindowTaskId":"22244444-5555-6666-7777-88888888"
```

```
}
```

Per ulteriori informazioni, vedere Registrare un'attività con la finestra di manutenzione (AWS CLI) nella Guida per l'utente di AWS Systems Manager.

Esempio 3: come registrare un'attività Run Command con una finestra di manutenzione

L'esempio seguente `register-task-with-maintenance-window` registra un'attività Run Command con una finestra di manutenzione destinata a un'istanza.

```
aws ssm register-task-with-maintenance-window \
    --window-id "mw-082dcd7649dee04e4" \
    --targets "Key=InstanceIds,Values=i-12344d305eEXAMPLE" \
    --service-role-arn "arn:aws:iam::111222333444:role/SSM" \
    --task-type "RUN_COMMAND" \
    --name "SSMInstallPowerShellModule" \
    --task-arn "AWS-InstallPowerShellModule" \
    --task-invocation-parameters "{\"RunCommand\":{\"Comment\":\"\",
\"OutputS3BucketName\":\"runcommandlogs\",\"Parameters\":{\"commands\":[\"Get-
Module -ListAvailable\"],\"executionTimeout\":[\"3600\"],\"source\":[\"https:\/
\/gallery.technet.microsoft.com\/EZOut-33ae0fb7\/file\/110351\/1\/EZOut.zip\"],
\"workingDirectory\":[\"\\\\\"]},\"TimeoutSeconds\":600}}" \
    --max-concurrency 1 \
    --max-errors 1 \
    --priority 10
```

Output:

```
{
    "WindowTaskId":"33344444-5555-6666-7777-88888888"
}
```

Per ulteriori informazioni, vedere Registrare un'attività con la finestra di manutenzione (AWS CLI) nella Guida per l'utente di AWS Systems Manager.

Esempio 4: come registrare un'attività Step Functions con una finestra di manutenzione

L'esempio seguente `register-task-with-maintenance-window` registra un'attività Step Functions con una finestra di manutenzione destinata a un'istanza.

```
aws ssm register-task-with-maintenance-window \
    --window-id "mw-1234d787d6EXAMPLE" \
```

```
    --targets Key=WindowTargetIds,Values=12347414-69c3-49f8-95b8-ed2dcEXAMPLE \
    --task-arn arn:aws:states:us-
east-1:111222333444:stateMachine:SSMTestStateMachine \
    --service-role-arn arn:aws:iam::111222333444:role/MaintenanceWindows \
    --task-type STEP_FUNCTIONS \
    --task-invocation-parameters '{"StepFunctions":{"Input":"{\"InstanceId\":
\"{{RESOURCE_ID}}\"}"}}' \
    --priority 0 \
    --max-concurrency 10 \
    --max-errors 5 \
    --name "Step_Functions_Example" \
    --description "My Step Functions Example"
```

Output:

```
{
    "WindowTaskId":"44444444-5555-6666-7777-88888888"
}
```

Per ulteriori informazioni, vedere Registrare un'attività con la finestra di manutenzione (AWS CLI) nella Guida per l'utente di AWS Systems Manager.

Esempio 5: come registrare un'attività utilizzando un ID di destinazione della finestra di manutenzione

L'esempio seguente `register-task-with-maintenance-window` registra un'attività utilizzando un ID di destinazione della finestra di manutenzione. L'ID di destinazione della finestra di manutenzione era nell'output del comando `aws ssm register-target-with-maintenance-window`. È inoltre possibile recuperarlo dall'output del comando `aws ssm describe-maintenance-window-targets`.

```
aws ssm register-task-with-maintenance-window \
    --targets "Key=WindowTargetIds,Values=350d44e6-28cc-44e2-951f-4b2c9EXAMPLE" \
    --task-arn "AWS-RunShellScript" \
    --service-role-arn "arn:aws:iam::111222333444:role/MaintenanceWindowsRole" \
    --window-id "mw-ab12cd34eEXAMPLE" \
    --task-type "RUN_COMMAND" \
    --task-parameters  "{\"commands\":{\"Values\":[\"df\"]}}" \
    --max-concurrency 1 \
    --max-errors 1 \
    --priority 10
```

Output:

```
{
    "WindowTaskId":"33344444-5555-6666-7777-88888888"
}
```

Per ulteriori informazioni, vedere Registrare un'attività con la finestra di manutenzione (AWS CLI) nella Guida per l'utente di AWS Systems Manager.

- Per i dettagli sull'API, vedere RegisterTaskWithMaintenanceWindowin AWS CLI Command Reference.

`remove-tags-from-resource`

Il seguente esempio di codice mostra come utilizzare`remove-tags-from-resource`.

AWS CLI

Come rimuovere un tag da una baseline delle patch

L'esempio seguente `remove-tags-from-resource` mostra come rimuovere i tag da una baseline delle patch.

```
aws ssm remove-tags-from-resource \
    --resource-type "PatchBaseline" \
    --resource-id "pb-0123456789abcdef0" \
    --tag-keys "Region"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, vedete Tagging AWS Resources nella Guida AWS generale.

- Per i dettagli sull'API, consulta RemoveTagsFromResource AWS CLICommand Reference.

`reset-service-setting`

Il seguente esempio di codice mostra come utilizzare`reset-service-setting`.

AWS CLI

Come ridefinire l'impostazione del servizio per il throughput dell'archivio parametri

L'esempio `reset-service-setting` seguente modifica l'impostazione del servizio per il throughput dell'archivio parametri nella Regione specificata, in modo da non utilizzare più un throughput maggiore.

```
aws ssm reset-service-setting \
    --setting-id arn:aws:ssm:us-east-1:123456789012:servicesetting/ssm/parameter-store/high-throughput-enabled
```

Output:

```
{
    "ServiceSetting": {
        "SettingId": "/ssm/parameter-store/high-throughput-enabled",
        "SettingValue": "false",
        "LastModifiedDate": 1555532818.578,
        "LastModifiedUser": "System",
        "ARN": "arn:aws:ssm:us-east-1:123456789012:servicesetting/ssm/parameter-store/high-throughput-enabled",
        "Status": "Default"
    }
}
```

Per ulteriori informazioni, consulta Incremento della capacità di throughput in Parameter Store nella Guida per l'utente di AWS Systems Manager.

- Per i dettagli sull'API, consulta ResetServiceSetting AWS CLICommand Reference.

`resume-session`

Il seguente esempio di codice mostra come utilizzare`resume-session`.

AWS CLI

Come riprendere una sessione di Session Manager

L'esempio `resume-session` seguente riprende l'esecuzione di una sessione di Session Manager con un'istanza dopo che quest'ultima è stata disconnessa. Si noti che questo comando interattivo richiede l'installazione del plug-in Session Manager sul computer client che effettua la chiamata.

```
aws ssm resume-session \
```

```
    --session-id Mary-Major-07a16060613c408b5
```

Output:

```
{
    "SessionId": "Mary-Major-07a16060613c408b5",
    "TokenValue":
 "AAEAAVbTGsaOnyvcUoNGqifbv5r/8lgxuQljCuY8qVcvOnoBAAAAAFxtd3jIXAFUUXGTJ7zF/
AWJPwDviOlF5p3dlAgrqVIVO6IEXhkHLz0/1gXKRKEME71E6TLOplLDJAMZ
+kREejkZu4c5AxMkrQjMF+gtHP1bYJKTwtHQd1wjulPLexO8SHl7g5R/
wekrj6WsDUpnEegFBfGftpAIz2GXQVfTJXKfkc5qepQ11C11DOIT2dozOqXgHwfQHfAKLErM5dWDZqKwyT1Z3iw7unQd
+ihfGa6MEJJ97Jmat/a2TspEnOjNn9Mvu5iwXIW2yCvWZrGUj+/
QI5Xr7s1XJBEnSKR54o4fN0GV9RWl0RZsZm1m1ki0JJtiwwgZ",
    "StreamUrl": "wss://ssmmessages.us-east-2.amazonaws.com/v1/data-channel/Mary-
Major-07a16060613c408b5?role=publish_subscribe"
}
```

Per ulteriori informazioni, vedere Installare il plug-in Session Manager per la AWS CLI nella Guida per l'utente di AWS Systems Manager.

- Per i dettagli sull'API, vedere ResumeSessionin AWS CLI Command Reference.

`send-automation-signal`

Il seguente esempio di codice mostra come utilizzare`send-automation-signal`.

AWS CLI

Come inviare un segnale a un'esecuzione di automazione

L'esempio `send-automation-signal` seguente invia un segnale di approvazione a un'esecuzione di automazione.

```
aws ssm send-automation-signal \
    --automation-execution-id 73c8eef8-f4ee-4a05-820c-e354fEXAMPLE \
    --signal-type "Approve"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Esecuzione di un flusso di lavoro di automazione con approvatori nella Guida per l'utente di AWS Systems Manager.

- Per i dettagli sull'API, consulta SendAutomationSignal AWS CLICommand Reference.

`send-command`

Il seguente esempio di codice mostra come utilizzare`send-command`.

AWS CLI

Esempio 1: eseguire un comando su una o più istanze remote

L'esempio seguente `send-command` esegue un comando `echo` su un'istanza di destinazione.

```
aws ssm send-command \
    --document-name "AWS-RunShellScript" \
    --parameters 'commands=["echo HelloWorld"]' \
    --targets "Key=instanceids,Values=i-1234567890abcdef0" \
    --comment "echo HelloWorld"
```

Output:

```
{
    "Command": {
        "CommandId": "92853adf-ba41-4cd6-9a88-142d1EXAMPLE",
        "DocumentName": "AWS-RunShellScript",
        "DocumentVersion": "",
        "Comment": "echo HelloWorld",
        "ExpiresAfter": 1550181014.717,
        "Parameters": {
            "commands": [
                "echo HelloWorld"
            ]
        },
        "InstanceIds": [
            "i-0f00f008a2dcbefe2"
        ],
        "Targets": [],
        "RequestedDateTime": 1550173814.717,
        "Status": "Pending",
        "StatusDetails": "Pending",
        "OutputS3BucketName": "",
        "OutputS3KeyPrefix": "",
        "MaxConcurrency": "50",
        "MaxErrors": "0",
```

```
        "TargetCount": 1,
        "CompletedCount": 0,
        "ErrorCount": 0,
        "DeliveryTimedOutCount": 0,
        "ServiceRole": "",
        "NotificationConfig": {
            "NotificationArn": "",
            "NotificationEvents": [],
            "NotificationType": ""
        },
        "CloudWatchOutputConfig": {
            "CloudWatchLogGroupName": "",
            "CloudWatchOutputEnabled": false
        }
    }
}
```

Per ulteriori informazioni, consulta Eseguire comandi utilizzando Systems Manager Run Command nella Guida per l'utente di AWS Systems Manager.

Esempio 2: come ottenere informazioni IP su un'istanza

L'esempio seguente `send-command` recupera le informazioni sull'IP di un'istanza.

```
aws ssm send-command \
    --instance-ids "i-1234567890abcdef0" \
    --document-name "AWS-RunShellScript" \
    --comment "IP config" \
    --parameters "commands=ifconfig"
```

Vedi l'esempio 1 per un output di esempio.

Per ulteriori informazioni, consulta Eseguire comandi utilizzando Systems Manager Run Command nella Guida per l'utente di AWS Systems Manager.

Esempio 3: eseguire un comando su istanze con tag specifici

L'esempio seguente `send-command` esegue un comando sulle istanze che hanno la chiave di tag "ENV" e il valore "Dev".

```
aws ssm send-command \
    --targets "Key=tag:ENV,Values=Dev" \
    --document-name "AWS-RunShellScript" \
```

```
    --parameters "commands=ifconfig"
```

Vedi l'esempio 1 per un output di esempio.

Per ulteriori informazioni, consulta Eseguire comandi utilizzando Systems Manager Run Command nella Guida per l'utente di AWS Systems Manager.

Esempio 4: come eseguire un comando che invia notifiche SNS

L'esempio seguente `send-command` esegue un comando che invia notifiche SNS per tutti gli eventi di notifica e il tipo di notifica `Command`.

```
aws ssm send-command \
    --instance-ids "i-1234567890abcdef0" \
    --document-name "AWS-RunShellScript" \
    --comment "IP config" \
    --parameters "commands=ifconfig" \
    --service-role-arn "arn:aws:iam::123456789012:role/SNS_Role" \
    --notification-config "NotificationArn=arn:aws:sns:us-
east-1:123456789012:SNSTopicName,NotificationEvents=All,NotificationType=Command"
```

Vedi l'esempio 1 per un output di esempio.

Per ulteriori informazioni, consulta Eseguire comandi utilizzando Systems Manager Run Command nella Guida per l'utente di AWS Systems Manager.

Esempio 5: per eseguire un comando che genera output su S3 e CloudWatch

L'`send-command`esempio seguente esegue un comando che invia i dettagli del comando a un bucket S3 e a un gruppo di log Logs. CloudWatch

```
aws ssm send-command \
    --instance-ids "i-1234567890abcdef0" \
    --document-name "AWS-RunShellScript" \
    --comment "IP config" \
    --parameters "commands=ifconfig" \
    --output-s3-bucket-name "s3-bucket-name" \
    --output-s3-key-prefix "runcommand" \
    --cloud-watch-output-
config "CloudWatchOutputEnabled=true,CloudWatchLogGroupName=CWLGroupName"
```

Vedi l'esempio 1 per un output di esempio.

Per ulteriori informazioni, consulta Eseguire comandi utilizzando Systems Manager Run Command nella Guida per l'utente di AWS Systems Manager.

Esempio 6: come eseguire comandi su più istanze con tag diversi

L'esempio seguente `send-command` esegue un comando su istanze con due chiavi e valori di tag diversi.

```
aws ssm send-command \
    --document-name "AWS-RunPowerShellScript" \
    --parameters commands=["echo helloWorld"] \
    --targets Key=tag:Env,Values=Dev Key=tag:Role,Values=WebServers
```

Vedi l'esempio 1 per un output di esempio.

Per ulteriori informazioni, consulta Eseguire comandi utilizzando Systems Manager Run Command nella Guida per l'utente di AWS Systems Manager.

Esempio 7: come indirizzare più istanze con la stessa chiave di tag

L'esempio seguente `send-command` esegue un comando su istanze che hanno la stessa chiave di tag ma con valori diversi.

```
aws ssm send-command \
    --document-name "AWS-RunPowerShellScript" \
    --parameters commands=["echo helloWorld"] \
    --targets Key=tag:Env,Values=Dev,Test
```

Vedi l'esempio 1 per un output di esempio.

Per ulteriori informazioni, consulta Eseguire comandi utilizzando Systems Manager Run Command nella Guida per l'utente di AWS Systems Manager.

Esempio 8: come eseguire un comando che utilizza un documento condiviso

L'esempio seguente `send-command` esegue un documento condiviso su un'istanza di destinazione.

```
aws ssm send-command \
    --document-name "arn:aws:ssm:us-east-1:123456789012:document/ExampleDocument" \
    --targets "Key=instanceids,Values=i-1234567890abcdef0"
```

Vedi l'esempio 1 per un output di esempio.

Per ulteriori informazioni, consulta Utilizzo di documenti condivisi SSM nella Guida per l'utente di AWS Systems Manager.

- Per i dettagli sull'API, consulta SendCommandCommand Reference.AWS CLI

`start-associations-once`

Il seguente esempio di codice mostra come utilizzare`start-associations-once`.

AWS CLI

Come eseguire un'associazione immediatamente e solo una volta

L'esempio `start-associations-once` seguente esegue l'associazione specificata immediatamente e solo una volta. Se il comando va a buon fine, non viene generato output.

```
aws ssm start-associations-once \
    --association-id "8dfe3659-4309-493a-8755-0123456789ab"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Visualizzazione della cronologia delle associazioni nella Guida per l'utente di AWS Systems Manager.

- Per i dettagli sull'API, consulta StartAssociationsOnce AWS CLICommand Reference.

`start-automation-execution`

Il seguente esempio di codice mostra come utilizzare`start-automation-execution`.

AWS CLI

Esempio 1: come eseguire un documento di automazione

L'esempio seguente `start-automation-execution` esegue un documento di automazione.

```
aws ssm start-automation-execution \
    --document-name "AWS-UpdateLinuxAmi" \
    --parameters "AutomationAssumeRole=arn:aws:iam::123456789012:role/
SSMAutomationRole,SourceAmiId=ami-EXAMPLE,IamInstanceProfileName=EC2InstanceRole"
```

Output:

```
{
  "AutomationExecutionId": "4105a4fc-f944-11e6-9d32-0a1b2EXAMPLE"
}
```

Per ulteriori informazioni, consulta Esecuzione manuale di un flusso di lavoro di automazione nella Guida per l'utente di AWS Systems Manager.

Esempio 2: come eseguire un documento di automazione condiviso

L'esempio seguente `start-automation-execution` esegue un documento di automazione condiviso.

```
aws ssm start-automation-execution \
    --document-name "arn:aws:ssm:us-east-1:123456789012:document/ExampleDocument"
```

Output:

```
{
  "AutomationExecutionId": "4105a4fc-f944-11e6-9d32-0a1b2EXAMPLE"
}
```

Per ulteriori informazioni, consulta Utilizzo di documenti condivisi SSM nella Guida per l'utente di AWS Systems Manager.

- Per i dettagli sull'API, consulta StartAutomationExecution AWS CLICommand Reference.

`start-change-request-execution`

Il seguente esempio di codice mostra come utilizzare`start-change-request-execution`.

AWS CLI

Esempio 1: come avviare una richiesta di modifica

L'esempio `start-change-request-execution` seguente avvia una richiesta di modifica con opzioni minime specificate.

```
aws ssm start-change-request-execution \
    --change-request-name MyChangeRequest \
```

```
    --document-name AWS-HelloWorldChangeTemplate \
    --runbooks '[{"DocumentName": "AWS-HelloWorld","Parameters":
 {"AutomationAssumeRole": ["arn:aws:iam:us-east-2:1112223233444:role/
MyChangeManagerAssumeRole"]}}]' \
    --parameters
 Approver="JohnDoe",ApproverType="IamUser",ApproverSnsTopicArn="arn:aws:sns:us-
east-2:1112223233444:MyNotificationTopic"
```

Output:

```
{
  "AutomationExecutionId": "9d32a4fc-f944-11e6-4105-0a1b2EXAMPLE"
}
```

Esempio 2: come avviare una richiesta di modifica utilizzando un file JSON esterno

L'esempio `start-automation-execution` seguente avvia una richiesta di modifica con più opzioni specificate in un file JSON.

```
aws ssm start-change-request-execution \
    --cli-input-json file://MyChangeRequest.json
```

Contenuto di `MyChangeRequest.json`:

```
{
    "ChangeRequestName": "MyChangeRequest",
    "DocumentName": "AWS-HelloWorldChangeTemplate",
    "DocumentVersion": "$DEFAULT",
    "ScheduledTime": "2021-12-30T03:00:00",
    "ScheduledEndTime": "2021-12-30T03:05:00",
    "Tags": [
        {
            "Key": "Purpose",
            "Value": "Testing"
        }
    ],
    "Parameters": {
        "Approver": [
            "JohnDoe"
        ],
        "ApproverType": [
            "IamUser"
```

```
        ],
        "ApproverSnsTopicArn": [
            "arn:aws:sns:us-east-2:111222333444;:MyNotificationTopic
        ]
    },
    "Runbooks": [
        {
            "DocumentName": "AWS-HelloWorld",
            "DocumentVersion": "1",
            "MaxConcurrency": "1",
            "MaxErrors": "1",
            "Parameters": {
                "AutomationAssumeRole": [
                    "arn:aws:iam::111222333444:role/MyChangeManagerAssumeRole"
                ]
            }
        }
    ],
    "ChangeDetails": "### Document Name: HelloWorldChangeTemplate\n\n## What does
 this document do?\nThis change template demonstrates the feature set available
 for creating change templates for Change Manager. This template starts a Runbook
 workflow for the Automation document called AWS-HelloWorld.\n\n## Input Parameters
\n* ApproverSnsTopicArn: (Required) Amazon Simple Notification Service ARN for
 approvers.\n* Approver: (Required) The name of the approver to send this request
 to.\n* ApproverType: (Required) The type of reviewer.\n  * Allowed Values: IamUser,
 IamGroup, IamRole, SSOGroup, SSOUser\n\n## Output Parameters\nThis document has no
 outputs \n"
}
```

Output:

```
{
  "AutomationExecutionId": "9d32a4fc-f944-11e6-4105-0a1b2EXAMPLE"
}
```

Per ulteriori informazioni, consulta Creazione di richieste di modifica nella Guida per l'utente di AWS Systems Manager.

- Per i dettagli sull'API, consulta StartChangeRequestExecution AWS CLICommand Reference.

`start-session`

Il seguente esempio di codice mostra come utilizzare`start-session`.

AWS CLI

Esempio 1: come avviare una sessione di Session Manager

Questo esempio `start-session` stabilisce una connessione con un'istanza per una sessione di Session Manager. Si noti che questo comando interattivo richiede l'installazione del plug-in Session Manager sul computer client che effettua la chiamata.

```
aws ssm start-session \
    --target "i-1234567890abcdef0"
```

Output:

```
Starting session with SessionId: Jane-Roe-07a16060613c408b5
```

Esempio 2: come avviare una sessione di Session Manager utilizzando SSH

Questo esempio `start-session` stabilisce una connessione con un'istanza per una sessione di Session Manager utilizzando SSH. Si noti che questo comando interattivo richiede l'installazione del plug-in Session Manager sul computer client che effettua la chiamata e che il comando utilizza l'utente predefinito sull'istanza, ad esempio `ec2-user` per le EC2 istanze per Linux.

```
ssh -i /path/my-key-pair.pem ec2-user@i-02573cafcfEXAMPLE
```

Output:

```
Starting session with SessionId: ec2-user-07a16060613c408b5
```

Per ulteriori informazioni, vedere Avvio di una sessione e installazione del plug-in Session Manager per la AWS CLI nella Guida per l'utente di AWS Systems Manager.

- Per i dettagli sull'API, vedere StartSessionin AWS CLI Command Reference.

`stop-automation-execution`

Il seguente esempio di codice mostra come utilizzare`stop-automation-execution`.

AWS CLI

Come arrestare l'esecuzione di un'automazione

L'esempio seguente `stop-automation-execution` interrompe un documento di automazione.

```
aws ssm stop-automation-execution
    --automation-execution-id "4105a4fc-f944-11e6-9d32-0a1b2EXAMPLE"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Esecuzione manuale di un flusso di lavoro di automazione nella Guida per l'utente di AWS Systems Manager.

- Per i dettagli sull'API, consulta StopAutomationExecution AWS CLICommand Reference.

`terminate-session`

Il seguente esempio di codice mostra come utilizzare`terminate-session`.

AWS CLI

Come terminare una sessione di Session Manager

L'esempio `terminate-session` seguente termina definitivamente una sessione creata dall'utente "Shirley-Rodriguez" e chiude la connessione dati tra il client Session Manager e l'agente SSM sull'istanza.

```
aws ssm terminate-session \
    --session-id "Shirley-Rodriguez-07a16060613c408b5"
```

Output:

```
{
    "SessionId": "Shirley-Rodriguez-07a16060613c408b5"
}
```

Per ulteriori informazioni, consulta Terminazione di una sessione nella Guida per l'utente di AWS Systems Manager.

- Per i dettagli sull'API, consulta TerminateSession AWS CLICommand Reference.

`unlabel-parameter-version`

Il seguente esempio di codice mostra come utilizzare`unlabel-parameter-version`.

AWS CLI

Come eliminare etichette dei parametri

L'esempio `unlabel-parameter-version` seguente elimina le etichette specificate dalla versione del parametro specificato.

```
aws ssm unlabel-parameter-version \
    --name "parameterName" \
    --parameter-version "version" \
    --labels "label_1" "label_2" "label_3"
```

Output:

```
{
    "RemovedLabels": [
        "label_1"
        "label_2"
        "label_3"
    ],
    "InvalidLabels": []
}
```

Per ulteriori informazioni, vedere Delete parameter labels (AWS CLI) nella AWS Systems Manager User Guide.

- Per i dettagli sull'API, vedere UnlabelParameterVersionin AWS CLI Command Reference.

`update-association-status`

Il seguente esempio di codice mostra come utilizzareu`pdate-association-status`.

AWS CLI

Come aggiornare lo stato dell'associazione

L'esempio seguente `update-association-status` aggiorna lo stato dell'associazione tra un'istanza e un documento.

```
aws ssm update-association-status \
    --name "AWS-UpdateSSMAgent" \
```

```
    --instance-id "i-1234567890abcdef0" \
    --association-
status "Date=1424421071.939,Name=Pending,Message=temp_status_change,AdditionalInfo=Additiona
Config-Needed"
```

Output:

```
{
    "AssociationDescription": {
        "Name": "AWS-UpdateSSMAgent",
        "InstanceId": "i-1234567890abcdef0",
        "AssociationVersion": "1",
        "Date": 1550507529.604,
        "LastUpdateAssociationDate": 1550507806.974,
        "Status": {
            "Date": 1424421071.0,
            "Name": "Pending",
            "Message": "temp_status_change",
            "AdditionalInfo": "Additional-Config-Needed"
        },
        "Overview": {
            "Status": "Success",
            "AssociationStatusAggregatedCount": {
                "Success": 1
            }
        },
        "DocumentVersion": "$DEFAULT",
        "AssociationId": "8dfe3659-4309-493a-8755-0123456789ab",
        "Targets": [
            {
                "Key": "InstanceIds",
                "Values": [
                    "i-1234567890abcdef0"
                ]
            }
        ],
        "LastExecutionDate": 1550507808.0,
        "LastSuccessfulExecutionDate": 1550507808.0
    }
}
```

Per ulteriori informazioni, consulta Utilizzo delle associazioni in Systems Manager nella AWS Guida per l'utente di Systems Manager.

- Per i dettagli sull'API, consulta UpdateAssociationStatus AWS CLICommand Reference.

`update-association`

Il seguente esempio di codice mostra come utilizzareu`update-association`.

AWS CLI

Esempio 1: come aggiornare un'associazione di documenti

L'esempio seguente `update-association` aggiorna l'associazione a una nuova versione del documento.

```
aws ssm update-association \
    --association-id "8dfe3659-4309-493a-8755-0123456789ab" \
    --document-version "\$LATEST"
```

Output:

```
{
    "AssociationDescription": {
        "Name": "AWS-UpdateSSMAgent",
        "AssociationVersion": "2",
        "Date": 1550508093.293,
        "LastUpdateAssociationDate": 1550508106.596,
        "Overview": {
            "Status": "Pending",
            "DetailedStatus": "Creating"
        },
        "DocumentVersion": "$LATEST",
        "AssociationId": "8dfe3659-4309-493a-8755-0123456789ab",
        "Targets": [
            {
                "Key": "tag:Name",
                "Values": [
                    "Linux"
                ]
            }
        ],
        "LastExecutionDate": 1550508094.879,
        "LastSuccessfulExecutionDate": 1550508094.879
    }
```

```
}
```

Per ulteriori informazioni, consulta Modifica e creazione di una nuova versione di un'associazione nella Guida per l'utente di AWS Systems Manager.

Esempio 2: come aggiornare l'espressione di pianificazione di un'associazione

L'esempio seguente `update-association` aggiorna l'espressione di pianificazione per l'associazione specificata.

```
aws ssm update-association \
    --association-id "8dfe3659-4309-493a-8755-0123456789ab" \
    --schedule-expression "cron(0 0 0/4 1/1 * ? *)"
```

Output:

```
{
    "AssociationDescription": {
        "Name": "AWS-HelloWorld",
        "AssociationVersion": "2",
        "Date": "2021-02-08T13:54:19.203000-08:00",
        "LastUpdateAssociationDate": "2021-06-29T11:51:07.933000-07:00",
        "Overview": {
            "Status": "Pending",
            "DetailedStatus": "Creating"
        },
        "DocumentVersion": "$DEFAULT",
        "AssociationId": "8dfe3659-4309-493a-8755-0123456789ab",
        "Targets": [
            {
                "Key": "aws:NoOpAutomationTag",
                "Values": [
                    "AWS-NoOpAutomationTarget-Value"
                ]
            }
        ],
        "ScheduleExpression": "cron(0 0 0/4 1/1 * ? *)",
        "LastExecutionDate": "2021-06-26T19:00:48.110000-07:00",
        "ApplyOnlyAtCronInterval": false
    }
}
```

Per ulteriori informazioni, consulta Modifica e creazione di una nuova versione di un'associazione nella Guida per l'utente di AWS Systems Manager.

- Per i dettagli sull'API, consulta UpdateAssociation AWS CLICommand Reference.

`update-document-default-version`

Il seguente esempio di codice mostra come utilizzare`update-document-default-version`.

AWS CLI

Come aggiornare la versione predefinita di un documento

L'esempio seguente `update-document-default-version` aggiorna la versione predefinita di un documento Systems Manager.

```
aws ssm update-document-default-version \
    --name "Example" \
    --document-version "2"
```

Output:

```
{
    "Description": {
        "Name": "Example",
        "DefaultVersion": "2"
    }
}
```

Per ulteriori informazioni, consulta Scrittura del contenuto del documento SSM nella Guida per l'utente di AWS Systems Manager.

- Per i dettagli sull'API, consulta UpdateDocumentDefaultVersion AWS CLICommand Reference.

`update-document-metadata`

Il seguente esempio di codice mostra come utilizzare`update-document-metadata`.

AWS CLI

Esempio: come approvare la versione più recente di un modello di modifica

Il comando `update-document-metadata` seguente fornisce un'approvazione per la versione più recente di un modello di modifica che è stato inviato per la revisione.

```
aws ssm update-document-metadata \
    --name MyChangeManagerTemplate \
    --document-reviews 'Action=Approve,Comment=[{Type=Comment,Content=Approved!}]'
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Revisione e approvazione o rifiuto dei modelli di modifica nella Guida per l'utente di AWS Systems Manager.

- Per i dettagli sull'API, consulta UpdateDocumentMetadata AWS CLICommand Reference.

`update-document`

Il seguente esempio di codice mostra come utilizzare`update-document`.

AWS CLI

Come creare una nuova versione di un documento

L'esempio seguente `update-document` crea una nuova versione di un documento quando viene eseguito su un computer Windows. Il documento specificato da `--document` deve essere in formato JSON. Nota che `file://` deve essere referenziato seguito dal percorso del file di contenuto. A causa di $ all'inizio del parametro `--document-version`, in Windows è necessario racchiudere il valore tra virgolette doppie. Su Linux, macOS o su PowerShell richiesta, è necessario racchiudere il valore tra virgolette singole.

Versione Windows:

```
aws ssm update-document \
    --name "RunShellScript" \
    --content "file://RunShellScript.json" \
    --document-version "$LATEST"
```

Versione Linux/Mac:

```
aws ssm update-document \
    --name "RunShellScript" \
```

```
    --content "file://RunShellScript.json" \
    --document-version '$LATEST'
```

Output:

```
{
  "DocumentDescription": {
      "Status": "Updating",
      "Hash": "f775e5df4904c6fa46686c4722fae9de1950dace25cd9608ff8d622046b68d9b",
      "Name": "RunShellScript",
      "Parameters": [
          {
              "Type": "StringList",
              "Name": "commands",
              "Description": "(Required) Specify a shell script or a command to
 run."
          }
      ],
      "DocumentType": "Command",
      "PlatformTypes": [
          "Linux"
      ],
      "DocumentVersion": "2",
      "HashType": "Sha256",
      "CreatedDate": 1487899655.152,
      "Owner": "809632081692",
      "SchemaVersion": "2.0",
      "DefaultVersion": "1",
      "LatestVersion": "2",
      "Description": "Run an updated script"
  }
}
```

- Per i dettagli sull'API, consulta Command UpdateDocumentReference AWS CLI .

`update-maintenance-window-target`

Il seguente esempio di codice mostra come utilizzare`update-maintenance-window-target`.

AWS CLI

Come aggiornare la destinazione di una finestra di manutenzione

L'esempio `update-maintenance-window-target` seguente aggiorna il nome di una finestra di manutenzione.

```
aws ssm update-maintenance-window-target \
    --window-id "mw-0c5ed765acEXAMPLE" \
    --window-target-id "57e8344e-fe64-4023-8191-6bf05EXAMPLE" \
    --name "NewName" \
    --no-replace
```

Output:

```
{
    "Description": "",
    "OwnerInformation": "",
    "WindowTargetId": "57e8344e-fe64-4023-8191-6bf05EXAMPLE",
    "WindowId": "mw-0c5ed765acEXAMPLE",
    "Targets": [
        {
            "Values": [
                "i-1234567890EXAMPLE"
            ],
            "Key": "InstanceIds"
        }
    ],
    "Name": "NewName"
}
```

Per ulteriori informazioni, vedere Update a Maintenance Window (AWS CLI) nella AWS Systems Manager User Guide.

- Per i dettagli sull'API, vedere UpdateMaintenanceWindowTargetin AWS CLI Command Reference.

`update-maintenance-window-task`

Il seguente esempio di codice mostra come utilizzare`update-maintenance-window-task`.

AWS CLI

Come aggiornare un'attività in una finestra di manutenzione

L'esempio `update-maintenance-window-task` seguente aggiorna il ruolo di servizio per un'attività della finestra di manutenzione.

```
aws ssm update-maintenance-window-task \
    --window-id "mw-0c5ed765acEXAMPLE" \
    --window-task-id "23d3809e-9fbe-4ddf-b41a-b49d7EXAMPLE" \
    --service-role-arn "arn:aws:iam::111222333444:role/aws-service-role/
ssm.amazonaws.com/AWSServiceRoleForAmazonSSM"
```

Output:

```
{
    "ServiceRoleArn": "arn:aws:iam::111222333444:role/aws-service-role/
ssm.amazonaws.com/AWSServiceRoleForAmazonSSM",
    "MaxErrors": "1",
    "TaskArn": "AWS-UpdateEC2Config",
    "MaxConcurrency": "1",
    "WindowTaskId": "23d3809e-9fbe-4ddf-b41a-b49d7EXAMPLE",
    "TaskParameters": {},
    "Priority": 1,
    "TaskInvocationParameters": {
        "RunCommand": {
            "TimeoutSeconds": 600,
            "Parameters": {
                "allowDowngrade": [
                    "false"
                ]
            }
        }
    },
    "WindowId": "mw-0c5ed765acEXAMPLE",
    "Description": "UpdateEC2Config",
    "Targets": [
        {
            "Values": [
                "57e8344e-fe64-4023-8191-6bf05EXAMPLE"
            ],
            "Key": "WindowTargetIds"
        }
    ],
    "Name": "UpdateEC2Config"
}
```

Per ulteriori informazioni, vedere Update a Maintenance Window (AWS CLI) nella AWS Systems Manager User Guide.

- Per i dettagli sull'API, vedere UpdateMaintenanceWindowTaskin AWS CLI Command Reference.

`update-maintenance-window`

Il seguente esempio di codice mostra come utilizzareupdate-maintenance-window.

AWS CLI

Esempio 1: come aggiornare una finestra di manutenzione

L'esempio seguente update-maintenance-window aggiorna il nome di una finestra di manutenzione.

```
aws ssm update-maintenance-window \
    --window-id "mw-1a2b3c4d5e6f7g8h9" \
    --name "My-Renamed-MW"
```

Output:

```
{
    "Cutoff": 1,
    "Name": "My-Renamed-MW",
    "Schedule": "cron(0 16 ? * TUE *)",
    "Enabled": true,
    "AllowUnassociatedTargets": true,
    "WindowId": "mw-1a2b3c4d5e6f7g8h9",
    "Duration": 4
}
```

Esempio 2: come disabilitare una finestra di manutenzione

L'esempio seguente update-maintenance-window disattiva una finestra di manutenzione.

```
aws ssm update-maintenance-window \
    --window-id "mw-1a2b3c4d5e6f7g8h9" \
    --no-enabled
```

Esempio 3: come abilitare una finestra di manutenzione

L'esempio seguente `update-maintenance-window` attiva una finestra di manutenzione.

```
aws ssm update-maintenance-window \
    --window-id "mw-1a2b3c4d5e6f7g8h9" \
    --enabled
```

Per ulteriori informazioni, vedere Update a Maintenance Window (AWS CLI) nella AWS Systems Manager User Guide.

- Per i dettagli sull'API, vedere UpdateMaintenanceWindowin AWS CLI Command Reference.

`update-managed-instance-role`

Il seguente esempio di codice mostra come utilizzare`update-managed-instance-role`.

AWS CLI

Come aggiornare il ruolo IAM di un'istanza gestita

L'esempio seguente `update-managed-instance-role` aggiorna il profilo dell'istanza IAM di un'istanza gestita.

```
aws ssm update-managed-instance-role \
    --instance-id "mi-08ab247cdfEXAMPLE" \
    --iam-role "ExampleRole"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Passaggio 4: creazione di un profilo dell'istanza IAM per Systems Manager nella Guida per l'utente di AWS Systems Manager.

- Per i dettagli sull'API, consulta UpdateManagedInstanceRole AWS CLICommand Reference.

`update-ops-item`

Il seguente esempio di codice mostra come utilizzare`update-ops-item`.

AWS CLI

Per aggiornare un OpsItem

L'`update-ops-item`esempio seguente aggiorna la descrizione, la priorità e la categoria di un OpsItem. Inoltre, il comando specifica un argomento SNS in cui vengono inviate le notifiche quando questo OpsItem viene modificato o modificato.

```
aws ssm update-ops-item \
    --ops-item-id "oi-287b5EXAMPLE" \
    --description "Primary OpsItem for failover event 2020-01-01-fh398yf" \
    --priority 2 \
    --category "Security" \
    --notifications "Arn=arn:aws:sns:us-east-2:111222333444:my-us-east-2-topic"
```

Output:

```
This command produces no output.
```

Per ulteriori informazioni, vedere Working with OpsItems nella AWS Systems Manager User Guide.

- Per i dettagli sull'API, vedere UpdateOpsItemin AWS CLI Command Reference.

`update-patch-baseline`

Il seguente esempio di codice mostra come utilizzareu`pdate-patch-baseline`.

AWS CLI

Esempio 1: come aggiornare una baseline delle patch

L'esempio seguente `update-patch-baseline` aggiunge le due patch specificate come rifiutate e una patch approvata alla baseline delle patch specificata.

```
aws ssm update-patch-baseline \
        --baseline-id "pb-0123456789abcdef0" \
        --rejected-patches "KB2032276" "MS10-048" \
        --approved-patches "KB2124261"
```

Output:

```
{
```

```
    "BaselineId": "pb-0123456789abcdef0",
    "Name": "WindowsPatching",
    "OperatingSystem": "WINDOWS",
    "GlobalFilters": {
        "PatchFilters": []
    },
    "ApprovalRules": {
        "PatchRules": [
            {
                "PatchFilterGroup": {
                    "PatchFilters": [
                        {
                            "Key": "PRODUCT",
                            "Values": [
                                "WindowsServer2016"
                            ]
                        }
                    ]
                },
                "ComplianceLevel": "CRITICAL",
                "ApproveAfterDays": 0,
                "EnableNonSecurity": false
            }
        ]
    },
    "ApprovedPatches": [
        "KB2124261"
    ],
    "ApprovedPatchesComplianceLevel": "UNSPECIFIED",
    "ApprovedPatchesEnableNonSecurity": false,
    "RejectedPatches": [
        "KB2032276",
        "MS10-048"
    ],
    "RejectedPatchesAction": "ALLOW_AS_DEPENDENCY",
    "CreatedDate": 1550244180.465,
    "ModifiedDate": 1550244180.465,
    "Description": "Patches for Windows Servers",
    "Sources": []
}
```


Esempio 2: come rinominare una baseline delle patch

L'esempio seguente `update-patch-baseline` rinomina la linea di baseline delle patch.

```
aws ssm update-patch-baseline \
    --baseline-id "pb-0713accee01234567" \
    --name "Windows-Server-2012-R2-Important-and-Critical-Security-Updates"
```

Per ulteriori informazioni, vedere Update or Delete a Patch Baseline < https://docs.aws.amazon.com/systems-manager/ latest/userguide/patch - baseline-update-or-delete .html>`__ nella Systems AWS Manager User Guide.

- Per i dettagli sull'API, vedere in Command Reference. UpdatePatchBaselineAWS CLI

`update-resource-data-sync`

Il seguente esempio di codice mostra come utilizzare`update-resource-data-sync`.

AWS CLI

Come aggiornare una sincronizzazione dati risorsa

L'`update-resource-data-sync`esempio seguente aggiorna la sincronizzazione dei dati di una SyncFromSource risorsa.

```
aws ssm update-resource-data-sync \
    --sync-name exampleSync \
    --sync-type SyncFromSource \
    --sync-source '{"SourceType":"SingleAccountMultiRegions", "SourceRegions":["us-east-1", "us-west-2"]}'
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Configurazione di Systems Manager Explorer per visualizzare dati da più account e regioni nella Guida per l'utente di AWS Systems Manager.

- Per i dettagli sull'API, vedere UpdateResourceDataSyncin AWS CLI Command Reference.

`update-service-setting`

Il seguente esempio di codice mostra come utilizzare`update-service-setting`.

AWS CLI

Come aggiornare l'impostazione del servizio per il throughput dell'archivio parametri

L'esempio `update-service-setting` seguente aggiorna l'impostazione del servizio corrente per il throughput dell'archivio parametri nella Regione specificata in modo da utilizzare un throughput maggiore.

```
aws ssm update-service-setting \
    --setting-id arn:aws:ssm:us-east-1:123456789012:servicesetting/ssm/parameter-store/high-throughput-enabled \
    --setting-value true
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Incremento della capacità di throughput in Parameter Store nella Guida per l'utente di AWS Systems Manager.

- Per i dettagli sull'API, consulta UpdateServiceSetting AWS CLICommand Reference.

Esempi di utilizzo di Amazon Textract AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando Amazon Textract. AWS Command Line Interface

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`analyze-document`

Il seguente esempio di codice mostra come usare`analyze-document`.

AWS CLI

Come analizzare il testo in un documento

L'esempio `analyze-document` seguente mostra come analizza il testo di un documento.

Linux/macOS:

```
aws textract analyze-document \
    --document '{"S3Object":{"Bucket":"bucket","Name":"document"}}' \
    --feature-types '["TABLES","FORMS"]'
```

Windows:

```
aws textract analyze-document \
    --document "{\"S3Object\":{\"Bucket\":\"bucket\",\"Name\":\"document\"}}" \
    --feature-types "[\"TABLES\",\"FORMS\"]" \
    --region region-name
```

Output:

```
{
    "Blocks": [
        {
            "Geometry": {
                "BoundingBox": {
                    "Width": 1.0,
                    "Top": 0.0,
                    "Left": 0.0,
                    "Height": 1.0
                },
                "Polygon": [
                    {
                        "Y": 0.0,
                        "X": 0.0
                    },
                    {
                        "Y": 0.0,
                        "X": 1.0
                    },
                    {
                        "Y": 1.0,
                        "X": 1.0
                    },
                    {
                        "Y": 1.0,
                        "X": 0.0
```

```
                    }
                ]
            },
            "Relationships": [
                {
                    "Type": "CHILD",
                    "Ids": [
                        "87586964-d50d-43e2-ace5-8a890657b9a0",
                        "a1e72126-21d9-44f4-a8d6-5c385f9002ba",
                        "e889d012-8a6b-4d2e-b7cd-7a8b327d876a"
                    ]
                }
            ],
            "BlockType": "PAGE",
            "Id": "c2227f12-b25d-4e1f-baea-1ee180d926b2"
        }
    ],
    "DocumentMetadata": {
        "Pages": 1
    }
}
```

Per ulteriori informazioni, consulta Analisi del testo dei documenti con Amazon Textract nella Guida per gli sviluppatori di Amazon Textract.

- Per i dettagli sull'API, consulta AnalyzeDocument AWS CLICommand Reference.

`detect-document-text`

Il seguente esempio di codice mostra come utilizzare`detect-document-text`.

AWS CLI

Come rilevare il testo in un documento

L'esempio `detect-document-text` seguente mostra come rileva il testo di un documento.

Linux/macOS:

```
aws textract detect-document-text \
    --document '{"S3Object":{"Bucket":"bucket","Name":"document"}}'
```

Windows:

```
aws textract detect-document-text \
    --document "{\"S3Object\":{\"Bucket\":\"bucket\",\"Name\":\"document\"}}" \
    --region region-name
```

Output:

```
{
    "Blocks": [
        {
            "Geometry": {
                "BoundingBox": {
                    "Width": 1.0,
                    "Top": 0.0,
                    "Left": 0.0,
                    "Height": 1.0
                },
                "Polygon": [
                    {
                        "Y": 0.0,
                        "X": 0.0
                    },
                    {
                        "Y": 0.0,
                        "X": 1.0
                    },
                    {
                        "Y": 1.0,
                        "X": 1.0
                    },
                    {
                        "Y": 1.0,
                        "X": 0.0
                    }
                ]
            },
            "Relationships": [
                {
                    "Type": "CHILD",
                    "Ids": [
                        "896a9f10-9e70-4412-81ce-49ead73ed881",
                        "0da18623-dc4c-463d-a3d1-9ac050e9e720",
                        "167338d7-d38c-4760-91f1-79a8ec457bb2"
                    ]
```

```
                }
            ],
            "BlockType": "PAGE",
            "Id": "21f0535e-60d5-4bc7-adf2-c05dd851fa25"
        },
        {
            "Relationships": [
                {
                    "Type": "CHILD",
                    "Ids": [
                        "62490c26-37ea-49fa-8034-7a9ff9369c9c",
                        "1e4f3f21-05bd-4da9-ba10-15d01e66604c"
                    ]
                }
            ],
            "Confidence": 89.11581420898438,
            "Geometry": {
                "BoundingBox": {
                    "Width": 0.33642634749412537,
                    "Top": 0.17169663310050964,
                    "Left": 0.13885067403316498,
                    "Height": 0.49159330129623413
                },
                "Polygon": [
                    {
                        "Y": 0.17169663310050964,
                        "X": 0.13885067403316498
                    },
                    {
                        "Y": 0.17169663310050964,
                        "X": 0.47527703642845154
                    },
                    {
                        "Y": 0.6632899641990662,
                        "X": 0.47527703642845154
                    },
                    {
                        "Y": 0.6632899641990662,
                        "X": 0.13885067403316498
                    }
                ]
            },
            "Text": "He llo,",
            "BlockType": "LINE",
```

```
            "Id": "896a9f10-9e70-4412-81ce-49ead73ed881"
        },
        {
            "Relationships": [
                {
                    "Type": "CHILD",
                    "Ids": [
                        "19b28058-9516-4352-b929-64d7cef29daf"
                    ]
                }
            ],
            "Confidence": 85.5694351196289,
            "Geometry": {
                "BoundingBox": {
                    "Width": 0.33182239532470703,
                    "Top": 0.23131252825260162,
                    "Left": 0.5091826915740967,
                    "Height": 0.3766750991344452
                },
                "Polygon": [
                    {
                        "Y": 0.23131252825260162,
                        "X": 0.5091826915740967
                    },
                    {
                        "Y": 0.23131252825260162,
                        "X": 0.8410050868988037
                    },
                    {
                        "Y": 0.607987642288208,
                        "X": 0.8410050868988037
                    },
                    {
                        "Y": 0.607987642288208,
                        "X": 0.5091826915740967
                    }
                ]
            },
            "Text": "worlc",
            "BlockType": "LINE",
            "Id": "0da18623-dc4c-463d-a3d1-9ac050e9e720"
        }
    ],
    "DocumentMetadata": {
```

```
        "Pages": 1
    }
}
```

Per ulteriori informazioni, consulta Rilevamento del testo dei documenti con Amazon Textract nella Guida per gli sviluppatori di Amazon Textract.

- Per i dettagli sull'API, consulta DetectDocumentText AWS CLICommand Reference.

`get-document-analysis`

Il seguente esempio di codice mostra come utilizzare`get-document-analysis`.

AWS CLI

Come ottenere i risultati dell'analisi asincrona del testo di un documento composto da più pagine

L'esempio `get-document-analysis` seguente mostra come ottiene i risultati dell'analisi asincrona del testo di un documento composto da più pagine.

```
aws textract get-document-analysis \
    --job-id df7cf32ebbd2a5de113535fcf4d921926a701b09b4e7d089f3aebadb41e0712b \
    --max-results 1000
```

Output:

```
{
    "Blocks": [
        {
            "Geometry": {
                "BoundingBox": {
                    "Width": 1.0,
                    "Top": 0.0,
                    "Left": 0.0,
                    "Height": 1.0
                },
                "Polygon": [
                    {
                        "Y": 0.0,
                        "X": 0.0
                    },
                    {
                        "Y": 0.0,
```

```
                        "X": 1.0
                    },
                    {
                        "Y": 1.0,
                        "X": 1.0
                    },
                    {
                        "Y": 1.0,
                        "X": 0.0
                    }
                ]
            },
            "Relationships": [
                {
                    "Type": "CHILD",
                    "Ids": [
                        "75966e64-81c2-4540-9649-d66ec341cd8f",
                        "bb099c24-8282-464c-a179-8a9fa0a057f0",
                        "5ebf522d-f9e4-4dc7-bfae-a288dc094595"
                    ]
                }
            ],
            "BlockType": "PAGE",
            "Id": "247c28ee-b63d-4aeb-9af0-5f7ea8ba109e",
            "Page": 1
        }
    ],
    "NextToken": "cY1W3eTFvoB0cH7YrKVudI4Gb0H8J0xAYLo8xI/JunCIPWCthaKQ+07n/
ElyutsSy0+1VOImoTRmP1zw4P0RFtaeV9Bzhnfedpx1YqwB4xaGDA==",
    "DocumentMetadata": {
        "Pages": 1
    },
    "JobStatus": "SUCCEEDED"
}
```

Per ulteriori informazioni, consulta Rilevamento e analisi del testo in documenti con più pagine nella Guida per gli sviluppatori di Amazon Textract.

- Per i dettagli sull'API, consulta GetDocumentAnalysis AWS CLICommand Reference.

`get-document-text-detection`

Il seguente esempio di codice mostra come utilizzare`get-document-text-detection`.

AWS CLI

Come ottenere i risultati dell'analisi asincrona del testo di un documento composto da più pagine

L'esempio `get-document-text-detection` seguente mostra come ottenere i risultati dell'analisi asincrona del testo di un documento composto da più pagine.

```
aws textract get-document-text-detection \
    --job-id 57849a3dc627d4df74123dca269d69f7b89329c870c65bb16c9fd63409d200b9 \
    --max-results 1000
```

Output

```
{
    "Blocks": [
        {
            "Geometry": {
                "BoundingBox": {
                    "Width": 1.0,
                    "Top": 0.0,
                    "Left": 0.0,
                    "Height": 1.0
                },
                "Polygon": [
                    {
                        "Y": 0.0,
                        "X": 0.0
                    },
                    {
                        "Y": 0.0,
                        "X": 1.0
                    },
                    {
                        "Y": 1.0,
                        "X": 1.0
                    },
                    {
                        "Y": 1.0,
                        "X": 0.0
                    }
                ]
            },
            "Relationships": [
```

```
                {
                    "Type": "CHILD",
                    "Ids": [
                        "1b926a34-0357-407b-ac8f-ec473160c6a9",
                        "0c35dc17-3605-4c9d-af1a-d9451059df51",
                        "dea3db8a-52c2-41c0-b50c-81f66f4aa758"
                    ]
                }
            ],
            "BlockType": "PAGE",
            "Id": "84671a5e-8c99-43be-a9d1-6838965da33e",
            "Page": 1
        }
    ],
    "NextToken": "GcqyoAJuZwujOT35EN4LCI3EUzMtiLq3nKyFFHvU5q1SaIdEBcSty+njNgoWwuMP/
muqc96S4o5NzDqehhXvhkodMyVO5OJGyms5lsrCxibWJw==",
    "DocumentMetadata": {
        "Pages": 1
    },
    "JobStatus": "SUCCEEDED"
}
```

Per ulteriori informazioni, consulta Rilevamento e analisi del testo in documenti con più pagine nella Guida per gli sviluppatori di Amazon Textract.

- Per i dettagli sull'API, consulta GetDocumentTextDetection AWS CLICommand Reference.

`start-document-analysis`

Il seguente esempio di codice mostra come utilizzare`start-document-analysis`.

AWS CLI

Come iniziare ad analizzare il testo in un documento composto da più pagine

L'esempio `start-document-analysis` seguente mostra come avviare l'analisi asincrona del testo in un documento composto da più pagine.

Linux/macOS:

```
aws textract start-document-analysis \
    --document-location '{"S3Object":{"Bucket":"bucket","Name":"document"}}' \
    --feature-types '["TABLES","FORMS"]' \
```

```
    --notification-channel "SNSTopicArn=arn:snsTopic,RoleArn=roleArn"
```

Windows:

```
aws textract start-document-analysis \
    --document-location "{\"S3Object\":{\"Bucket\":\"bucket\",\"Name\":\"document
\"}}" \
    --feature-types "[\"TABLES\", \"FORMS\"]" \
    --region region-name \
    --notification-channel "SNSTopicArn=arn:snsTopic,RoleArn=roleArn"
```

Output:

```
{
    "JobId": "df7cf32ebbd2a5de113535fcf4d921926a701b09b4e7d089f3aebadb41e0712b"
}
```

Per ulteriori informazioni, consulta Rilevamento e analisi del testo in documenti con più pagine nella Guida per gli sviluppatori di Amazon Textract.

- Per i dettagli sull'API, consulta StartDocumentAnalysis AWS CLICommand Reference.

`start-document-text-detection`

Il seguente esempio di codice mostra come utilizzare`start-document-text-detection`.

AWS CLI

Come iniziare a rilevare il testo in un documento composto da più pagine

L'esempio `start-document-text-detection` seguente mostra come avviare il rilevamento asincrono del testo in un documento composto da più pagine.

Linux/macOS:

```
aws textract start-document-text-detection \
        --document-location '{"S3Object":{"Bucket":"bucket","Name":"document"}}' \
        --notification-channel "SNSTopicArn=arn:snsTopic,RoleArn=roleARN"
```

Windows:

```
aws textract start-document-text-detection \
```

```
    --document-location "{\"S3Object\":{\"Bucket\":\"bucket\",\"Name\":\"document
\"}}" \
    --region region-name \
    --notification-channel "SNSTopicArn=arn:snsTopic,RoleArn=roleArn"
```

Output:

```
{
    "JobId": "57849a3dc627d4df74123dca269d69f7b89329c870c65bb16c9fd63409d200b9"
}
```

Per ulteriori informazioni, consulta Rilevamento e analisi del testo in documenti con più pagine nella Guida per gli sviluppatori di Amazon Textract.

- Per i dettagli sull'API, consulta StartDocumentTextDetection AWS CLICommand Reference.

Esempi di Amazon Transcribe utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando Amazon Transcribe. AWS Command Line Interface

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`create-language-model`

Il seguente esempio di codice mostra come usare. `create-language-model`

AWS CLI

Esempio 1: come creare un modello linguistico personalizzato utilizzando i dati di addestramento e di ottimizzazione.

L'esempio `create-language-model` seguente crea un modello linguistico personalizzato. È possibile utilizzare un modello linguistico personalizzato per migliorare le prestazioni di trascrizione dei domini come quello legale, alberghiero, finanziario e assicurativo. Per language-code, immetti un codice lingua valido. Ad esempio base-model-name, specificate un modello di base più adatto alla frequenza di campionamento dell'audio che desiderate trascrivere con il vostro modello linguistico personalizzato. Per model-name, specifica il nome che si desidera assegnare al modello linguistico personalizzato.

```
aws transcribe create-language-model \
    --language-code language-code \
    --base-model-name base-model-name \
    --model-name cli-clm-example \
    --input-data-config S3Uri="s3://amzn-s3-demo-bucket/Amazon-S3-Prefix-for-
training-data",TuningDataS3Uri="s3://amzn-s3-demo-bucket/Amazon-S3-Prefix-for-
tuning-data",DataAccessRoleArn="arn:aws:iam::AWS-account-number:role/IAM-role-with-
permissions-to-create-a-custom-language-model"
```

Output:

```
{
    "LanguageCode": "language-code",
    "BaseModelName": "base-model-name",
    "ModelName": "cli-clm-example",
    "InputDataConfig": {
        "S3Uri": "s3://amzn-s3-demo-bucket/Amazon-S3-Prefix/",
        "TuningDataS3Uri": "s3://amzn-s3-demo-bucket/Amazon-S3-Prefix/",
        "DataAccessRoleArn": "arn:aws:iam::AWS-account-number:role/IAM-role-with-
permissions-create-a-custom-language-model"
    },
    "ModelStatus": "IN_PROGRESS"
}
```

Per ulteriori informazioni, consulta Modelli linguistici personalizzati nella Guida per gli sviluppatori di Amazon Transcribe.

Esempio 2: come creare un modello linguistico personalizzato utilizzando solo i dati di addestramento.

L'esempio `create-language-model` seguente trascrive il file audio. È possibile utilizzare un modello linguistico personalizzato per migliorare le prestazioni di trascrizione dei domini come quello legale, alberghiero, finanziario e assicurativo. Per language-code, immetti un

codice lingua valido. Ad esempio base-model-name, specifica un modello di base più adatto alla frequenza di campionamento dell'audio che desideri trascrivere con il tuo modello linguistico personalizzato. Per model-name, specifica il nome che si desidera assegnare al modello linguistico personalizzato.

```
aws transcribe create-language-model \
    --language-code en-US \
    --base-model-name base-model-name \
    --model-name cli-clm-example \
    --input-data-config S3Uri="s3://amzn-s3-demo-bucket/Amazon-S3-Prefix-For-
Training-Data",DataAccessRoleArn="arn:aws:iam::AWS-account-number:role/IAM-role-
with-permissions-to-create-a-custom-language-model"
```

Output:

```
{
    "LanguageCode": "en-US",
    "BaseModelName": "base-model-name",
    "ModelName": "cli-clm-example",
    "InputDataConfig": {
        "S3Uri": "s3://amzn-s3-demo-bucket/Amazon-S3-Prefix-For-Training-Data/",
        "DataAccessRoleArn": "arn:aws:iam::your-AWS-account-number:role/IAM-role-
with-permissions-to-create-a-custom-language-model"
    },
    "ModelStatus": "IN_PROGRESS"
}
```

Per ulteriori informazioni, consulta Modelli linguistici personalizzati nella Guida per gli sviluppatori di Amazon Transcribe.

- Per i dettagli sull'API, consulta AWS CLI Command CreateLanguageModelReference.

`create-medical-vocabulary`

Il seguente esempio di codice mostra come utilizzare`create-medical-vocabulary`.

AWS CLI

Come creare un vocabolario personalizzato medico

L'esempio `create-medical-vocabulary` seguente crea un vocabolario personalizzato. Per creare un vocabolario personalizzato, devi aver creato un file di testo con tutti i termini che

desideri trascrivere in modo più accurato. Per vocabulary-file-uri, specifica l'URI Amazon Simple Storage Service (Amazon S3) di quel file di testo. Per language-code, specifica un codice di lingua corrispondente alla lingua del vocabolario personalizzato. Per vocabulary-name, specifica come vuoi denominare il vocabolario personalizzato.

```
aws transcribe create-medical-vocabulary \
    --vocabulary-name cli-medical-vocab-example \
    --language-code language-code \
    --vocabulary-file-uri https://amzn-s3-demo-bucket.AWS-Region.amazonaws.com/the-text-file-for-the-medical-custom-vocabulary.txt
```

Output:

```
{
    "VocabularyName": "cli-medical-vocab-example",
    "LanguageCode": "language-code",
    "VocabularyState": "PENDING"
}
```

Per ulteriori informazioni, consulta Vocabolari medici personalizzati nella Guida per gli sviluppatori di Amazon Transcribe.

- Per i dettagli sull'API, consulta CreateMedicalVocabulary AWS CLICommand Reference.

`create-vocabulary-filter`

Il seguente esempio di codice mostra come utilizzare`create-vocabulary-filter`.

AWS CLI

Come creare un filtro di vocabolario

L'esempio `create-vocabulary-filter` seguente crea un filtro di vocabolario che utilizza un file di testo contenente un elenco di parole che non devono essere presenti in una trascrizione. Per language-code, specifica il codice lingua corrispondente alla lingua del filtro di vocabolario. Per vocabulary-filter-file-uri, specifica l'URI Amazon Simple Storage Service (Amazon S3) del file di testo. Per vocabulary-filter-name, specifica il nome del filtro del vocabolario.

```
aws transcribe create-vocabulary-filter \
    --language-code language-code \
```

```
    --vocabulary-filter-file-uri s3://amzn-s3-demo-bucket/vocabulary-filter.txt \
    --vocabulary-filter-name cli-vocabulary-filter-example
```

Output:

```
{
    "VocabularyFilterName": "cli-vocabulary-filter-example",
    "LanguageCode": "language-code"
}
```

Per ulteriori informazioni, consulta Filtraggio di parole indesiderate nella Guida per gli sviluppatori di Amazon Transcribe.

- Per i dettagli sull'API, consulta Command CreateVocabularyFilterReference AWS CLI .

`create-vocabulary`

Il seguente esempio di codice mostra come utilizzare`create-vocabulary`.

AWS CLI

Creazione di un vocabolario personalizzato

L'esempio `create-vocabulary` seguente crea un vocabolario personalizzato. Per creare un vocabolario personalizzato, devi aver creato un file di testo con tutti i termini che desideri trascrivere in modo più accurato. Per vocabulary-file-uri, specifica l'URI Amazon Simple Storage Service (Amazon S3) di quel file di testo. Per language-code, specifica un codice di lingua corrispondente alla lingua del vocabolario personalizzato. Per vocabulary-name, specifica come vuoi denominare il vocabolario personalizzato.

```
aws transcribe create-vocabulary \
    --language-code language-code \
    --vocabulary-name cli-vocab-example \
    --vocabulary-file-uri s3://amzn-s3-demo-bucket/Amazon-S3-prefix/the-text-file-for-the-custom-vocabulary.txt
```

Output:

```
{
    "VocabularyName": "cli-vocab-example",
    "LanguageCode": "language-code",
```

```
    "VocabularyState": "PENDING"
}
```

Per ulteriori informazioni, consulta Vocabolari personalizzati nella Guida per gli sviluppatori di Amazon Transcribe.

- Per i dettagli sull'API, consulta CreateVocabulary AWS CLICommand Reference.

`delete-language-model`

Il seguente esempio di codice mostra come utilizzare`delete-language-model`.

AWS CLI

Come eliminare un modello linguistico personalizzato

L'esempio `delete-language-model` seguente elimina un modello linguistico personalizzato.

```
aws transcribe delete-language-model \
    --model-name model-name
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Modelli linguistici personalizzati nella Guida per gli sviluppatori di Amazon Transcribe.

- Per i dettagli sull'API, consulta DeleteLanguageModel AWS CLICommand Reference.

`delete-medical-transcription-job`

Il seguente esempio di codice mostra come utilizzare`delete-medical-transcription-job`.

AWS CLI

Eliminazione di un processo di trascrizione medica

L'esempio `delete-medical-transcription-job` seguente elimina un processo di trascrizione medica.

```
aws transcribe delete-medical-transcription-job \
    --medical-transcription-job-name medical-transcription-job-name
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta DeleteMedicalTranscriptionJobla Amazon Transcribe Developer Guide.

- Per i dettagli sull'API, consulta AWS CLI Command DeleteMedicalTranscriptionJobReference.

`delete-medical-vocabulary`

Il seguente esempio di codice mostra come utilizzare`delete-medical-vocabulary`.

AWS CLI

Come eliminare un vocabolario medico personalizzato

L'esempio `delete-medical-vocabulary` seguente elimina un vocabolario personalizzato medico. Per vocabulary-name, specifica il nome del vocabolario personalizzato medico.

```
aws transcribe delete-vocabulary \
    --vocabulary-name medical-custom-vocabulary-name
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Vocabolari medici personalizzati nella Guida per gli sviluppatori di Amazon Transcribe.

- Per i dettagli sull'API, consulta DeleteMedicalVocabulary AWS CLICommand Reference.

`delete-transcription-job`

Il seguente esempio di codice mostra come utilizzare`delete-transcription-job`.

AWS CLI

Eliminazione di un processo di trascrizione

L'esempio `delete-transcription-job` seguente elimina uno dei processi di trascrizione.

```
aws transcribe delete-transcription-job \
    --transcription-job-name your-transcription-job
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta DeleteTranscriptionJobla Amazon Transcribe Developer Guide.

- Per i dettagli sull'API, consulta AWS CLI Command DeleteTranscriptionJobReference.

`delete-vocabulary-filter`

Il seguente esempio di codice mostra come utilizzare`delete-vocabulary-filter`.

AWS CLI

Come eliminare un filtro di vocabolario

L'esempio `delete-vocabulary-filter` seguente elimina un filtro di vocabolario.

```
aws transcribe delete-vocabulary-filter \
    --vocabulary-filter-name vocabulary-filter-name
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Filtraggio di parole indesiderate nella Guida per gli sviluppatori di Amazon Transcribe.

- Per i dettagli sull'API, consulta DeleteVocabularyFilter AWS CLICommand Reference.

`delete-vocabulary`

Il seguente esempio di codice mostra come utilizzare`delete-vocabulary`.

AWS CLI

Eliminazione di un vocabolario personalizzato

L'esempio `delete-vocabulary` seguente elimina un vocabolario personalizzato.

```
aws transcribe delete-vocabulary \
    --vocabulary-name vocabulary-name
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Vocabolari personalizzati nella Guida per gli sviluppatori di Amazon Transcribe.

- Per i dettagli sull'API, consulta DeleteVocabulary AWS CLICommand Reference.

`describe-language-model`

Il seguente esempio di codice mostra come utilizzare`describe-language-model`.

AWS CLI

Come ottenere informazioni su uno specifico modello linguistico personalizzato

L'esempio `describe-language-model` seguente ottiene informazioni su uno specifico modello linguistico personalizzato. Ad esempio, sotto BaseModelName puoi vedere se il tuo modello è stato addestrato utilizzando un WideBand modello NarrowBand o. I modelli linguistici personalizzati con un modello NarrowBand base possono trascrivere l'audio con una frequenza di campionamento inferiore a 16 kHz. I modelli linguistici che utilizzano un modello WideBand base possono trascrivere l'audio con una frequenza di campionamento superiore a 16 kHz. Il parametro S3Uri indica il prefisso Amazon S3 che è stato utilizzato per accedere ai dati di addestramento per creare il modello linguistico personalizzato.

```
aws transcribe describe-language-model \
    --model-name cli-clm-example
```

Output:

```
{
    "LanguageModel": {
        "ModelName": "cli-clm-example",
        "CreateTime": "2020-09-25T17:57:38.504000+00:00",
        "LastModifiedTime": "2020-09-25T17:57:48.585000+00:00",
        "LanguageCode": "language-code",
        "BaseModelName": "base-model-name",
        "ModelStatus": "IN_PROGRESS",
        "UpgradeAvailability": false,
        "InputDataConfig": {
            "S3Uri": "s3://amzn-s3-demo-bucket/Amazon-S3-Prefix/",
            "TuningDataS3Uri": "s3://amzn-s3-demo-bucket/Amazon-S3-Prefix/",
            "DataAccessRoleArn": "arn:aws:iam::AWS-account-number:role/IAM-role-
with-permissions-to-create-a-custom-language-model"
        }
    }
}
```

Per ulteriori informazioni, consulta Modelli linguistici personalizzati nella Guida per gli sviluppatori di Amazon Transcribe.

- Per i dettagli sull'API, consulta Command DescribeLanguageModelReference AWS CLI .

`get-medical-transcription-job`

Il seguente esempio di codice mostra come utilizzare`get-medical-transcription-job`.

AWS CLI

Come ottenere informazioni su uno specifico processo di trascrizione medica

L'esempio `get-medical-transcription-job` seguente ottiene informazioni su uno specifico processo di trascrizione medica. Per accedere ai risultati della trascrizione, utilizzate il TranscriptFileUri parametro. Se sono state abilitate funzionalità aggiuntive per il processo di trascrizione, è possibile visualizzarle nell'oggetto Impostazioni. Il parametro Specialty mostra la specializzazione medica del provider. Il parametro Type indica se il discorso nel processo di trascrizione è relativo a una conversazione medica o una dettatura medica.

```
aws transcribe get-medical-transcription-job \
    --medical-transcription-job-name vocabulary-dictation-medical-transcription-job
```

Output:

```
{
    "MedicalTranscriptionJob": {
        "MedicalTranscriptionJobName": "vocabulary-dictation-medical-transcription-
job",
        "TranscriptionJobStatus": "COMPLETED",
        "LanguageCode": "en-US",
        "MediaSampleRateHertz": 48000,
        "MediaFormat": "mp4",
        "Media": {
            "MediaFileUri": "s3://Amazon-S3-Prefix/your-audio-file.file-extension"
        },
        "Transcript": {
            "TranscriptFileUri": "https://s3.Region.amazonaws.com/Amazon-S3-Prefix/
vocabulary-dictation-medical-transcription-job.json"
        },
        "StartTime": "2020-09-21T21:17:27.045000+00:00",
        "CreationTime": "2020-09-21T21:17:27.016000+00:00",
```

```
        "CompletionTime": "2020-09-21T21:17:59.561000+00:00",
        "Settings": {
            "ChannelIdentification": false,
            "ShowAlternatives": false,
            "VocabularyName": "cli-medical-vocab-example"
        },
        "Specialty": "PRIMARYCARE",
        "Type": "DICTATION"
    }
}
```

Per ulteriori informazioni, consulta Trascrizione in batch nella Guida per gli sviluppatori di Amazon Transcribe.

- Per i dettagli sull'API, consulta AWS CLI Command GetMedicalTranscriptionJobReference.

`get-medical-vocabulary`

Il seguente esempio di codice mostra come utilizzare`get-medical-vocabulary`.

AWS CLI

Come ottenere informazioni su un vocabolario medico personalizzato

L'esempio `get-medical-vocabulary` seguente ottiene informazioni su un vocabolario personalizzato medico. È possibile utilizzare il VocabularyState parametro per visualizzare lo stato di elaborazione del vocabolario. Se è PRONTO, puoi usarlo durante l'operazione. StartMedicalTranscriptionJob :

```
aws transcribe get-medical-vocabulary \
    --vocabulary-name medical-vocab-example
```

Output:

```
{
    "VocabularyName": "medical-vocab-example",
    "LanguageCode": "en-US",
    "VocabularyState": "READY",
    "LastModifiedTime": "2020-09-19T23:59:04.349000+00:00",
    "DownloadUri": "https://link-to-download-the-text-file-used-to-create-your-
medical-custom-vocabulary"
}
```

Per ulteriori informazioni, consulta Vocabolari medici personalizzati nella Guida per gli sviluppatori di Amazon Transcribe.

- Per i dettagli sull'API, consulta GetMedicalVocabulary AWS CLICommand Reference.

`get-transcription-job`

Il seguente esempio di codice mostra come utilizzare`get-transcription-job`.

AWS CLI

Ottenere informazioni su un processo di trascrizione specifico

L'esempio `get-transcription-job` seguente ottiene informazioni su un processo di trascrizione specifico. Per accedere ai risultati della trascrizione, utilizzate il TranscriptFileUri parametro. Utilizzate il MediaFileUri parametro per vedere quale file audio avete trascritto con questo lavoro. Puoi usare l'oggetto Settings per visualizzare le funzionalità opzionali che hai abilitato nel processo di trascrizione.

```
aws transcribe get-transcription-job \
    --transcription-job-name your-transcription-job
```

Output:

```
{
    "TranscriptionJob": {
        "TranscriptionJobName": "your-transcription-job",
        "TranscriptionJobStatus": "COMPLETED",
        "LanguageCode": "language-code",
        "MediaSampleRateHertz": 48000,
        "MediaFormat": "mp4",
        "Media": {
            "MediaFileUri": "s3://amzn-s3-demo-bucket/your-audio-file.file-
extension"
        },
        "Transcript": {
            "TranscriptFileUri": "https://Amazon-S3-file-location-of-transcription-
output"
        },
        "StartTime": "2020-09-18T22:27:23.970000+00:00",
        "CreationTime": "2020-09-18T22:27:23.948000+00:00",
        "CompletionTime": "2020-09-18T22:28:21.197000+00:00",
```

```
        "Settings": {
            "ChannelIdentification": false,
            "ShowAlternatives": false
        },
        "IdentifyLanguage": true,
        "IdentifiedLanguageScore": 0.8672199249267578
    }
}
```

Per ulteriori informazioni, consulta Getting Started (AWS Command Line Interface) nella Amazon Transcribe Developer Guide.

- Per i dettagli sull'API, consulta AWS CLI Command GetTranscriptionJobReference.

`get-vocabulary-filter`

Il seguente esempio di codice mostra come utilizzare`get-vocabulary-filter`.

AWS CLI

Come ottenere informazioni su un filtro di vocabolario

L'esempio `get-vocabulary-filter` seguente ottiene informazioni su un filtro di vocabolario. È possibile utilizzare il DownloadUri parametro per ottenere l'elenco delle parole utilizzate per creare il filtro del vocabolario.

```
aws transcribe get-vocabulary-filter \
    --vocabulary-filter-name testFilter
```

Output:

```
{
    "VocabularyFilterName": "testFilter",
    "LanguageCode": "language-code",
    "LastModifiedTime": "2020-05-07T22:39:32.147000+00:00",
    "DownloadUri": "https://Amazon-S3-location-to-download-your-vocabulary-filter"
}
```

Per ulteriori informazioni, consulta Filtrare le parole indesiderate nella Guida per gli sviluppatori di Amazon Transcribe.

- Per i dettagli sull'API, consulta Command GetVocabularyFilterReference AWS CLI .

`get-vocabulary`

Il seguente esempio di codice mostra come utilizzare`get-vocabulary`.

AWS CLI

Ottenere informazioni su un vocabolario personalizzato

L'esempio `get-vocabulary` seguente ottiene informazioni su un vocabolario personalizzato creato in precedenza.

```
aws transcribe get-vocabulary \
    --vocabulary-name cli-vocab-1
```

Output:

```
{
    "VocabularyName": "cli-vocab-1",
    "LanguageCode": "language-code",
    "VocabularyState": "READY",
    "LastModifiedTime": "2020-09-19T23:22:32.836000+00:00",
    "DownloadUri": "https://link-to-download-the-text-file-used-to-create-your-
custom-vocabulary"
}
```

Per ulteriori informazioni, consulta Vocabolari personalizzati nella Guida per gli sviluppatori di Amazon Transcribe.

- Per i dettagli sull'API, consulta GetVocabulary AWS CLICommand Reference.

`list-language-models`

Il seguente esempio di codice mostra come utilizzare`list-language-models`.

AWS CLI

Come elencare i modelli linguistici personalizzati

L'`list-language-models`esempio seguente elenca i modelli linguistici personalizzati associati all' AWS account e alla regione. È possibile utilizzare i parametri `S3Uri` e `TuningDataS3Uri` per trovare i prefissi Amazon S3 utilizzati come dati di addestramento o dati di ottimizzazione.

BaseModelName Indica se hai utilizzato un WideBand modello o un NarrowBand modello per creare un modello linguistico personalizzato. È possibile trascrivere l'audio con una frequenza di campionamento inferiore a 16 kHz con un modello linguistico personalizzato utilizzando un modello base. NarrowBand È possibile trascrivere l'audio a 16 kHz o superiore con un modello linguistico personalizzato utilizzando un modello base. WideBand Il parametro `ModelStatus` mostra se è possibile utilizzare il modello linguistico personalizzato in un processo di trascrizione. Se il valore è COMPLETED, è possibile utilizzare il modello in un processo di trascrizione.

```
aws transcribe list-language-models
```

Output:

```
{
    "Models": [
        {
            "ModelName": "cli-clm-2",
            "CreateTime": "2020-09-25T17:57:38.504000+00:00",
            "LastModifiedTime": "2020-09-25T17:57:48.585000+00:00",
            "LanguageCode": "language-code",
            "BaseModelName": "WideBand",
            "ModelStatus": "IN_PROGRESS",
            "UpgradeAvailability": false,
            "InputDataConfig": {
                "S3Uri": "s3://amzn-s3-demo-bucket/clm-training-data/",
                "TuningDataS3Uri": "s3://amzn-s3-demo-bucket/clm-tuning-data/",
                "DataAccessRoleArn": "arn:aws:iam::AWS-account-number:role/IAM-role-
used-to-create-the-custom-language-model"
            }
        },
        {
            "ModelName": "cli-clm-1",
            "CreateTime": "2020-09-25T17:16:01.835000+00:00",
            "LastModifiedTime": "2020-09-25T17:16:15.555000+00:00",
            "LanguageCode": "language-code",
            "BaseModelName": "WideBand",
            "ModelStatus": "IN_PROGRESS",
            "UpgradeAvailability": false,
            "InputDataConfig": {
                "S3Uri": "s3://amzn-s3-demo-bucket/clm-training-data/",
                "DataAccessRoleArn": "arn:aws:iam::AWS-account-number:role/IAM-role-
used-to-create-the-custom-language-model"
            }
```

```
        },
        {
            "ModelName": "clm-console-1",
            "CreateTime": "2020-09-24T19:26:28.076000+00:00",
            "LastModifiedTime": "2020-09-25T04:25:22.271000+00:00",
            "LanguageCode": "language-code",
            "BaseModelName": "NarrowBand",
            "ModelStatus": "COMPLETED",
            "UpgradeAvailability": false,
            "InputDataConfig": {
                "S3Uri": "s3://amzn-s3-demo-bucket/clm-training-data/",
                "DataAccessRoleArn": "arn:aws:iam::AWS-account-number:role/IAM-role-
used-to-create-the-custom-language-model"
            }
        }
    ]
}
```

Per ulteriori informazioni, consulta Modelli linguistici personalizzati nella Guida per gli sviluppatori di Amazon Transcribe.

- Per i dettagli sull'API, consulta Command ListLanguageModelsReference AWS CLI .

`list-medical-transcription-jobs`

Il seguente esempio di codice mostra come utilizzare`list-medical-transcription-jobs`.

AWS CLI

Elencare i processi di trascrizione medica

L'`list-medical-transcription-jobs`esempio seguente elenca i lavori di trascrizione medica associati al tuo AWS account e alla tua regione. Per ottenere ulteriori informazioni su un particolare processo di trascrizione, copiate il valore di un MedicalTranscriptionJobName parametro nell'output della trascrizione e specificate quel valore per l'opzione del comando. `MedicalTranscriptionJobName` `get-medical-transcription-job` Per visualizzare altri lavori di trascrizione, copiate il valore del NextToken parametro, eseguite nuovamente il `list-medical-transcription-jobs` comando e specificate quel valore nell'opzione. `--next-token`

```
aws transcribe list-medical-transcription-jobs
```

Output:


```
{
    "NextToken": "3/PblzkiGhzjER3KHuQt2fmbPLF7cDYafjFMEoGn44ON/
gsuUSTIkGyanvRE6WMXFd/ZTEc2EZj+P9eii/
z1O2FDYli6RLI0WoRX4RwMisVrh9G0Kie0Y8ikBCdtqlZB10Wa9McC+ebOl
+LaDtZPC4u6ttoHLRlEfzqstHXSgapXg3tEBtm9piIaPB6MOM5BB6t86+qtmocTR/
qrteHZBBudhTfbCwhsxaqujHiiUvFdm3BQbKKWIW06yV9b+4f38oD2lVIan
+vfUs3gBYAl5VTDmXXzQPBQOHPjtwmFI+IWX15nSUjWuN3TUylHgPWzDaYT8qBtu0Z+3UG4V6b
+K2CC0XszXg5rBq9hYgNzy4XoFh/6s5DoSnzq49Q9xHgHdT2yBADFmvFK7myZBsj75+2vQZOSVpWUPy3WT/32zFAcoEL
+mFYfUjtTZ8n/jq7aQEjQ42A
+X/7K6JgOcdVPtEg8PlDr5kgYYG3q3OmYXX37U3FZuJmnTI63VtIXsNnOU5eGoYObtpk00Nq9UkzgSJxqj84ZD5n
+S0EGy9ZUYBJRRcGeYUM3Q4DbSJfUwSAqcFdLIWZdp8qIREMQIBWy7BLwSdyqsQo2vRrd53hm5aWM7SVf6pPq6X/
IXR5+1eUOOD8/coaTT4ES2DerbV6RkV4o0VT1d0SdVX/
MmtkNG8nYj8PqU07w7988quh1ZP6D80veJS1q73tUUR9MjnGernW2tAnvnLNhdefBcD
+sZVfYq3iBMFY7wTy1P1G6NqW9GrYDYoX3tTPWlD7phpbVSyKrh/
PdYrps5UxnsGoA1b7L/FfAXDfUoGrGUB4N3JsPYXX9D++g+6gV1qBBs/
WfF934aKqfD6UTggm/zV3GAOWiBpfvAZRvEb924i6yGHyMC7y54O1ZAwSBupmI
+FFd13CaPO4kN1vJlth6aM5vUPXg4BpyUhtbRhwD/KxCvf9K0tLJGyL1A==",
    "MedicalTranscriptionJobSummaries": [
        {
            "MedicalTranscriptionJobName": "vocabulary-dictation-medical-
transcription-job",
            "CreationTime": "2020-09-21T21:17:27.016000+00:00",
            "StartTime": "2020-09-21T21:17:27.045000+00:00",
            "CompletionTime": "2020-09-21T21:17:59.561000+00:00",
            "LanguageCode": "en-US",
            "TranscriptionJobStatus": "COMPLETED",
            "OutputLocationType": "CUSTOMER_BUCKET",
            "Specialty": "PRIMARYCARE",
            "Type": "DICTATION"
        },
        {
            "MedicalTranscriptionJobName": "alternatives-dictation-medical-
transcription-job",
            "CreationTime": "2020-09-21T21:01:14.569000+00:00",
            "StartTime": "2020-09-21T21:01:14.592000+00:00",
            "CompletionTime": "2020-09-21T21:01:43.606000+00:00",
            "LanguageCode": "en-US",
            "TranscriptionJobStatus": "COMPLETED",
            "OutputLocationType": "CUSTOMER_BUCKET",
            "Specialty": "PRIMARYCARE",
            "Type": "DICTATION"
        },
```

```
        {
            "MedicalTranscriptionJobName": "alternatives-conversation-medical-
transcription-job",
            "CreationTime": "2020-09-21T19:09:18.171000+00:00",
            "StartTime": "2020-09-21T19:09:18.199000+00:00",
            "CompletionTime": "2020-09-21T19:10:22.516000+00:00",
            "LanguageCode": "en-US",
            "TranscriptionJobStatus": "COMPLETED",
            "OutputLocationType": "CUSTOMER_BUCKET",
            "Specialty": "PRIMARYCARE",
            "Type": "CONVERSATION"
        },
        {
            "MedicalTranscriptionJobName": "speaker-id-conversation-medical-
transcription-job",
            "CreationTime": "2020-09-21T18:43:37.157000+00:00",
            "StartTime": "2020-09-21T18:43:37.265000+00:00",
            "CompletionTime": "2020-09-21T18:44:21.192000+00:00",
            "LanguageCode": "en-US",
            "TranscriptionJobStatus": "COMPLETED",
            "OutputLocationType": "CUSTOMER_BUCKET",
            "Specialty": "PRIMARYCARE",
            "Type": "CONVERSATION"
        },
        {
            "MedicalTranscriptionJobName": "multichannel-conversation-medical-
transcription-job",
            "CreationTime": "2020-09-20T23:46:44.053000+00:00",
            "StartTime": "2020-09-20T23:46:44.081000+00:00",
            "CompletionTime": "2020-09-20T23:47:35.851000+00:00",
            "LanguageCode": "en-US",
            "TranscriptionJobStatus": "COMPLETED",
            "OutputLocationType": "CUSTOMER_BUCKET",
            "Specialty": "PRIMARYCARE",
            "Type": "CONVERSATION"
        }
    ]
}
```

Per ulteriori informazioni, consulta https://docs.aws.amazon.com/transcribe/ latest/dg/batch -med-transcription.html> nella Amazon Transcribe Developer Guide.

- Per i dettagli sull'ListMedicalTranscriptionJobsAPI, consulta AWS CLI Command Reference.

`list-medical-vocabularies`

Il seguente esempio di codice mostra come utilizzare`list-medical-vocabularies`.

AWS CLI

Come elencare i vocabolari medici personalizzati

L'`list-medical-vocabularies`esempio seguente elenca i vocabolari medici personalizzati associati al tuo AWS account e alla tua regione. Per ottenere ulteriori informazioni su un particolare processo di trascrizione, copia il valore di un parametro `MedicalTranscriptionJobName` nell'output di trascrizione e specifica tale valore per l'opzione `MedicalTranscriptionJobName` del comando `get-medical-transcription-job`. Per visualizzare ulteriori processi di trascrizione, copia il valore del parametro `NextToken`, esegui nuovamente il comando `list-medical-transcription-jobs` e specifica quel valore nell'opzione `--next-token`.

```
aws transcribe list-medical-vocabularies
```

Output:

```
{
    "Vocabularies": [
        {
            "VocabularyName": "cli-medical-vocab-2",
            "LanguageCode": "en-US",
            "LastModifiedTime": "2020-09-21T21:44:59.521000+00:00",
            "VocabularyState": "READY"
        },
        {
            "VocabularyName": "cli-medical-vocab-1",
            "LanguageCode": "en-US",
            "LastModifiedTime": "2020-09-19T23:59:04.349000+00:00",
            "VocabularyState": "READY"
        }
    ]
}
```

Per ulteriori informazioni, consulta Vocabolari medici personalizzati nella Guida per gli sviluppatori di Amazon Transcribe.

- Per i dettagli sull'API, consulta Command ListMedicalVocabulariesReference AWS CLI .

`list-transcription-jobs`

Il seguente esempio di codice mostra come utilizzare`list-transcription-jobs`.

AWS CLI

Elencare i processi di trascrizione

L'`list-transcription-jobs`esempio seguente elenca i lavori di trascrizione associati al tuo AWS account e alla tua regione.

```
aws transcribe list-transcription-jobs
```

Output:

```
{
    "NextToken": "NextToken",
    "TranscriptionJobSummaries": [
        {
            "TranscriptionJobName": "speak-id-job-1",
            "CreationTime": "2020-08-17T21:06:15.391000+00:00",
            "StartTime": "2020-08-17T21:06:15.416000+00:00",
            "CompletionTime": "2020-08-17T21:07:05.098000+00:00",
            "LanguageCode": "language-code",
            "TranscriptionJobStatus": "COMPLETED",
            "OutputLocationType": "SERVICE_BUCKET"
        },
        {
            "TranscriptionJobName": "job-1",
            "CreationTime": "2020-08-17T20:50:24.207000+00:00",
            "StartTime": "2020-08-17T20:50:24.230000+00:00",
            "CompletionTime": "2020-08-17T20:52:18.737000+00:00",
            "LanguageCode": "language-code",
            "TranscriptionJobStatus": "COMPLETED",
            "OutputLocationType": "SERVICE_BUCKET"
        },
        {
            "TranscriptionJobName": "sdk-test-job-4",
            "CreationTime": "2020-08-17T20:32:27.917000+00:00",
            "StartTime": "2020-08-17T20:32:27.956000+00:00",
            "CompletionTime": "2020-08-17T20:33:15.126000+00:00",
            "LanguageCode": "language-code",
            "TranscriptionJobStatus": "COMPLETED",
```

```
            "OutputLocationType": "SERVICE_BUCKET"
        },
        {
            "TranscriptionJobName": "Diarization-speak-id",
            "CreationTime": "2020-08-10T22:10:09.066000+00:00",
            "StartTime": "2020-08-10T22:10:09.116000+00:00",
            "CompletionTime": "2020-08-10T22:26:48.172000+00:00",
            "LanguageCode": "language-code",
            "TranscriptionJobStatus": "COMPLETED",
            "OutputLocationType": "SERVICE_BUCKET"
        },
        {
            "TranscriptionJobName": "your-transcription-job-name",
            "CreationTime": "2020-07-29T17:45:09.791000+00:00",
            "StartTime": "2020-07-29T17:45:09.826000+00:00",
            "CompletionTime": "2020-07-29T17:46:20.831000+00:00",
            "LanguageCode": "language-code",
            "TranscriptionJobStatus": "COMPLETED",
            "OutputLocationType": "SERVICE_BUCKET"
        }
    ]
}
```

Per ulteriori informazioni, consulta Getting Started (AWS Command Line Interface) nella Amazon Transcribe Developer Guide.

- Per i dettagli sull'API, consulta AWS CLI Command ListTranscriptionJobsReference.

`list-vocabularies`

Il seguente esempio di codice mostra come utilizzare`list-vocabularies`.

AWS CLI

Elenco dei vocabolari personalizzati

L'`list-vocabularies`esempio seguente elenca i vocabolari personalizzati associati all' AWS account e alla regione.

```
aws transcribe list-vocabularies
```

Output:

```
{
    "NextToken": "NextToken",
    "Vocabularies": [
        {
            "VocabularyName": "ards-test-1",
            "LanguageCode": "language-code",
            "LastModifiedTime": "2020-04-27T22:00:27.330000+00:00",
            "VocabularyState": "READY"
        },
        {
            "VocabularyName": "sample-test",
            "LanguageCode": "language-code",
            "LastModifiedTime": "2020-04-24T23:04:11.044000+00:00",
            "VocabularyState": "READY"
        },
        {
            "VocabularyName": "CRLF-to-LF-test-3-1",
            "LanguageCode": "language-code",
            "LastModifiedTime": "2020-04-24T22:12:22.277000+00:00",
            "VocabularyState": "READY"
        },
        {
            "VocabularyName": "CRLF-to-LF-test-2",
            "LanguageCode": "language-code",
            "LastModifiedTime": "2020-04-24T21:53:50.455000+00:00",
            "VocabularyState": "READY"
        },
        {
            "VocabularyName": "CRLF-to-LF-1-1",
            "LanguageCode": "language-code",
            "LastModifiedTime": "2020-04-24T21:39:33.356000+00:00",
            "VocabularyState": "READY"
        }
    ]
}
```

Per ulteriori informazioni, consulta Vocabolari personalizzati nella Guida per gli sviluppatori di Amazon Transcribe.

- Per i dettagli sull'API, consulta Command ListVocabulariesReference AWS CLI .

`list-vocabulary-filters`

Il seguente esempio di codice mostra come utilizzare`list-vocabulary-filters`.

AWS CLI

Come elencare i filtri di vocabolario

L'`list-vocabulary-filters`esempio seguente elenca i filtri del vocabolario associati al tuo AWS account e alla tua regione.

```
aws transcribe list-vocabulary-filters
```

Output:

```
{
    "NextToken": "NextToken": [
        {
            "VocabularyFilterName": "testFilter",
            "LanguageCode": "language-code",
            "LastModifiedTime": "2020-05-07T22:39:32.147000+00:00"
        },
        {
            "VocabularyFilterName": "testFilter2",
            "LanguageCode": "language-code",
            "LastModifiedTime": "2020-05-21T23:29:35.174000+00:00"
        },
        {
            "VocabularyFilterName": "filter2",
            "LanguageCode": "language-code",
            "LastModifiedTime": "2020-05-08T20:18:26.426000+00:00"
        },
        {
            "VocabularyFilterName": "filter-review",
            "LanguageCode": "language-code",
            "LastModifiedTime": "2020-06-03T18:52:30.448000+00:00"
        },
        {
            "VocabularyFilterName": "crlf-filt",
            "LanguageCode": "language-code",
            "LastModifiedTime": "2020-05-22T19:42:42.737000+00:00"
        }
    ]
```

```
}
```

Per ulteriori informazioni, consulta Filtraggio di parole indesiderate nella Guida per gli sviluppatori di Amazon Transcribe.

- Per i dettagli sull'API, consulta Command ListVocabularyFiltersReference AWS CLI .

`start-medical-transcription-job`

Il seguente esempio di codice mostra come utilizzare`start-medical-transcription-job`.

AWS CLI

Esempio 1: trascrivere una dettatura medica archiviata come file audio

L'esempio `start-medical-transcription-job` seguente trascrive un file audio. Specifica la posizione dell'output della trascrizione nel parametro `OutputBucketName`.

```
aws transcribe start-medical-transcription-job \
    --cli-input-json file://myfile.json
```

Contenuto di `myfile.json`:

```
{
    "MedicalTranscriptionJobName": "simple-dictation-medical-transcription-job",
    "LanguageCode": "language-code",
    "Specialty": "PRIMARYCARE",
    "Type": "DICTATION",
    "OutputBucketName":"amzn-s3-demo-bucket",
    "Media": {
        "MediaFileUri": "s3://amzn-s3-demo-bucket/your-audio-file.extension"
    }
}
```

Output:

```
{
    "MedicalTranscriptionJob": {
        "MedicalTranscriptionJobName": "simple-dictation-medical-transcription-job",
        "TranscriptionJobStatus": "IN_PROGRESS",
        "LanguageCode": "language-code",
```

```
        "Media": {
            "MediaFileUri": "s3://amzn-s3-demo-bucket/your-audio-file.extension"
        },
        "StartTime": "2020-09-20T00:35:22.256000+00:00",
        "CreationTime": "2020-09-20T00:35:22.218000+00:00",
        "Specialty": "PRIMARYCARE",
        "Type": "DICTATION"
    }
}
```

Per ulteriori informazioni, consulta la panoramica della trascrizione in batch nella Guida per gli sviluppatori di Amazon Transcribe.

Esempio 2: trascrivere un dialogo medico-paziente archiviato come file audio

L'esempio `start-medical-transcription-job` seguente trascrive un file audio contenente un dialogo medico-paziente. Specificate la posizione dell'output della trascrizione nel OutputBucketName parametro.

```
aws transcribe start-medical-transcription-job \
    --cli-input-json file://mysecondfile.json
```

Contenuto di `mysecondfile.json`:

```
{
    "MedicalTranscriptionJobName": "simple-dictation-medical-transcription-job",
    "LanguageCode": "language-code",
    "Specialty": "PRIMARYCARE",
    "Type": "CONVERSATION",
    "OutputBucketName":"amzn-s3-demo-bucket",
    "Media": {
        "MediaFileUri": "s3://amzn-s3-demo-bucket/your-audio-file.extension"
    }
}
```

Output:

```
{
    "MedicalTranscriptionJob": {
        "MedicalTranscriptionJobName": "simple-conversation-medical-transcription-
job",
```

```
        "TranscriptionJobStatus": "IN_PROGRESS",
        "LanguageCode": "language-code",
        "Media": {
            "MediaFileUri": "s3://amzn-s3-demo-bucket/your-audio-file.extension"
        },
        "StartTime": "2020-09-20T23:19:49.965000+00:00",
        "CreationTime": "2020-09-20T23:19:49.941000+00:00",
        "Specialty": "PRIMARYCARE",
        "Type": "CONVERSATION"
    }
}
```

Per ulteriori informazioni, consulta la panoramica della trascrizione in batch nella Guida per gli sviluppatori di Amazon Transcribe.

Esempio 3: trascrivere un file audio multicanale di un dialogo medico-paziente

L'esempio `start-medical-transcription-job` seguente trascrive l'audio da ogni canale del file audio e unisce le trascrizioni separate di ciascun canale in un unico batch di trascrizione. Specifica la posizione dell'output della trascrizione nel parametro `OutputBucketName`.

```
aws transcribe start-medical-transcription-job \
    --cli-input-json file://mythirdfile.json
```

Contenuto di `mythirdfile.json`:

```
{
    "MedicalTranscriptionJobName": "multichannel-conversation-medical-transcription-
job",
    "LanguageCode": "language-code",
    "Specialty": "PRIMARYCARE",
    "Type": "CONVERSATION",
    "OutputBucketName":"amzn-s3-demo-bucket",
        "Media": {
          "MediaFileUri": "s3://amzn-s3-demo-bucket/your-audio-file.extension"
        },
        "Settings":{
          "ChannelIdentification": true
        }
}
```

Output:

```
{
    "MedicalTranscriptionJob": {
        "MedicalTranscriptionJobName": "multichannel-conversation-medical-
transcription-job",
        "TranscriptionJobStatus": "IN_PROGRESS",
        "LanguageCode": "language-code",
        "Media": {
            "MediaFileUri": "s3://amzn-s3-demo-bucket/your-audio-file.extension"
        },
        "StartTime": "2020-09-20T23:46:44.081000+00:00",
        "CreationTime": "2020-09-20T23:46:44.053000+00:00",
        "Settings": {
            "ChannelIdentification": true
        },
        "Specialty": "PRIMARYCARE",
        "Type": "CONVERSATION"
    }
}
```

Per ulteriori informazioni, consulta Identificazione del canale nella Guida per gli sviluppatori di Amazon Transcribe.

Esempio 4: trascrivere un file audio di un dialogo tra medico e paziente e identificare le persone che parlano nell'output della trascrizione

L'esempio `start-medical-transcription-job` seguente trascrive un file audio ed etichetta il discorso di ciascuna persona che parla nell'output di trascrizione. Specifica la posizione dell'output della trascrizione nel parametro `OutputBucketName`.

```
aws transcribe start-medical-transcription-job \
    --cli-input-json file://myfourthfile.json
```

Contenuto di `myfourthfile.json`:

```
{
    "MedicalTranscriptionJobName": "speaker-id-conversation-medical-transcription-
job",
    "LanguageCode": "language-code",
    "Specialty": "PRIMARYCARE",
    "Type": "CONVERSATION",
    "OutputBucketName":"amzn-s3-demo-bucket",
    "Media": {
```

```
        "MediaFileUri": "s3://amzn-s3-demo-bucket/your-audio-file.extension"
        },
    "Settings":{
        "ShowSpeakerLabels": true,
        "MaxSpeakerLabels": 2
        }
}
```

Output:

```
{
    "MedicalTranscriptionJob": {
        "MedicalTranscriptionJobName": "speaker-id-conversation-medical-
transcription-job",
        "TranscriptionJobStatus": "IN_PROGRESS",
        "LanguageCode": "language-code",
        "Media": {
            "MediaFileUri": "s3://amzn-s3-demo-bucket/your-audio-file.extension"
        },
        "StartTime": "2020-09-21T18:43:37.265000+00:00",
        "CreationTime": "2020-09-21T18:43:37.157000+00:00",
        "Settings": {
            "ShowSpeakerLabels": true,
            "MaxSpeakerLabels": 2
        },
        "Specialty": "PRIMARYCARE",
        "Type": "CONVERSATION"
    }
}
```

Per ulteriori informazioni, consulta Identificazione delle persone che parlano nella Guida per gli sviluppatori di Amazon Transcribe.

Esempio 5: trascrivere una conversazione medica archiviata come file audio con un massimo di due alternative di trascrizione

L'esempio `start-medical-transcription-job` seguente crea fino a due trascrizioni alternative da un singolo file audio. A ogni trascrizione è associato un livello di affidabilità. Per impostazione predefinita, Amazon Transcribe restituisce la trascrizione con il livello di affidabilità più elevato. Puoi specificare che Amazon Transcribe restituisca trascrizioni aggiuntive con livelli di affidabilità inferiori. Specifica la posizione dell'output della trascrizione nel parametro `OutputBucketName`.

```
aws transcribe start-medical-transcription-job \
    --cli-input-json file://myfifthfile.json
```

Contenuto di `myfifthfile.json`:

```
{
    "MedicalTranscriptionJobName": "alternatives-conversation-medical-transcription-
job",
    "LanguageCode": "language-code",
    "Specialty": "PRIMARYCARE",
    "Type": "CONVERSATION",
    "OutputBucketName":"amzn-s3-demo-bucket",
    "Media": {
        "MediaFileUri": "s3://amzn-s3-demo-bucket/your-audio-file.extension"
    },
    "Settings":{
        "ShowAlternatives": true,
        "MaxAlternatives": 2
    }
}
```

Output:

```
{
    "MedicalTranscriptionJob": {
        "MedicalTranscriptionJobName": "alternatives-conversation-medical-
transcription-job",
        "TranscriptionJobStatus": "IN_PROGRESS",
        "LanguageCode": "language-code",
        "Media": {
            "MediaFileUri": "s3://amzn-s3-demo-bucket/your-audio-file.extension"
        },
        "StartTime": "2020-09-21T19:09:18.199000+00:00",
        "CreationTime": "2020-09-21T19:09:18.171000+00:00",
        "Settings": {
            "ShowAlternatives": true,
            "MaxAlternatives": 2
        },
        "Specialty": "PRIMARYCARE",
        "Type": "CONVERSATION"
    }
}
```

Per ulteriori informazioni, consulta Trascrizioni alternative nella Guida per gli sviluppatori di Amazon Transcribe.

Esempio 6: trascrivere un file audio di una dettatura medica con un massimo di due trascrizioni alternative

L'esempio `start-medical-transcription-job` seguente trascrive un file audio e utilizza un filtro del vocabolario per mascherare eventuali parole indesiderate. Si specifica la posizione dell'output della trascrizione nel parametro. OutputBucketName

```
aws transcribe start-medical-transcription-job \
    --cli-input-json file://mysixthfile.json
```

Contenuto di `mysixthfile.json`:

```
{
    "MedicalTranscriptionJobName": "alternatives-conversation-medical-transcription-
job",
    "LanguageCode": "language-code",
    "Specialty": "PRIMARYCARE",
    "Type": "DICTATION",
    "OutputBucketName":"amzn-s3-demo-bucket",
    "Media": {
        "MediaFileUri": "s3://amzn-s3-demo-bucket/your-audio-file.extension"
    },
    "Settings":{
          "ShowAlternatives": true,
          "MaxAlternatives": 2
    }
}
```

Output:

```
{
    "MedicalTranscriptionJob": {
        "MedicalTranscriptionJobName": "alternatives-dictation-medical-
transcription-job",
        "TranscriptionJobStatus": "IN_PROGRESS",
        "LanguageCode": "language-code",
        "Media": {
            "MediaFileUri": "s3://amzn-s3-demo-bucket/your-audio-file.extension"
        },
```

```
        "StartTime": "2020-09-21T21:01:14.592000+00:00",
        "CreationTime": "2020-09-21T21:01:14.569000+00:00",
        "Settings": {
            "ShowAlternatives": true,
            "MaxAlternatives": 2
        },
        "Specialty": "PRIMARYCARE",
        "Type": "DICTATION"
    }
}
```

Per ulteriori informazioni, consulta Trascrizioni alternative nella Guida per gli sviluppatori di Amazon Transcribe.

Esempio 7: trascrivere un file audio di una dettatura medica con maggiore precisione utilizzando un vocabolario personalizzato

L'esempio `start-medical-transcription-job` seguente trascrive un file audio e utilizza un vocabolario medico personalizzato creato in precedenza per aumentare la precisione della trascrizione. Specifica la posizione dell'output della trascrizione nel parametro. `OutputBucketName`

```
aws transcribe start-transcription-job \
    --cli-input-json file://myseventhfile.json
```

Contenuto di `mysixthfile.json`:

```
{
    "MedicalTranscriptionJobName": "vocabulary-dictation-medical-transcription-job",
    "LanguageCode": "language-code",
    "Specialty": "PRIMARYCARE",
    "Type": "DICTATION",
    "OutputBucketName":"amzn-s3-demo-bucket",
    "Media": {
        "MediaFileUri": "s3://amzn-s3-demo-bucket/your-audio-file.extension"
    },
    "Settings":{
        "VocabularyName": "cli-medical-vocab-1"
    }
}
```

Output:

```
{
    "MedicalTranscriptionJob": {
        "MedicalTranscriptionJobName": "vocabulary-dictation-medical-transcription-
job",
        "TranscriptionJobStatus": "IN_PROGRESS",
        "LanguageCode": "language-code",
        "Media": {
            "MediaFileUri": "s3://amzn-s3-demo-bucket/your-audio-file.extension"
        },
        "StartTime": "2020-09-21T21:17:27.045000+00:00",
        "CreationTime": "2020-09-21T21:17:27.016000+00:00",
        "Settings": {
            "VocabularyName": "cli-medical-vocab-1"
        },
        "Specialty": "PRIMARYCARE",
        "Type": "DICTATION"
    }
}
```

Per ulteriori informazioni, consulta Vocabolari medici personalizzati nella Guida per gli sviluppatori di Amazon Transcribe.

- Per i dettagli sull'API, consulta AWS CLI Command StartMedicalTranscriptionJobReference.

`start-transcription-job`

Il seguente esempio di codice mostra come utilizzare`start-transcription-job`.

AWS CLI

Esempio 1: trascrivere un file audio

L'esempio `start-transcription-job` seguente trascrive il file audio.

```
aws transcribe start-transcription-job \
    --cli-input-json file://myfile.json
```

Contenuto di `myfile.json`:

```
{
    "TranscriptionJobName": "cli-simple-transcription-job",
    "LanguageCode": "the-language-of-your-transcription-job",
```

```
    "Media": {
        "MediaFileUri": "s3://amzn-s3-demo-bucket/Amazon-S3-prefix/your-media-file-
name.file-extension"
    }
}
```

Per ulteriori informazioni, consulta Getting Started (AWS Command Line Interface) nella Amazon Transcribe Developer Guide.

Esempio 2: trascrivere un file audio multicanale

L'esempio `start-transcription-job` seguente trascrive il file audio multicanale.

```
aws transcribe start-transcription-job \
    --cli-input-json file://mysecondfile.json
```

Contenuto di `mysecondfile.json`:

```
{
    "TranscriptionJobName": "cli-channelid-job",
    "LanguageCode": "the-language-of-your-transcription-job",
    "Media": {
        "MediaFileUri": "s3://amzn-s3-demo-bucket/Amazon-S3-prefix/your-media-file-
name.file-extension"
    },
    "Settings":{
        "ChannelIdentification":true
    }
}
```

Output:

```
{
    "TranscriptionJob": {
        "TranscriptionJobName": "cli-channelid-job",
        "TranscriptionJobStatus": "IN_PROGRESS",
        "LanguageCode": "the-language-of-your-transcription-job",
        "Media": {
            "MediaFileUri": "s3://amzn-s3-demo-bucket/Amazon-S3-prefix/your-media-
file-name.file-extension"
        },
        "StartTime": "2020-09-17T16:07:56.817000+00:00",
        "CreationTime": "2020-09-17T16:07:56.784000+00:00",
```

```
        "Settings": {
            "ChannelIdentification": true
        }
    }
}
```

Per ulteriori informazioni, consulta Trascrizione dell'audio multicanale nella Guida per gli sviluppatori di Amazon Transcribe.

Esempio 3: trascrivere un file audio e identificare le diverse persone che parlano

L'esempio `start-transcription-job` seguente trascrive il file audio e identifica le persone che parlano nell'output di trascrizione.

```
aws transcribe start-transcription-job \
    --cli-input-json file://mythirdfile.json
```

Contenuto di `mythirdfile.json`:

```
{
    "TranscriptionJobName": "cli-speakerid-job",
    "LanguageCode": "the-language-of-your-transcription-job",
    "Media": {
        "MediaFileUri": "s3://amzn-s3-demo-bucket/Amazon-S3-prefix/your-media-file-
name.file-extension"
    },
    "Settings":{
    "ShowSpeakerLabels": true,
    "MaxSpeakerLabels": 2
    }
}
```

Output:

```
{
    "TranscriptionJob": {
        "TranscriptionJobName": "cli-speakerid-job",
        "TranscriptionJobStatus": "IN_PROGRESS",
        "LanguageCode": "the-language-of-your-transcription-job",
        "Media": {
            "MediaFileUri": "s3://amzn-s3-demo-bucket/Amazon-S3-prefix/your-media-
file-name.file-extension"
```

```
        },
        "StartTime": "2020-09-17T16:22:59.696000+00:00",
        "CreationTime": "2020-09-17T16:22:59.676000+00:00",
        "Settings": {
            "ShowSpeakerLabels": true,
            "MaxSpeakerLabels": 2
        }
    }
}
```

Per ulteriori informazioni, consulta Identificazione delle persone che parlano nella Guida per gli sviluppatori di Amazon Transcribe.

Esempio 4: trascrivere un file audio e mascherare eventuali parole indesiderate nell'output della trascrizione

L'esempio `start-transcription-job` seguente trascrive il file audio e utilizza un filtro del vocabolario creato in precedenza per mascherare eventuali parole indesiderate.

```
aws transcribe start-transcription-job \
    --cli-input-json file://myfourthfile.json
```

Contenuto di `myfourthfile.json`:

```
{
    "TranscriptionJobName": "cli-filter-mask-job",
    "LanguageCode": "the-language-of-your-transcription-job",
    "Media": {
          "MediaFileUri": "s3://amzn-s3-demo-bucket/Amazon-S3-prefix/your-media-
file-name.file-extension"
    },
    "Settings":{
        "VocabularyFilterName": "your-vocabulary-filter",
        "VocabularyFilterMethod": "mask"
    }
}
```

Output:

```
{
    "TranscriptionJob": {
        "TranscriptionJobName": "cli-filter-mask-job",
```

```
        "TranscriptionJobStatus": "IN_PROGRESS",
        "LanguageCode": "the-language-of-your-transcription-job",
        "Media": {
            "MediaFileUri": "s3://Amazon-S3-Prefix/your-media-file.file-extension"
        },
        "StartTime": "2020-09-18T16:36:18.568000+00:00",
        "CreationTime": "2020-09-18T16:36:18.547000+00:00",
        "Settings": {
            "VocabularyFilterName": "your-vocabulary-filter",
            "VocabularyFilterMethod": "mask"
        }
    }
}
```

Per ulteriori informazioni, consulta Filtraggio delle trascrizioni nella Guida per gli sviluppatori di Amazon Transcribe.

Esempio 5: trascrivere un file audio e rimuovere eventuali parole indesiderate nell'output della trascrizione

L'esempio `start-transcription-job` seguente trascrive il file audio e utilizza un filtro del vocabolario creato in precedenza per mascherare eventuali parole indesiderate.

```
aws transcribe start-transcription-job \
    --cli-input-json file://myfifthfile.json
```

Contenuto di `myfifthfile.json`:

```
{
    "TranscriptionJobName": "cli-filter-remove-job",
    "LanguageCode": "the-language-of-your-transcription-job",
    "Media": {
        "MediaFileUri": "s3://amzn-s3-demo-bucket/Amazon-S3-prefix/your-media-file-
name.file-extension"
    },
    "Settings":{
        "VocabularyFilterName": "your-vocabulary-filter",
        "VocabularyFilterMethod": "remove"
    }
}
```

Output:

```
{
    "TranscriptionJob": {
        "TranscriptionJobName": "cli-filter-remove-job",
        "TranscriptionJobStatus": "IN_PROGRESS",
        "LanguageCode": "the-language-of-your-transcription-job",
        "Media": {
            "MediaFileUri": "s3://amzn-s3-demo-bucket/Amazon-S3-prefix/your-media-
file-name.file-extension"
        },
        "StartTime": "2020-09-18T16:36:18.568000+00:00",
        "CreationTime": "2020-09-18T16:36:18.547000+00:00",
        "Settings": {
            "VocabularyFilterName": "your-vocabulary-filter",
            "VocabularyFilterMethod": "remove"
        }
    }
}
```

Per ulteriori informazioni, consulta Filtraggio delle trascrizioni nella Guida per gli sviluppatori di Amazon Transcribe.

Esempio 6: trascrivere un file audio con maggiore precisione utilizzando un vocabolario personalizzato

L'esempio `start-transcription-job` seguente trascrive il file audio e utilizza un filtro del vocabolario creato in precedenza per mascherare eventuali parole indesiderate.

```
aws transcribe start-transcription-job \
    --cli-input-json file://mysixthfile.json
```

Contenuto di `mysixthfile.json`:

```
{
    "TranscriptionJobName": "cli-vocab-job",
    "LanguageCode": "the-language-of-your-transcription-job",
    "Media": {
        "MediaFileUri": "s3://amzn-s3-demo-bucket/Amazon-S3-prefix/your-media-file-
name.file-extension"
    },
    "Settings":{
        "VocabularyName": "your-vocabulary"
    }
```

```
}
```

Output:

```
{
    "TranscriptionJob": {
        "TranscriptionJobName": "cli-vocab-job",
        "TranscriptionJobStatus": "IN_PROGRESS",
        "LanguageCode": "the-language-of-your-transcription-job",
        "Media": {
            "MediaFileUri": "s3://amzn-s3-demo-bucket/Amazon-S3-prefix/your-media-
file-name.file-extension"
        },
        "StartTime": "2020-09-18T16:36:18.568000+00:00",
        "CreationTime": "2020-09-18T16:36:18.547000+00:00",
        "Settings": {
            "VocabularyName": "your-vocabulary"
        }
    }
}
```

Per ulteriori informazioni, consulta Filtraggio delle trascrizioni nella Guida per gli sviluppatori di Amazon Transcribe.

Esempio 7: identificare la lingua di un file audio e trascriverla

L'esempio `start-transcription-job` seguente trascrive il file audio e utilizza un filtro del vocabolario creato in precedenza per mascherare eventuali parole indesiderate.

```
aws transcribe start-transcription-job \
    --cli-input-json file://myseventhfile.json
```

Contenuto di `myseventhfile.json`:

```
{
    "TranscriptionJobName": "cli-identify-language-transcription-job",
    "IdentifyLanguage": true,
    "Media": {
        "MediaFileUri": "s3://amzn-s3-demo-bucket/Amazon-S3-prefix/your-media-file-
name.file-extension"
    }
```

```
}
```

Output:

```
{
    "TranscriptionJob": {
        "TranscriptionJobName": "cli-identify-language-transcription-job",
        "TranscriptionJobStatus": "IN_PROGRESS",
        "Media": {
            "MediaFileUri": "s3://amzn-s3-demo-bucket/Amazon-S3-prefix/your-media-
file-name.file-extension"
        },
        "StartTime": "2020-09-18T22:27:23.970000+00:00",
        "CreationTime": "2020-09-18T22:27:23.948000+00:00",
        "IdentifyLanguage": true
    }
}
```

Per ulteriori informazioni, consulta Identificazione della lingua nella Guida per gli sviluppatori di Amazon Transcribe.

Esempio 8: trascrivere un file audio con informazioni di identificazione personale nascoste

L'esempio `start-transcription-job` seguente trascrive il file audio e nasconde le informazioni di identificazione personale nell'output di trascrizione.

```
aws transcribe start-transcription-job \
    --cli-input-json file://myeighthfile.json
```

Contenuto di `myeigthfile.json`:

```
{
    "TranscriptionJobName": "cli-redaction-job",
    "LanguageCode": "language-code",
    "Media": {
        "MediaFileUri": "s3://Amazon-S3-Prefix/your-media-file.file-extension"
    },
    "ContentRedaction": {
        "RedactionOutput":"redacted",
        "RedactionType":"PII"
    }
}
```

Output:

```
{
    "TranscriptionJob": {
        "TranscriptionJobName": "cli-redaction-job",
        "TranscriptionJobStatus": "IN_PROGRESS",
        "LanguageCode": "language-code",
        "Media": {
            "MediaFileUri": "s3://Amazon-S3-Prefix/your-media-file.file-extension"
        },
        "StartTime": "2020-09-25T23:49:13.195000+00:00",
        "CreationTime": "2020-09-25T23:49:13.176000+00:00",
        "ContentRedaction": {
            "RedactionType": "PII",
            "RedactionOutput": "redacted"
        }
    }
}
```

Per ulteriori informazioni, consulta Redazione dei contenuti automatici nella Guida per gli sviluppatori di Amazon Transcribe.

Esempio 9: generare una trascrizione con informazioni di identificazione personale (PII) redatte e una trascrizione non redatta

L'esempio `start-transcription-job` seguente genera due trascrizioni del file audio, una con le informazioni di identificazione personale redatte e l'altra senza alcuna redazione.

```
aws transcribe start-transcription-job \
    --cli-input-json file://myninthfile.json
```

Contenuto di `myninthfile.json`:

```
{
    "TranscriptionJobName": "cli-redaction-job-with-unredacted-transcript",
    "LanguageCode": "language-code",
    "Media": {
          "MediaFileUri": "s3://Amazon-S3-Prefix/your-media-file.file-extension"
        },
    "ContentRedaction": {
        "RedactionOutput":"redacted_and_unredacted",
        "RedactionType":"PII"
```

```
    }
}
```

Output:

```
{
    "TranscriptionJob": {
        "TranscriptionJobName": "cli-redaction-job-with-unredacted-transcript",
        "TranscriptionJobStatus": "IN_PROGRESS",
        "LanguageCode": "language-code",
        "Media": {
            "MediaFileUri": "s3://Amazon-S3-Prefix/your-media-file.file-extension"
        },
        "StartTime": "2020-09-25T23:59:47.677000+00:00",
        "CreationTime": "2020-09-25T23:59:47.653000+00:00",
        "ContentRedaction": {
            "RedactionType": "PII",
            "RedactionOutput": "redacted_and_unredacted"
        }
    }
}
```

Per ulteriori informazioni, consulta Redazione dei contenuti automatici nella Guida per gli sviluppatori di Amazon Transcribe.

Esempio 10: utilizzare un modello linguistico personalizzato creato in precedenza per trascrivere un file audio.

L'esempio `start-transcription-job` seguente trascrive il file audio con un modello linguistico personalizzato creato in precedenza.

```
aws transcribe start-transcription-job \
    --cli-input-json file://mytenthfile.json
```

Contenuto di `mytenthfile.json`:

```
{
    "TranscriptionJobName": "cli-clm-2-job-1",
    "LanguageCode": "language-code",
    "Media": {
        "MediaFileUri": "s3://amzn-s3-demo-bucket/your-audio-file.file-extension"
    },
```

```
    "ModelSettings": {
        "LanguageModelName":"cli-clm-2"
    }
}
```

Output:

```
{
    "TranscriptionJob": {
        "TranscriptionJobName": "cli-clm-2-job-1",
        "TranscriptionJobStatus": "IN_PROGRESS",
        "LanguageCode": "language-code",
        "Media": {
            "MediaFileUri": "s3://amzn-s3-demo-bucket/your-audio-file.file-
extension"
        },
        "StartTime": "2020-09-28T17:56:01.835000+00:00",
        "CreationTime": "2020-09-28T17:56:01.801000+00:00",
        "ModelSettings": {
            "LanguageModelName": "cli-clm-2"
        }
    }
}
```

Per ulteriori informazioni, consulta Modelli linguistici personalizzati nella Guida per gli sviluppatori di Amazon Transcribe.

- Per i dettagli sull'API, consulta AWS CLI Command StartTranscriptionJobReference.

`update-medical-vocabulary`

Il seguente esempio di codice mostra come utilizzare`update-medical-vocabulary`.

AWS CLI

Come aggiornare un vocabolario medico personalizzato con nuovi termini.

L'esempio `update-medical-vocabulary` seguente sostituisce i termini utilizzati in un vocabolario personalizzato medico con quelli nuovi. Prerequisito: per sostituire i termini in un vocabolario medico personalizzato, è necessario disporre di un file con i nuovi termini.

```
aws transcribe update-medical-vocabulary \
```

```
    --vocabulary-file-uri s3://amzn-s3-demo-bucket/Amazon-S3-Prefix/medical-custom-
vocabulary.txt \
    --vocabulary-name medical-custom-vocabulary \
    --language-code language
```

Output:

```
{
    "VocabularyName": "medical-custom-vocabulary",
    "LanguageCode": "en-US",
    "VocabularyState": "PENDING"
}
```

Per ulteriori informazioni, consulta Vocabolari medici personalizzati nella Guida per gli sviluppatori di Amazon Transcribe.

- Per i dettagli sull'API, consulta UpdateMedicalVocabulary AWS CLICommand Reference.

`update-vocabulary-filter`

Il seguente esempio di codice mostra come utilizzare`update-vocabulary-filter`.

AWS CLI

Come sostituire le parole in un filtro di vocabolario

L'esempio `update-vocabulary-filter` seguente sostituisce le parole in un filtro di vocabolario con nuove parole. Prerequisito: per aggiornare un filtro di vocabolario con le nuove parole, è necessario che tali parole siano salvate come file di testo.

```
aws transcribe update-vocabulary-filter \
    --vocabulary-filter-file-uri s3://amzn-s3-demo-bucket/Amazon-S3-Prefix/your-
text-file-to-update-your-vocabulary-filter.txt \
    --vocabulary-filter-name vocabulary-filter-name
```

Output:

```
{
    "VocabularyFilterName": "vocabulary-filter-name",
    "LanguageCode": "language-code",
    "LastModifiedTime": "2020-09-23T18:40:35.139000+00:00"
}
```

Per ulteriori informazioni, consulta Filtraggio di parole indesiderate nella Guida per gli sviluppatori di Amazon Transcribe.

- Per i dettagli sull'API, consulta UpdateVocabularyFilter AWS CLICommand Reference.

`update-vocabulary`

Il seguente esempio di codice mostra come utilizzare`update-vocabulary`.

AWS CLI

Aggiornamento di un vocabolario personalizzato con nuovi termini.

L'esempio `update-vocabulary` seguente sovrascrive i termini utilizzati per creare un vocabolario personalizzato con quelli nuovi forniti dall'utente. Prerequisito: per sostituire i termini in un vocabolario personalizzato, è necessario disporre di un file con nuovi termini.

```
aws transcribe update-vocabulary \
    --vocabulary-file-uri s3://amzn-s3-demo-bucket/Amazon-S3-Prefix/custom-vocabulary.txt \
    --vocabulary-name custom-vocabulary \
    --language-code language-code
```

Output:

```
{
    "VocabularyName": "custom-vocabulary",
    "LanguageCode": "language",
    "VocabularyState": "PENDING"
}
```

Per ulteriori informazioni, consulta Vocabolari personalizzati nella Guida per gli sviluppatori di Amazon Transcribe.

- Per i dettagli sull'API, consulta UpdateVocabulary AWS CLICommand Reference.

Esempi di Amazon Translate con AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando Amazon Translate. AWS Command Line Interface

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`import-terminology`

Il seguente esempio di codice mostra come usare`import-terminology`.

AWS CLI

Come importare una terminologia personalizzata da un file

L'esempio `import-terminology` seguente crea una terminologia chiamata `MyTestTerminology` dal file `test-terminology.csv`:

```
aws translate import-terminology \
    --name MyTestTerminology \
    --description "Creating a test terminology in AWS Translate" \
    --merge-strategy OVERWRITE \
    --data-file fileb://test-terminology.csv \
    --terminology-data Format=CSV
```

Contenuto di `test-terminology.csv`:

en,fr,es,zh Hello world!,Bonjour tout le monde!,Hola Mundo!,????
Amazon,Amazon,Amazon,Amazon

Output:

```
{
    "TerminologyProperties": {
        "SourceLanguageCode": "en",
        "Name": "MyTestTerminology",
```

```
        "TargetLanguageCodes": [
            "fr",
            "es",
            "zh"
        ],
        "SizeBytes": 97,
        "LastUpdatedAt": 1571089500.851,
        "CreatedAt": 1571089500.851,
        "TermCount": 6,
        "Arn": "arn:aws:translate:us-west-2:123456789012:terminology/
MyTestTerminology/LATEST",
        "Description": "Creating a test terminology in AWS Translate"
    }
}
```

- Per i dettagli sull'API, consulta ImportTerminology AWS CLICommand Reference.

Trusted Advisor esempi utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with Trusted Advisor.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`get-organization-recommendation`

Il seguente esempio di codice mostra come utilizzare`get-organization-recommendation`.

AWS CLI

Come ottenere una raccomandazione sull'organizzazione

L'esempio `get-organization-recommendation` seguente ottiene una raccomandazione a livello di organizzazione in base al relativo identificatore.

```
aws trustedadvisor get-organization-recommendation \
    --organization-recommendation-identifier arn:aws:trustedadvisor:::organization-recommendation/9534ec9b-bf3a-44e8-8213-2ed68b39d9d5
```

Output:

```
{
    "organizationRecommendation": {
        "arn": "arn:aws:trustedadvisor:::organization-recommendation/9534ec9b-bf3a-44e8-8213-2ed68b39d9d5",
        "name": "Lambda Runtime Deprecation Warning",
        "description": "One or more lambdas are using a deprecated runtime",
        "awsServices": [
            "lambda"
        ],
        "checkArn": "arn:aws:trustedadvisor:::check/L4dfs2Q4C5",
        "id": "9534ec9b-bf3a-44e8-8213-2ed68b39d9d5",
        "lifecycleStage": "resolved",
        "pillars": [
            "security"
        ],
        "resourcesAggregates": {
            "errorCount": 0,
            "okCount": 0,
            "warningCount": 0
        },
        "source": "ta_check",
        "status": "warning",
        "type": "priority"
    }
}
```

Per ulteriori informazioni, consulta Nozioni di base sull'API Trusted Advisor nella Guida per l'utente di AWS Trusted Advisor.

- Per i dettagli sull'API, consulta GetOrganizationRecommendation AWS CLICommand Reference.

`get-recommendation`

Il seguente esempio di codice mostra come utilizzare`get-recommendation`.

AWS CLI

Come ottenere una raccomandazione

L'esempio `get-recommendation` seguente ottiene una raccomandazione in base al relativo identificatore.

```
aws trustedadvisor get-recommendation \
    --recommendation-
identifier arn:aws:trustedadvisor::000000000000:recommendation/55fa4d2e-
bbb7-491a-833b-5773e9589578
```

Output:

```
{
    "recommendation": {
        "arn": "arn:aws:trustedadvisor::000000000000:recommendation/55fa4d2e-
bbb7-491a-833b-5773e9589578",
        "name": "MFA Recommendation",
        "description": "Enable multi-factor authentication",
        "awsServices": [
            "iam"
        ],
        "checkArn": "arn:aws:trustedadvisor:::check/7DAFEmoDos",
        "id": "55fa4d2e-bbb7-491a-833b-5773e9589578",
        "lastUpdatedAt": "2023-11-01T15:57:58.673Z",
        "pillarSpecificAggregates": {
            "costOptimizing": {
                "estimatedMonthlySavings": 0.0,
                "estimatedPercentMonthlySavings": 0.0
            }
        },
        "pillars": [
            "security"
        ],
        "resourcesAggregates": {
            "errorCount": 1,
            "okCount": 0,
            "warningCount": 0
```

```
        },
        "source": "ta_check",
        "status": "error",
        "type": "standard"
    }
}
```

Per ulteriori informazioni, consulta Nozioni di base sull'API Trusted Advisor nella Guida per l'utente di AWS Trusted Advisor.

- Per i dettagli sull'API, consulta GetRecommendation AWS CLICommand Reference.

`list-checks`

Il seguente esempio di codice mostra come utilizzare`list-checks`.

AWS CLI

Come elencare i controlli di Trusted Advisor

L'esempio `list-checks` seguente elenca tutti i controlli di Trusted Advisor.

```
aws trustedadvisor list-checks
```

Output:

```
{
    "checkSummaries": [
        {
            "arn": "arn:aws:trustedadvisor:::check/1iG5NDGVre",
            "awsServices": [
                "EC2"
            ],
            "description": "Checks security groups for rules that allow unrestricted
 access to a resource. Unrestricted access increases opportunities for malicious
 activity (hacking, denial-of-service attacks, loss of data)",
            "id": "1iG5NDGVre",
            "metadata": {
                "0": "Region",
                "1": "Security Group Name",
                "2": "Security Group ID",
                "3": "Protocol",
                "4": "Port",
```

```
                "5": "Status",
                "6": "IP Range"
            },
            "name": "Security Groups - Unrestricted Access",
            "pillars": [
                "security"
            ],
            "source": "ta_check"
        },
        {
            "arn": "arn:aws:trustedadvisor:::check/1qazXsw23e",
            "awsServices": [
                "RDS"
            ],
            "description": "Checks your usage of RDS and provides recommendations
on purchase of Reserved Instances to help reduce costs incurred from using RDS
On-Demand. AWS generates these recommendations by analyzing your On-Demand usage
for the past 30 days. We then simulate every combination of reservations in the
generated category of usage in order to identify the best number of each type
of Reserved Instance to purchase to maximize your savings. This check covers
recommendations based on partial upfront payment option with 1-year or 3-year
commitment. This check is not available to accounts linked in Consolidated Billing.
Recommendations are only available for the Paying Account.",
            "id": "1qazXsw23e",
            "metadata": {
                "0": "Region",
                "1": "Family",
                "2": "Instance Type",
                "3": "License Model",
                "4": "Database Edition",
                "5": "Database Engine",
                "6": "Deployment Option",
                "7": "Recommended number of Reserved Instances to purchase",
                "8": "Expected Average Reserved Instance Utilization",
                "9": "Estimated Savings with Recommendation (monthly)"
                "10": "Upfront Cost of Reserved Instances",
                "11": "Estimated cost of Reserved Instances (monthly)",
                "12": "Estimated On-Demand Cost Post Recommended Reserved Instance
Purchase (monthly)",
                "13": "Estimated Break Even (months)",
                "14": "Lookback Period (days)",
                "15": "Term (years)"
            },
```

```
            "name": "Amazon Relational Database Service (RDS) Reserved Instance
Optimization",
            "pillars": [
                "cost_optimizing"
            ],
            "source": "ta_check"
        },
        {
            "arn": "arn:aws:trustedadvisor:::check/1qw23er45t",
            "awsServices": [
                "Redshift"
            ],
            "description": "Checks your usage of Redshift and provides
recommendations on purchase of Reserved Nodes to help reduce costs incurred from
using Redshift On-Demand. AWS generates these recommendations by analyzing your
On-Demand usage for the past 30 days. We then simulate every combination of
reservations in the generated category of usage in order to identify the best
number of each type of Reserved Nodes to purchase to maximize your savings. This
check covers recommendations based on partial upfront payment option with 1-year or
3-year commitment. This check is not available to accounts linked in Consolidated
Billing. Recommendations are only available for the Paying Account.",
            "id": "1qw23er45t",
            "metadata": {
                "0": "Region",
                "1": "Family",
                "2": "Node Type",
                "3": "Recommended number of Reserved Nodes to purchase",
                "4": "Expected Average Reserved Node Utilization",
                "5": "Estimated Savings with Recommendation (monthly)",
                "6": "Upfront Cost of Reserved Nodes",
                "7": "Estimated cost of Reserved Nodes (monthly)",
                "8": "Estimated On-Demand Cost Post Recommended Reserved Nodes
Purchase (monthly)",
                "9": "Estimated Break Even (months)",
                "10": "Lookback Period (days)",
                "11": "Term (years)",
            },
            "name": "Amazon Redshift Reserved Node Optimization",
            "pillars": [
                "cost_optimizing"
            ],
            "source": "ta_check"
        },
    ],
```

```
    "nextToken": "REDACTED"
}
```

Per ulteriori informazioni, consulta Nozioni di base sull'API Trusted Advisor nella Guida per l'utente di AWS Trusted Advisor.

- Per i dettagli sull'API, consulta ListChecks AWS CLICommand Reference.

`list-organization-recommendation-accounts`

Il seguente esempio di codice mostra come utilizzare`list-organization-recommendation-accounts`.

AWS CLI

Come elencare gli account di raccomandazione dell'organizzazione

L'esempio `list-organization-recommendation-accounts` seguente elenca tutti i riepiloghi delle raccomandazioni relative agli account per una raccomandazione a livello di organizzazione in base al relativo identificatore.

```
aws trustedadvisor list-organization-recommendation-accounts \
    --organization-recommendation-identifier arn:aws:trustedadvisor:::organization-recommendation/9534ec9b-bf3a-44e8-8213-2ed68b39d9d5
```

Output:

```
{
    "accountRecommendationLifecycleSummaries": [{
        "accountId": "000000000000",
        "accountRecommendationArn":
 "arn:aws:trustedadvisor::000000000000:recommendation/9534ec9b-
bf3a-44e8-8213-2ed68b39d9d5",
        "lifecycleStage": "resolved",
        "updateReason": "Resolved issue",
        "updateReasonCode": "valid_business_case",
        "lastUpdatedAt": "2023-01-17T18:25:44.552Z"
    }],
    "nextToken": "REDACTED"
}
```

Per ulteriori informazioni, consulta Nozioni di base sull'API Trusted Advisor nella Guida per l'utente di AWS Trusted Advisor.

- Per i dettagli sull'API, consulta ListOrganizationRecommendationAccounts AWS CLICommand Reference.

`list-organization-recommendation-resources`

Il seguente esempio di codice mostra come utilizzare`list-organization-recommendation-resources`.

AWS CLI

Come elencare le risorse di raccomandazione dell'organizzazione

L'esempio `list-organization-recommendation-resources` seguente elenca tutte le risorse per una raccomandazione a livello di organizzazione in base al relativo identificatore.

```
aws trustedadvisor list-organization-recommendation-resources \
    --organization-recommendation-identifier arn:aws:trustedadvisor:::organization-recommendation/5a694939-2e54-45a2-ae72-730598fa89d0
```

Output:

```
{
    "organizationRecommendationResourceSummaries": [
        {
            "arn": "arn:aws:trustedadvisor::000000000000:recommendation-resource/5a694939-2e54-45a2-ae72-730598fa89d0/bb38affc0ce0681d9a6cd13f30238ba03a8f63dfe7a379dc403c619119d86af",
            "awsResourceId": "database-1-instance-1",
            "id": "bb38affc0ce0681d9a6cd13f302383ba03a8f63dfe7a379dc403c619119d86af",
            "lastUpdatedAt": "2023-11-01T15:09:51.891Z",
            "metadata": {
                "0": "14",
                "1": "208.79999999999998",
                "2": "database-1-instance-1",
                "3": "db.r5.large",
                "4": "false",
                "5": "us-west-2",
                "6": "arn:aws:rds:us-west-2:000000000000:db:database-1-instance-1",
                "7": "1"
```

```
            },
            "recommendationArn": "arn:aws:trustedadvisor:::organization-
recommendation/5a694939-2e54-45a2-ae72-730598fa89d0",
            "regionCode": "us-west-2",
            "status": "warning"
        },
        {
            "arn": "arn:aws:trustedadvisor::000000000000:recommendation-
resource/5a694939-2e54-45a2-
ae72-730598fa89d0/51fded4d7a3278818df9cfe344ff5762cec46c095a6763d1ba1ba53bd0e1b0e6",
            "awsResourceId": "database-1",
            "id":
 "51fded4d7a3278818df9cfe344ff5762cec46c095a6763d1ba1ba53bd0e1b0e6",
            "lastUpdatedAt": "2023-11-01T15:09:51.891Z",
            "metadata": {
                "0": "14",
                "1": "31.679999999999996",
                "2": "database-1",
                "3": "db.t3.small",
                "4": "false",
                "5": "us-west-2",
                "6": "arn:aws:rds:us-west-2:000000000000:db:database-1",
                "7": "20"
            },
            "recommendationArn": "arn:aws:trustedadvisor:::organization-
recommendation/5a694939-2e54-45a2-ae72-730598fa89d0",
            "regionCode": "us-west-2",
            "status": "warning"
        },
        {
            "arn": "arn:aws:trustedadvisor::000000000000:recommendation-
resource/5a694939-2e54-45a2-ae72-730598fa89d0/
f4d01bd20f4cd5372062aafc8786c489e48f0ead7cdab121463bf9f89e40a36b",
            "awsResourceId": "database-2-instance-1-us-west-2a",
            "id":
 "f4d01bd20f4cd5372062aafc8786c489e48f0ead7cdab121463bf9f89e40a36b",
            "lastUpdatedAt": "2023-11-01T15:09:51.891Z",
            "metadata": {
                "0": "14",
                "1": "187.20000000000002",
                "2": "database-2-instance-1-us-west-2a",
                "3": "db.r6g.large",
                "4": "true",
                "5": "us-west-2",
```

```
                "6": "arn:aws:rds:us-west-2:000000000000:db:database-2-instance-1-
us-west-2a",
                "7": "1"
            },
            "recommendationArn": "arn:aws:trustedadvisor:::organization-
recommendation/5a694939-2e54-45a2-ae72-730598fa89d0",
            "regionCode": "us-west-2",
            "status": "warning"
        },
    ],
    "nextToken": "REDACTED"
}
```

Per ulteriori informazioni, consulta Nozioni di base sull'API Trusted Advisor nella Guida per l'utente di AWS Trusted Advisor.

- Per i dettagli sull'API, consulta ListOrganizationRecommendationResources AWS CLICommand Reference.

`list-organization-recommendations`

Il seguente esempio di codice mostra come utilizzare`list-organization-recommendations`.

AWS CLI

Esempio 1: come elencare le raccomandazioni a livello di l'organizzazione

L'esempio `list-organization-recommendations` seguente elenca tutte le raccomandazioni a livello di organizzazione e non include un filtro.

```
aws trustedadvisor list-organization-recommendations
```

Output:

```
{
    "organizationRecommendationSummaries": [
        {
            "arn": "arn:aws:trustedadvisor:::organization-recommendation/9534ec9b-
bf3a-44e8-8213-2ed68b39d9d5",
            "name": "Lambda Runtime Deprecation Warning",
            "awsServices": [
                "lambda"
            ],
```

```
            "checkArn": "arn:aws:trustedadvisor:::check/L4dfs2Q4C5",
            "id": "9534ec9b-bf3a-44e8-8213-2ed68b39d9d5",
            "lifecycleStage": "resolved",
            "pillars": [
                "security"
            ],
            "resourcesAggregates": {
                "errorCount": 0,
                "okCount": 0,
                "warningCount": 0
            },
            "source": "ta_check",
            "status": "warning",
            "type": "priority"
        },
        {
            "arn": "arn:aws:trustedadvisor:::organization-
recommendation/4ecff4d4-1bc1-4c99-a5b8-0fff9ee500d6",
            "name": "Lambda Runtime Deprecation Warning",
            "awsServices": [
                "lambda"
            ],
            "checkArn": "arn:aws:trustedadvisor:::check/L4dfs2Q4C5",
            "id": "4ecff4d4-1bc1-4c99-a5b8-0fff9ee500d6",
            "lifecycleStage": "resolved",
            "pillars": [
                "security"
            ],
            "resourcesAggregates": {
                "errorCount": 0,
                "okCount": 0,
                "warningCount": 0
            },
            "source": "ta_check",
            "status": "warning",
            "type": "priority"
        },
    ],
    "nextToken": "REDACTED"
}
```


Per ulteriori informazioni, consulta Nozioni di base sull'API Trusted Advisor nella Guida per l'utente di AWS Trusted Advisor.

Esempio 2: come elencare le raccomandazioni a livello di organizzazione con un filtro

L'esempio `list-organization-recommendations` seguente filtra e restituisce al massimo una raccomandazione a livello di organizzazione appartenente al pilastro della sicurezza.

```
aws trustedadvisor list-organization-recommendations \
    --pillar security \
    --max-items 100
```

Output:

```
{
    "organizationRecommendationSummaries": [{
        "arn": "arn:aws:trustedadvisor:::organization-recommendation/9534ec9b-
bf3a-44e8-8213-2ed68b39d9d5",
        "name": "Lambda Runtime Deprecation Warning",
        "awsServices": [
            "lambda"
        ],
        "checkArn": "arn:aws:trustedadvisor:::check/L4dfs2Q4C5",
        "id": "9534ec9b-bf3a-44e8-8213-2ed68b39d9d5",
        "lifecycleStage": "resolved",
        "pillars": [
            "security"
        ],
        "resourcesAggregates": {
            "errorCount": 0,
            "okCount": 0,
            "warningCount": 0
        },
        "source": "ta_check",
        "status": "warning",
        "type": "priority"
    }],
    "nextToken": "REDACTED"
}
```

Per ulteriori informazioni, consulta Nozioni di base sull'API Trusted Advisor nella Guida per l'utente di AWS Trusted Advisor.

Esempio 3: come elencare le raccomandazioni a livello di organizzazione con un token di paginazione

L'esempio `list-organization-recommendations` seguente utilizza il valore "nextToken" restituito da una richiesta precedente per recuperare la pagina successiva delle raccomandazioni a livello di organizzazione.

```
aws trustedadvisor list-organization-recommendations \
    --pillar security \
    --max-items 100 \
    --starting-token <next-token>
```

Output:

```
{
    "organizationRecommendationSummaries": [{
        "arn": "arn:aws:trustedadvisor:::organization-
recommendation/4ecff4d4-1bc1-4c99-a5b8-0fff9ee500d6",
        "name": "Lambda Runtime Deprecation Warning",
        "awsServices": [
            "lambda"
        ],
        "checkArn": "arn:aws:trustedadvisor:::check/L4dfs2Q4C5",
        "id": "4ecff4d4-1bc1-4c99-a5b8-0fff9ee500d6",
        "lifecycleStage": "resolved",
        "pillars": [
            "security"
        ],
        "resourcesAggregates": {
            "errorCount": 0,
            "okCount": 0,
            "warningCount": 0
        },
        "source": "ta_check",
        "status": "warning",
        "type": "priority"
    }]
}
```

Per ulteriori informazioni, consulta Nozioni di base sull'API Trusted Advisor nella Guida per l'utente di AWS Trusted Advisor.

- Per i dettagli sull'API, consulta ListOrganizationRecommendations AWS CLICommand Reference.

`list-recommendation-resources`

Il seguente esempio di codice mostra come utilizzare`list-recommendation-resources`.

AWS CLI

Come elencare le risorse di raccomandazione

L'esempio `list-recommendation-resources` seguente elenca tutte le risorse per una raccomandazione in base al relativo identificatore.

```
aws trustedadvisor list-recommendation-resources \
    --recommendation-
identifier arn:aws:trustedadvisor::000000000000:recommendation/55fa4d2e-
bbb7-491a-833b-5773e9589578
```

Output:

```
{
    "recommendationResourceSummaries": [
        {
            "arn": "arn:aws:trustedadvisor::000000000000:recommendation-
resource/55fa4d2e-
bbb7-491a-833b-5773e9589578/18959a1f1973cff8e706e9d9bde28bba36cd602a6b2cb86c8b61252835236010
            "id":
 "18959a1f1973cff8e706e9d9bde28bba36cd602a6b2cb86c8b61252835236010",
            "awsResourceId": "webcms-dev-01",
            "lastUpdatedAt": "2023-11-01T15:09:51.891Z",
            "metadata": {
                "0": "14",
                "1": "123.12000000000002",
                "2": "webcms-dev-01",
                "3": "db.m6i.large",
                "4": "false",
                "5": "us-east-1",
                "6": "arn:aws:rds:us-east-1:000000000000:db:webcms-dev-01",
                "7": "20"
            },
            "recommendationArn":
 "arn:aws:trustedadvisor::000000000000:recommendation/55fa4d2e-
bbb7-491a-833b-5773e9589578",
            "regionCode": "us-east-1",
            "status": "warning"
```

```
        },
        {
            "arn": "arn:aws:trustedadvisor::000000000000:recommendation-
resource/55fa4d2e-bbb7-491a-833b-5773e9589578/
e6367ff500ac90db8e4adeb4892e39ee9c36bbf812dcbce4b9e4fefcec9eb63e",
            "id":
 "e6367ff500ac90db8e4adeb4892e39ee9c36bbf812dcbce4b9e4fefcec9eb63e",
            "awsResourceId": "aws-dev-db-stack-instance-1",
            "lastUpdatedAt": "2023-11-01T15:09:51.891Z",
            "metadata": {
                "0": "14",
                "1": "29.52",
                "2": "aws-dev-db-stack-instance-1",
                "3": "db.t2.small",
                "4": "false",
                "5": "us-east-1",
                "6": "arn:aws:rds:us-east-1:000000000000:db:aws-dev-db-stack-
instance-1",
                "7": "1"
            },
            "recommendationArn":
 "arn:aws:trustedadvisor::000000000000:recommendation/55fa4d2e-
bbb7-491a-833b-5773e9589578",
            "regionCode": "us-east-1",
            "status": "warning"
        },
        {
            "arn": "arn:aws:trustedadvisor::000000000000:recommendation-
resource/55fa4d2e-
bbb7-491a-833b-5773e9589578/31aa78ba050a5015d2d38cca7f5f1ce88f70857c4e1c3ad03f8f9fd95dad7459
            "id":
 "31aa78ba050a5015d2d38cca7f5f1ce88f70857c4e1c3ad03f8f9fd95dad7459",
            "awsResourceId": "aws-awesome-apps-stack-db",
            "lastUpdatedAt": "2023-11-01T15:09:51.891Z",
            "metadata": {
                "0": "14",
                "1": "114.48000000000002",
                "2": "aws-awesome-apps-stack-db",
                "3": "db.m6g.large",
                "4": "false",
                "5": "us-east-1",
                "6": "arn:aws:rds:us-east-1:000000000000:db:aws-awesome-apps-stack-
db",
                "7": "100"
```

```
            },
            "recommendationArn":
 "arn:aws:trustedadvisor::000000000000:recommendation/55fa4d2e-
bbb7-491a-833b-5773e9589578",
            "regionCode": "us-east-1",
            "status": "warning"
        }
    ],
    "nextToken": "REDACTED"
}
```

Per ulteriori informazioni, consulta Nozioni di base sull'API Trusted Advisor nella Guida per l'utente di AWS Trusted Advisor.

- Per i dettagli sull'API, consulta ListRecommendationResources AWS CLICommand Reference.

`list-recommendations`

Il seguente esempio di codice mostra come utilizzare`list-recommendations`.

AWS CLI

Esempio 1: come elencare raccomandazioni

L'esempio `list-recommendations` seguente elenca tutte le raccomandazioni e non include un filtro.

```
aws trustedadvisor list-recommendations
```

Output:

```
{
    "recommendationSummaries": [
        {
            "arn": "arn:aws:trustedadvisor::000000000000:recommendation/55fa4d2e-
bbb7-491a-833b-5773e9589578",
            "name": "MFA Recommendation",
            "awsServices": [
                "iam"
            ],
            "checkArn": "arn:aws:trustedadvisor:::check/7DAFEmoDos",
            "id": "55fa4d2e-bbb7-491a-833b-5773e9589578",
            "lastUpdatedAt": "2023-11-01T15:57:58.673Z",
```

```
            "pillarSpecificAggregates": {
                "costOptimizing": {
                    "estimatedMonthlySavings": 0.0,
                    "estimatedPercentMonthlySavings": 0.0
                }
            },
            "pillars": [
                "security"
            ],
            "resourcesAggregates": {
                "errorCount": 1,
                "okCount": 0,
                "warningCount": 0
            },
            "source": "ta_check",
            "status": "error",
            "type": "standard"
        },
        {
            "arn":
 "arn:aws:trustedadvisor::000000000000:recommendation/8b602b6f-452d-4cb2-8a9e-
c7650955d9cd",
            "name": "RDS clusters quota warning",
            "awsServices": [
                "rds"
            ],
            "checkArn": "arn:aws:trustedadvisor:::check/gjqMBn6pjz",
            "id": "8b602b6f-452d-4cb2-8a9e-c7650955d9cd",
            "lastUpdatedAt": "2023-11-01T15:58:17.397Z",
            "pillarSpecificAggregates": {
                "costOptimizing": {
                    "estimatedMonthlySavings": 0.0,
                    "estimatedPercentMonthlySavings": 0.0
                }
            },
            "pillars": [
                "service_limits"
            ],
            "resourcesAggregates": {
                "errorCount": 0,
                "okCount": 3,
                "warningCount": 6
            },
            "source": "ta_check",
```

```
            "status": "warning",
            "type": "standard"
        }
    ],
    "nextToken": "REDACTED"
}
```

Per ulteriori informazioni, consulta Nozioni di base sull'API Trusted Advisor nella Guida per l'utente di AWS Trusted Advisor.

Esempio 2: come elencare raccomandazioni con un filtro

L'esempio `list-recommendations` seguente elenca le raccomandazioni e include un filtro.

```
aws trustedadvisor list-recommendations \
    --aws-service iam \
    --max-items 100
```

Output:

```
{
    "recommendationSummaries": [{
        "arn": "arn:aws:trustedadvisor::000000000000:recommendation/55fa4d2e-
bbb7-491a-833b-5773e9589578",
        "name": "MFA Recommendation",
        "awsServices": [
            "iam"
        ],
        "checkArn": "arn:aws:trustedadvisor:::check/7DAFEmoDos",
        "id": "55fa4d2e-bbb7-491a-833b-5773e9589578",
        "lastUpdatedAt": "2023-11-01T15:57:58.673Z",
        "pillarSpecificAggregates": {
            "costOptimizing": {
                "estimatedMonthlySavings": 0.0,
                "estimatedPercentMonthlySavings": 0.0
            }
        },
        "pillars": [
            "security"
        ],
        "resourcesAggregates": {
            "errorCount": 1,
```

```
            "okCount": 0,
            "warningCount": 0
        },
        "source": "ta_check",
        "status": "error",
        "type": "standard"
    }],
    "nextToken": "REDACTED"
}
```

Per ulteriori informazioni, consulta Nozioni di base sull'API Trusted Advisor nella Guida per l'utente di AWS Trusted Advisor.

Esempio 3: come elencare le raccomandazioni con un token di paginazione

L'esempio `list-recommendations` seguente utilizza il valore "nextToken" restituito da una richiesta precedente per recuperare la pagina successiva delle raccomandazioni filtrate.

```
aws trustedadvisor list-recommendations \
    --aws-service rds \
    --max-items 100 \
    --starting-token <next-token>
```

Output:

```
{
    "recommendationSummaries": [{
        "arn":
 "arn:aws:trustedadvisor::000000000000:recommendation/8b602b6f-452d-4cb2-8a9e-
c7650955d9cd",
        "name": "RDS clusters quota warning",
        "awsServices": [
            "rds"
        ],
        "checkArn": "arn:aws:trustedadvisor:::check/gjqMBn6pjz",
        "id": "8b602b6f-452d-4cb2-8a9e-c7650955d9cd",
        "lastUpdatedAt": "2023-11-01T15:58:17.397Z",
        "pillarSpecificAggregates": {
            "costOptimizing": {
                "estimatedMonthlySavings": 0.0,
                "estimatedPercentMonthlySavings": 0.0
            }
```

```
        },
        "pillars": [
            "service_limits"
        ],
        "resourcesAggregates": {
            "errorCount": 0,
            "okCount": 3,
            "warningCount": 6
        },
        "source": "ta_check",
        "status": "warning",
        "type": "standard"
    }]
}
```

Per ulteriori informazioni, consulta Nozioni di base sull'API Trusted Advisor nella Guida per l'utente di AWS Trusted Advisor.

- Per i dettagli sull'API, consulta ListRecommendations AWS CLICommand Reference.

`update-organization-recommendation-lifecycle`

Il seguente esempio di codice mostra come utilizzare`update-organization-recommendation-lifecycle`.

AWS CLI

Come aggiornare il ciclo di vita delle raccomandazioni di un'organizzazione

L'esempio `update-organization-recommendation-lifecycle` seguente aggiorna il ciclo di vita di una raccomandazione a livello di organizzazione in base al relativo identificatore.

```
aws trustedadvisor update-organization-recommendation-lifecycle \
    --organization-recommendation-identifier arn:aws:trustedadvisor:::organization-recommendation/96b5e5ca-7930-444c-90c6-06d386128100 \
    --lifecycle-stage dismissed \
    --update-reason-code not_applicable
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Nozioni di base sull'API Trusted Advisor nella Guida per l'utente di AWS Trusted Advisor.

- Per i dettagli sull'API, consulta UpdateOrganizationRecommendationLifecycle AWS CLICommand Reference.

`update-recommendation-lifecycle`

Il seguente esempio di codice mostra come utilizzare`update-recommendation-lifecycle`.

AWS CLI

Come aggiornare il ciclo di vita di una raccomandazione

L'esempio `update-recommendation-lifecycle` seguente aggiorna il ciclo di vita di una raccomandazione in base al relativo identificatore.

```
aws trustedadvisor update-recommendation-lifecycle \
    --recommendation-
identifier arn:aws:trustedadvisor::000000000000:recommendation/861c9c6e-
f169-405a-8b59-537a8caccd7a \
    --lifecycle-stage resolved \
    --update-reason-code valid_business_case
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Nozioni di base sull'API Trusted Advisor nella Guida per l'utente di AWS Trusted Advisor.

- Per i dettagli sull'API, consulta UpdateRecommendationLifecycle AWS CLICommand Reference.

Esempi di autorizzazioni verificate utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with Verified Permissions.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`create-identity-source`

Il seguente esempio di codice mostra come utilizzare`create-identity-source`.

AWS CLI

Come creare un'origine dell'identità

L'esempio `create-identity-source` seguente crea un'origine di identità che consente di fare riferimento alle identità archiviate nel pool di utenti Amazon Cognito specificato. Queste identità sono disponibili nell'area Autorizzazioni verificate come entità di tipo `User`.

```
aws verifiedpermissions create-identity-source \
    --configuration file://config.txt \
    --principal-entity-type "User" \
    --policy-store-id PSEXAMPLEabcdefg111111
```

Contenuto di `config.txt`:

```
{
    "cognitoUserPoolConfiguration": {
        "userPoolArn": "arn:aws:cognito-idp:us-west-2:123456789012:userpool/us-
west-2_1a2b3c4d5",
        "clientIds":["a1b2c3d4e5f6g7h8i9j0kalbmc"]
    }
}
```

Output:

```
{
    "createdDate": "2023-05-19T20:30:28.214829+00:00",
    "identitySourceId": "ISEXAMPLEabcdefg111111",
    "lastUpdatedDate": "2023-05-19T20:30:28.214829+00:00",
    "policyStoreId": "PSEXAMPLEabcdefg111111"
}
```

Per ulteriori informazioni sulle origini dell'identità, consulta Utilizzo delle autorizzazioni verificate con i provider di identità nella Guida per l'utente di Autorizzazioni verificate da Amazon.

- Per i dettagli sull'API, consulta CreateIdentitySource AWS CLICommand Reference.

`create-policy-store`

Il seguente esempio di codice mostra come utilizzare`create-policy-store`.

AWS CLI

Come creare un archivio di policy

L'`create-policy-store`esempio seguente crea un archivio delle politiche nella AWS regione corrente.

```
aws verifiedpermissions create-policy-store \
    --validation-settings "mode=STRICT"
```

Output:

```
{
    "arn": "arn:aws:verifiedpermissions::123456789012:policy-store/
PSEXAMPLEabcdefg111111",
    "createdDate": "2023-05-16T17:41:29.103459+00:00",
    "lastUpdatedDate": "2023-05-16T17:41:29.103459+00:00",
    "policyStoreId": "PSEXAMPLEabcdefg111111"
}
```

Per ulteriori informazioni sugli archivi di policy, consulta Archivi di policy di Autorizzazioni verificate da Amazon nella Guida per l'utente di Autorizzazioni verificate da Amazon.

- Per i dettagli sull'API, vedere CreatePolicyStorein AWS CLI Command Reference.

`create-policy-template`

Il seguente esempio di codice mostra come utilizzare`create-policy-template`.

AWS CLI

Come creare un modello di policy

L'esempio `create-policy-template` seguente crea un modello di policy con un'istruzione contenente un segnaposto per il principale.

```
aws verifiedpermissions create-policy-template \
    --statement file://template1.txt \
    --policy-store-id PSEXAMPLEabcdefg111111
```

Contenuto di `template1.txt`:

```
permit(
    principal in ?principal,
    action == Action::"view",
    resource == Photo::"VacationPhoto94.jpg"
);
```

Output:

```
{
    "createdDate": "2023-06-12T20:47:42.804511+00:00",
    "lastUpdatedDate": "2023-06-12T20:47:42.804511+00:00",
    "policyStoreId": "PSEXAMPLEabcdefg111111",
    "policyTemplateId": "PTEXAMPLEabcdefg111111"
}
```

Per ulteriori informazioni sui modelli di policy, consulta Modelli di policy di Autorizzazioni verificate da Amazon nella Guida per l'utente di Autorizzazioni verificate da Amazon.

- Per i dettagli sull'API, consulta CreatePolicyTemplate AWS CLICommand Reference.

`create-policy`

Il seguente esempio di codice mostra come utilizzare`create-policy`.

AWS CLI

Esempio 1: come creare una policy statica

L'esempio `create-policy` seguente crea una policy statica con un ambito che specifica sia un principale che una risorsa.

```
aws verifiedpermissions create-policy \
```

```
    --definition file://definition1.txt \
    --policy-store-id PSEXAMPLEabcdefg111111
```

Contenuto del file `definition1.txt`:

```
{
    "static": {
        "description":  "Grant everyone of janeFriends UserGroup access to the
 vacationFolder Album",
        "statement": "permit(principal in UserGroup::\"janeFriends\", action,
 resource in Album::\"vacationFolder\" );"
    }
}
```

Output:

```
{
    "createdDate": "2023-06-12T20:33:37.382907+00:00",
    "lastUpdatedDate": "2023-06-12T20:33:37.382907+00:00",
    "policyId": "SPEXAMPLEabcdefg111111",
    "policyStoreId": "PSEXAMPLEabcdefg111111",
    "policyType": "STATIC",
    "principal": {
        "entityId": "janeFriends",
        "entityType": "UserGroup"
    },
    "resource": {
        "entityId": "vacationFolder",
        "entityType": "Album"
    }
}
```

Esempio 2: come creare una policy statica che garantisca a tutti l'accesso a una risorsa

L'esempio `create-policy` seguente crea una policy statica con un ambito che specifica solo una risorsa.

```
aws verifiedpermissions create-policy \
    --definition file://definition2.txt \
    --policy-store-id PSEXAMPLEabcdefg111111
```

Contenuto del file `definition2.txt`:

```
{
    "static": {
        "description":  "Grant everyone access to the publicFolder Album",
        "statement": "permit(principal, action, resource in Album::\"publicFolder
\");"
    }
}
```

Output:

```
{
    "createdDate": "2023-06-12T20:39:44.975897+00:00",
    "lastUpdatedDate": "2023-06-12T20:39:44.975897+00:00",
    "policyId": "PbfR73F8oh5MMfr9uRtFDB",
    "policyStoreId": "PSEXAMPLEabcdefg222222",
    "policyType": "STATIC",
    "resource": {
        "entityId": "publicFolder",
        "entityType": "Album"
    }
}
```

Esempio 3: come creare una policy basata su modello associata al modello specificato

L'esempio `create-policy` seguente crea una policy collegata al modello utilizzando il modello di policy specificato e associa il principale specificato da utilizzare con la nuova policy collegata al modello.

```
aws verifiedpermissions create-policy \
    --definition file://definition.txt \
    --policy-store-id PSEXAMPLEabcdefg111111
```

Contenuto di `definition.txt`:

```
{
    "templateLinked": {
        "policyTemplateId": "PTEXAMPLEabcdefg111111",
        "principal": {
            "entityType": "User",
            "entityId": "alice"
        }
```

```
    }
}
```

Output:

```
{
    "createdDate": "2023-06-12T20:49:51.490211+00:00",
    "lastUpdatedDate": "2023-06-12T20:49:51.490211+00:00",
    "policyId": "TPEXAMPLEabcdefg111111",
    "policyStoreId": "PSEXAMPLEabcdefg111111",
    "policyType": "TEMPLATE_LINKED",
    "principal": {
        "entityId": "alice",
        "entityType": "User"
    },
    "resource": {
        "entityId": "VacationPhoto94.jpg",
        "entityType": "Photo"
    }
}
```

Per ulteriori informazioni sulle policy, consulta Policy di Autorizzazioni verificate da Amazon nella Guida per l'utente di Autorizzazioni verificate da Amazon.

- Per i dettagli sull'API, consulta CreatePolicy AWS CLICommand Reference.

`delete-identity-source`

Il seguente esempio di codice mostra come utilizzare`delete-identity-source`.

AWS CLI

Come eliminare un'origine dell'identità

L'esempio `delete-identity-source` seguente elimina l'origine dell'identità con l'ID specificato.

```
aws verifiedpermissions delete-identity-source \
    --identity-source-id ISEXAMPLEabcdefg111111 \
    --policy-store-id PSEXAMPLEabcdefg111111
```

Questo comando non produce alcun output.

Per ulteriori informazioni sulle origini dell'identità, consulta Utilizzo delle autorizzazioni verificate con i provider di identità nella Guida per l'utente di Autorizzazioni verificate da Amazon.

- Per i dettagli sull'API, consulta DeleteIdentitySource AWS CLICommand Reference.

`delete-policy-store`

Il seguente esempio di codice mostra come utilizzare`delete-policy-store`.

AWS CLI

Come eliminare un archivio di policy

L'esempio `delete-policy-store` seguente elimina l'archivio di policy con l'ID specificato.

```
aws verifiedpermissions delete-policy-store \
    --policy-store-id PSEXAMPLEabcdefg111111
```

Questo comando non produce alcun output.

Per ulteriori informazioni sugli archivi di policy, consulta Archivi di policy di Autorizzazioni verificate da Amazon nella Guida per l'utente di Autorizzazioni verificate da Amazon.

- Per i dettagli sull'API, consulta DeletePolicyStore AWS CLICommand Reference.

`delete-policy-template`

Il seguente esempio di codice mostra come utilizzare`delete-policy-template`.

AWS CLI

Come eliminare un modello di policy

L'esempio `delete-policy-template` seguente elimina il modello di policy con l'ID specificato.

```
aws verifiedpermissions delete-policy \
    --policy-template-id PTEXAMPLEabcdefg111111 \
    --policy-store-id PSEXAMPLEabcdefg111111
```

Questo comando non produce alcun output.

Per ulteriori informazioni sui modelli di policy, consulta Modelli di policy di Autorizzazioni verificate da Amazon nella Guida per l'utente di Autorizzazioni verificate da Amazon.

- Per i dettagli sull'API, consulta DeletePolicyTemplate AWS CLICommand Reference.

`delete-policy`

Il seguente esempio di codice mostra come utilizzare`delete-policy`.

AWS CLI

Come eliminare una policy statica o basata su modello

L'esempio `delete-policy` seguente elimina la policy con l'ID specificato.

```
aws verifiedpermissions delete-policy \
    --policy-id SPEXAMPLEabcdefg111111 \
    --policy-store-id PSEXAMPLEabcdefg111111
```

Questo comando non produce alcun output.

Per ulteriori informazioni sulle policy, consulta Policy di Autorizzazioni verificate da Amazon nella Guida per l'utente di Autorizzazioni verificate da Amazon.

- Per i dettagli sull'API, consulta DeletePolicy AWS CLICommand Reference.

`get-identity-source`

Il seguente esempio di codice mostra come utilizzare`get-identity-source`.

AWS CLI

Come recuperare i dettagli su un'origine dell'identità

L'esempio `get-identity-source` seguente visualizza i dettagli dell'origine dell'identità con l'ID specificato.

```
aws verifiedpermissions get-identity-source \
    --identity-source  ISEXAMPLEabcdefg111111 \
    --policy-store-id PSEXAMPLEabcdefg111111
```

Output:

```
{
    "createdDate": "2023-06-12T22:27:49.150035+00:00",
    "details": {
        "clientIds": [ "a1b2c3d4e5f6g7h8i9j0kalbmc" ],
        "discoveryUrl": "https://cognito-idp.us-west-2.amazonaws.com/us-
west-2_1a2b3c4d5",
        "openIdIssuer": "COGNITO",
        "userPoolArn": "arn:aws:cognito-idp:us-west-2:123456789012:userpool/us-
west-2_1a2b3c4d5"
    },
    "identitySourceId": "ISEXAMPLEabcdefg111111",
    "lastUpdatedDate": "2023-06-12T22:27:49.150035+00:00",
    "policyStoreId": "PSEXAMPLEabcdefg111111",
    "principalEntityType": "User"
}
```

Per ulteriori informazioni sulle origini dell'identità, consulta Utilizzo delle autorizzazioni verificate con i provider di identità nella Guida per l'utente di Autorizzazioni verificate da Amazon.

- Per i dettagli sull'API, consulta GetIdentitySource AWS CLICommand Reference.

`get-policy-store`

Il seguente esempio di codice mostra come utilizzare`get-policy-store`.

AWS CLI

Come recuperare i dettagli relativi a un archivio di policy

L'esempio `get-policy-store` seguente visualizza i dettagli dell'archivio di policy con l'ID specificato.

```
aws verifiedpermissions get-policy-store \
    --policy-store-id PSEXAMPLEabcdefg111111
```

Output:

```
{
    "arn": "arn:aws:verifiedpermissions::123456789012:policy-store/
PSEXAMPLEabcdefg111111",
```

```
    "createdDate": "2023-06-05T20:16:46.225598+00:00",
    "lastUpdatedDate": "2023-06-08T20:40:23.173691+00:00",
    "policyStoreId": "PSEXAMPLEabcdefg111111",
    "validationSettings": { "mode": "OFF" }
}
```

Per ulteriori informazioni sugli archivi di policy, consulta Archivi di policy di Autorizzazioni verificate da Amazon nella Guida per l'utente di Autorizzazioni verificate da Amazon.

- Per i dettagli sull'API, consulta GetPolicyStore AWS CLICommand Reference.

`get-policy-template`

Il seguente esempio di codice mostra come utilizzare`get-policy-template`.

AWS CLI

Come recuperare i dettagli relativi a un modello di policy

L'esempio `get-policy-template` seguente visualizza i dettagli del modello di policy con l'ID specificato.

```
aws verifiedpermissions get-policy-template \
    --policy-template-id  PTEXAMPLEabcdefg111111 \
    --policy-store-id PSEXAMPLEabcdefg111111
```

Output:

```
{
    "createdDate": "2023-06-12T20:47:42.804511+00:00",
    "lastUpdatedDate": "2023-06-12T20:47:42.804511+00:00",
    "policyStoreId": "PSEXAMPLEabcdefg111111",
    "policyTemplateId": "PTEXAMPLEabcdefg111111",
    "statement": "permit(\n    principal in ?principal,\n    action == Action::
\"view\",\n    resource == Photo::\"VacationPhoto94.jpg\"\n);"
}
```

Per ulteriori informazioni sui modelli di policy, consulta Modelli di policy di Autorizzazioni verificate da Amazon nella Guida per l'utente di Autorizzazioni verificate da Amazon.

- Per i dettagli sull'API, consulta GetPolicyTemplate AWS CLICommand Reference.

`get-policy`

Il seguente esempio di codice mostra come utilizzare`get-policy`.

AWS CLI

Come recuperare i dettagli relativi a una policy

L'esempio `get-policy` seguente mostra i dettagli della policy con l'ID specificato.

```
aws verifiedpermissions get-policy \
    --policy-id PSEXAMPLEabcdefg111111 \
    --policy-store-id PSEXAMPLEabcdefg111111
```

Output:

```
{
    "createdDate": "2023-06-12T20:33:37.382907+00:00",
    "definition": {
        "static": {
            "description": "Grant everyone of janeFriends UserGroup access to the
 vacationFolder Album",
            "statement": "permit(principal in UserGroup::\"janeFriends\", action,
 resource in Album::\"vacationFolder\" );"
        }
    },
    "lastUpdatedDate": "2023-06-12T20:33:37.382907+00:00",
    "policyId": "SPEXAMPLEabcdefg111111",
    "policyStoreId": "PSEXAMPLEabcdefg111111",
    "policyType": "STATIC",
    "principal": {
        "entityId": "janeFriends",
        "entityType": "UserGroup"
    },
    "resource": {
        "entityId": "vacationFolder",
        "entityType": "Album"
    }
}
```

Per ulteriori informazioni sulle policy, consulta Policy di Autorizzazioni verificate da Amazon nella Guida per l'utente di Autorizzazioni verificate da Amazon.

- Per i dettagli sull'API, consulta GetPolicy AWS CLICommand Reference.

`get-schema`

Il seguente esempio di codice mostra come utilizzare`get-schema`.

AWS CLI

Come recuperare lo schema in un archivio di policy

L'esempio `get-schema` seguente mostra i dettagli dello schema nell'archivio di policy specificato.

```
aws verifiedpermissions get-schema \
    --policy-store-id PSEXAMPLEabcdefg111111
```

Output:

```
{
    "policyStoreId": "PSEXAMPLEabcdefg111111",
    "schema": "{\"MySampleNamespace\":{\"entityTypes\":{\"Employee\":{\"shape
\":{\"attributes\":{\"jobLevel\":{\"type\":\"Long\"},\"name\":{\"type\":\"String
\"}},\"type\":\"Record\"}}},\"actions\":{\"remoteAccess\":{\"appliesTo\":
{\"principalTypes\":[\"Employee\"]}}}}}",
    "createdDate": "2023-06-14T17:47:13.999885+00:00",
    "lastUpdatedDate": "2023-06-14T17:47:13.999885+00:00"
}
```

Per ulteriori informazioni sullo schema, consulta Schema di archivio di policy nella Guida per l'utente di Autorizzazioni verificate da Amazon.

- Per i dettagli sull'API, consulta GetSchema AWS CLICommand Reference.

`is-authorized-with-token`

Il seguente esempio di codice mostra come utilizzare`is-authorized-with-token`.

AWS CLI

Esempio 1: come richiedere una decisione di autorizzazione per una richiesta utente (consentire)

L'esempio `is-authorized-with-token` seguente richiede una decisione di autorizzazione per un utente autenticato da Amazon Cognito. La richiesta utilizza il token di identità fornito da Cognito anziché il token di accesso. In questo esempio, l'archivio di informazioni specificato è configurato per restituire i principali come entità di tipo `CognitoUser`.

```
aws verifiedpermissions is-authorized-with-token \
    --action actionId="View",actionType="Action" \
    --resource entityId="vacationPhoto94.jpg",entityType="Photo" \
    --policy-store-id PSEXAMPLEabcdefg111111 \
    --identity-token "AbCdE12345...long.string...54321EdCbA"
```

L'archivio delle policy contiene una policy con la seguente istruzione che accetta le identità dal pool di utenti e dall'ID applicazione Cognito specificati.

```
permit(
    principal == CognitoUser::"us-east-1_1a2b3c4d5|a1b2c3d4e5f6g7h8i9j0kalbmc",
    action,
    resource == Photo::"VacationPhoto94.jpg"
);
```

Output:

```
{
    "decision":"Allow",
    "determiningPolicies":[
        {
        "determiningPolicyId":"SPEXAMPLEabcdefg111111"
        }
    ],
    "errors":[]
}
```

Per ulteriori informazioni sull'utilizzo delle identità di un pool di utenti Cognito, consulta Utilizzo delle autorizzazioni verificate con i provider di identità nella Guida per l'utente di Autorizzazioni verificate da Amazon.

- Per i dettagli sull'API, consulta IsAuthorizedWithToken AWS CLICommand Reference.

`is-authorized`

Il seguente esempio di codice mostra come utilizzare`is-authorized`.

AWS CLI

Esempio 1: come richiedere una decisione di autorizzazione per una richiesta utente (consentire)

L'esempio `is-authorized` seguente richiede una decisione di autorizzazione per un principale di tipo `User` denominato `Alice` che desidera eseguire l'operazione `updatePhoto` su una risorsa di tipo `Photo` denominata `VacationPhoto94.jpg`.

La risposta mostra che la richiesta è consentita da una policy.

```
aws verifiedpermissions is-authorized \
    --principal entityType=User,entityId=alice \
    --action actionType=Action,actionId=view \
    --resource entityType=Photo,entityId=VactionPhoto94.jpg \
    --policy-store-id PSEXAMPLEabcdefg111111
```

Output:

```
{
    "decision": "ALLOW",
    "determiningPolicies": [
        {
            "policyId": "SPEXAMPLEabcdefg111111"
        }
    ],
    "errors": []
}
```

Esempio 2: come richiedere una decisione di autorizzazione per una richiesta utente (negare)

L'esempio seguente è uguale all'esempio precedente tranne per il fatto che il principale è `User::"Bob"`. L'archivio delle policy non contiene alcuna policy che consenta a tale utente l'accesso a `Album::"alice_folder"`.

L'output indica che `Deny` era implicito perché l'elenco di `DeterminingPolicies` è vuoto.

```
aws verifiedpermissions create-policy \
    --definition file://definition2.txt \
    --policy-store-id PSEXAMPLEabcdefg111111
```

Output:

```
{
    "decision": "DENY",
    "determiningPolicies": [],
```

```
    "errors": []
}
```

Per ulteriori informazioni, consulta la Guida per l'utente di Autorizzazioni verificate da Amazon.

- Per i dettagli sull'API, consulta IsAuthorized AWS CLICommand Reference.

`list-identity-sources`

Il seguente esempio di codice mostra come utilizzare`list-identity-sources`.

AWS CLI

Come elencare le origini dell'identità disponibili

L'esempio `list-identity-sources` seguente elenca tutte le origini dell'identità nell'archivio delle policy specificato.

```
aws verifiedpermissions list-identity-sources \
    --policy-store-id PSEXAMPLEabcdefg111111
```

Output:

```
{
    "identitySources": [
        {
            "createdDate": "2023-06-12T22:27:49.150035+00:00",
            "details": {
                "clientIds": [ "a1b2c3d4e5f6g7h8i9j0kalbmc" ],
                "discoveryUrl": "https://cognito-idp.us-west-2.amazonaws.com/us-
west-2_1a2b3c4d5",
                "openIdIssuer": "COGNITO",
                "userPoolArn": "arn:aws:cognito-idp:us-west-2:123456789012:userpool/
us-west-2_1a2b3c4d5"
            },
            "identitySourceId": "ISEXAMPLEabcdefg111111",
            "lastUpdatedDate": "2023-06-12T22:27:49.150035+00:00",
            "policyStoreId": "PSEXAMPLEabcdefg111111",
            "principalEntityType": "User"
        }
    ]
}
```

Per ulteriori informazioni sulle origini dell'identità, consulta Utilizzo delle autorizzazioni verificate con i provider di identità nella Guida per l'utente di Autorizzazioni verificate da Amazon.

- Per i dettagli sull'API, consulta ListIdentitySources AWS CLICommand Reference.

`list-policies`

Il seguente esempio di codice mostra come utilizzare`list-policies`.

AWS CLI

Come elencare le policy disponibili

L'esempio `list-policies` seguente elenca tutte le policy nell'archivio di policy specificato.

```
aws verifiedpermissions list-policies \
    --policy-store-id PSEXAMPLEabcdefg111111
```

Output:

```
{
    "policies": [
        {
            "createdDate": "2023-06-12T20:33:37.382907+00:00",
            "definition": {
                "static": {
                    "description": "Grant everyone of janeFriends UserGroup access
 to the vacationFolder Album"
                }
            },
            "lastUpdatedDate": "2023-06-12T20:33:37.382907+00:00",
            "policyId": "SPEXAMPLEabcdefg111111",
            "policyStoreId": "PSEXAMPLEabcdefg111111",
            "policyType": "STATIC",
            "principal": {
                "entityId": "janeFriends",
                "entityType": "UserGroup"
            },
            "resource": {
                "entityId": "vacationFolder",
                "entityType": "Album"
            }
```

```
        },
        {
            "createdDate": "2023-06-12T20:39:44.975897+00:00",
            "definition": {
                "static": {
                    "description": "Grant everyone access to the publicFolder Album"
                }
            },
            "lastUpdatedDate": "2023-06-12T20:39:44.975897+00:00",
            "policyId": "SPEXAMPLEabcdefg222222",
            "policyStoreId": "PSEXAMPLEabcdefg111111",
            "policyType": "STATIC",
            "resource": {
                "entityId": "publicFolder",
                "entityType": "Album"
            }
        },
        {
            "createdDate": "2023-06-12T20:49:51.490211+00:00",
            "definition": {
                "templateLinked": {
                    "policyTemplateId": "PTEXAMPLEabcdefg111111"
                }
            },
            "lastUpdatedDate": "2023-06-12T20:49:51.490211+00:00",
            "policyId": "SPEXAMPLEabcdefg333333",
            "policyStoreId": "PSEXAMPLEabcdefg111111",
            "policyType": "TEMPLATE_LINKED",
            "principal": {
                "entityId": "alice",
                "entityType": "User"
            },
            "resource": {
                "entityId": "VacationPhoto94.jpg",
                "entityType": "Photo"
            }
        }
    ]
}
```


Per ulteriori informazioni sulle policy, consulta Policy di Autorizzazioni verificate da Amazon nella Guida per l'utente di Autorizzazioni verificate da Amazon.

- Per i dettagli sull'API, consulta ListPolicies AWS CLICommand Reference.

`list-policy-stores`

Il seguente esempio di codice mostra come utilizzare`list-policy-stores`.

AWS CLI

Come elencare gli archivi di policy disponibili

L'`list-policy-stores`esempio seguente elenca tutti gli archivi delle politiche nella AWS regione. Tutti i comandi per Autorizzazioni verificate da Amazon tranne `create-policy-store` e `list-policy-stores` richiedono la specificazione dell'ID del archivio delle policy con cui si desidera lavorare.

```
aws verifiedpermissions list-policy-stores
```

Output:

```
{
    "policyStores": [
        {
            "arn": "arn:aws:verifiedpermissions::123456789012:policy-store/
PSEXAMPLEabcdefg111111",
            "createdDate": "2023-06-05T20:16:46.225598+00:00",
            "policyStoreId": "PSEXAMPLEabcdefg111111"
        },
        {
            "arn": "arn:aws:verifiedpermissions::123456789012:policy-store/
PSEXAMPLEabcdefg222222",
            "createdDate": "2023-06-08T18:09:37.364356+00:00",
            "policyStoreId": "PSEXAMPLEabcdefg222222"
        },
        {
            "arn": "arn:aws:verifiedpermissions::123456789012:policy-store/
PSEXAMPLEabcdefg333333",
            "createdDate": "2023-06-08T18:09:46.920600+00:00",
            "policyStoreId": "PSEXAMPLEabcdefg333333"
        }
    ]
}
```

Per ulteriori informazioni sugli archivi di policy, consulta Archivi di policy di Autorizzazioni verificate da Amazon nella Guida per l'utente di Autorizzazioni verificate da Amazon.

- Per i dettagli sull'API, vedere ListPolicyStoresin AWS CLI Command Reference.

`list-policy-templates`

Il seguente esempio di codice mostra come utilizzare`list-policy-templates`.

AWS CLI

Come elencare i modelli di policy disponibili

L'esempio `list-policy-templates` seguente elenca tutti i modelli di policy nell'archivio di policy specificato.

```
aws verifiedpermissions list-policy-templates \
    --policy-store-id PSEXAMPLEabcdefg111111
```

Output:

```
{
    "policyTemplates": [
        {
            "createdDate": "2023-06-12T20:47:42.804511+00:00",
            "lastUpdatedDate": "2023-06-12T20:47:42.804511+00:00",
            "policyStoreId": "PSEXAMPLEabcdefg111111",
            "policyTemplateId": "PTEXAMPLEabcdefg111111"
        }
    ]
}
```

Per ulteriori informazioni sui modelli di policy, consulta Modelli di policy di Autorizzazioni verificate da Amazon nella Guida per l'utente di Autorizzazioni verificate da Amazon.

- Per i dettagli sull'API, consulta ListPolicyTemplates AWS CLICommand Reference.

`put-schema`

Il seguente esempio di codice mostra come utilizzare`put-schema`.

AWS CLI

Come salvare uno schema in un archivio di policy

L'esempio `put-schema` seguente crea o sostituisce lo schema nell'archivio di policy specificato.

Il parametro `cedarJson` nel file di input accetta una rappresentazione di stringa di un oggetto JSON. Contiene virgolette incorporate (“”) all'interno della coppia di virgolette più esterna. A tale scopo è necessario convertire il codice JSON in una stringa facendo precedere tutte le virgolette incorporate da una barra rovesciata (“”) e combinando tutte le righe in un'unica riga di testo senza interruzioni di riga.

Le stringhe di esempio possono essere visualizzate su più righe per motivi di leggibilità, ma l'operazione richiede che i parametri vengano inviati come stringhe a riga singola.

aws verifiedpermissions put-schema --definition file: //schema.txt -- policy-store-id PSEXAMPLEabcdefg111111

Contenuto di `schema.txt`:

```
{
    "cedarJson": "{\"MySampleNamespace\": {\"actions\": {\"remoteAccess\": {
            \"appliesTo\": {\"principalTypes\": [\"Employee\"]}},\"entityTypes\": {
            \"Employee\": {\"shape\": {\"attributes\": {\"jobLevel\": {\"type\":
            \"Long\"},\"name\": {\"type\": \"String\"}},\"type\": \"Record\"}}}}}"
}
```

Output:

```
{
    "policyStoreId": "PSEXAMPLEabcdefg111111",
    "namespaces": [
        "MySampleNamespace"
    ],
    "createdDate": "2023-06-14T17:47:13.999885+00:00",
    "lastUpdatedDate": "2023-06-14T17:47:13.999885+00:00"
}
```

Per ulteriori informazioni sullo schema, consulta Schema di archivio di policy nella Guida per l'utente di Autorizzazioni verificate da Amazon.

- Per i dettagli sull'API, consulta Command Reference. PutSchemaAWS CLI

`update-identity-source`

Il seguente esempio di codice mostra come utilizzare`update-identity-source`.

AWS CLI

Come aggiornare un'origine dell'identità

L'esempio `update-identity-source` seguente modifica l'origine dell'identità specificata fornendo una nuova configurazione del pool di utenti Cognito e cambiando il tipo di entità restituito dall'origine dell'identità.

```
aws verifiedpermissions update-identity-source
    --identity-source-id ISEXAMPLEabcdefg111111 \
    --update-configuration file://config.txt \
    --principal-entity-type "Employee" \
    --policy-store-id PSEXAMPLEabcdefg111111
```

Contenuto di `config.txt`:

```
{
        "cognitoUserPoolConfiguration": {
                "userPoolArn": "arn:aws:cognito-idp:us-west-2:123456789012:userpool/
us-west-2_1a2b3c4d5",
                "clientIds":["a1b2c3d4e5f6g7h8i9j0kalbmc"]
        }
}
```

Output:

```
{
    "createdDate": "2023-05-19T20:30:28.214829+00:00",
    "identitySourceId": "ISEXAMPLEabcdefg111111",
    "lastUpdatedDate": "2023-05-19T20:30:28.214829+00:00",
    "policyStoreId": "PSEXAMPLEabcdefg111111"
}
```

Per ulteriori informazioni sulle origini dell'identità, consulta Utilizzo delle autorizzazioni verificate con i provider di identità nella Guida per l'utente di Autorizzazioni verificate da Amazon.

- Per i dettagli sull'API, consulta UpdateIdentitySource AWS CLICommand Reference.

`update-policy-store`

Il seguente esempio di codice mostra come utilizzare`update-policy-store`.

AWS CLI

Come aggiornare un archivio di policy

L'esempio `update-policy-store` seguente modifica un archivio di policy cambiandone l'impostazione di convalida.

```
aws verifiedpermissions update-policy-store \
    --validation-settings "mode=STRICT" \
    --policy-store-id PSEXAMPLEabcdefg111111
```

Output:

```
{
    "arn": "arn:aws:verifiedpermissions::123456789012:policy-store/
PSEXAMPLEabcdefg111111",
    "createdDate": "2023-05-16T17:41:29.103459+00:00",
    "lastUpdatedDate": "2023-05-16T17:41:29.103459+00:00",
    "policyStoreId": "PSEXAMPLEabcdefg111111"
}
```

Per ulteriori informazioni sugli archivi di policy, consulta Archivi di policy di Autorizzazioni verificate da Amazon nella Guida per l'utente di Autorizzazioni verificate da Amazon.

- Per i dettagli sull'API, consulta UpdatePolicyStore AWS CLICommand Reference.

`update-policy-template`

Il seguente esempio di codice mostra come utilizzare`update-policy-template`.

AWS CLI

Esempio 1: come aggiornare un modello di policy

L'esempio `update-policy-template` seguente modifica la policy specificata collegata al modello per sostituire la relativa dichiarazione di policy.

```
aws verifiedpermissions update-policy-template \
    --policy-template-id PTEXAMPLEabcdefg111111 \
    --statement file://template1.txt \
    --policy-store-id PSEXAMPLEabcdefg111111
```

Contenuto del file `template1.txt`:

```
permit(
    principal in ?principal,
    action == Action::"view",
    resource == Photo::"VacationPhoto94.jpg"
);
```

Output:

```
{
    "createdDate": "2023-06-12T20:47:42.804511+00:00",
    "lastUpdatedDate": "2023-06-12T20:47:42.804511+00:00",
    "policyStoreId": "PSEXAMPLEabcdefg111111",
    "policyTemplateId": "PTEXAMPLEabcdefg111111"
}
```

Per ulteriori informazioni sui modelli di policy, consulta Modelli di policy di Autorizzazioni verificate da Amazon nella Guida per l'utente di Autorizzazioni verificate da Amazon.

- Per i dettagli sull'API, consulta UpdatePolicyTemplate AWS CLICommand Reference.

`update-policy`

Il seguente esempio di codice mostra come utilizzare`update-policy`.

AWS CLI

Come aggiornare una policy statica

L'esempio `update-policy` seguente modifica una policy statica esistente aggiornandone la descrizione e l'istruzione.

```
aws verifiedpermissions update-policy \
    --policy-id SPEXAMPLEabcdefg111111 \
    --definition file://updated-definition.txt \
    --policy-store-id PSEXAMPLEabcdefg111111
```

Il parametro `statement` accetta una rappresentazione di stringa di un oggetto JSON. Contiene virgolette incorporate ("") all'interno della coppia di virgolette più esterna. A tale scopo è

necessario convertire il codice JSON in una stringa facendo precedere tutte le virgolette incorporate da una barra rovesciata (“”) e combinando tutte le righe in un'unica riga di testo senza interruzioni di riga.

È possibile visualizzare stringhe di esempio su più righe per motivi di leggibilità, ma l'operazione richiede che i parametri vengano inviati come stringhe a riga singola.

Contenuto del file `updated-definition.txt`:

```
{
    "static": {
        "description": "Updated policy to grant janeFriends UserGroup access to the
 vacationFolder Album with view action only",
        "statement": "permit(principal in UserGroup::\"janeFriends\", action ==
 Action::\"view\", resource in Album::\"vacationFolder\" );"
    }
}
```

Output:

```
{
    "createdDate": "2023-06-12T20:33:37.382907+00:00",
    "lastUpdatedDate": "2023-06-12T20:33:37.382907+00:00",
    "policyId": "SPEXAMPLEabcdefg111111",
    "policyStoreId": "PSEXAMPLEabcdefg111111",
    "policyType": "STATIC",
    "principal": {
        "entityId": "janeFriends",
        "entityType": "UserGroup"
    },
    "resource": {
        "entityId": "vacationFolder",
        "entityType": "Album"
    }
}
```

Per ulteriori informazioni sulle policy, consulta Policy di Autorizzazioni verificate da Amazon nella Guida per l'utente di Autorizzazioni verificate da Amazon.

- Per i dettagli sull'API, consulta UpdatePolicy AWS CLICommand Reference.

Esempi di VPC Lattice utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with VPC Lattice.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`create-listener`

Il seguente esempio di codice mostra come utilizzare. `create-listener`

AWS CLI

Come creare un listener

L'esempio `create-listener` seguente crea un listener HTTPS con una regola predefinita che inoltra il traffico al gruppo di destinazione VPC Lattice specificato.

```
aws vpc-lattice create-listener \
    --name my-service-listener \
    --protocol HTTPS \
    --port 443 \
    --service-identifier svc-0285b53b2eEXAMPLE \
    --default-action file://listener-config.json
```

Contenuto di `listener-config.json`:

```
{
    "forward": {
        "targetGroups": [
            {
```

```
                "targetGroupIdentifier": "tg-0eaa4b9ab4EXAMPLE"
            }
        ]
    }
}
```

Output:

```
{
    "arn": "arn:aws:vpc-lattice:us-east-2:123456789012:service/
svc-0285b53b2eEXAMPLE/listener/listener-07cc7fb0abEXAMPLE",
    "defaultAction": {
        "forward": {
            "targetGroups": [
                {
                    "targetGroupIdentifier": "tg-0eaa4b9ab4EXAMPLE",
                    "weight": 100
                }
            ]
        }
    },
    "id": "listener-07cc7fb0abEXAMPLE",
    "name": "my-service-listener",
    "port": 443,
    "protocol": "HTTPS",
    "serviceArn": "arn:aws:vpc-lattice:us-east-2:123456789012:service/
svc-0285b53b2eEXAMPLE",
    "serviceId": "svc-0285b53b2eEXAMPLE"
}
```

Per ulteriori informazioni, consulta Listener nella Guida per l'utente di Amazon VPC Lattice.

- Per i dettagli sull'API, consulta CreateListener AWS CLICommand Reference.

`create-resource-configuration`

Il seguente esempio di codice mostra come utilizzare`create-resource-configuration`.

AWS CLI

Come creare una configurazione delle risorse

L'`create-resource-configuration`esempio seguente crea una configurazione delle risorse che specifica un singolo IPv4 indirizzo.

```
aws vpc-lattice create-resource-configuration \
    --name my-resource-config \
    --type SINGLE \
    --resource-gateway-identifier rgw-0bba03f3d56060135 \
    --resource-configuration-definition 'ipResource={ipAddress=10.0.14.85}'
```

Output:

```
{
    "allowAssociationToShareableServiceNetwork": true,
    "arn": "arn:aws:vpc-lattice:us-east-1:123456789012:resourceconfiguration/
rcfg-07129f3acded87625",
    "id": "rcfg-07129f3acded87625",
    "name": "my-resource-config",
    "portRanges": [
        "1-65535"
    ],
    "protocol": "TCP",
    "resourceConfigurationDefinition": {
        "ipResource": {
            "ipAddress": "10.0.14.85"
        }
    },
    "resourceGatewayId": "rgw-0bba03f3d56060135",
    "status": "ACTIVE",
    "type": "SINGLE"
}
```

Per ulteriori informazioni, consulta Configurazione delle risorse per risorse VPC nella Guida per l'utente di Amazon VPC Lattice.

- Per i dettagli sull'API, vedere CreateResourceConfigurationin AWS CLI Command Reference.

`create-resource-gateway`

Il seguente esempio di codice mostra come utilizzare`create-resource-gateway`.

AWS CLI

Come creare un gateway per le risorse

L'esempio `create-resource-gateway` seguente crea un gateway per le risorse per la sottorete specificata.

```
aws vpc-lattice create-resource-gateway \
    --name my-resource-gateway \
    --vpc-identifier vpc-0bf4c2739bc05a69 \
    --subnet-ids subnet-08e8943905b63a683
```

Output:

```
{
    "arn": "arn:aws:vpc-lattice:us-east-1:123456789012:resourcegateway/
rgw-0bba03f3d56060135",
    "id": "rgw-0bba03f3d56060135",
    "ipAddressType": "IPV4",
    "name": "my-resource-gateway",
    "securityGroupIds": [
        "sg-087ffd596c5fe962c"
    ],
    "status": "ACTIVE",
    "subnetIds": [
        "subnet-08e8943905b63a683"
    ],
    "vpcIdentifier": "vpc-0bf4c2739bc05a694"
}
```

Per ulteriori informazioni, consulta Gateway per le risorse in VPC Lattice nella Guida per l'utente di Amazon VPC Lattice.

- Per i dettagli sull'API, consulta CreateResourceGateway AWS CLICommand Reference.

`create-service-network-service-association`

Il seguente esempio di codice mostra come utilizzare`create-service-network-service-association`.

AWS CLI

Come creare un'associazione a un servizio

L'esempio `create-service-network-service-association` seguente associa il servizio specificato alla rete di servizi specificata.

```
aws vpc-lattice create-service-network-service-association \
    --service-identifier svc-0285b53b2eEXAMPLE \
    --service-network-identifier sn-080ec7dc93EXAMPLE
```

Output:

```
{
    "arn": "arn:aws:vpc-lattice:us-
east-2:123456789012:servicenetworkserviceassociation/snsa-0e16955a8cEXAMPLE",
    "createdBy": "123456789012",
    "dnsEntry": {
        "domainName": "my-lattice-service-0285b53b2eEXAMPLE.7d67968.vpc-lattice-
svcs.us-east-2.on.aws",
        "hostedZoneId": "Z09127221KTH2CEXAMPLE"
    },
    "id": "snsa-0e16955a8cEXAMPLE",
    "status": "CREATE_IN_PROGRESS"
}
```

Per ulteriori informazioni, consulta Gestire le associazioni dei servizi nella Guida per l'utente di Amazon VPC Lattice.

- Per i dettagli sull'API, consulta CreateServiceNetworkServiceAssociation AWS CLICommand Reference.

`create-service-network-vpc-association`

Il seguente esempio di codice mostra come utilizzare`create-service-network-vpc-association`.

AWS CLI

Come creare un'associazione a un VPC

L'esempio `create-service-network-vpc-association` seguente associa il VPC specificato alla rete di servizi specificata. Il gruppo di sicurezza specificato controlla quali risorse nel VPC possono accedere alla rete di servizi e ai relativi servizi.

```
aws vpc-lattice create-service-network-vpc-association \
    --vpc-identifier vpc-0a1b2c3d4eEXAMPLE \
    --service-network-identifier sn-080ec7dc93EXAMPLE \
    --security-group-ids sg-0aee16bc6cEXAMPLE
```

Output:

```
{
    "arn": "arn:aws:vpc-lattice:us-east-2:123456789012:servicenetworkvpcassociation/
snva-0821fc8631EXAMPLE",
    "createdBy": "123456789012",
    "id": "snva-0821fc8631EXAMPLE",
    "securityGroupIds": [
        "sg-0aee16bc6cEXAMPLE"
    ],
    "status": "CREATE_IN_PROGRESS"
}
```

Per ulteriori informazioni, consulta Gestire le associazioni VPC nella Guida per l'utente di Amazon VPC Lattice.

- Per i dettagli sull'API, consulta CreateServiceNetworkVpcAssociation AWS CLICommand Reference.

`create-service-network`

Il seguente esempio di codice mostra come utilizzare`create-service-network`.

AWS CLI

Come creare una rete di servizi

L'esempio `create-service-network` seguente crea una rete di servizi con il nome specificato.

```
aws vpc-lattice create-service-network \
    --name my-service-network
```

Output:

```
{
    "arn": "arn:aws:vpc-lattice:us-east-2:123456789012:servicenetwork/
sn-080ec7dc93EXAMPLE",
    "authType": "NONE",
    "id": "sn-080ec7dc93EXAMPLE",
    "name": "my-service-network"
}
```

Per ulteriori informazioni, consulta Reti di servizio nella Guida per l'utente di Amazon VPC Lattice.

- Per i dettagli sull'API, consulta CreateServiceNetwork AWS CLICommand Reference.

`create-service`

Il seguente esempio di codice mostra come utilizzare`create-service`.

AWS CLI

Come creare un servizio

L'esempio `create-service` seguente crea un servizio con il nome specificato.

```
aws vpc-lattice create-service \
    --name my-lattice-service
```

Output:

```
{
    "arn": "arn:aws:vpc-lattice:us-east-2:123456789012:service/
svc-0285b53b2eEXAMPLE",
    "authType": "NONE",
    "dnsEntry": {
        "domainName": "my-lattice-service-0285b53b2eEXAMPLE.1a2b3c4.vpc-lattice-
svcs.us-east-2.on.aws",
        "hostedZoneId": "Z09127221KTH2CEXAMPLE"
    },
    "id": "svc-0285b53b2eEXAMPLE",
    "name": "my-lattice-service",
    "status": "CREATE_IN_PROGRESS"
}
```

Per ulteriori informazioni, consulta Servizi in VPC Lattice nella Guida per l'utente di Amazon VPC Lattice.

- Per i dettagli sull'API, consulta CreateService AWS CLICommand Reference.

`create-target-group`

Il seguente esempio di codice mostra come utilizzare`create-target-group`.

AWS CLI

Esempio 1: come creare un gruppo di destinazione di tipo INSTANCE

L'esempio `create-target-group` seguente crea un gruppo di destinazione con il nome, il tipo e la configurazione specificati.

```
aws vpc-lattice create-target-group \
    --name my-lattice-target-group-instance \
    --type INSTANCE \
    --config file://tg-config.json
```

Contenuto di `tg-config.json`:

```
{
    "port": 443,
    "protocol": "HTTPS",
    "protocolVersion": "HTTP1",
    "vpcIdentifier": "vpc-f1663d9868EXAMPLE"
}
```

Output:

```
{
    "arn": "arn:aws:vpc-lattice:us-east-2:123456789012:targetgroup/
tg-0eaa4b9ab4EXAMPLE",
    "config": {
        "healthCheck": {
            "enabled": true,
            "healthCheckIntervalSeconds": 30,
            "healthCheckTimeoutSeconds": 5,
            "healthyThresholdCount": 5,
```

```
            "matcher": {
                "httpCode": "200"
            },
            "path": "/",
            "protocol": "HTTPS",
            "protocolVersion": "HTTP1",
            "unhealthyThresholdCount": 2
        },
        "port": 443,
        "protocol": "HTTPS",
        "protocolVersion": "HTTP1",
        "vpcIdentifier": "vpc-f1663d9868EXAMPLE"
    },
    "id": "tg-0eaa4b9ab4EXAMPLE",
    "name": "my-lattice-target-group-instance",
    "status": "CREATE_IN_PROGRESS",
    "type": "INSTANCE"
}
```

Esempio 2: come creare un gruppo di destinazione di tipo IP

L'esempio `create-target-group` seguente crea un gruppo di destinazione con il nome, il tipo e la configurazione specificati.

```
aws vpc-lattice create-target-group \
    --name my-lattice-target-group-ip \
    --type IP \
    --config file://tg-config.json
```

Contenuto di `tg-config.json`:

```
{
    "ipAddressType": "IPV4",
    "port": 443,
    "protocol": "HTTPS",
    "protocolVersion": "HTTP1",
    "vpcIdentifier": "vpc-f1663d9868EXAMPLE"
}
```

Output:

```
{
```

```
    "arn": "arn:aws:vpc-lattice:us-east-2:123456789012:targetgroup/
tg-0eaa4b9ab4EXAMPLE",
    "config": {
        "healthCheck": {
            "enabled": true,
            "healthCheckIntervalSeconds": 30,
            "healthCheckTimeoutSeconds": 5,
            "healthyThresholdCount": 5,
            "matcher": {
                "httpCode": "200"
            },
            "path": "/",
            "protocol": "HTTPS",
            "protocolVersion": "HTTP1",
            "unhealthyThresholdCount": 2
        },
        "ipAddressType": "IPV4",
        "port": 443,
        "protocol": "HTTPS",
        "protocolVersion": "HTTP1",
        "vpcIdentifier": "vpc-f1663d9868EXAMPLE"
    },
    "id": "tg-0eaa4b9ab4EXAMPLE",
    "name": "my-lattice-target-group-ip",
    "status": "CREATE_IN_PROGRESS",
    "type": "IP"
}
```

Esempio 3: come creare un gruppo di destinazione di tipo LAMBDA

L'esempio `create-target-group` seguente crea un gruppo di destinazione con il nome, il tipo e la configurazione specificati.

```
aws vpc-lattice create-target-group \
    --name my-lattice-target-group-lambda \
    --type LAMBDA
```

Output:

```
{
    "arn": "arn:aws:vpc-lattice:us-east-2:123456789012:targetgroup/
tg-0eaa4b9ab4EXAMPLE",
```

```
    "id": "tg-0eaa4b9ab4EXAMPLE",
    "name": "my-lattice-target-group-lambda",
    "status": "CREATE_IN_PROGRESS",
    "type": "LAMBDA"
}
```

Esempio 4: come creare un gruppo di destinazione di tipo ALB

L'esempio `create-target-group` seguente crea un gruppo di destinazione con il nome, il tipo e la configurazione specificati.

```
aws vpc-lattice create-target-group \
    --name my-lattice-target-group-alb \
    --type ALB \
    --config file://tg-config.json
```

Contenuto di `tg-config.json`:

```
{
    "port": 443,
    "protocol": "HTTPS",
    "protocolVersion": "HTTP1",
    "vpcIdentifier": "vpc-f1663d9868EXAMPLE"
}
```

Output:

```
{
    "arn": "arn:aws:vpc-lattice:us-east-2:123456789012:targetgroup/
tg-0eaa4b9ab4EXAMPLE",
    "config": {
        "port": 443,
        "protocol": "HTTPS",
        "protocolVersion": "HTTP1",
        "vpcIdentifier": "vpc-f1663d9868EXAMPLE"
    },
    "id": "tg-0eaa4b9ab4EXAMPLE",
    "name": "my-lattice-target-group-alb",
    "status": "CREATE_IN_PROGRESS",
    "type": "ALB"
}
```

Per ulteriori informazioni, consulta Gruppi di destinazione nella Guida per l'utente di Amazon VPC Lattice.

- Per i dettagli sull'API, consulta CreateTargetGroup AWS CLICommand Reference.

`delete-auth-policy`

Il seguente esempio di codice mostra come utilizzare`delete-auth-policy`.

AWS CLI

Come eliminare una policy di autenticazione

L'esempio `delete-auth-policy` seguente elimina la policy di autenticazione per il servizio specificato.

```
aws vpc-lattice delete-auth-policy \
    --resource-identifier svc-0285b53b2eEXAMPLE
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Policy di autenticazione nella Guida per l'utente di Amazon VPC Lattice.

- Per i dettagli sull'API, consulta DeleteAuthPolicy AWS CLICommand Reference.

`delete-listener`

Il seguente esempio di codice mostra come utilizzare`delete-listener`.

AWS CLI

Come eliminare un listener

L'esempio `delete-listener` seguente elimina il listener specificato.

```
aws vpc-lattice delete-listener \
    --listener-identifier listener-07cc7fb0abEXAMPLE \
    --service-identifier svc-0285b53b2eEXAMPLE
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Listener nella Guida per l'utente di Amazon VPC Lattice.

- Per i dettagli sull'API, consulta DeleteListener AWS CLICommand Reference.

`delete-resource-configuration`

Il seguente esempio di codice mostra come utilizzare`delete-resource-configuration`.

AWS CLI

Come eliminare una configurazione di risorse

L'esempio `delete-resource-configuration` seguente elimina la configurazione di risorse specificata.

```
aws vpc-lattice delete-resource-configuration \
    --resource-configuration-identifier rcfg-07129f3acded87625
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Gateway per le risorse in VPC Lattice nella Guida per l'utente di Amazon VPC Lattice.

- Per i dettagli sull'API, consulta DeleteResourceConfiguration AWS CLICommand Reference.

`delete-resource-gateway`

Il seguente esempio di codice mostra come utilizzare`delete-resource-gateway`.

AWS CLI

Come eliminare un gateway per le risorse

L'esempio `delete-resource-gateway` seguente elimina il gateway per le risorse specificato.

```
aws vpc-lattice delete-resource-gateway \
    --resource-gateway-identifier rgw-0bba03f3d56060135
```

Output:

```
{
```

```
    "arn": "arn:aws:vpc-lattice:us-east-1:123456789012:resourcegateway/
rgw-0bba03f3d56060135",
    "id": "rgw-0bba03f3d56060135",
    "name": "my-resource-gateway",
    "status": "DELETE_IN_PROGRESS"
}
```

Per ulteriori informazioni, consulta Gateway per le risorse in VPC Lattice nella Guida per l'utente di Amazon VPC Lattice.

- Per i dettagli sull'API, consulta DeleteResourceGateway AWS CLICommand Reference.

`delete-service-network-service-association`

Il seguente esempio di codice mostra come utilizzare`delete-service-network-service-association`.

AWS CLI

Come eliminare un'associazione di servizi

L'esempio `delete-service-network-service-association` seguente elimina l'associazione di servizi specificata.

```
aws vpc-lattice delete-service-network-service-association \
    --service-network-service-association-identifier snsa-031fabb4d8EXAMPLE
```

Output:

```
{
    "arn": "arn:aws:vpc-lattice:us-
east-2:123456789012:servicenetworkserviceassociation/snsa-031fabb4d8EXAMPLE",
    "id": "snsa-031fabb4d8EXAMPLE",
    "status": "DELETE_IN_PROGRESS"
}
```

Per ulteriori informazioni, consulta Gestire le associazioni dei servizi nella Guida per l'utente di Amazon VPC Lattice.

- Per i dettagli sull'API, consulta DeleteServiceNetworkServiceAssociation AWS CLICommand Reference.

`delete-service-network-vpc-association`

Il seguente esempio di codice mostra come utilizzare`delete-service-network-vpc-association`.

AWS CLI

Come eliminare un'associazione VPC

L'esempio `delete-service-network-vpc-association` seguente elimina l'associazione VPC specificata.

```
aws vpc-lattice delete-service-network-vpc-association \
    --service-network-vpc-association-identifier snva-0821fc8631EXAMPLE
```

Output:

```
{
    "arn": "arn:aws:vpc-lattice:us-east-2:123456789012:servicenetworkvpcassociation/
snva-0821fc8631EXAMPLE",
    "id": "snva-0821fc8631EXAMPLE",
    "status": "DELETE_IN_PROGRESS"
}
```

Per ulteriori informazioni, consulta Gestire le associazioni VPC nella Guida per l'utente di Amazon VPC Lattice.

- Per i dettagli sull'API, consulta DeleteServiceNetworkVpcAssociation AWS CLICommand Reference.

`delete-service-network`

Il seguente esempio di codice mostra come utilizzare`delete-service-network`.

AWS CLI

Come eliminare una rete di servizi

L'esempio `delete-service-network` seguente elimina la rete di servizi specificata.

```
aws vpc-lattice delete-service-network \
    --service-network-identifier sn-080ec7dc93EXAMPLE
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Reti di servizio nella Guida per l'utente di Amazon VPC Lattice.

- Per i dettagli sull'API, consulta DeleteServiceNetwork AWS CLICommand Reference.

`delete-service`

Il seguente esempio di codice mostra come utilizzare`delete-service`.

AWS CLI

Come eliminare un servizio

L'esempio `delete-service` seguente elimina il servizio specificato.

```
aws vpc-lattice delete-service \
    --service-identifier svc-0285b53b2eEXAMPLE
```

Output:

```
{
    "arn": "arn:aws:vpc-lattice:us-west-2:123456789012:service/
svc-0285b53b2eEXAMPLE",
    "id": "svc-0285b53b2eEXAMPLE",
    "name": "my-lattice-service",
    "status": "DELETE_IN_PROGRESS"
}
```

Per ulteriori informazioni, consulta Servizi in VPC Lattice nella Guida per l'utente di Amazon VPC Lattice.

- Per i dettagli sull'API, consulta DeleteService AWS CLICommand Reference.

`delete-target-group`

Il seguente esempio di codice mostra come utilizzare`delete-target-group`.

AWS CLI

Come eliminare un gruppo di destinazione

L'esempio `delete-target-group` seguente elimina il gruppo di destinazione specificato.

```
aws vpc-lattice delete-target-group \
    --target-group-identifier tg-0eaa4b9ab4EXAMPLE
```

Output:

```
{
    "arn": "arn:aws:vpc-lattice:us-east-2:123456789012:targetgroup/
tg-0eaa4b9ab4EXAMPLE",
    "id": "tg-0eaa4b9ab4EXAMPLE",
    "status": "DELETE_IN_PROGRESS"
}
```

Per ulteriori informazioni, consulta Gruppi di destinazione nella Guida per l'utente di Amazon VPC Lattice.

- Per i dettagli sull'API, consulta DeleteTargetGroup AWS CLICommand Reference.

`deregister-targets`

Il seguente esempio di codice mostra come utilizzare`deregister-targets`.

AWS CLI

Come annullare la registrazione di una destinazione

L'esempio `deregister-targets` seguente annulla la registrazione della destinazione specificata nel gruppo di destinazione specificato.

```
aws vpc-lattice deregister-targets \
    --targets i-07dd579bc5EXAMPLE \
    --target-group-identifier tg-0eaa4b9ab4EXAMPLE
```

Output:

```
{
    "successful": [
        {
            "id": "i-07dd579bc5EXAMPLE",
            "port": 443
        }
    ],
```

```
    "unsuccessful": []
}
```

Per ulteriori informazioni, consulta Registrare le destinazioni nella Guida per l'utente di Amazon VPC Lattice.

- Per i dettagli sull'API, consulta DeregisterTargets AWS CLICommand Reference.

`get-auth-policy`

Il seguente esempio di codice mostra come utilizzare`get-auth-policy`.

AWS CLI

Come ottenere informazioni su una policy di autorizzazione

L'esempio `get-auth-policy` seguente ottiene informazioni sulla policy di autenticazione per il servizio specificato.

```
aws vpc-lattice get-auth-policy \
    --resource-identifier svc-0285b53b2eEXAMPLE
```

Output:

```
{
    "createdAt": "2023-06-07T03:51:20.266Z",
    "lastUpdatedAt": "2023-06-07T04:39:27.082Z",
    "policy": "{\"Version\":\"2012-10-17\",\"Statement\":[{\"Effect\":\"Allow
\",\"Principal\":{\"AWS\":\"arn:aws:iam::123456789012:role/my-clients\"},
\"Action\":\"vpc-lattice-svcs:Invoke\",\"Resource\":\"arn:aws:vpc-lattice:us-
east-2:123456789012:service/svc-0285b53b2eEXAMPLE\"}]}",
    "state": "Active"
}
```

Per ulteriori informazioni, consulta Policy di autenticazione nella Guida per l'utente di Amazon VPC Lattice.

- Per i dettagli sull'API, consulta GetAuthPolicy AWS CLICommand Reference.

`get-listener`

Il seguente esempio di codice mostra come utilizzare`get-listener`.

AWS CLI

Come ottenere informazioni su un listener di un servizio

L'esempio `get-listener` seguente ottiene informazioni sul listener specificato per il servizio specificato.

```
aws vpc-lattice get-listener \
    --listener-identifier listener-0ccf55918cEXAMPLE \
    --service-identifier svc-0285b53b2eEXAMPLE
```

Output:

```
{
    "arn": "arn:aws:vpc-lattice:us-east-2:123456789012:service/
svc-0285b53b2eEXAMPLE/listener/listener-0ccf55918cEXAMPLE",
    "createdAt": "2023-05-07T05:08:45.192Z",
    "defaultAction": {
        "forward": {
            "targetGroups": [
                {
                    "targetGroupIdentifier": "tg-0ff213abb6EXAMPLE",
                    "weight": 1
                }
            ]
        }
    },
    "id": "listener-0ccf55918cEXAMPLE",
    "lastUpdatedAt": "2023-05-07T05:08:45.192Z",
    "name": "http-80",
    "port": 80,
    "protocol": "HTTP",
    "serviceArn": "arn:aws:vpc-lattice:us-east-2:123456789012:service/
svc-0285b53b2eEXAMPLE",
    "serviceId": "svc-0285b53b2eEXAMPLE"
}
```

Per ulteriori informazioni, consulta Definire il routing nella Guida per l'utente di Amazon VPC Lattice.

- Per i dettagli sull'API, consulta GetListener AWS CLICommand Reference.

`get-resource-configuration`

Il seguente esempio di codice mostra come utilizzare`get-resource-configuration`.

AWS CLI

Come ottenere informazioni su una configurazione delle risorse

L'esempio `get-resource-configuration` seguente ottiene informazioni sulla configurazione delle risorse specificata.

```
aws vpc-lattice get-resource-configuration \
    --resource-configuration-identifier rcfg-07129f3acded87625
```

Output:

```
{
    "allowAssociationToShareableServiceNetwork": true,
    "amazonManaged": false,
    "arn": "arn:aws:vpc-lattice:us-east-1:123456789012:resourceconfiguration/
rcfg-07129f3acded87625",
    "createdAt": "2025-02-01T00:57:35.871000+00:00",
    "id": "rcfg-07129f3acded87625",
    "lastUpdatedAt": "2025-02-01T00:57:46.874000+00:00",
    "name": "my-resource-config",
    "portRanges": [
        "1-65535"
    ],
    "protocol": "TCP",
    "resourceConfigurationDefinition": {
        "ipResource": {
            "ipAddress": "10.0.14.85"
        }
    },
    "resourceGatewayId": "rgw-0bba03f3d56060135",
    "status": "ACTIVE",
    "type": "SINGLE"
}
```

Per ulteriori informazioni, consulta Gateway per le risorse in VPC Lattice nella Guida per l'utente di Amazon VPC Lattice.

- Per i dettagli sull'API, consulta GetResourceConfiguration AWS CLICommand Reference.

`get-resource-gateway`

Il seguente esempio di codice mostra come utilizzare`get-resource-gateway`.

AWS CLI

Come ottenere informazioni su un gateway per le risorse

L'esempio `get-resource-gateway` seguente ottiene informazioni sul gateway per le risorse specificato.

```
aws vpc-lattice get-resource-gateway \
    --resource-gateway-identifier rgw-0bba03f3d56060135
```

Output:

```
{
    "arn": "arn:aws:vpc-lattice:us-east-1:123456789012:resourcegateway/
rgw-0bba03f3d56060135",
    "createdAt": "2025-02-01T00:57:33.241000+00:00",
    "id": "rgw-0bba03f3d56060135",
    "ipAddressType": "IPV4",
    "lastUpdatedAt": "2025-02-01T00:57:44.351000+00:00",
    "name": "my-resource-gateway",
    "securityGroupIds": [
        "sg-087ffd596c5fe962c"
    ],
    "status": "ACTIVE",
    "subnetIds": [
        "subnet-08e8943905b63a683"
    ],
    "vpcId": "vpc-0bf4c2739bc05a694"
}
```

Per ulteriori informazioni, consulta Gateway per le risorse in VPC Lattice nella Guida per l'utente di Amazon VPC Lattice.

- Per i dettagli sull'API, consulta GetResourceGateway AWS CLICommand Reference.

`get-service-network-service-association`

Il seguente esempio di codice mostra come utilizzare`get-service-network-service-association`.

AWS CLI

Come ottenere informazioni su un'associazione di servizi

L'esempio `get-service-network-service-association` seguente ottiene informazioni sull'associazione di servizi specificata.

```
aws vpc-lattice get-service-network-service-association \
    --service-network-service-association-identifier snsa-031fabb4d8EXAMPLE
```

Output:

```
{
    "arn": "arn:aws:vpc-lattice:us-
east-2:123456789012:servicenetworkserviceassociation/snsa-031fabb4d8EXAMPLE",
    "createdAt": "2023-05-05T21:48:16.076Z",
    "createdBy": "123456789012",
    "dnsEntry": {
        "domainName": "my-lattice-service-0285b53b2eEXAMPLE.7d67968.vpc-lattice-
svcs.us-east-2.on.aws",
        "hostedZoneId": "Z09127221KTH2CEXAMPLE"
    },
    "id": "snsa-031fabb4d8EXAMPLE",
    "serviceArn": "arn:aws:vpc-lattice:us-east-2:123456789012:service/
svc-0285b53b2eEXAMPLE",
    "serviceId": "svc-0285b53b2eEXAMPLE",
    "serviceName": "my-lattice-service",
    "serviceNetworkArn": "arn:aws:vpc-lattice:us-east-2:123456789012:servicenetwork/
sn-080ec7dc93EXAMPLE",
    "serviceNetworkId": "sn-080ec7dc93EXAMPLE",
    "serviceNetworkName": "my-service-network",
    "status": "ACTIVE"
}
```

Per ulteriori informazioni, consulta Gestire le associazioni dei servizi nella Guida per l'utente di Amazon VPC Lattice.

- Per i dettagli sull'API, consulta GetServiceNetworkServiceAssociation AWS CLICommand Reference.

`get-service-network-vpc-association`

Il seguente esempio di codice mostra come utilizzare`get-service-network-vpc-association`.

AWS CLI

Come ottenere informazioni su un'associazione VPC

L'esempio `get-service-network-vpc-association` seguente ottiene informazioni sull'associazione VPC specificata.

```
aws vpc-lattice get-service-network-vpc-association \
    --service-network-vpc-association-identifier snva-0821fc8631EXAMPLE
```

Output:

```
{
    "arn": "arn:aws:vpc-lattice:us-east-2:123456789012:servicenetworkvpcassociation/
snva-0821fc8631EXAMPLE",
    "createdAt": "2023-06-06T23:41:08.421Z",
    "createdBy": "123456789012",
    "id": "snva-0c5dcb60d6EXAMPLE",
    "lastUpdatedAt": "2023-06-06T23:41:08.421Z",
    "securityGroupIds": [
        "sg-0aee16bc6cEXAMPLE"
    ],
    "serviceNetworkArn": "arn:aws:vpc-lattice:us-east-2:123456789012:servicenetwork/
sn-080ec7dc93EXAMPLE",
    "serviceNetworkId": "sn-080ec7dc93EXAMPLE",
    "serviceNetworkName": "my-service-network",
    "status": "ACTIVE",
    "vpcId": "vpc-0a1b2c3d4eEXAMPLE"
}
```

Per ulteriori informazioni, consulta Gestire le associazioni VPC nella Guida per l'utente di Amazon VPC Lattice.

- Per i dettagli sull'API, consulta GetServiceNetworkVpcAssociation AWS CLICommand Reference.

`get-service-network`

Il seguente esempio di codice mostra come utilizzare`get-service-network`.

AWS CLI

Come ottenere informazioni su una rete di servizi

L'esempio `get-service-network` seguente ottiene informazioni sulla rete di servizi specificata.

```
aws vpc-lattice get-service-network \
    --service-network-identifier sn-080ec7dc93EXAMPLE
```

Output:

```
{
   "arn": "arn:aws:vpc-lattice:us-east-2:123456789012:servicenetwork/
sn-080ec7dc93EXAMPLE",
   "authType": "AWS_IAM",
   "createdAt": "2023-05-05T15:26:08.417Z",
   "id": "sn-080ec7dc93EXAMPLE",
   "lastUpdatedAt": "2023-05-05T15:26:08.417Z",
   "name": "my-service-network",
   "numberOfAssociatedServices": 2,
   "numberOfAssociatedVPCs": 3
}
```

Per ulteriori informazioni, consulta Reti di servizio nella Guida per l'utente di Amazon VPC Lattice.

- Per i dettagli sull'API, consulta GetServiceNetwork AWS CLICommand Reference.

`get-service`

Il seguente esempio di codice mostra come utilizzare`get-service`.

AWS CLI

Come ottenere informazioni su un servizio

L'esempio `get-service` seguente ottiene informazioni sul servizio specificato.

```
aws vpc-lattice get-service \
    --service-identifier svc-0285b53b2eEXAMPLE
```

Output:

```
{
    "arn": "arn:aws:vpc-lattice:us-east-2:123456789012:service/
svc-0285b53b2eEXAMPLE",
    "authType": "AWS_IAM",
    "createdAt": "2023-05-05T21:35:29.339Z",
    "dnsEntry": {
        "domainName": "my-lattice-service-0285b53b2eEXAMPLE.7d67968.vpc-lattice-
svcs.us-east-2.on.aws",
        "hostedZoneId": "Z09127221KTH2CFUOHIZH"
    },
    "id": "svc-0285b53b2eEXAMPLE",
    "lastUpdatedAt": "2023-05-05T21:35:29.339Z",
    "name": "my-lattice-service",
    "status": "ACTIVE"
}
```

Per ulteriori informazioni, consulta Servizi nella Guida per l'utente di Amazon VPC Lattice.

- Per i dettagli sull'API, consulta GetService AWS CLICommand Reference.

`get-target-group`

Il seguente esempio di codice mostra come utilizzare`get-target-group`.

AWS CLI

Come ottenere informazioni su un gruppo di destinazione

L'esempio `get-target-group` seguente ottiene informazioni sul gruppo di destinazione specificato, che ha un tipo di destinazione `INSTANCE`.

```
aws vpc-lattice get-target-group \
    --target-group-identifier tg-0eaa4b9ab4EXAMPLE
```

Output:

```
{
    "arn": "arn:aws:vpc-lattice:us-east-2:123456789012:targetgroup/
tg-0eaa4b9ab4EXAMPLE",
    "config": {
```

```
        "healthCheck": {
            "enabled": true,
            "healthCheckIntervalSeconds": 30,
            "healthCheckTimeoutSeconds": 5,
            "healthyThresholdCount": 5,
            "matcher": {
                "httpCode": "200"
            },
            "path": "/",
            "protocol": "HTTPS",
            "protocolVersion": "HTTP1",
            "unhealthyThresholdCount": 2
        },
        "port": 443,
        "protocol": "HTTPS",
        "protocolVersion": "HTTP1",
        "vpcIdentifier": "vpc-f1663d9868EXAMPLE"
    },
    "createdAt": "2023-05-06T04:41:04.122Z",
    "id": "tg-0eaa4b9ab4EXAMPLE",
    "lastUpdatedAt": "2023-05-06T04:41:04.122Z",
    "name": "my-target-group",
    "serviceArns": [
        "arn:aws:vpc-lattice:us-east-2:123456789012:service/svc-0285b53b2eEXAMPLE"
    ],
    "status": "ACTIVE",
    "type": "INSTANCE"
}
```


Per ulteriori informazioni, consulta Gruppi di destinazione nella Guida per l'utente di Amazon VPC Lattice.

- Per i dettagli sull'API, consulta GetTargetGroup AWS CLICommand Reference.

`list-listeners`

Il seguente esempio di codice mostra come utilizzare`list-listeners`.

AWS CLI

Come elencare i listener per un servizio

L'esempio `list-listeners` seguente elenca i listener per il servizio specificato.

```
aws vpc-lattice list-listeners \
    --service-identifier svc-0285b53b2eEXAMPLE
```

Output:

```
{
    "items": [
        {
            "arn": "arn:aws:vpc-lattice:us-east-2:123456789012:service/
svc-0285b53b2eEXAMPLE/listener/listener-0ccf55918cEXAMPLE",
            "createdAt": "2023-05-07T05:08:45.192Z",
            "id": "listener-0ccf55918cEXAMPLE",
            "lastUpdatedAt": "2023-05-07T05:08:45.192Z",
            "name": "http-80",
            "port": 80,
            "protocol": "HTTP"
        }
    ]
}
```

Per ulteriori informazioni, consulta Definire il routing nella Guida per l'utente di Amazon VPC Lattice.

- Per i dettagli sull'API, consulta ListListeners AWS CLICommand Reference.

`list-resource-configurations`

Il seguente esempio di codice mostra come utilizzare`list-resource-configurations`.

AWS CLI

Come elencare le configurazioni delle risorse

L'esempio `list-resource-configurations` seguente elenca le configurazioni delle risorse.

```
aws vpc-lattice list-resource-configurations
```

Output:

```
{
    "items": [
```

```
        {
            "amazonManaged": false,
            "arn": "arn:aws:vpc-lattice:us-
east-1:123456789012:resourceconfiguration/rcfg-07129f3acded87625",
            "createdAt": "2025-02-01T00:57:35.871000+00:00",
            "id": "rcfg-07129f3acded87625",
            "lastUpdatedAt": "2025-02-01T00:57:46.874000+00:00",
            "name": "my-resource-config",
            "resourceGatewayId": "rgw-0bba03f3d56060135",
            "status": "ACTIVE",
            "type": "SINGLE"
        }
    ]
}
```

Per ulteriori informazioni, consulta Configurazione delle risorse nella Guida per l'utente di Amazon VPC Lattice.

- Per i dettagli sull'API, consulta ListResourceConfigurations AWS CLICommand Reference.

`list-resource-endpoint-associations`

Il seguente esempio di codice mostra come utilizzare`list-resource-endpoint-associations`.

AWS CLI

Come elencare le associazioni di endpoint VPC

L'esempio `list-resource-endpoint-associations` seguente elenca gli endpoint VPC associati alla configurazione delle risorse specificata.

```
aws vpc-lattice list-resource-endpoint-associations \
    --resource-configuration-identifier rcfg-07129f3acded87625
```

Output:

```
{
    "items": [
        {
            "arn": "arn:aws:vpc-lattice:us-
east-1:123456789012:resourceendpointassociation/rea-0956a7435baf89326",
            "createdAt": "2025-02-01T00:57:38.998000+00:00",
```

```
            "id": "rea-0956a7435baf89326",
            "resourceConfigurationArn": "arn:aws:vpc-lattice:us-
east-1:123456789012:resourceconfiguration/rcfg-07129f3acded87625",
            "resourceConfigurationId": "rcfg-07129f3acded87625",
            "vpcEndpointId": "vpce-019b90d6f16d4f958",
            "vpcEndpointOwner": "123456789012"
        }
    ]
}
```

Per ulteriori informazioni, consulta Gestire le associazioni per una configurazione di risorse VPC Lattice nella Guida per l'utente di Amazon VPC Lattice.

- Per i dettagli sull'API, consulta ListResourceEndpointAssociations AWS CLICommand Reference.

`list-resource-gateways`

Il seguente esempio di codice mostra come utilizzare`list-resource-gateways`.

AWS CLI

Come elencare i gateway per le risorse

L'esempio `list-resource-gateways` seguente elenca tutti i gateway per le risorse in uso.

```
aws vpc-lattice list-resource-gateways
```

Output:

```
{
    "items": [
        {
            "arn": "arn:aws:vpc-lattice:us-east-1:123456789012:resourcegateway/
rgw-0bba03f3d56060135",
            "createdAt": "2025-02-01T00:57:33.241000+00:00",
            "id": "rgw-0bba03f3d56060135",
            "ipAddressType": "IPV4",
            "lastUpdatedAt": "2025-02-01T00:57:44.351000+00:00",
            "name": "my-resource-gateway",
            "seurityGroupIds": [
                "sg-087ffd596c5fe962c"
```

```
            ],
            "status": "ACTIVE",
            "subnetIds": [
                "subnet-08e8943905b63a683"
            ],
            "vpcIdentifier": "vpc-0bf4c2739bc05a694"
        }
    ]
}
```

Per ulteriori informazioni, consulta Gateway per le risorse in VPC Lattice nella Guida per l'utente di Amazon VPC Lattice.

- Per i dettagli sull'API, consulta ListResourceGateways AWS CLICommand Reference.

`list-service-network-service-associations`

Il seguente esempio di codice mostra come utilizzare`list-service-network-service-associations`.

AWS CLI

Come elencare le associazioni di servizi

L'esempio `list-service-network-service-associations` seguente elenca le associazioni di servizi per la rete di servizi specificata. L'`--query`opzione indirizza l'output alle associazioni IDs di servizi.

```
aws vpc-lattice list-service-network-service-associations \
    --service-network-identifier sn-080ec7dc93EXAMPLE \
    --query items[*].id
```

Output:

```
[
    "snsa-031fabb4d8EXAMPLE",
    "snsa-0e16955a8cEXAMPLE"
]
```

Per ulteriori informazioni, consulta Gestire le associazioni dei servizi nella Guida per l'utente di Amazon VPC Lattice.

- Per i dettagli sull'API, vedere ListServiceNetworkServiceAssociationsin AWS CLI Command Reference.

`list-service-network-vpc-associations`

Il seguente esempio di codice mostra come utilizzare`list-service-network-vpc-associations`.

AWS CLI

Come elencare le associazioni VPC

L'esempio `list-service-network-vpc-associations` seguente elenca le associazioni VPC per la rete di servizi specificata. L'`--query`opzione indirizza l'output alle IDs associazioni VPC.

```
aws vpc-lattice list-service-network-vpc-associations \
    --service-network-identifier sn-080ec7dc93EXAMPLE \
    --query items[*].id
```

Output:

```
[
    "snva-0821fc8631EXAMPLE",
    "snva-0c5dcb60d6EXAMPLE"
]
```

Per ulteriori informazioni, consulta Gestire le associazioni VPC nella Guida per l'utente di Amazon VPC Lattice.

- Per i dettagli sull'API, vedere ListServiceNetworkVpcAssociationsin AWS CLI Command Reference.

`list-service-network-vpc-endpoint-associations`

Il seguente esempio di codice mostra come utilizzare`list-service-network-vpc-endpoint-associations`.

AWS CLI

Come elencare le associazioni di endpoint VPC

L'esempio `list-service-network-vpc-endpoint-associations` seguente elenca gli endpoint VPC associati alla rete di servizi specificata.

```
aws vpc-lattice list-service-network-vpc-endpoint-associations \
    --service-network-identifier sn-0808d1748faee0c1e
```

Output:

```
{
    "items": [
        {
            "createdAt": "2025-02-01T01:21:36.667000+00:00",
            "serviceNetworkArn": "arn:aws:vpc-lattice:us-
east-1:123456789012:servicenetwork/sn-0808d1748faee0c1e",
            "state": "ACTIVE",
            "vpcEndpointId": "vpce-0cc199f605eaeace7",
            "vpcEndpointOwnerId": "123456789012"
        }
    ]
}
```

Per ulteriori informazioni, consulta Gestire le associazioni per una rete di servizio di VPC Lattice nella Guida per l'utente di Amazon VPC Lattice.

- Per i dettagli sull'API, consulta ListServiceNetworkVpcEndpointAssociations AWS CLICommand Reference.

`list-service-networks`

Il seguente esempio di codice mostra come utilizzare`list-service-networks`.

AWS CLI

Come elencare le reti di servizi

L'esempio `list-service-networks` seguente elenca le reti di servizi di proprietà o condivise con l'account chiamante. L'opzione `--query` definisce i nomi delle risorse Amazon (ARN) delle reti di servizi come ambito dei risultati.

```
aws vpc-lattice list-service-networks \
    --query items[*].arn
```

Output:

```
[
    "arn:aws:vpc-lattice:us-east-2:123456789012:servicenetwork/
sn-080ec7dc93EXAMPLE",
    "arn:aws:vpc-lattice:us-east-2:111122223333:servicenetwork/sn-0ec4d436cfEXAMPLE"
]
```

Per ulteriori informazioni, consulta Reti di servizio nella Guida per l'utente di Amazon VPC Lattice.

- Per i dettagli sull'API, consulta ListServiceNetworks AWS CLICommand Reference.

`list-services`

Il seguente esempio di codice mostra come utilizzare`list-services`.

AWS CLI

Come elencare i servizi

L'esempio `list-services` seguente elenca i servizi di proprietà o condivisi con l'account chiamante. L'opzione `--query` definisce i nomi delle risorse Amazon (ARN) dei servizi come ambito dei risultati.

```
aws vpc-lattice list-services \
    --query items[*].arn
```

Output:

```
[
    "arn:aws:vpc-lattice:us-east-2:123456789012:service/svc-0285b53b2eEXAMPLE",
    "arn:aws:vpc-lattice:us-east-2:111122223333:service/svc-0b8ac96550EXAMPLE"
]
```

Per ulteriori informazioni, consulta Servizi nella Guida per l'utente di Amazon VPC Lattice.

- Per i dettagli sull'API, consulta ListServices AWS CLICommand Reference.

`list-target-groups`

Il seguente esempio di codice mostra come utilizzare`list-target-groups`.

AWS CLI

Come elencare i gruppi di destinazione

L'esempio `list-target-groups` seguente elenca i gruppi di destinazione con un tipo di destinazione LAMBDA.

```
aws vpc-lattice list-target-groups \
    --target-group-type LAMBDA
```

Output:

```
{
    "items": [
        {
            "arn": "arn:aws:vpc-lattice:us-east-2:123456789012:targetgroup/
tg-045c1b7d9dEXAMPLE",
            "createdAt": "2023-05-06T05:22:16.637Z",
            "id": "tg-045c1b7d9dEXAMPLE",
            "lastUpdatedAt": "2023-05-06T05:22:16.637Z",
            "name": "my-target-group-lam",
            "serviceArns": [
                "arn:aws:vpc-lattice:us-east-2:123456789012:service/
svc-0285b53b2eEXAMPLE"
            ],
            "status": "ACTIVE",
            "type": "LAMBDA"
        }
    ]
}
```

Per ulteriori informazioni, consulta Gruppi di destinazione nella Guida per l'utente di Amazon VPC Lattice.

- Per i dettagli sull'API, consulta ListTargetGroups AWS CLICommand Reference.

`list-targets`

Il seguente esempio di codice mostra come utilizzare`list-targets`.

AWS CLI

Come elencare le destinazioni per un gruppo di destinazione

L'esempio `list-targets` seguente elenca le destinazioni per il gruppo di destinazione specificato.

```
aws vpc-lattice list-targets \
    --target-group-identifier tg-0eaa4b9ab4EXAMPLE
```

Output:

```
{
    "items": [
        {
            "id": "i-07dd579bc5EXAMPLE",
            "port": 443,
            "status": "HEALTHY"
        },
        {
            "id": "i-047b3c9078EXAMPLE",
            "port": 443,
            "reasonCode": "HealthCheckFailed",
            "status": "UNHEALTHY"
        }
    ]
}
```

Per ulteriori informazioni, consulta Gruppi di destinazione nella Guida per l'utente di Amazon VPC Lattice.

- Per i dettagli sull'API, consulta ListTargets AWS CLICommand Reference.

`put-auth-policy`

Il seguente esempio di codice mostra come utilizzare`put-auth-policy`.

AWS CLI

Come creare una policy di autenticazione per un servizio

L'esempio `put-auth-policy` seguente concede l'accesso alle richieste da qualsiasi principale autenticato che utilizza il ruolo IAM specificato. La risorsa è l'ARN del servizio a cui è collegata la policy.

```
aws vpc-lattice put-auth-policy \
```

```
    --resource-identifier svc-0285b53b2eEXAMPLE \
    --policy file://auth-policy.json
```

Contenuto di `auth-policy.json`:

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::123456789012:role/my-clients"
            },
            "Action": "vpc-lattice-svcs:Invoke",
            "Resource": "arn:aws:vpc-lattice:us-east-2:123456789012:service/
svc-0285b53b2eEXAMPLE"
        }
    ]
}
```

Output:

```
{
    "policy": "{\"Version\":\"2012-10-17\",\"Statement\":[{\"Effect\":\"Allow
\",\"Principal\":{\"AWS\":\"arn:aws:iam::123456789012:role/my-clients\"},
\"Action\":\"vpc-lattice-svcs:Invoke\",\"Resource\":\"arn:aws:vpc-lattice:us-
east-2:123456789012:service/svc-0285b53b2eEXAMPLE\"}]}",
    "state": "Active"
}
```

Per ulteriori informazioni, consulta Policy di autenticazione nella Guida per l'utente di Amazon VPC Lattice.

- Per i dettagli sull'API, consulta PutAuthPolicy AWS CLICommand Reference.

`register-targets`

Il seguente esempio di codice mostra come utilizzare`register-targets`.

AWS CLI

Come registrare una destinazione

L'esempio `register-targets` seguente registra la destinazione specificata nel gruppo di destinazione specificato.

```
aws vpc-lattice register-targets \
    --targets id=i-047b3c9078EXAMPLE id=i-07dd579bc5EXAMPLE \
    --target-group-identifier tg-0eaa4b9ab4EXAMPLE
```

Output:

```
{
    "successful": [
        {
            "id": "i-07dd579bc5EXAMPLE",
            "port": 443
        }
    ],
    "unsuccessful": [
        {
            "failureCode": "UnsupportedTarget",
            "failureMessage": "Instance targets must be in the same VPC as their
 target group",
            "id": "i-047b3c9078EXAMPLE",
            "port": 443
        }
    ]
}
```

Per ulteriori informazioni, consulta Registrare le destinazioni nella Guida per l'utente di Amazon VPC Lattice.

- Per i dettagli sull'API, consulta RegisterTargets AWS CLICommand Reference.

AWS WAF Classic esempi utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with AWS WAF Classic.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`put-logging-configuration`

Il seguente esempio di codice mostra come utilizzare`put-logging-configuration`.

AWS CLI

Come creare una configurazione di registrazione di log per l'ARN WebACL con il flusso ARN di Kinesis Firehose specificato

L'`put-logging-configuration`esempio seguente mostra la configurazione di registrazione per WAF con. CloudFront

```
aws waf put-logging-configuration \
    --logging-configuration ResourceArn=arn:aws:waf::123456789012:webacl/3bffd3ed-fa2e-445e-869f-a6a7cf153fd3,LogDestinationConfigs=arn:aws:firehose:us-east-1:123456789012:deliverystream/aws-waf-logs-firehose-stream,RedactedFields=[]
```

Output:

```
{
    "LoggingConfiguration": {
        "ResourceArn": "arn:aws:waf::123456789012:webacl/3bffd3ed-fa2e-445e-869f-a6a7cf153fd3",
        "LogDestinationConfigs": [
            "arn:aws:firehose:us-east-1:123456789012:deliverystream/aws-waf-logs-firehose-stream"
        ]
    }
}
```

- Per i dettagli sull'API, vedere PutLoggingConfigurationin AWS CLI Command Reference.

update-byte-match-set

Il seguente esempio di codice mostra come utilizzareupdate-byte-match-set.

AWS CLI

Come aggiornare un set di corrispondenze di byte

Il update-byte-match-set comando seguente elimina un ByteMatchTuple oggetto (filtro) in un ByteMatchSet:

```
aws waf update-byte-match-set --byte-match-set-id a123fae4-
b567-8e90-1234-5ab67ac8ca90 --change-token 12cs345-67cd-890b-1cd2-c3a4567d89f1 --
updates
 Action="DELETE",ByteMatchTuple={FieldToMatch={Type="HEADER",Data="referer"},TargetString="b
```

Per ulteriori informazioni, consulta Utilizzo di condizioni di corrispondenza per stringa nella Guida per gli sviluppatori di AWS WAF.

- Per i dettagli sull'API, vedere UpdateByteMatchSetin AWS CLI Command Reference.

update-ip-set

Il seguente esempio di codice mostra come utilizzareupdate-ip-set.

AWS CLI

Come creare un set di IP

Il update-ip-set comando seguente aggiorna un IPSet con un IPv4 indirizzo ed elimina un IPv6 indirizzo:

```
aws waf update-ip-set --ip-set-id a123fae4-b567-8e90-1234-5ab67ac8ca90
 --change-token 12cs345-67cd-890b-1cd2-c3a4567d89f1 --updates
 Action="INSERT",IPSetDescriptor={Type="IPV4",Value="12.34.56.78/16"},Action="DELETE",IPSetD
```

In alternativa, è possibile utilizzare un file JSON per specificare l'input. Esempio:

```
aws waf update-ip-set --ip-set-id a123fae4-b567-8e90-1234-5ab67ac8ca90 --change-
token 12cs345-67cd-890b-1cd2-c3a4567d89f1  --updates file://change.json
```

Dove si trova il contenuto del file:

```
[
{
"Action": "INSERT",
"IPSetDescriptor":
{
"Type": "IPV4",
"Value": "12.34.56.78/16"
}
},
{
"Action": "DELETE",
"IPSetDescriptor":
{
"Type": "IPV6",
"Value": "1111:0000:0000:0000:0000:0000:0000:0111/128"
}
}
]
```

Per ulteriori informazioni, consulta Utilizzo di condizioni di corrispondenza IP nella Guida per gli sviluppatori di AWS WAF.

- Per i dettagli sull'API, vedere UpdateIpSetin AWS CLI Command Reference.

`update-rule`

Il seguente esempio di codice mostra come utilizzare`update-rule`.

AWS CLI

Come aggiornare una regola

Il comando `update-rule` seguente elimina un oggetto Predicate in una regola:

```
aws waf update-rule --rule-id a123fae4-b567-8e90-1234-5ab67ac8ca90
 --change-token 12cs345-67cd-890b-1cd2-c3a4567d89f1 --updates
 Action="DELETE",Predicate={Negated=false,Type="ByteMatch",DataId="MyByteMatchSetID"}
```

Per ulteriori informazioni, consulta Utilizzo delle regole nella Guida per gli sviluppatori di AWS WAF.

- Per i dettagli sull'API, consulta UpdateRule AWS CLICommand Reference.

`update-size-constraint-set`

Il seguente esempio di codice mostra come utilizzare`update-size-constraint-set`.

AWS CLI

Come aggiornare un set di vincoli di dimensione

Il `update-size-constraint-set` comando seguente elimina un SizeConstraint oggetto (filtri) in un set di vincoli di dimensione:

```
aws waf update-size-constraint-set --size-constraint-set-id a123fae4-
b567-8e90-1234-5ab67ac8ca90 --change-token 12cs345-67cd-890b-1cd2-c3a4567d89f1 --
updates
 Action="DELETE",SizeConstraint={FieldToMatch={Type="QUERY_STRING"},TextTransformation="NONE
```

Per ulteriori informazioni, consulta Utilizzo di condizioni per i vincoli di dimensioni nella Guida per gli sviluppatori di AWS WAF.

- Per i dettagli sull'API, consultate Command UpdateSizeConstraintSetReference AWS CLI .

`update-sql-injection-match-set`

Il seguente esempio di codice mostra come utilizzare`update-sql-injection-match-set`.

AWS CLI

Come aggiornare un set di corrispondenze dell'iniezione SQL

Il `update-sql-injection-match-set` comando seguente elimina un SqlInjectionMatchTuple oggetto (filtri) in un set di match di SQL Injection:

```
aws waf update-sql-injection-match-set --sql-injection-
match-set-id a123fae4-b567-8e90-1234-5ab67ac8ca90 --
change-token 12cs345-67cd-890b-1cd2-c3a4567d89f1 --updates
 Action="DELETE",SqlInjectionMatchTuple={FieldToMatch={Type="QUERY_STRING"},TextTransformati
```

Per ulteriori informazioni, consulta Utilizzo di condizioni di corrispondenza SQL injection nella Guida per gli sviluppatori di AWS WAF.

- Per i dettagli sull'API, vedere UpdateSqlInjectionMatchSetin AWS CLI Command Reference.

`update-web-acl`

Il seguente esempio di codice mostra come utilizzareupdate-web-acl.

AWS CLI

Come aggiornare una lista di controllo degli accessi (ACL) web

Il comando update-web-acl seguente elimina un oggetto ActivatedRule in un WebACL.

aws waf update-web-acl -- web-acl-id a123fae4-b567-8e90-1234-5ab67ac8ca90 --change-token 12cs345-67cd-890b-1cd2-c3a4567d89f1 --updates action="Delete», =' {Priority=1, =» -1-Example», Action= {type="Allow "}, type="Regular"} 'ActivatedRuleRuleIdWAFRule

Output:

```
{
    "ChangeToken": "12cs345-67cd-890b-1cd2-c3a4567d89f1"
}
```

Per ulteriori informazioni, consulta Working with Web ACLs nella AWS WAF, AWS Firewall Manager and AWS Shield Advanced Developer Guide.

- Per i dettagli sull'API, consulta UpdateWebAcl AWS CLICommand Reference.

`update-xss-match-set`

Il seguente esempio di codice mostra come utilizzareupdate-xss-match-set.

AWS CLI

Per aggiornare un XSSMatch set

Il update-xss-match-set comando seguente elimina un XssMatchTuple oggetto (filtri) in un XssMatchSet:

```
aws waf update-xss-match-set --xss-match-set-id a123fae4-b567-8e90-1234-5ab67ac8ca90
 --change-token 12cs345-67cd-890b-1cd2-c3a4567d89f1 --updates
 Action="DELETE",XssMatchTuple={FieldToMatch={Type="QUERY_STRING"},TextTransformation="URL_D
```

Per ulteriori informazioni, consulta Utilizzo di condizioni di corrispondenza Cross-site scripting nella Guida per gli sviluppatori di AWS WAF.

- Per i dettagli sull'API, vedere UpdateXssMatchSetin AWS CLI Command Reference.

AWS WAF Classic regionale esempi utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with AWS WAF Classic regionale.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`associate-web-acl`

Il seguente esempio di codice mostra come utilizzare`associate-web-acl`.

AWS CLI

Come associare una lista di controllo degli accessi (ACL) a una risorsa

Il `associate-web-acl` comando seguente associa un ACL Web, specificato da, a una risorsa specificata da resource-arn. web-acl-id L'ARN della risorsa può fare riferimento a un Application Load Balancer o a un Gateway API:

```
aws waf-regional associate-web-acl \
    --web-acl-id a123fae4-b567-8e90-1234-5ab67ac8ca90 \
    --resource-arn 12cs345-67cd-890b-1cd2-c3a4567d89f1
```

Per ulteriori informazioni, vedere Working with Web nella WAF Developer Guide. ACLs AWS

- Per i dettagli sull'API, consulta AssociateWebAcl AWS CLICommand Reference.

`put-logging-configuration`

Il seguente esempio di codice mostra come utilizzare`put-logging-configuration`.

AWS CLI

Come creare una configurazione di registrazione di log per l'ARN WebACL con il flusso ARN di Kinesis Firehose specificato

L'`put-logging-configuration`esempio seguente mostra la configurazione di registrazione per WAF con ALB/APIGateway in Region. `us-east-1`

```
aws waf-regional put-logging-configuration \
    --logging-configuration ResourceArn=arn:aws:waf-
regional:us-east-1:123456789012:webacl/3bffd3ed-fa2e-445e-869f-
a6a7cf153fd3,LogDestinationConfigs=arn:aws:firehose:us-
east-1:123456789012:deliverystream/aws-waf-logs-firehose-stream,RedactedFields=[] \
    --region us-east-1
```

Output:

```
{
    "LoggingConfiguration": {
        "ResourceArn": "arn:aws:waf-regional:us-east-1:123456789012:webacl/3bffd3ed-
fa2e-445e-869f-a6a7cf153fd3",
        "LogDestinationConfigs": [
            "arn:aws:firehose:us-east-1:123456789012:deliverystream/aws-waf-logs-
firehose-stream"
        ]
    }
}
```

- Per i dettagli sull'API, vedere PutLoggingConfigurationin AWS CLI Command Reference.

`update-byte-match-set`

Il seguente esempio di codice mostra come utilizzare`update-byte-match-set`.

AWS CLI

Come aggiornare un set di corrispondenze di byte

Il comando `update-byte-match-set` seguente elimina un oggetto `ByteMatchTuple` (filtro) in un `ByteMatchSet`. Poiché il valore `updates` ha virgolette doppie incorporate, è necessario racchiudere il valore tra virgolette singole.

```
aws waf-regional update-byte-match-set \
    --byte-match-set-id a123fae4-b567-8e90-1234-5ab67ac8ca90 \
    --change-token 12cs345-67cd-890b-1cd2-c3a4567d89f1 \
    --updates
 'Action="DELETE",ByteMatchTuple={FieldToMatch={Type="HEADER",Data="referer"},TargetString="
```

Per ulteriori informazioni, consulta Utilizzo di condizioni di corrispondenza per stringa nella Guida per gli sviluppatori di AWS WAF.

- Per i dettagli sull'API, consulta UpdateByteMatchSet AWS CLICommand Reference.

`update-ip-set`

Il seguente esempio di codice mostra come utilizzare`update-ip-set`.

AWS CLI

Come creare un set di IP

Il `update-ip-set` comando seguente aggiorna un IPSet con un IPv4 indirizzo ed elimina un IPv6 indirizzo. Ottieni il valore per `change-token` eseguendo il comando `get-change-token`. Poiché il valore per gli aggiornamenti include virgolette doppie incorporate, è necessario racchiudere il valore tra virgolette singole.

```
aws waf update-ip-set \
    --ip-set-id a123fae4-b567-8e90-1234-5ab67ac8ca90 \
    --change-token 12cs345-67cd-890b-1cd2-c3a4567d89f1 \
    --updates
 'Action="INSERT",IPSetDescriptor={Type="IPV4",Value="12.34.56.78/16"},Action="DELETE",IPSet
```

In alternativa, è possibile utilizzare un file JSON per specificare l'input. Esempio:

```
aws waf-regional update-ip-set \
    --ip-set-id a123fae4-b567-8e90-1234-5ab67ac8ca90 \
    --change-token 12cs345-67cd-890b-1cd2-c3a4567d89f1  \
    --updates file://change.json
```

Contenuto di `change.json`

```
[
    {
        "Action": "INSERT",
        "IPSetDescriptor":
        {
            "Type": "IPV4",
            "Value": "12.34.56.78/16"
        }
    },
    {
        "Action": "DELETE",
        "IPSetDescriptor":
        {
            "Type": "IPV6",
            "Value": "1111:0000:0000:0000:0000:0000:0000:0111/128"
        }
    }
]
```

Per ulteriori informazioni, consulta Utilizzo di condizioni di corrispondenza IP nella Guida per gli sviluppatori di AWS WAF.

- Per i dettagli sull'API, vedere UpdateIpSetin AWS CLI Command Reference.

`update-rule`

Il seguente esempio di codice mostra come utilizzare`update-rule`.

AWS CLI

Come aggiornare una regola

Il comando `update-rule` seguente elimina un oggetto `Predicate` in una regola. Poiché il valore `updates` ha virgolette doppie incorporate, è necessario racchiudere l'intero valore tra virgolette singole.

```
aws waf-regional update-rule \
    --rule-id a123fae4-b567-8e90-1234-5ab67ac8ca90 \
    --change-token 12cs345-67cd-890b-1cd2-c3a4567d89f1 \
```

```
    --updates
'Action="DELETE",Predicate={Negated=false,Type="ByteMatch",DataId="MyByteMatchSetID"}'
```

Per ulteriori informazioni, consulta Utilizzo delle regole nella Guida per gli sviluppatori di AWS WAF.

- Per i dettagli sull'API, consulta UpdateRule AWS CLICommand Reference.

`update-size-constraint-set`

Il seguente esempio di codice mostra come utilizzare`update-size-constraint-set`.

AWS CLI

Come aggiornare un set di vincoli di dimensione

Il `update-size-constraint-set` comando seguente elimina un oggetto SizeConstraint `(filtri) in un set di vincoli di dimensione. Poiché il valore `updates` contiene virgolette doppie incorporate, è necessario racchiudere l'intero valore tra virgolette singole.

```
aws waf-regional update-size-constraint-set \
    --size-constraint-set-id a123fae4-b567-8e90-1234-5ab67ac8ca90 \
    --change-token 12cs345-67cd-890b-1cd2-c3a4567d89f1 \
    --updates
 'Action="DELETE",SizeConstraint={FieldToMatch={Type="QUERY_STRING"},TextTransformation="NON
```

Per ulteriori informazioni, consulta Utilizzo di condizioni per i vincoli di dimensioni nella Guida per gli sviluppatori di AWS WAF.

- Per i dettagli sull'API, vedere UpdateSizeConstraintSetin AWS CLI Command Reference.

`update-sql-injection-match-set`

Il seguente esempio di codice mostra come utilizzare`update-sql-injection-match-set`.

AWS CLI

Come aggiornare un set di corrispondenze dell'iniezione SQL

Il comando `update-sql-injection-match-set` seguente elimina un oggetto `SqlInjectionMatchTuple` (filtri) in un set di corrispondenze dell'iniezione SQL. Poiché il

valore `updates` contiene virgolette doppie incorporate, è necessario racchiudere l'intero valore tra virgolette singole.

aws waf-regional update-sql-injection-match -set -- sql-injection-match-set -id a123fae4-b567-8e90-1234-5ab67ac8ca90 --change-token 12cs345-67cd-890b-1cd2-c3a4567d89f1 --updates 'action="DELETE», = {{type="QUERY_STRING "}, ="URL_DECODE"} 'SqlInjectionMatchTupleFieldToMatchTextTransformation

Per ulteriori informazioni, consulta Utilizzo di condizioni di corrispondenza SQL injection nella Guida per gli sviluppatori di AWS WAF.

- Per i dettagli sull'API, consulta UpdateSqlInjectionMatchSet AWS CLICommand Reference.

`update-web-acl`

Il seguente esempio di codice mostra come utilizzare`update-web-acl`.

AWS CLI

Come aggiornare una lista di controllo degli accessi (ACL) web

Il comando `update-web-acl` seguente elimina un oggetto `ActivatedRule` in un WebACL. Poiché il valore `updates` contiene virgolette doppie incorporate, è necessario racchiudere l'intero valore tra virgolette singole.

```
aws waf-regional update-web-acl \
    --web-acl-id a123fae4-b567-8e90-1234-5ab67ac8ca90 \
    --change-token 12cs345-67cd-890b-1cd2-c3a4567d89f1 \
    --updates Action="DELETE",ActivatedRule='{Priority=1,RuleId="WAFRule-1-
Example",Action={Type="ALLOW"},Type="ALLOW"}'
```

Per ulteriori informazioni, vedere Working with Web ACLs nella AWS WAF Developer Guide.

- Per i dettagli sull'API, consulta UpdateWebAcl AWS CLICommand Reference.

`update-xss-match-set`

Il seguente esempio di codice mostra come utilizzare`update-xss-match-set`.

AWS CLI

Per aggiornare un XSSMatch set

Il comando `update-xss-match-set` seguente elimina un oggetto `XssMatchTuple` (filtri) in un `XssMatchSet`. Poiché il valore `updates` contiene virgolette doppie incorporate, è necessario racchiudere l'intero valore tra virgolette singole.

```
aws waf-regional update-xss-match-set \
    --xss-match-set-id a123fae4-b567-8e90-1234-5ab67ac8ca90 \
    --change-token 12cs345-67cd-890b-1cd2-c3a4567d89f1 \
    --updates
 'Action="DELETE",XssMatchTuple={FieldToMatch={Type="QUERY_STRING"},TextTransformation="URL_
```

Per ulteriori informazioni, consulta Utilizzo di condizioni di corrispondenza Cross-site scripting nella Guida per gli sviluppatori di AWS WAF.

- Per i dettagli sull'API, consulta UpdateXssMatchSet AWS CLICommand Reference.

AWS WAFV2 esempi utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with AWS WAFV2.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`associate-web-acl`

Il seguente esempio di codice mostra come utilizzare`associate-web-acl`.

AWS CLI

Per associare un ACL Web a una risorsa regionale AWS

L'esempio `associate-web-acl` seguente associa la lista di controllo degli accessi web specificata con un Application Load Balancer.

```
aws wafv2 associate-web-acl \
    --web-acl-arn arn:aws:wafv2:us-west-2:123456789012:regional/webacl/test-cli/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 \
    --resource-arn arn:aws:elasticloadbalancing:us-west-2:123456789012:loadbalancer/app/waf-cli-alb/1ea17125f8b25a2a \
    --region us-west-2
```

Questo comando non produce alcun output.

Per ulteriori informazioni, vedere Associare o dissociare un ACL Web con una AWS risorsa nella WAF AWS , Firewall AWS Manager, and AWS Shield Advanced Developer Guide.

- Per i dettagli sull'API, consulta Command Reference. AssociateWebAclAWS CLI

`check-capacity`

Il seguente esempio di codice mostra come utilizzare`check-capacity`.

AWS CLI

Come ottenere la capacità utilizzata da un set di regole

Il comando `check-capacity` seguente recupera i requisiti di capacità per un set di regole che include un'istruzione di regola basata sulla frequenza e un'istruzione di regola AND contenente regole annidate.

```
aws wafv2 check-capacity \
    --scope REGIONAL \
    --rules file://waf-rule-list.json \
    --region us-west-2
```

Contenuto del file://waf-rule-list.json:

```
[
    {
        "Name":"basic-rule",
        "Priority":0,
        "Statement":{
            "AndStatement":{
```

```
                "Statements":[
                    {
                        "ByteMatchStatement":{
                            "SearchString":"example.com",
                            "FieldToMatch":{
                                "SingleHeader":{
                                    "Name":"host"
                                }
                            },
                            "TextTransformations":[
                                {
                                    "Priority":0,
                                    "Type":"LOWERCASE"
                                }
                            ],
                            "PositionalConstraint":"EXACTLY"
                        }
                    },
                    {
                        "GeoMatchStatement":{
                            "CountryCodes":[
                                "US",
                                "IN"
                            ]
                        }
                    }
                ]
            }
        },
        "Action":{
            "Allow":{

            }
        },
        "VisibilityConfig":{
            "SampledRequestsEnabled":true,
            "CloudWatchMetricsEnabled":true,
            "MetricName":"basic-rule"
        }
    },
    {
        "Name":"rate-rule",
        "Priority":1,
        "Statement":{
```

```
            "RateBasedStatement":{
                "Limit":1000,
                "AggregateKeyType":"IP"
            }
        },
        "Action":{
            "Block":{

            }
        },
        "VisibilityConfig":{
            "SampledRequestsEnabled":true,
            "CloudWatchMetricsEnabled":true,
            "MetricName":"rate-rule"
        }
    }
]
```

Output:

```
{
    "Capacity":15
}
```

Per ulteriori informazioni, vedere AWS WAF Web ACL Capacity Units (WCU) nella WAF AWS, Firewall AWS Manager, and AWS Shield Advanced Developer Guide.

- Per i dettagli sull'API, consulta Command Reference. CheckCapacityAWS CLI

`create-ip-set`

Il seguente esempio di codice mostra come utilizzare`create-ip-set`.

AWS CLI

Per creare un set IP da utilizzare nel Web ACLs e nei gruppi di regole

Il comando `create-ip-set` seguente crea un set IP con una singola specifica relativa all'intervallo di indirizzi.

```
aws wafv2 create-ip-set \
    --name testip \
    --scope REGIONAL \
```

```
    --ip-address-version IPV4 \
    --addresses 198.51.100.0/16
```

Output:

```
{
    "Summary":{
        "ARN":"arn:aws:wafv2:us-west-2:123456789012:regional/ipset/testip/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
        "Description":"",
        "Name":"testip",
        "LockToken":"447e55ac-0000-0000-0000-86b67c17f8b5",
        "Id":"a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
    }
}
```

Per ulteriori informazioni, consulta IP Sets e Regex Pattern Sets nella AWS WAF, AWS Firewall Manager, and AWS Shield Advanced Developer Guide.

- Per i dettagli sulle API, consulta AWS CLI Command CreateIpSetReference.

`create-regex-pattern-set`

Il seguente esempio di codice mostra come utilizzare`create-regex-pattern-set`.

AWS CLI

Per creare un set di pattern regex da utilizzare nel Web ACLs e nei gruppi di regole

Il comando `create-regex-pattern-set` seguente crea un set di modelli di espressioni regolari con due modelli regex specificati.

```
aws wafv2 create-regex-pattern-set \
    --name regexPatterSet01 \
    --scope REGIONAL \
    --description 'Test web-acl' \
    --regular-expression-list '[{"RegexString": "/[0-9]*/"},{"RegexString": "/[a-
z]*/"}]'
```

Output:

```
{
```

```
    "Summary":{
        "ARN":"arn:aws:wafv2:us-west-2:123456789012:regional/regexpatternset/
regexPatterSet01/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
        "Description":"Test web-acl",
        "Name":"regexPatterSet01",
        "LockToken":"0bc01e21-03c9-4b98-9433-6229cbf1ef1c",
        "Id":"a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
    }
}
```

Per ulteriori informazioni, consulta IP Sets e Regex Pattern Sets nella AWS WAF, AWS Firewall Manager, and AWS Shield Advanced Developer Guide.

- Per i dettagli sulle API, consulta AWS CLI Command CreateRegexPatternSetReference.

`create-rule-group`

Il seguente esempio di codice mostra come utilizzare`create-rule-group`.

AWS CLI

Per creare un gruppo di regole personalizzato da utilizzare nel Web ACLs

Il comando `create-rule-group` seguente crea un gruppo di regole personalizzato per uso regionale. Le istruzioni delle regole per il gruppo sono fornite in un file in formato JSON.

```
aws wafv2 create-rule-group \
    --name "TestRuleGroup" \
    --scope REGIONAL \
    --capacity 250 \
    --rules file://waf-rule.json \
    --visibility-
config SampledRequestsEnabled=true,CloudWatchMetricsEnabled=true,MetricName=TestRuleGroupMet
 \
    --region us-west-2
```

Contenuto di file://waf-rule.json:

```
[
    {
        "Name":"basic-rule",
        "Priority":0,
        "Statement":{
```

```
            "AndStatement":{
                "Statements":[
                    {
                        "ByteMatchStatement":{
                            "SearchString":"example.com",
                            "FieldToMatch":{
                                "SingleHeader":{
                                    "Name":"host"
                                }
                            },
                            "TextTransformations":[
                                {
                                    "Priority":0,
                                    "Type":"LOWERCASE"
                                }
                            ],
                            "PositionalConstraint":"EXACTLY"
                        }
                    },
                    {
                        "GeoMatchStatement":{
                            "CountryCodes":[
                                "US",
                                "IN"
                            ]
                        }
                    }
                ]
            }
        },
        "Action":{
            "Allow":{

            }
        },
        "VisibilityConfig":{
            "SampledRequestsEnabled":true,
            "CloudWatchMetricsEnabled":true,
            "MetricName":"basic-rule"
        }
    }
]
```

Output:

```
{
    "Summary":{
        "ARN":"arn:aws:wafv2:us-west-2:123456789012:regional/rulegroup/
TestRuleGroup/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
        "Description":"",
        "Name":"TestRuleGroup",
        "LockToken":"7b3bcec2-374e-4c5a-b2b9-563bf47249f0",
        "Id":"a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
    }
}
```

Per ulteriori informazioni, consulta Managing Your Own Rule Groups nella AWS WAF, AWS Firewall Manager e AWS Shield Advanced Developer Guide.

- Per i dettagli sull'API, consulta CreateRuleGroup AWS CLICommand Reference.

`create-web-acl`

Il seguente esempio di codice mostra come utilizzare`create-web-acl`.

AWS CLI

Come creare una lista di controllo degli accessi (ACL) web

Il comando `create-web-acl` seguente crea una lista di controllo degli accessi (ACL) web per uso regionale. Le istruzioni delle regole per la lista di controllo degli accessi web sono fornite in un file in formato JSON.

```
aws wafv2 create-web-acl \
    --name TestWebAcl \
    --scope REGIONAL \
    --default-action Allow={} \
    --visibility-
config SampledRequestsEnabled=true,CloudWatchMetricsEnabled=true,MetricName=TestWebAclMetric
 \
    --rules file://waf-rule.json \
    --region us-west-2
```

Contenuto di file://waf-rule.json:

```
[
    {
        "Name":"basic-rule",
        "Priority":0,
        "Statement":{
            "AndStatement":{
                "Statements":[
                    {
                        "ByteMatchStatement":{
                            "SearchString":"example.com",
                            "FieldToMatch":{
                                "SingleHeader":{
                                    "Name":"host"
                                }
                            },
                            "TextTransformations":[
                                {
                                    "Priority":0,
                                    "Type":"LOWERCASE"
                                }
                            ],
                            "PositionalConstraint":"EXACTLY"
                        }
                    },
                    {
                        "GeoMatchStatement":{
                            "CountryCodes":[
                                "US",
                                "IN"
                            ]
                        }
                    }
                ]
            }
        },
        "Action":{
            "Allow":{

            }
        },
        "VisibilityConfig":{
            "SampledRequestsEnabled":true,
            "CloudWatchMetricsEnabled":true,
```

```
            "MetricName":"basic-rule"
        }
    }
]
```

Output:

```
{
    "Summary":{
        "ARN":"arn:aws:wafv2:us-west-2:123456789012:regional/webacl/TestWebAcl/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
        "Description":"",
        "Name":"TestWebAcl",
        "LockToken":"2294b3a1-eb60-4aa0-a86f-a3ae04329de9",
        "Id":"a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
    }
}
```

Per ulteriori informazioni, vedere Managing and Using a Web Access Control List (Web ACL) nella AWS WAF, AWS Firewall Manager, and AWS Shield Advanced Developer Guide.

- Per i dettagli sull'API, consulta AWS CLI Command CreateWebAclReference.

`delete-ip-set`

Il seguente esempio di codice mostra come utilizzare`delete-ip-set`.

AWS CLI

Come eliminare un set di IP

L'esempio `delete-ip-set` seguente elimina il set IP specificato. Questa chiamata richiede un ID, che è possibile ottenere dalla chiamata `list-ip-sets`, e un token di blocco, ottenibile dalle chiamate `list-ip-sets` e `get-ip-set`.

```
aws wafv2 delete-ip-set \
    --name test1 \
    --scope REGIONAL \
    --id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 \
    --lock-token 46851772-db6f-459d-9385-49428812e357
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta IP Sets e Regex Pattern Sets nella AWS WAF, AWS Firewall Manager, and AWS Shield Advanced Developer Guide.

- Per i dettagli sulle API, consulta AWS CLI Command DeleteIpSetReference.

`delete-logging-configuration`

Il seguente esempio di codice mostra come utilizzare`delete-logging-configuration`.

AWS CLI

Come disabilitare la registrazione per una lista di controllo degli accessi (ACL) web

Il comando `delete-logging-configuration` seguente rimuove eventuali configurazioni di registrazione di log dalla lista di controllo degli accessi web specificata.

```
aws wafv2 delete-logging-configuration \
    --resource-arn arn:aws:wafv2:us-west-2:123456789012:regional/webacl/test/
a1b2c3d4-5678-90ab-cdef-EXAMPLE22222
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Logging Web ACL Traffic Information nella AWS WAF, AWS Firewall Manager e AWS Shield Advanced Developer Guide.

- Per i dettagli sull'API, consulta Command DeleteLoggingConfigurationReference AWS CLI .

`delete-regex-pattern-set`

Il seguente esempio di codice mostra come utilizzare`delete-regex-pattern-set`.

AWS CLI

Come eliminare un set di modelli di espressioni regolari

Il comando `delete-regex-pattern-set` seguente aggiorna le impostazioni per il set di modelli di espressioni regolari specificato. Questa chiamata richiede un ID, che è possibile ottenere dalla chiamata `list-regex-pattern-sets`, e un token di blocco, ottenibile dalla chiamata `list-regex-pattern-sets` o dalla chiamata `get-regex-pattern-set`.

```
aws wafv2 delete-regex-pattern-set \
```

```
    --name regexPatterSet01 \
    --scope REGIONAL \
    --id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 \
    --lock-token 0bc01e21-03c9-4b98-9433-6229cbf1ef1c
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta IP Sets e Regex Pattern Sets nella AWS WAF, AWS Firewall Manager, and AWS Shield Advanced Developer Guide.

- Per i dettagli sulle API, consulta AWS CLI Command DeleteRegexPatternSetReference.

`delete-rule-group`

Il seguente esempio di codice mostra come utilizzare`delete-rule-group`.

AWS CLI

Come eliminare un gruppo di regole personalizzato

Il comando `delete-rule-group` seguente elimina il gruppo di regole personalizzato specificato. Questa chiamata richiede un ID, che è possibile ottenere dalla chiamata `list-rule-groups`, e un token di blocco, ottenibile dalla chiamata `list-rule-groups` o dalla chiamata `get-rule-group`.

```
aws wafv2 delete-rule-group \
    --name TestRuleGroup \
    --scope REGIONAL \
    --id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 \
    --lock-token 7b3bcec2-0000-0000-0000-563bf47249f0
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Managing Your Own Rule Groups nella AWS WAF, AWS Firewall Manager e AWS Shield Advanced Developer Guide.

- Per i dettagli sull'API, consulta DeleteRuleGroup AWS CLICommand Reference.

`delete-web-acl`

Il seguente esempio di codice mostra come utilizzare`delete-web-acl`.

AWS CLI

Come eliminare una lista di controllo degli accessi web

Il comando `delete-web-acl` seguente elimina la lista di controllo degli accessi web specificata dall'account. Una lista di controllo degli accessi web può essere eliminata solo se non è associata a una risorsa. Questa chiamata richiede un ID, che è possibile ottenere dalla chiamata `list-web-acls`, e un token di blocco, ottenibile dalla chiamata `list-web-acls` o dalla chiamata `get-web-acl`.

```
aws wafv2 delete-web-acl \
    --name test \
    --scope REGIONAL \
    --id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 \
    --lock-token ebab4ed2-155e-4c9a-9efb-e4c45665b1f5
```

Questo comando non produce alcun output.

Per ulteriori informazioni, vedere Managing and Using a Web Access Control List (Web ACL) nella AWS WAF, AWS Firewall Manager, and AWS Shield Advanced Developer Guide.

- Per i dettagli sull'API, consulta AWS CLI Command DeleteWebAclReference.

`describe-managed-rule-group`

Il seguente esempio di codice mostra come utilizzare`describe-managed-rule-group`.

AWS CLI

Come recuperare la descrizione per un gruppo di regole gestito

Di seguito viene `describe-managed-rule-group` recuperata la descrizione di un gruppo di regole AWS gestito.

```
aws wafv2 describe-managed-rule-group \
    --vendor-name AWS \
    --name AWSManagedRulesCommonRuleSet \
    --scope REGIONAL
```

Output:

```
{
```

```
    "Capacity": 700,
    "Rules": [
        {
            "Name": "NoUserAgent_HEADER",
            "Action": {
                "Block": {}
            }
        },
        {
            "Name": "UserAgent_BadBots_HEADER",
            "Action": {
                "Block": {}
            }
        },
        {
            "Name": "SizeRestrictions_QUERYSTRING",
            "Action": {
                "Block": {}
            }
        },
        {
            "Name": "SizeRestrictions_Cookie_HEADER",
            "Action": {
                "Block": {}
            }
        },
        {
            "Name": "SizeRestrictions_BODY",
            "Action": {
                "Block": {}
            }
        },
        {
            "Name": "SizeRestrictions_URIPATH",
            "Action": {
                "Block": {}
            }
        },
        {
            "Name": "EC2MetaDataSSRF_BODY",
            "Action": {
                "Block": {}
            }
        },
```

```
        {
            "Name": "EC2MetaDataSSRF_COOKIE",
            "Action": {
                "Block": {}
            }
        },
        {
            "Name": "EC2MetaDataSSRF_URIPATH",
            "Action": {
                "Block": {}
            }
        },
        {
            "Name": "EC2MetaDataSSRF_QUERYARGUMENTS",
            "Action": {
                "Block": {}
            }
        },
        {
            "Name": "GenericLFI_QUERYARGUMENTS",
            "Action": {
                "Block": {}
            }
        },
        {
            }
            "Name": "GenericLFI_URIPATH",
            "Action": {
                "Block": {}
            }
        },
        {
            "Name": "GenericLFI_BODY",
            "Action": {
                "Block": {}
            }
        },
        {
            "Name": "RestrictedExtensions_URIPATH",
            "Action": {
                "Block": {}
            }
        },
        {
```

```
            "Name": "RestrictedExtensions_QUERYARGUMENTS",
            "Action": {
                "Block": {}
            }
        },
        {
            "Name": "GenericRFI_QUERYARGUMENTS",
            "Action": {
                "Block": {}
            }
        },
        {
            "Name": "GenericRFI_BODY",
            "Action": {
                "Block": {}
            }
        },
        {
            "Name": "GenericRFI_URIPATH",
            "Action": {
                "Block": {}
            }
        },
        {
            "Name": "CrossSiteScripting_COOKIE",
            "Action": {
                "Block": {}
            }
        },
        {
            "Name": "CrossSiteScripting_QUERYARGUMENTS",
            "Action": {
                "Block": {}
            }
        },
        {
            "Name": "CrossSiteScripting_BODY",
            "Action": {
                "Block": {}
            }
        },
        {
            "Name": "CrossSiteScripting_URIPATH",
            "Action": {
```

```
                "Block": {}
            }
        }
    ]
}
```

Per ulteriori informazioni, consulta Managed Rule Groups nella AWS WAF, AWS Firewall Manager and AWS Shield Advanced Developer Guide.

- Per i dettagli sull'API, consulta DescribeManagedRuleGroup AWS CLICommand Reference.

`disassociate-web-acl`

Il seguente esempio di codice mostra come utilizzare`disassociate-web-acl`.

AWS CLI

Per dissociare un ACL Web da una risorsa regionale AWS

L'esempio `disassociate-web-acl` seguente rimuove qualsiasi associazione tra una lista di controllo degli accessi web esistente e l'Application Load Balancer specificato.

```
aws wafv2 disassociate-web-acl \
    --resource-arn arn:aws:elasticloadbalancing:us-west-2:123456789012:loadbalancer/app/waf-cli-alb/1ea17125f8b25a2a \
    --region us-west-2
```

Questo comando non produce alcun output.

Per ulteriori informazioni, vedere Associare o dissociare un ACL Web con una AWS risorsa nella WAF AWS , Firewall AWS Manager, and AWS Shield Advanced Developer Guide.

- Per i dettagli sull'API, consulta Command Reference. DisassociateWebAclAWS CLI

`get-ip-set`

Il seguente esempio di codice mostra come utilizzare`get-ip-set`.

AWS CLI

Come recuperare un set IP specifico

Il comando `get-ip-set` seguente recupera il set IP in base al nome, all'ambito e all'ID specificati. È possibile ottenere l'ID per un set IP dai comandi `create-ip-set` e `list-ip-sets`.

```
aws wafv2 get-ip-set \
    --name testip \
    --scope REGIONAL \
    --id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Output:

```
{
    "IPSet":{
        "Description":"",
        "Name":"testip",
        "IPAddressVersion":"IPV4",
        "Id":"a1b2c3d4-5678-90ab-cdef-EXAMPLE1111",
        "ARN":"arn:aws:wafv2:us-west-2:123456789012:regional/ipset/testip/
a1b2c3d4-5678-90ab-cdef-EXAMPLE1111",
        "Addresses":[
            "192.0.2.0/16"
        ]
    },
    "LockToken":"447e55ac-2396-4c6d-b9f9-86b67c17f8b5"
}
```

Per ulteriori informazioni, consulta IP Sets e Regex Pattern Sets nella AWS WAF, AWS Firewall Manager, and AWS Shield Advanced Developer Guide.

- Per i dettagli sulle API, consulta AWS CLI Command GetIpSetReference.

`get-logging-configuration`

Il seguente esempio di codice mostra come utilizzare`get-logging-configuration`.

AWS CLI

Come recuperare le configurazioni di registrazione di log per una lista di controllo degli accessi web

Il comando `get-logging-configuration` seguente recupera la configurazione di registrazione di log per la lista di controllo degli accessi web specificata.

```
aws wafv2 get-logging-configuration \
    --resource-arn arn:aws:wafv2:us-west-2:123456789012:regional/webacl/test/
a1b2c3d4-5678-90ab-cdef-EXAMPLE22222 \
    --region us-west-2
```

Output:

```
{
    "LoggingConfiguration":{
        "ResourceArn":"arn:aws:wafv2:us-west-2:123456789012:regional/webacl/test/
a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
        "RedactedFields":[
            {
                "Method":{

                }
            }
        ],
        "LogDestinationConfigs":[
            "arn:aws:firehose:us-west-2:123456789012:deliverystream/aws-waf-logs-
custom-transformation"
        ]
    }
}
```

Per ulteriori informazioni, consulta Logging Web ACL Traffic Information nella AWS WAF, AWS Firewall Manager e AWS Shield Advanced Developer Guide.

- Per i dettagli sull'API, consulta Command GetLoggingConfigurationReference AWS CLI .

`get-rate-based-statement-managed-keys`

Il seguente esempio di codice mostra come utilizzare`get-rate-based-statement-managed-keys`.

AWS CLI

Come recuperare un elenco di indirizzi IP bloccati da una regola basata sulla frequenza

Il comando `get-rate-based-statement-managed-keys` seguente recupera gli indirizzi IP attualmente bloccati da una regola basata sulla frequenza utilizzata per un'applicazione regionale.

```
aws wafv2 get-rate-based-statement-managed-keys \
    --scope REGIONAL \
    --web-acl-name testwebacl2 \
    --web-acl-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 \
    --rule-name ratebasedtest
```

Output:

```
{
    "ManagedKeysIPV4":{
        "IPAddressVersion":"IPV4",
        "Addresses":[
            "198.51.100.0/32"
        ]
    },
    "ManagedKeysIPV6":{
        "IPAddressVersion":"IPV6",
        "Addresses":[

        ]
    }
}
```

Per ulteriori informazioni, consulta Rate-Based Rule Statement nella AWS WAF, AWS Firewall Manager, and AWS Shield Advanced Developer Guide.

- Per i dettagli sull'API, consulta Command GetRateBasedStatementManagedKeysReference AWS CLI .

`get-regex-pattern-set`

Il seguente esempio di codice mostra come utilizzare`get-regex-pattern-set`.

AWS CLI

Come recuperare un set di modelli di espressioni regolari specifico

Il comando `get-regex-pattern-set` seguente recupera il set di modelli di espressioni regolari in base al nome, all'ambito, alla Regione e all'ID specificati. È possibile ottenere l'ID per un set di modelli di espressioni regolari tramite i comandi `create-regex-pattern-set` e `list-regex-pattern-sets`.

```
aws wafv2 get-regex-pattern-set \
    --name regexPatterSet01 \
    --scope REGIONAL \
    --id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 \
    --region us-west-2
```

Output:

```
{
    "RegexPatternSet":{
        "Description":"Test web-acl",
        "RegularExpressionList":[
            {
                "RegexString":"/[0-9]*/"
            },
            {
                "RegexString":"/[a-z]*/"
            }
        ],
        "Name":"regexPatterSet01",
        "ARN":"arn:aws:wafv2:us-west-2:123456789012:regional/regexpatternset/
regexPatterSet01/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
        "Id":"a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
    },
    "LockToken":"c8abf33f-b6fc-46ae-846e-42f994d57b29"
}
```

Per ulteriori informazioni, consulta IP Sets e Regex Pattern Sets nella AWS WAF, AWS Firewall Manager, and AWS Shield Advanced Developer Guide.

- Per i dettagli sulle API, consulta AWS CLI Command GetRegexPatternSetReference.

`get-rule-group`

Il seguente esempio di codice mostra come utilizzare`get-rule-group`.

AWS CLI

Come recuperare un gruppo di regole personalizzato specifico

Il comando `get-rule-group` seguente recupera il gruppo di regole personalizzato in base al nome, all'ambito e all'ID specificati. È possibile ottenere l'ID per un gruppo di regole tramite i comandi `create-rule-group` e `list-rule-groups`.

```
aws wafv2 get-rule-group \
    --name ff \
    --scope REGIONAL \
    --id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Output:

```
{
    "RuleGroup":{
        "Capacity":1,
        "Description":"",
        "Rules":[
            {
                "Priority":0,
                "Action":{
                    "Block":{

                    }
                },
                "VisibilityConfig":{
                    "SampledRequestsEnabled":true,
                    "CloudWatchMetricsEnabled":true,
                    "MetricName":"jj"
                },
                "Name":"jj",
                "Statement":{
                    "SizeConstraintStatement":{
                        "ComparisonOperator":"LE",
                        "TextTransformations":[
                            {
                                "Priority":0,
                                "Type":"NONE"
                            }
                        ],
                        "FieldToMatch":{
                            "UriPath":{

                            }
                        },
```

```
                        "Size":7
                    }
                }
            }
        ],
        "VisibilityConfig":{
            "SampledRequestsEnabled":true,
            "CloudWatchMetricsEnabled":true,
            "MetricName":"ff"
        },
        "Id":"a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
        "ARN":"arn:aws:wafv2:us-west-2:123456789012:regional/rulegroup/ff/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
        "Name":"ff"
    },
    "LockToken":"485458c9-1830-4234-af31-ec4d52ced1b3"
}
```

Per ulteriori informazioni, consulta Managing Your Own Rule Groups nella AWS WAF, AWS Firewall Manager e AWS Shield Advanced Developer Guide.

- Per i dettagli sull'API, consulta GetRuleGroup AWS CLICommand Reference.

`get-sampled-requests`

Il seguente esempio di codice mostra come utilizzare`get-sampled-requests`.

AWS CLI

Come recuperare un campione di richieste web per una lista di controllo degli accessi web

Il comando `get-sampled-requests` seguente recupera le richieste web campione per la lista di controllo degli accessi web, la metrica della regola e l'intervallo di tempo specificati.

```
aws wafv2 get-sampled-requests \
    --web-acl-arn arn:aws:wafv2:us-west-2:123456789012:regional/webacl/test-cli/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 \
    --rule-metric-name AWS-AWSManagedRulesSQLiRuleSet \
    --scope=REGIONAL \
    --time-window StartTime=2020-02-12T20:00Z,EndTime=2020-02-12T21:10Z \
    --max-items 100
```

Output:

```
{
    "TimeWindow": {
    "EndTime": 1581541800.0,
    "StartTime": 1581537600.0
    },
    "SampledRequests": [
        {
            "Action": "BLOCK",
            "Timestamp": 1581541799.564,
            "RuleNameWithinRuleGroup": "AWS#AWSManagedRulesSQLiRuleSet#SQLi_BODY",
            "Request": {
                "Country": "US",
                "URI": "/",
                "Headers": [
                    {
                        "Name": "Host",
                        "Value": "alb-test-1EXAMPLE1.us-east-1.elb.amazonaws.com"
                    },
                    {
                        "Name": "Content-Length",
                        "Value": "7456"
                    },
                    {
                        "Name": "User-Agent",
                        "Value": "curl/7.53.1"
                    },
                    {
                        "Name": "Accept",
                        "Value": "/"
                    },
                    {
                        "Name": "Content-Type",
                        "Value": "application/x-www-form-urlencoded"
                    }
                ],
                "ClientIP": "198.51.100.08",
                "Method": "POST",
                "HTTPVersion": "HTTP/1.1"
            },
            "Weight": 1
        },
        {
            "Action": "BLOCK",
```

```
            "Timestamp": 1581541799.988,
            "RuleNameWithinRuleGroup": "AWS#AWSManagedRulesSQLiRuleSet#SQLi_BODY",
            "Request": {
                "Country": "US",
                "URI": "/",
                "Headers": [
                    {
                        "Name": "Host",
                        "Value": "alb-test-1EXAMPLE1.us-east-1.elb.amazonaws.com"
                    },
                    {
                        "Name": "Content-Length",
                        "Value": "7456"
                    },
                    {
                        "Name": "User-Agent",
                        "Value": "curl/7.53.1"
                    },
                    {
                        "Name": "Accept",
                        "Value": "/"
                    },
                    {
                        "Name": "Content-Type",
                        "Value": "application/x-www-form-urlencoded"
                    }
                ],
                "ClientIP": "198.51.100.08",
                "Method": "POST",
                "HTTPVersion": "HTTP/1.1"
            },
            "Weight": 3
        },
        {
            "Action": "BLOCK",
            "Timestamp": 1581541799.846,
            "RuleNameWithinRuleGroup": "AWS#AWSManagedRulesSQLiRuleSet#SQLi_BODY",
            "Request": {
                "Country": "US",
                "URI": "/",
                "Headers": [
                    {
                        "Name": "Host",
                        "Value": "alb-test-1EXAMPLE1.us-east-1.elb.amazonaws.com"
```

```
                    },
                    {
                        "Name": "Content-Length",
                        "Value": "7456"
                    },
                    {
                        "Name": "User-Agent",
                        "Value": "curl/7.53.1"
                    },
                    {
                        "Name": "Accept",
                        "Value": "/"
                    },
                    {
                        "Name": "Content-Type",
                        "Value": "application/x-www-form-urlencoded"
                    }
                ],
                "ClientIP": "198.51.100.08",
                "Method": "POST",
                "HTTPVersion": "HTTP/1.1"
            },
            "Weight": 1
        },
        {
            "Action": "BLOCK",
            "Timestamp": 1581541799.4,
            "RuleNameWithinRuleGroup": "AWS#AWSManagedRulesSQLiRuleSet#SQLi_BODY",
            "Request": {
                "Country": "US",
                "URI": "/",
                "Headers": [
                    {
                        "Name": "Host",
                        "Value": "alb-test-1EXAMPLE1.us-east-1.elb.amazonaws.com"
                    },
                    {
                        "Name": "Content-Length",
                        "Value": "7456"
                    },
                    {
                        "Name": "User-Agent",
                        "Value": "curl/7.53.1"
                    },
```

```
                    {
                        "Name": "Accept",
                        "Value": "/"
                    },
                    {
                        "Name": "Content-Type",
                        "Value": "application/x-www-form-urlencoded"
                    }
                ],
                "ClientIP": "198.51.100.08",
                "Method": "POST",
                "HTTPVersion": "HTTP/1.1"
            },
            "Weight": 1
        }
    ],
    "PopulationSize": 4
}
```

Per ulteriori informazioni, vedere Visualizzazione di un esempio di richieste Web nella AWS WAF, AWS Firewall Manager and AWS Shield Advanced Developer Guide.

- Per i dettagli sulle API, consulta GetSampledRequests AWS CLICommand Reference.

`get-web-acl-for-resource`

Il seguente esempio di codice mostra come utilizzare`get-web-acl-for-resource`.

AWS CLI

Per recuperare l'ACL web associato a una risorsa AWS

Il comando `get-web-acl-for-resource` seguente recupera il file JSON per la lista di controllo degli accessi web associata alla risorsa specificata.

```
aws wafv2 get-web-acl-for-resource \
    --resource-arn arn:aws:elasticloadbalancing:us-west-2:123456789012:loadbalancer/
app/waf-cli-alb/1ea17125f8b25a2a
```

Output:

```
{
    "WebACL":{
```

```
        "Capacity":3,
        "Description":"",
        "Rules":[
            {
                "Priority":1,
                "Action":{
                    "Block":{

                    }
                },
                "VisibilityConfig":{
                    "SampledRequestsEnabled":true,
                    "CloudWatchMetricsEnabled":true,
                    "MetricName":"testrule01"
                },
                "Name":"testrule01",
                "Statement":{
                    "AndStatement":{
                        "Statements":[
                            {
                                "ByteMatchStatement":{
                                    "PositionalConstraint":"EXACTLY",
                                    "TextTransformations":[
                                        {
                                            "Priority":0,
                                            "Type":"NONE"
                                        }
                                    ],
                                    "SearchString":"dGVzdHN0cmluZw==",
                                    "FieldToMatch":{
                                        "UriPath":{

                                        }
                                    }
                                }
                            },
                            {
                                "SizeConstraintStatement":{
                                    "ComparisonOperator":"EQ",
                                    "TextTransformations":[
                                        {
                                            "Priority":0,
                                            "Type":"NONE"
                                        }
```

```
                                    ],
                                    "FieldToMatch":{
                                        "QueryString":{

                                       }
                                    },
                                    "Size":0
                                }
                            }
                        ]
                    }
                }
            }
        ],
        "VisibilityConfig":{
            "SampledRequestsEnabled":true,
            "CloudWatchMetricsEnabled":true,
            "MetricName":"test01"
        },
        "DefaultAction":{
            "Allow":{

            }
        },
        "Id":"9a1b2c3d4-5678-90ab-cdef-EXAMPLE11111  ",
        "ARN":"arn:aws:wafv2:us-west-2:123456789012:regional/webacl/test01/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111  ",
        "Name":"test01"
    }
}
```

Per ulteriori informazioni, vedere Associare o dissociare un ACL Web con una AWS risorsa nella WAF AWS , Firewall AWS Manager, and AWS Shield Advanced Developer Guide.

- Per i dettagli sull'API, consulta Command Reference. GetWebAclForResourceAWS CLI

`get-web-acl`

Il seguente esempio di codice mostra come utilizzare`get-web-acl`.

AWS CLI

Come recuperare una lista di controllo degli accessi web

Il comando `get-web-acl` seguente recupera la lista di controllo degli accessi web in base al nome, all'ambito e all'ID specificati. È possibile ottenere l'ID per una lista di controllo degli accessi web tramite i comandi `create-web-acl` e `list-web-acls`.

```
aws wafv2 get-web-acl \
    --name test01 \
    --scope REGIONAL \
    --id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Output:

```
{
    "WebACL":{
        "Capacity":3,
        "Description":"",
        "Rules":[
            {
                "Priority":1,
                "Action":{
                    "Block":{

                    }
                },
                "VisibilityConfig":{
                   "SampledRequestsEnabled":true,
                    "CloudWatchMetricsEnabled":true,
                    "MetricName":"testrule01"
                },
                "Name":"testrule01",
                "Statement":{
                    "AndStatement":{
                        "Statements":[
                            {
                                "ByteMatchStatement":{
                                    "PositionalConstraint":"EXACTLY",
                                    "TextTransformations":[
                                        {
                                            "Priority":0,
                                            "Type":"NONE"
                                        }
                                    ],
                                    "SearchString":"dGVzdHN0cmluZw==",
                                    "FieldToMatch":{
```

```
                                    "UriPath":{

                                    }
                                }
                            }
                        },
                        {
                            "SizeConstraintStatement":{
                                "ComparisonOperator":"EQ",
                                "TextTransformations":[
                                    {
                                        "Priority":0,
                                        "Type":"NONE"
                                    }
                                ],
                                "FieldToMatch":{
                                    "QueryString":{

                                    }
                                },
                                "Size":0
                            }
                        }
                    ]
                }
            }
        }
    ],
    "VisibilityConfig":{
        "SampledRequestsEnabled":true,
        "CloudWatchMetricsEnabled":true,
        "MetricName":"test01"
    },
    "DefaultAction":{
        "Allow":{

        }
    },
    "Id":"a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "ARN":"arn:aws:wafv2:us-west-2:123456789012:regional/webacl/test01/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "Name":"test01"
},
"LockToken":"e3db7e2c-d58b-4ee6-8346-6aec5511c6fb"
```

```
}
```

Per ulteriori informazioni, vedere Managing and Using a Web Access Control List (Web ACL) nella AWS WAF, AWS Firewall Manager, and AWS Shield Advanced Developer Guide.

- Per i dettagli sull'API, consulta AWS CLI Command GetWebAclReference.

`list-available-managed-rule-groups`

Il seguente esempio di codice mostra come utilizzare`list-available-managed-rule-groups`.

AWS CLI

Come recuperare i gruppi di regole gestiti

Quanto segue `list-available-managed-rule-groups` restituisce l'elenco di tutti i gruppi di regole gestiti attualmente disponibili per l'uso nel Web ACLs.

```
aws wafv2 list-available-managed-rule-groups \
    --scope REGIONAL
```

Output:

```
{
   "ManagedRuleGroups": [
       {
           "VendorName": "AWS",
           "Name": "AWSManagedRulesCommonRuleSet",
           "Description": "Contains rules that are generally applicable to web
applications. This provides protection against exploitation of a wide range of
vulnerabilities, including those described in OWASP publications and common Common
Vulnerabilities and Exposures (CVE)."
       },
       {
           "VendorName": "AWS",
           "Name": "AWSManagedRulesAdminProtectionRuleSet",
           "Description": "Contains rules that allow you to block external access
to exposed admin pages. This may be useful if you are running third-party software
or would like to reduce the risk of a malicious actor gaining administrative access
to your application."
       },
       {
           "VendorName": "AWS",
```

```
            "Name": "AWSManagedRulesKnownBadInputsRuleSet",
            "Description": "Contains rules that allow you to block request patterns
that are known to be invalid and are associated with exploitation or discovery of
vulnerabilities. This can help reduce the risk of a malicious actor discovering a
vulnerable application."
        },
        {
            "VendorName": "AWS",
            "Name": "AWSManagedRulesSQLiRuleSet",
            "Description": "Contains rules that allow you to block request patterns
associated with exploitation of SQL databases, like SQL injection attacks. This can
help prevent remote injection of unauthorized queries."
        },
        {
            "VendorName": "AWS",
            "Name": "AWSManagedRulesLinuxRuleSet",
            "Description": "Contains rules that block request patterns associated
with exploitation of vulnerabilities specific to Linux, including LFI attacks. This
can help prevent attacks that expose file contents or execute code for which the
attacker should not have had access."
        },
        {
            "VendorName": "AWS",
            "Name": "AWSManagedRulesUnixRuleSet",
            "Description": "Contains rules that block request patterns associated
with exploiting vulnerabilities specific to POSIX/POSIX-like OS, including LFI
attacks. This can help prevent attacks that expose file contents or execute code
for which access should not been allowed."
        },
        {
            "VendorName": "AWS",
            "Name": "AWSManagedRulesWindowsRuleSet",
            "Description": "Contains rules that block request patterns associated
with exploiting vulnerabilities specific to Windows, (e.g., PowerShell commands).
This can help prevent exploits that allow attacker to run unauthorized commands or
execute malicious code."
        },
        {
            "VendorName": "AWS",
            "Name": "AWSManagedRulesPHPRuleSet",
            "Description": "Contains rules that block request patterns associated
with exploiting vulnerabilities specific to the use of the PHP, including injection
of unsafe PHP functions. This can help prevent exploits that allow an attacker to
remotely execute code or commands."
```

```
        },
        {
            "VendorName": "AWS",
            "Name": "AWSManagedRulesWordPressRuleSet",
            "Description": "The WordPress Applications group contains rules that
 block request patterns associated with the exploitation of vulnerabilities specific
 to WordPress sites."
        },
        {
            "VendorName": "AWS",
            "Name": "AWSManagedRulesAmazonIpReputationList",
            "Description": "This group contains rules that are based on Amazon
 threat intelligence. This is useful if you would like to block sources associated
 with bots or other threats."
        }
    ]
}
```

Per ulteriori informazioni, consulta Managed Rule Groups nella AWS WAF, AWS Firewall Manager and AWS Shield Advanced Developer Guide.

- Per i dettagli sull'API, consulta ListAvailableManagedRuleGroups AWS CLICommand Reference.

`list-ip-sets`

Il seguente esempio di codice mostra come utilizzare`list-ip-sets`.

AWS CLI

Come recuperare un elenco di set IP

Il comando `list-ip-sets` seguente recupera tutti i set IP dell'account che hanno un ambito regionale.

```
aws wafv2 list-ip-sets \
    --scope REGIONAL
```

Output:

```
{
```

```
    "IPSets":[
        {
            "ARN":"arn:aws:wafv2:us-west-2:123456789012:regional/ipset/testip/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
            "Description":"",
            "Name":"testip",
            "LockToken":"0674c84b-0304-47fe-8728-c6bff46af8fc",
            "Id":"a1b2c3d4-5678-90ab-cdef-EXAMPLE11111  "
        }
    ],
    "NextMarker":"testip"
}
```

Per ulteriori informazioni, consulta IP Sets e Regex Pattern Sets nella AWS WAF, AWS Firewall Manager, and AWS Shield Advanced Developer Guide.

- Per i dettagli sulle API, consulta AWS CLI Command ListIpSetsReference.

`list-logging-configurations`

Il seguente esempio di codice mostra come utilizzare`list-logging-configurations`.

AWS CLI

Come recuperare un elenco di tutte le configurazioni di registrazione di log per una Regione

Di seguito `list-logging-configurations` vengono recuperate tutte le configurazioni di registrazione per il Web ACLs destinate all'uso regionale nella regione. `us-west-2`

```
aws wafv2 list-logging-configurations \
    --scope REGIONAL \
    --region us-west-2
```

Output:

```
{
    "LoggingConfigurations":[
        {
            "ResourceArn":"arn:aws:wafv2:us-west-2:123456789012:regional/webacl/
test-2/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
            "RedactedFields":[
                {
```

```
                    "QueryString":{

                    }
                }
            ],
            "LogDestinationConfigs":[
                "arn:aws:firehose:us-west-2:123456789012:deliverystream/aws-waf-
logs-test"
            ]
        },
        {
            "ResourceArn":"arn:aws:wafv2:us-west-2:123456789012:regional/webacl/
test/a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
            "RedactedFields":[
                {
                    "Method":{

                    }
                }
            ],
            "LogDestinationConfigs":[
                "arn:aws:firehose:us-west-2:123456789012:deliverystream/aws-waf-
logs-custom-transformation"
            ]
        }
    ]
}
```

Per ulteriori informazioni, consulta Logging Web ACL Traffic Information nella AWS WAF, AWS Firewall Manager e AWS Shield Advanced Developer Guide.

- Per i dettagli sull'API, consulta Command ListLoggingConfigurationsReference AWS CLI .

`list-regex-pattern-sets`

Il seguente esempio di codice mostra come utilizzare`list-regex-pattern-sets`.

AWS CLI

Come recuperare un elenco di set di modelli di espressioni regolari

Il comando `list-regex-pattern-sets` seguente recupera tutti i set di modelli di espressioni regolari per l'account definiti nella Regione `us-west-2`.

```
aws wafv2 list-regex-pattern-sets \
--scope REGIONAL \
--region us-west-2
```

Output:

```
{
    "NextMarker":"regexPatterSet01",
    "RegexPatternSets":[
        {
            "ARN":"arn:aws:wafv2:us-west-2:123456789012:regional/regexpatternset/
regexPatterSet01/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
            "Description":"Test web-acl",
            "Name":"regexPatterSet01",
            "LockToken":"f17743f7-0000-0000-0000-19a8b93bfb01",
            "Id":"a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
        }
    ]
}
```

Per ulteriori informazioni, consulta IP Sets e Regex Pattern Sets nella AWS WAF, AWS Firewall Manager, and AWS Shield Advanced Developer Guide.

- Per i dettagli sulle API, consulta AWS CLI Command ListRegexPatternSetsReference.

`list-resources-for-web-acl`

Il seguente esempio di codice mostra come utilizzare`list-resources-for-web-acl`.

AWS CLI

Come recuperare le risorse associate a una lista di controllo degli accessi web

Il comando `list-resources-for-web-acl` seguente recupera le risorse REST API del gateway API attualmente associate alla lista di controllo degli accessi web specificata nella Regione `us-west-2`.

```
aws wafv2 list-resources-for-web-acl \
    --web-acl-arn arn:aws:wafv2:us-west-2:123456789012:regional/webacl/TestWebAcl/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 \
    --resource-type API_GATEWAY \
```

```
    --region us-west-2
```

Output:

```
{
    "ResourceArns": [
        "arn:aws:apigateway:us-west-2::/restapis/EXAMPLE111/stages/testing"
    ]
}
```

Per ulteriori informazioni, vedere Associare o dissociare un ACL Web con una AWS risorsa nella WAF AWS , Firewall AWS Manager, and AWS Shield Advanced Developer Guide.

- Per i dettagli sull'API, consulta Command Reference. ListResourcesForWebAclAWS CLI

`list-rule-groups`

Il seguente esempio di codice mostra come utilizzare`list-rule-groups`.

AWS CLI

Come recuperare un elenco di gruppi di regole personalizzati

Il comando `list-rule-groups` seguente recupera tutti i gruppi di regole personalizzati definiti per l'account in base all'ambito e alla Regione specificati.

```
aws wafv2 list-rule-groups \
    --scope REGIONAL \
    --region us-west-2
```

Output:

```
{
    "RuleGroups":[
        {
            "ARN":"arn:aws:wafv2:us-west-2:123456789012:regional/rulegroup/
TestRuleGroup/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
            "Description":"",
            "Name":"TestRuleGroup",
            "LockToken":"1eb5ec48-0000-0000-0000-ee9b906c541e",
            "Id":"a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
        },
```

```
        {
            "ARN":"arn:aws:wafv2:us-west-2:123456789012:regional/rulegroup/test/
a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
            "Description":"",
            "Name":"test",
            "LockToken":"b0f4583e-998b-4880-9069-3fbe45738b43",
            "Id":"a1b2c3d4-5678-90ab-cdef-EXAMPLE22222"
        }
    ],
    "NextMarker":"test"
}
```

Per ulteriori informazioni, consulta Managing Your Own Rule Groups nella AWS WAF, AWS Firewall Manager e AWS Shield Advanced Developer Guide.

- Per i dettagli sull'API, consulta ListRuleGroups AWS CLICommand Reference.

`list-tags-for-resource`

Il seguente esempio di codice mostra come utilizzare`list-tags-for-resource`.

AWS CLI

Per recuperare tutti i tag di una risorsa AWS WAF

Il comando `list-tags-for-resource` seguente recupera l'elenco di tutte le coppie di chiave-valore dei tag per la lista di controllo degli accessi web specificata.

```
aws wafv2 list-tags-for-resource \
    --resource-arn arn:aws:wafv2:us-west-2:123456789012:regional/webacl/testwebacl2/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Output:

```
{
    "NextMarker":"",
    "TagInfoForResource":{
        "ResourceARN":"arn:aws:wafv2:us-west-2:123456789012:regional/webacl/
testwebacl2/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
        "TagList":[

        ]
```

```
    }
}
```

Per ulteriori informazioni, consulta Getting Started with AWS WAF nella AWS WAF, AWS Firewall Manager and AWS Shield Advanced Developer Guide.

- Per i dettagli sull'API, consulta AWS CLI Command ListTagsForResourceReference.

`list-web-acls`

Il seguente esempio di codice mostra come utilizzare`list-web-acls`.

AWS CLI

Per recuperare il Web da un ACLs cannocchiale

Quanto segue `list-web-acls` recupera tutti ACLs i Web definiti per l'account per l'ambito specificato.

```
aws wafv2 list-web-acls \
    --scope REGIONAL
```

Output:

```
{
    "NextMarker":"Testt",
    "WebACLs":[
        {
            "ARN":"arn:aws:wafv2:us-west-2:123456789012:regional/webacl/Testt/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
            "Description":"sssss",
            "Name":"Testt",
            "LockToken":"7f36cb30-74ef-4cff-8cd4-a77e1aba1746",
            "Id":"a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
        }
    ]
}
```

Per ulteriori informazioni, vedere Managing and Using a Web Access Control List (Web ACL) nella AWS WAF, AWS Firewall Manager, and AWS Shield Advanced Developer Guide.

- Per i dettagli sull'API, consulta AWS CLI Command ListWebAclsReference.

`put-logging-configuration`

Il seguente esempio di codice mostra come utilizzare`put-logging-configuration`.

AWS CLI

Come aggiungere una configurazione di registrazione di log a una lista di controllo degli accessi web

Il comando `put-logging-configuration` seguente aggiunge la configurazione di registrazione di log di Amazon Kinesis Data Firehose `aws-waf-logs-custom-transformation` alla lista di controllo degli accessi web specificata, senza campi oscurati nei log.

```
aws wafv2 put-logging-configuration \
    --logging-configuration ResourceArn=arn:aws:wafv2:us-
west-2:123456789012:regional/webacl/test-cli/a1b2c3d4-5678-90ab-
cdef-EXAMPLE11111,LogDestinationConfigs=arn:aws:firehose:us-
west-2:123456789012:deliverystream/aws-waf-logs-custom-transformation \
            --region us-west-2
```

Output:

```
{
    "LoggingConfiguration":{
        "ResourceArn":"arn:aws:wafv2:us-west-2:123456789012:regional/webacl/test-
cli/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
        "LogDestinationConfigs":[
            "arn:aws:firehose:us-west-2:123456789012:deliverystream/aws-waf-logs-
custom-transformation"
        ]
    }
}
```

Per ulteriori informazioni, consulta Logging Web ACL Traffic Information nella AWS WAF, AWS Firewall Manager e AWS Shield Advanced Developer Guide.

- Per i dettagli sull'API, consulta Command PutLoggingConfigurationReference AWS CLI .

`tag-resource`

Il seguente esempio di codice mostra come utilizzare`tag-resource`.

AWS CLI

Per aggiungere tag a una risorsa AWS WAF

L'esempio `tag-resource` seguente aggiunge un tag con una chiave di `Name` e il valore impostato su `AWSWAF` alla lista di controllo degli accessi web specificata.

```
aws wafv2 tag-resource \
    --resource-arn arn:aws:wafv2:us-west-2:123456789012:regional/webacl/
apiGatewayWebAcl/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 \
    --tags Key=Name,Value=AWSWAF
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Getting Started with AWS WAF nella AWS WAF, AWS Firewall Manager and AWS Shield Advanced Developer Guide.

- Per i dettagli sull'API, consulta AWS CLI Command TagResourceReference.

`untag-resource`

Il seguente esempio di codice mostra come utilizzare`untag-resource`.

AWS CLI

Per rimuovere tag da una risorsa AWS WAF

L'esempio `untag-resource` seguente rimuove i tag con la chiave `KeyName` dalla lista di controllo degli accessi web specificata.

```
aws wafv2 untag-resource \
    --resource-arn arn:aws:wafv2:us-west-2:123456789012:regional/webacl/
apiGatewayWebAcl/a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 \
    --tag-keys "KeyName"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Getting Started with AWS WAF nella AWS WAF, AWS Firewall Manager and AWS Shield Advanced Developer Guide.

- Per i dettagli sull'API, consulta AWS CLI Command UntagResourceReference.

update-ip-set

Il seguente esempio di codice mostra come utilizzareupdate-ip-set.

AWS CLI

Come modificare le impostazioni di un set IP esistente

Il comando update-ip-set seguente aggiorna le impostazioni per il set IP specificato. Questa chiamata richiede un ID, che è possibile ottenere dalla chiamata list-ip-sets, e un token di blocco, ottenibile dalle chiamate list-ip-sets e get-ip-set. Questa chiamata restituisce anche un token di blocco che è possibile utilizzare per un aggiornamento successivo.

```
aws wafv2 update-ip-set \
    --name testip \
    --scope REGIONAL \
    --id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 \
    --addresses 198.51.100.0/16 \
    --lock-token 447e55ac-2396-4c6d-b9f9-86b67c17f8b5
```

Output:

```
{
    "NextLockToken": "0674c84b-0304-47fe-8728-c6bff46af8fc"
}
```

Per ulteriori informazioni, consulta IP Sets e Regex Pattern Sets nella AWS WAF, AWS Firewall Manager, and AWS Shield Advanced Developer Guide.

- Per i dettagli sulle API, consulta AWS CLI Command UpdateIpSetReference.

update-regex-pattern-set

Il seguente esempio di codice mostra come utilizzareupdate-regex-pattern-set.

AWS CLI

Come modificare le impostazioni per un set di modelli di espressioni regolari esistenti

Il comando update-regex-pattern-set seguente aggiorna le impostazioni per il set di modelli di espressioni regolari specificato. Questa chiamata richiede un ID, che è possibile

ottenere dalla chiamata `list-regex-pattern-sets`, e un token di blocco, ottenibile dalle chiamate `list-regex-pattern-sets` e `get-regex-pattern-set`. Questa chiamata restituisce anche un token di blocco che è possibile utilizzare per un aggiornamento successivo.

```
aws wafv2 update-regex-pattern-set \
    --name ExampleRegex \
    --scope REGIONAL \
    --id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 \
    --regular-expression-list RegexString="^.+$"  \
    --lock-token ed207e9c-82e9-4a77-aadd-81e6173ab7eb
```

Output:

```
{
    "NextLockToken": "12ebc73e-fa68-417d-a9b8-2bdd761a4fa5"
}
```

Per ulteriori informazioni, consulta IP Sets e Regex Pattern Sets nella AWS WAF, AWS Firewall Manager, and AWS Shield Advanced Developer Guide.

- Per i dettagli sulle API, consulta AWS CLI Command UpdateRegexPatternSetReference.

update-rule-group

Il seguente esempio di codice mostra come utilizzareu`pdate-rule-group`.

AWS CLI

Come aggiornare un gruppo di regole personalizzato

Il comando `update-rule-group` seguente modifica la configurazione di visibilità per un gruppo di regole personalizzato esistente. Questa chiamata richiede un ID, che è possibile ottenere dalla chiamata `list-rule-groups`, e un token di blocco, ottenibile dalle chiamate `list-rule-groups` e `get-rule-group`. Questa chiamata restituisce anche un token di blocco che è possibile utilizzare per un aggiornamento successivo.

```
aws wafv2 update-rule-group \
    --name TestRuleGroup \
    --scope REGIONAL \
    --id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 \
```

```
    --lock-token 7b3bcec2-0000-0000-0000-563bf47249f0 \
    --visibility-
config SampledRequestsEnabled=false,CloudWatchMetricsEnabled=false,MetricName=TestMetricsFor
 \
    --region us-west-2
```

Output:

```
{
    "NextLockToken": "1eb5ec48-0000-0000-0000-ee9b906c541e"
}
```

Per ulteriori informazioni, consulta Managing Your Own Rule Groups nella AWS WAF, AWS Firewall Manager e AWS Shield Advanced Developer Guide.

- Per i dettagli sull'API, consulta UpdateRuleGroup AWS CLICommand Reference.

`update-web-acl`

Il seguente esempio di codice mostra come utilizzare`update-web-acl`.

AWS CLI

Come aggiornare una lista di controllo degli accessi (ACL) web

Il comando `update-web-acl` seguente modifica le impostazioni per una lista di controllo degli accessi web esistente. Questa chiamata richiede un ID, che è possibile ottenere dalla chiamata `list-web-acls`, nonché un token di blocco e altre impostazioni, ottenibili dalla chiamata `get-web-acl`. Questa chiamata restituisce anche un token di blocco che è possibile utilizzare per un aggiornamento successivo.

```
aws wafv2 update-web-acl \
    --name TestWebAcl \
    --scope REGIONAL \
    --id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 \
    --lock-token 2294b3a1-0000-0000-0000-a3ae04329de9 \
    --default-action Block={} \
    --visibility-
config SampledRequestsEnabled=false,CloudWatchMetricsEnabled=false,MetricName=NewMetricTestW
 \
    --rules file://waf-rule.json \
```

```
    --region us-west-2
```

Output:

```
{
    "NextLockToken": "714a0cfb-0000-0000-0000-2959c8b9a684"
}
```

Per ulteriori informazioni, vedere Managing and Using a Web Access Control List (Web ACL) nella AWS WAF, AWS Firewall Manager, and AWS Shield Advanced Developer Guide.

- Per i dettagli sull'API, consulta AWS CLI Command UpdateWebAclReference.

WorkDocs esempi utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with WorkDocs.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`abort-document-version-upload`

Il seguente esempio di codice mostra come utilizzare`abort-document-version-upload`.

AWS CLI

Come arrestare il caricamento di una versione del documento

Questo esempio arresta il caricamento di una versione del documento avviato in precedenza.

Comando:

```
aws workdocs abort-document-version-upload --document-
id feaba64d4efdf271c2521b60a2a44a8f057e84beaabbe22f012673132098 35f2 --version-
id 1536773972914-ddb67663e782e7ce8455ebc962217cf9f9e47b5a9a702e5c84dcccd417da9313
```

Output:

```
None
```

- Per i dettagli sull'API, consulta AbortDocumentVersionUpload AWS CLICommand Reference.

`activate-user`

Il seguente esempio di codice mostra come utilizzare`activate-user`.

AWS CLI

Come attivare un utente

Questo esempio attiva un utente inattivo.

Comando:

```
aws workdocs activate-user --user-
id "S-1-1-11-1111111111-2222222222-3333333333-3333&d-926726012c"
```

Output:

```
{
  "User": {
      "Id": "S-1-1-11-1111111111-2222222222-3333333333-3333&d-926726012c",
      "Username": "exampleUser",
      "EmailAddress": "exampleUser@site.awsapps.com",
      "GivenName": "Example",
      "Surname": "User",
      "OrganizationId": "d-926726012c",
      "RootFolderId":
 "75f67c183aa1217409ac87576a45c03a5df5e6d8c51c35c01669970538e86cd0",
      "RecycleBinFolderId":
 "642b7dd3e60b14204534f3df7b1959e01b5d170f8c2707f410e40a8149120a57",
```

```
        "Status": "ACTIVE",
        "Type": "MINIMALUSER",
        "CreatedTimestamp": 1521226107.747,
        "ModifiedTimestamp": 1525297406.462,
        "Storage": {
            "StorageUtilizedInBytes": 0,
            "StorageRule": {
                "StorageAllocatedInBytes": 0,
                "StorageType": "QUOTA"
            }
        }
    }
}
```

- Per i dettagli sull'API, consulta ActivateUser AWS CLICommand Reference.

`add-resource-permissions`

Il seguente esempio di codice mostra come utilizzare`add-resource-permissions`.

AWS CLI

Come aggiungere autorizzazioni per una risorsa

Questo esempio aggiunge le autorizzazioni alla risorsa per i principali specificati.

Comando:

```
aws workdocs add-resource-permissions --resource-
id d90d93c1fe44bad0c8471e973ebaab339090401a95e777cffa58e977d2983b65 --
principals Id=anonymous,Type=ANONYMOUS,Role=VIEWER
```

Output:

```
{
    "ShareResults": [
        {
            "PrincipalId": "anonymous",
            "Role": "VIEWER",
            "Status": "SUCCESS",
            "ShareId":
 "d90d93c1fe44bad0c8471e973ebaab339090401a95e777cffa58e977d2983b65",
```

```
            "StatusMessage": ""
        }
    ]
}
```

- Per i dettagli sull'API, consulta AddResourcePermissions AWS CLICommand Reference.

`create-comment`

Il seguente esempio di codice mostra come utilizzare`create-comment`.

AWS CLI

Come aggiungere un nuovo commento

In questo esempio viene aggiunto un nuovo commento alla versione del documento specificata.

Comando:

```
aws workdocs create-comment --document-
id 15df51e0335cfcc6a2e4de9dd8be9f22ee40545ad9176f54758dcf903be982d3 --version-
id 1521672507741-9f7df0ea5dd0b121c4f3564a0c7c0b4da95cd12c635d3c442af337a88e297920 --
text "This is a comment."
```

Output:

```
{
    "Comment": {
        "CommentId": "1534799058197-
c7f5c84de9115875bbca93e0367bbebac609541d461636b760849b88b1609dd5",
        "ThreadId": "1534799058197-
c7f5c84de9115875bbca93e0367bbebac609541d461636b760849b88b1609dd5",
        "Text": "This is a comment.",
        "Contributor": {
            "Id": "arn:aws:iam::123456789123:user/exampleUser",
            "Username": "exampleUser",
            "GivenName": "Example",
            "Surname": "User",
            "Status": "ACTIVE"
        },
        "CreatedTimestamp": 1534799058.197,
```

```
        "Status": "PUBLISHED",
        "Visibility": "PUBLIC"
    }
}
```

- Per i dettagli sull'API, consulta CreateComment AWS CLICommand Reference.

`create-custom-metadata`

Il seguente esempio di codice mostra come utilizzare`create-custom-metadata`.

AWS CLI

Come creare metadati personalizzati

In questo esempio vengono creati metadati personalizzati per il documento specificato.

Comando:

```
aws workdocs create-custom-metadata --resource-
id d90d93c1fe44bad0c8471e973ebaab339090401a95e777cffa58e977d2983b65 --custom-
metadata KeyName1=example,KeyName2=example2
```

Output:

```
None
```

- Per i dettagli sull'API, consulta CreateCustomMetadata AWS CLICommand Reference.

`create-folder`

Il seguente esempio di codice mostra come utilizzare`create-folder`.

AWS CLI

Come creare una cartella

Questo esempio crea una cartella.

Comando:

```
aws workdocs create-folder --name documents --parent-folder-
id 1ece93e5fe75315c7407c4967918b4fd9da87ddb2a588e67b7fdaf4a98fde678
```

Output:

```
{
  "Metadata": {
      "Id": "50893c0af679524d1a0e0651130ed6d073e1a05f95bd12c42dcde5d35634ed08",
      "Name": "documents",
      "CreatorId": "S-1-1-11-1111111111-2222222222-3333333333-3333&d-926726012c",
      "ParentFolderId":
 "1ece93e5fe75315c7407c4967918b4fd9da87ddb2a588e67b7fdaf4a98fde678",
      "CreatedTimestamp": 1534450467.622,
      "ModifiedTimestamp": 1534450467.622,
      "ResourceState": "ACTIVE",
      "Signature": "",
      "Size": 0,
      "LatestVersionSize": 0
  }
}
```

- Per i dettagli sull'API, consulta CreateFolder AWS CLICommand Reference.

`create-labels`

Il seguente esempio di codice mostra come utilizzare`create-labels`.

AWS CLI

Come creare etichette

Questo esempio crea una serie di etichette per un documento.

Comando:

```
aws workdocs create-labels --resource-
id d90d93c1fe44bad0c8471e973ebaab339090401a95e777cffa58e977d2983b65 --
labels "documents" "examples" "my_documents"
```

Output:

```
None
```

- Per i dettagli sull'API, consulta CreateLabels AWS CLICommand Reference.

`create-notification-subscription`

Il seguente esempio di codice mostra come utilizzare`create-notification-subscription`.

AWS CLI

Come creare una sottoscrizione alle notifiche

L'`create-notification-subscription`esempio seguente configura un abbonamento alle notifiche per l' WorkDocs organizzazione Amazon specificata.

```
aws workdocs create-notification-subscription \
    --organization-id d-123456789c \
    --protocol HTTPS \
    --subscription-type ALL \
    --notification-endpoint "https://example.com/example"
```

Output:

```
{
    "Subscription": {
        "SubscriptionId": "123ab4c5-678d-901e-f23g-45h6789j0123",
        "EndPoint": "https://example.com/example",
        "Protocol": "HTTPS"
    }
}
```

Per ulteriori informazioni, consulta Abbonarsi alle notifiche nell'Amazon WorkDocs Developer Guide.

- Per i dettagli sull'API, consulta CreateNotificationSubscription AWS CLICommand Reference.

`create-user`

Il seguente esempio di codice mostra come utilizzare`create-user`.

AWS CLI

Come creare un nuovo utente

In questo esempio viene creato un nuovo utente in una directory Simple AD o Microsoft Active Directory.

Comando:

```
aws workdocs create-user --organization-id d-926726012c --username exampleUser2
 --email-address exampleUser2@site.awsapps.com --given-name example2Name --
surname example2Surname --password examplePa$$w0rd
```

Output:

```
{
  "User": {
      "Id": "S-1-1-11-1111111111-2222222222-3333333333-3333&d-926726012c",
      "Username": "exampleUser2",
      "EmailAddress": "exampleUser2@site.awsapps.com",
      "GivenName": "example2Name",
      "Surname": "example2Surname",
      "OrganizationId": "d-926726012c",
      "RootFolderId":
 "35b886cb17198cbd547655e58b025dff0cf34aaed638be52009567e23dc67390",
      "RecycleBinFolderId":
 "9858c3e9ed4c2460dde9aadb4c69fde998070dd46e5e985bd08ec6169ea249ff",
      "Status": "ACTIVE",
      "Type": "MINIMALUSER",
      "CreatedTimestamp": 1535478836.584,
      "ModifiedTimestamp": 1535478836.584,
      "Storage": {
          "StorageUtilizedInBytes": 0,
          "StorageRule": {
              "StorageAllocatedInBytes": 0,
              "StorageType": "QUOTA"
          }
      }
  }
}
```

- Per i dettagli sull'API, consulta CreateUser AWS CLICommand Reference.

`deactivate-user`

Il seguente esempio di codice mostra come utilizzare`deactivate-user`.

AWS CLI

Come disattivare un utente

Questo esempio disattiva un utente attivo.

Comando:

```
aws workdocs deactivate-user --user-
id "S-1-1-11-1111111111-2222222222-3333333333-3333&d-926726012c"
```

Output:

```
None
```

- Per i dettagli sull'API, consulta DeactivateUser AWS CLICommand Reference.

`delete-comment`

Il seguente esempio di codice mostra come utilizzare`delete-comment`.

AWS CLI

Come eliminare un commento specificato da una versione del documento

In questo esempio viene eliminato il commento specificato dalla versione del documento specificata.

Comando:

```
aws workdocs delete-comment --document-
id 15df51e0335cfcc6a2e4de9dd8be9f22ee40545ad9176f54758dcf903be982d3 --version-
id 1521672507741-9f7df0ea5dd0b121c4f3564a0c7c0b4da95cd12c635d3c442af337a88e297920 --
comment-id 1534799058197-
c7f5c84de9115875bbca93e0367bbebac609541d461636b760849b88b1609dd5
```

Output:

```
None
```

- Per i dettagli sull'API, consulta DeleteComment AWS CLICommand Reference.

`delete-custom-metadata`

Il seguente esempio di codice mostra come utilizzare`delete-custom-metadata`.

AWS CLI

Come eliminare i metadati personalizzati da una risorsa

In questo esempio vengono eliminati tutti i metadati personalizzati dalla risorsa specificata.

Comando:

```
aws workdocs delete-custom-metadata --resource-
id d90d93c1fe44bad0c8471e973ebaab339090401a95e777cffa58e977d2983b65 --delete-all
```

Output:

```
None
```

- Per i dettagli sull'API, consulta DeleteCustomMetadata AWS CLICommand Reference.

`delete-document`

Il seguente esempio di codice mostra come utilizzare`delete-document`.

AWS CLI

Come eliminare un documento

In questo esempio viene eliminato il documento specificato.

Comando:

```
aws workdocs delete-document --document-
id b83ed5e5b167b65ef69de9d597627ff1a0d4f07a45e67f1fab7d26b54427de0a
```

Output:

```
None
```

- Per i dettagli sull'API, consulta DeleteDocument AWS CLICommand Reference.

`delete-folder-contents`

Il seguente esempio di codice mostra come utilizzare`delete-folder-contents`.

AWS CLI

Come eliminare il contenuto di una cartella

In questo esempio viene eliminato il contenuto della cartella specificata.

Comando:

```
aws workdocs delete-folder-contents --folder-
id 26fa8aa4ba2071447c194f7b150b07149dbdb9e1c8a301872dcd93a4735ce65d
```

Output:

```
None
```

- Per i dettagli sull'API, consulta DeleteFolderContents AWS CLICommand Reference.

`delete-folder`

Il seguente esempio di codice mostra come utilizzare`delete-folder`.

AWS CLI

Come eliminare una cartella

Questo esempio elimina la cartella specificata.

Comando:

```
aws workdocs delete-folder --folder-
id 26fa8aa4ba2071447c194f7b150b07149dbdb9e1c8a301872dcd93a4735ce65d
```

Output:

```
None
```

- Per i dettagli sull'API, consulta DeleteFolder AWS CLICommand Reference.

`delete-labels`

Il seguente esempio di codice mostra come utilizzare`delete-labels`.

AWS CLI

Come eliminare etichette

In questo esempio vengono eliminate le etichette specificate da un documento.

Comando:

```
aws workdocs delete-labels --resource-
id d90d93c1fe44bad0c8471e973ebaab339090401a95e777cffa58e977d2983b65 --
labels "documents" "examples"
```

Output:

```
None
```

- Per i dettagli sull'API, consulta DeleteLabels AWS CLICommand Reference.

`delete-notification-subscription`

Il seguente esempio di codice mostra come utilizzare`delete-notification-subscription`.

AWS CLI

Come eliminare una sottoscrizione alle notifiche

L'esempio `delete-notification-subscription` seguente elimina la sottoscrizione alle notifiche specificata.

```
aws workdocs delete-notification-subscription \
    --subscription-id 123ab4c5-678d-901e-f23g-45h6789j0123 \
    --organization-id d-123456789c
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Abbonarsi alle notifiche nell'Amazon WorkDocs Developer Guide.

- Per i dettagli sull'API, consulta DeleteNotificationSubscription AWS CLICommand Reference.

`delete-user`

Il seguente esempio di codice mostra come utilizzare`delete-user`.

AWS CLI

Come eliminare un utente

Questo esempio elimina un utente.

Comando:

```
aws workdocs delete-user --user-
id "S-1-1-11-1111111111-2222222222-3333333333-3333&d-926726012c"
```

Output:

```
None
```

- Per i dettagli sull'API, consulta DeleteUser AWS CLICommand Reference.

`describe-activities`

Il seguente esempio di codice mostra come utilizzare`describe-activities`.

AWS CLI

Come ottenere un elenco delle attività degli utenti

Questo esempio restituisce un elenco delle attività degli utenti più recenti per l'organizzazione specificata, con un limite impostato per le ultime due attività.

Comando:

```
aws workdocs describe-activities --organization-id d-926726012c --limit 2
```

Output:

```
{
  "UserActivities": [
      {
          "Type": "DOCUMENT_VERSION_DOWNLOADED",
          "TimeStamp": 1534800122.17,
          "Initiator": {
              "Id": "arn:aws:iam::123456789123:user/exampleUser"
          },
          "ResourceMetadata": {
              "Type": "document",
              "Name": "updatedDoc",
              "Id":
 "15df51e0335cfcc6a2e4de9dd8be9f22ee40545ad9176f54758dcf903be982d3",
              "Owner": {
                  "Id":
 "S-1-1-11-1111111111-2222222222-3333333333-3333&d-926726012c",
                  "GivenName": "exampleName",
                  "Surname": "exampleSurname"
              }
          }
      },
      {
          "Type": "DOCUMENT_VERSION_VIEWED",
          "TimeStamp": 1534799079.207,
          "Initiator": {
              "Id": "S-1-1-11-1111111111-2222222222-3333333333-3333&d-926726012c",
              "GivenName": "exampleName",
              "Surname": "exampleSurname"
          },
          "ResourceMetadata": {
              "Type": "document",
              "Name": "updatedDoc",
              "Id":
 "15df51e0335cfcc6a2e4de9dd8be9f22ee40545ad9176f54758dcf903be982d3",
              "Owner": {
                  "Id":
 "S-1-1-11-1111111111-2222222222-3333333333-3333&d-926726012c",
                  "GivenName": "exampleName",
                  "Surname": "exampleSurname"
              }
          }
      }
```

```
  ],
  "Marker":
 "DnF1ZXJ5VGhlbkZldGNoAgAAAAAAAAS7FmlTaU1OdlFTU1h1UU00VVFIbDlRWHcAAAAAAAAAJTRY3bWh5eUgzaVF1ZX
}
```

- Per i dettagli sull'API, consulta DescribeActivities AWS CLICommand Reference.

`describe-comments`

Il seguente esempio di codice mostra come utilizzare`describe-comments`.

AWS CLI

Come elencare tutti i commenti di una versione del documento specificata

In questo esempio vengono elencati tutti i commenti per la versione del documento specificata.

Comando:

```
aws workdocs describe-comments --document-
id 15df51e0335cfcc6a2e4de9dd8be9f22ee40545ad9176f54758dcf903be982d3 --version-
id 1521672507741-9f7df0ea5dd0b121c4f3564a0c7c0b4da95cd12c635d3c442af337a88e297920
```

Output:

```
{
  "Comments": [
      {
          "CommentId": "1534799058197-
c7f5c84de9115875bbca93e0367bbebac609541d461636b760849b88b1609dd5",
          "ThreadId": "1534799058197-
c7f5c84de9115875bbca93e0367bbebac609541d461636b760849b88b1609dd5",
          "Text": "This is a comment.",
          "Contributor": {
              "Username": "arn:aws:iam::123456789123:user/exampleUser",
              "Type": "USER"
          },
          "CreatedTimestamp": 1534799058.197,
          "Status": "PUBLISHED",
          "Visibility": "PUBLIC"
      }
  ]
```

```
}
```

- Per i dettagli sull'API, consulta DescribeComments AWS CLICommand Reference.

`describe-document-versions`

Il seguente esempio di codice mostra come utilizzare`describe-document-versions`.

AWS CLI

Come recuperare le versioni di un documento

Questo esempio recupera le versioni del documento specificato, incluse le versioni inizializzate e un URL per il documento di origine.

Comando:

```
aws workdocs describe-document-versions --document-
id d90d93c1fe44bad0c8471e973ebaab339090401a95e777cffa58e977d2983b65 --fields SOURCE
```

Output:

```
{
  "DocumentVersions": [
      {
          "Id":
 "1534452029587-15e129dfc187505c407588df255be83de2920d733859f1d2762411d22a83e3ef",
          "Name": "exampleDoc.docx",
          "ContentType": "application/vnd.openxmlformats-
officedocument.wordprocessingml.document",
          "Size": 13922,
          "Signature": "1a23456b78901c23d4ef56gh7EXAMPLE",
          "Status": "ACTIVE",
          "CreatedTimestamp": 1534452029.587,
          "ModifiedTimestamp": 1534452029.849,
          "CreatorId":
 "S-1-1-11-1111111111-2222222222-3333333333-3333&d-926726012c",
          "Source": {
              "ORIGINAL": "https://gb-us-west-2-prod-doc-source.s3.us-
west-2.amazonaws.com/
d90d93c1fe44bad0c8471e973ebaab339090401a95e777cffa58e977d2983b65/1534452029587-15e129dfc1875
```

```
response-content-disposition=attachment%3B%20filename%2A
%3DUTF-8%27%27exampleDoc29.docx&X-Amz-Algorithm=AWS1-ABCD-EFG234&X-Amz-
Date=20180816T204149Z&X-Amz-SignedHeaders=host&X-Amz-Expires=900&X-Amz-
Credential=AKIAIOSFODNN7EXAMPLE%2F20180816%2Fus-west-2%2Fs3%2Faws1_request&X-Amz-
Signature=01Ab2c34d567e8f90123g456hi78j901k2345678l901234mno56pqr78EXAMPLE"
          }
      },
      {
          "Id": "1529005196082-
bb75fa19abc287699cb07147f75816dce43a53a10f28dc001bf61ef2fab01c59",
          "Name": "exampleDoc.pdf",
          "ContentType": "application/pdf",
          "Size": 425916,
          "Signature": "1a23456b78901c23d4ef56gh7EXAMPLE",
          "Status": "ACTIVE",
          "CreatedTimestamp": 1529005196.082,
          "ModifiedTimestamp": 1529005196.796,
          "CreatorId":
 "S-1-1-11-1111111111-2222222222-3333333333-3333&d-926726012c",
          "Source": {
              "ORIGINAL": "https://gb-us-west-2-prod-doc-source.s3.us-
west-2.amazonaws.com/
d90d93c1fe44bad0c8471e973ebaab339090401a95e777cffa58e977d2983b65/1529005196082-
bb75fa19abc287699cb07147f75816dce43a53a10f28dc001bf61ef2fab01c59?
response-content-disposition=attachment%3B%20filename%2A
%3DUTF-8%27%27exampleDoc29.pdf&X-Amz-Algorithm=AWS1-ABCD-EFG234&X-Amz-
Date=20180816T204149Z&X-Amz-SignedHeaders=host&X-Amz-Expires=900&X-Amz-
Credential=AKIAIOSFODNN7EXAMPLE%2F20180816%2Fus-west-2%2Fs3%2Faws1_request&X-Amz-
Signature=01Ab2c34d567e8f90123g456hi78j901k2345678l901234mno56pqr78EXAMPLE"
          }
      }
  ]
}
```

- Per i dettagli sull'API, consulta DescribeDocumentVersions AWS CLICommand Reference.

`describe-folder-contents`

Il seguente esempio di codice mostra come utilizzare`describe-folder-contents`.

AWS CLI

Come descrivere il contenuto di una cartella

Questo esempio descrive tutto il contenuto attivo della cartella specificata, inclusi i suoi documenti e le sue sottocartelle, in ordine crescente per data.

Comando:

```
aws workdocs describe-folder-contents --folder-
id 1ece93e5fe75315c7407c4967918b4fd9da87ddb2a588e67b7fdaf4a98fde678 --sort DATE --
order ASCENDING --type ALL
```

Output:

```
{
  "Folders": [
      {
          "Id": "50893c0af679524d1a0e0651130ed6d073e1a05f95bd12c42dcde5d35634ed08",
          "Name": "testing",
          "CreatorId":
 "S-1-1-11-1111111111-2222222222-3333333333-3333&d-926726012c",
          "ParentFolderId":
 "1ece93e5fe75315c7407c4967918b4fd9da87ddb2a588e67b7fdaf4a98fde678",
          "CreatedTimestamp": 1534450467.622,
          "ModifiedTimestamp": 1534451113.504,
          "ResourceState": "ACTIVE",
          "Signature": "1a23456b78901c23d4ef56gh7EXAMPLE",
          "Size": 23019,
          "LatestVersionSize": 11537
      }
  ],
  "Documents": [
      {
          "Id": "d90d93c1fe44bad0c8471e973ebaab339090401a95e777cffa58e977d2983b65",
          "CreatorId":
 "S-1-1-11-1111111111-2222222222-3333333333-3333&d-926726012c",
          "ParentFolderId":
 "1ece93e5fe75315c7407c4967918b4fd9da87ddb2a588e67b7fdaf4a98fde678",
          "CreatedTimestamp": 1529005196.082,
          "ModifiedTimestamp": 1534452483.01,
          "LatestVersionMetadata": {
              "Id":
 "1534452029587-15e129dfc187505c407588df255be83de2920d733859f1d2762411d22a83e3ef",
              "Name": "exampleDoc.docx",
              "ContentType": "application/vnd.openxmlformats-
officedocument.wordprocessingml.document",
```

```
            "Size": 13922,
            "Signature": "1a23456b78901c23d4ef56gh7EXAMPLE",
            "Status": "ACTIVE",
            "CreatedTimestamp": 1534452029.587,
            "ModifiedTimestamp": 1534452029.587,
            "CreatorId":
 "S-1-1-11-1111111111-2222222222-3333333333-3333&d-926726012c"
        },
        "ResourceState": "ACTIVE"
    }
  ]
}
```

- Per i dettagli sull'API, consulta DescribeFolderContents AWS CLICommand Reference.

`describe-groups`

Il seguente esempio di codice mostra come utilizzare`describe-groups`.

AWS CLI

Come recuperare un elenco di gruppi

L'`describe-groups`esempio seguente elenca i gruppi associati all' WorkDocs organizzazione Amazon specificata.

```
aws workdocs describe-groups \
    --search-query "e" \
    --organization-id d-123456789c
```

Output:

```
{
    "Groups": [
        {
            "Id": "S-1-1-11-1122222222-2222233333-3333334444-4444&d-123456789c",
            "Name": "Example Group 1"
        },
        {
            "Id": "S-1-1-11-1122222222-2222233333-3333334444-5555&d-123456789c",
            "Name": "Example Group 2"
```

```
        }
    ]
}
```

Per ulteriori informazioni, consulta la sezione Getting Started with Amazon WorkDocs nella Amazon WorkDocs Administration Guide.

- Per i dettagli sull'API, consulta DescribeGroups AWS CLICommand Reference.

`describe-notification-subscriptions`

Il seguente esempio di codice mostra come utilizzare`describe-notification-subscriptions`.

AWS CLI

Come recuperare un elenco di sottoscrizioni alle notifiche

L'`describe-notification-subscriptions`esempio seguente recupera gli abbonamenti alle notifiche per l'organizzazione Amazon WorkDocs specificata.

```
aws workdocs describe-notification-subscriptions \
    --organization-id d-123456789c
```

Output:

```
{
    "Subscriptions": [
        {
            "SubscriptionId": "123ab4c5-678d-901e-f23g-45h6789j0123",
            "EndPoint": "https://example.com/example",
            "Protocol": "HTTPS"
        }
    ]
}
```

Per ulteriori informazioni, consulta Abbonarsi alle notifiche nell'Amazon WorkDocs Developer Guide.

- Per i dettagli sull'API, consulta DescribeNotificationSubscriptions AWS CLICommand Reference.

`describe-resource-permissions`

Il seguente esempio di codice mostra come utilizzare`describe-resource-permissions`.

AWS CLI

Come ottenere un elenco di autorizzazioni per una risorsa

Questo esempio restituisce un elenco delle autorizzazioni per la risorsa specificata (documento o cartella).

Comando:

```
aws workdocs describe-resource-permissions --resource-
id 15df51e0335cfcc6a2e4de9dd8be9f22ee40545ad9176f54758dcf903be982d3
```

Output:

```
{
  "Principals": [
      {
          "Id": "anonymous",
          "Type": "ANONYMOUS",
          "Roles": [
              {
                  "Role": "VIEWER",
                  "Type": "DIRECT"
              }
          ]
      },
      {
          "Id": "S-1-1-11-1111111111-2222222222-3333333333-3333&d-926726012c",
          "Type": "USER",
          "Roles": [
              {
                  "Role": "OWNER",
                  "Type": "DIRECT"
              }
          ]
      },
      {
          "Id": "d-926726012c",
          "Type": "ORGANIZATION",
```

```
            "Roles": [
                {
                    "Role": "VIEWER",
                    "Type": "INHERITED"
                }
            ]
        }
    ]
}
```

- Per i dettagli sull'API, consulta DescribeResourcePermissions AWS CLICommand Reference.

`describe-users`

Il seguente esempio di codice mostra come utilizzare`describe-users`.

AWS CLI

Come recuperare i dettagli per utenti specificati

Questo esempio recupera i dettagli di tutti gli utenti dell'organizzazione specificata.

Comando:

```
aws workdocs describe-users --organization-id d-926726012c
```

Output:

```
{
  "Users": [
      {
          "Id": "S-1-1-11-1111111111-2222222222-3333333333-3333&d-926726012c",
          "Username": "example1User",
          "OrganizationId": "d-926726012c",
          "RootFolderId":
 "3c0e3f849dd20a9771d937b9bbcc97e18796150ae56c26d64a4fa0320a2dedc9",
          "RecycleBinFolderId":
 "c277f4c4d647be1f5147b3184ffa96e1e2bf708278b696cacba68ba13b91f4fe",
          "Status": "INACTIVE",
          "Type": "USER",
          "CreatedTimestamp": 1535478999.452,
          "ModifiedTimestamp": 1535478999.452
      },
```

```
        {
            "Id": "S-1-1-11-1111111111-2222222222-3333333333-4444&d-926726012c",
            "Username": "example2User",
            "EmailAddress": "example2User@site.awsapps.com",
            "GivenName": "example2Name",
            "Surname": "example2Surname",
            "OrganizationId": "d-926726012c",
            "RootFolderId":
 "35b886cb17198cbd547655e58b025dff0cf34aaed638be52009567e23dc67390",
            "RecycleBinFolderId":
 "9858c3e9ed4c2460dde9aadb4c69fde998070dd46e5e985bd08ec6169ea249ff",
            "Status": "ACTIVE",
            "Type": "MINIMALUSER",
            "CreatedTimestamp": 1535478836.584,
            "ModifiedTimestamp": 1535478836.584
        }
    ]
}
```

- Per i dettagli sull'API, consulta DescribeUsers AWS CLICommand Reference.

`get-document-path`

Il seguente esempio di codice mostra come utilizzare`get-document-path`.

AWS CLI

Come recuperare le informazioni sul percorso di un documento

Questo esempio recupera le informazioni sul percorso (gerarchia a partire dalla cartella radice) per il documento specificato e include i nomi delle cartelle principali.

Comando:

```
aws workdocs get-document-path --document-
id d90d93c1fe44bad0c8471e973ebaab339090401a95e777cffa58e977d2983b65 --fields NAME
```

Output:

```
{
    "Path": {
        "Components": [
            {
```

```
                "Id":
 "a43d29cbb8e7c4d25cfee8b803a504b0dc63e760b55ad0c611c6b87691eb6ff3",
                "Name": "/"
            },
            {
                "Id":
 "1ece93e5fe75315c7407c4967918b4fd9da87ddb2a588e67b7fdaf4a98fde678",
                "Name": "Top Level Folder"
            },
            {
                "Id":
 "d90d93c1fe44bad0c8471e973ebaab339090401a95e777cffa58e977d2983b65",
                "Name": "exampleDoc.docx"
            }
        ]
    }
}
```

- Per i dettagli sull'API, consulta GetDocumentPath AWS CLICommand Reference.

`get-document-version`

Il seguente esempio di codice mostra come utilizzare`get-document-version`.

AWS CLI

Come recuperare i metadati della versione per un documento specificato

Questo esempio recupera i metadati della versione per il documento specificato, inclusi un URL di origine e i metadati personalizzati.

Comando:

```
aws workdocs get-document-version --document-
id 15df51e0335cfcc6a2e4de9dd8be9f22ee40545ad9176f54758dcf903be982d3 --version-
id 1521672507741-9f7df0ea5dd0b121c4f3564a0c7c0b4da95cd12c635d3c442af337a88e297920 --
fields SOURCE --include-custom-metadata
```

Output:

```
{
  "Metadata": {
```

```
      "Id":
 "1521672507741-9f7df0ea5dd0b121c4f3564a0c7c0b4da95cd12c635d3c442af337a88e297920",
      "Name": "exampleDoc",
      "ContentType": "application/vnd.openxmlformats-
officedocument.wordprocessingml.document",
      "Size": 11537,
      "Signature": "1a23456b78901c23d4ef56gh7EXAMPLE",
      "Status": "ACTIVE",
      "CreatedTimestamp": 1521672507.741,
      "ModifiedTimestamp": 1534451113.504,
      "CreatorId": "S-1-1-11-1111111111-2222222222-3333333333-3333&d-926726012c",
      "Source": {
          "ORIGINAL": "https://gb-us-west-2-prod-doc-source.s3.us-
west-2.amazonaws.com/15df51e0335cfcc6a2e4de9dd8be9f22ee40545ad9176f54758dcf903be982d3/152167
response-content-disposition=attachment%3B%20filename%2A
%3DUTF-8%27%27exampleDoc&X-Amz-Algorithm=AWS1-ABCD-EFG234&X-Amz-
Date=20180820T212202Z&X-Amz-SignedHeaders=host&X-Amz-Expires=900&X-Amz-
Credential=AKIAIOSFODNN7EXAMPLE%2F20180820%2Fus-west-2%2Fs3%2Faws1_request&X-Amz-
Signature=01Ab2c34d567e8f90123g456hi78j901k2345678l901234mno56pqr78EXAMPLE"
      }
  }
}
```

- Per i dettagli sull'API, consulta GetDocumentVersion AWS CLICommand Reference.

`get-document`

Il seguente esempio di codice mostra come utilizzare`get-document`.

AWS CLI

Come recuperare i dettagli del documento

In questo esempio vengono recuperati i dettagli del documento specificato.

Comando:

```
aws workdocs get-document --document-
id d90d93c1fe44bad0c8471e973ebaab339090401a95e777cffa58e977d2983b65
```

Output:

```
{
```

```
  "Metadata": {
      "Id": "d90d93c1fe44bad0c8471e973ebaab339090401a95e777cffa58e977d2983b65",
      "CreatorId": "S-1-1-11-1111111111-2222222222-3333333333-3333&d-926726012c",
      "ParentFolderId":
 "1ece93e5fe75315c7407c4967918b4fd9da87ddb2a588e67b7fdaf4a98fde678",
      "CreatedTimestamp": 1529005196.082,
      "ModifiedTimestamp": 1534452483.01,
      "LatestVersionMetadata": {
          "Id":
 "1534452029587-15e129dfc187505c407588df255be83de2920d733859f1d2762411d22a83e3ef",
          "Name": "exampleDoc.docx",
          "ContentType": "application/vnd.openxmlformats-
officedocument.wordprocessingml.document",
          "Size": 13922,
          "Signature": "1a23456b78901c23d4ef56gh7EXAMPLE",
          "Status": "ACTIVE",
          "CreatedTimestamp": 1534452029.587,
          "ModifiedTimestamp": 1534452029.587,
          "CreatorId": "S-1-1-11-1111111111-2222222222-3333333333-3333&d-926726012c"
      },
      "ResourceState": "ACTIVE"
  }
}
```

- Per i dettagli sull'API, consulta GetDocument AWS CLICommand Reference.

`get-folder-path`

Il seguente esempio di codice mostra come utilizzare`get-folder-path`.

AWS CLI

Come recuperare le informazioni sul percorso per una cartella

Questo esempio recupera le informazioni sul percorso (gerarchia a partire dalla cartella radice) per la cartella specificata e include i nomi delle cartelle principali.

Comando:

```
aws workdocs get-folder-path --folder-
id 50893c0af679524d1a0e0651130ed6d073e1a05f95bd12c42dcde5d35634ed08 --fields NAME
```

Output:

```
{
  "Path": {
      "Components": [
          {
              "Id":
 "a43d29cbb8e7c4d25cfee8b803a504b0dc63e760b55ad0c611c6b87691eb6ff3",
              "Name": "/"
          },
          {
              "Id":
 "1ece93e5fe75315c7407c4967918b4fd9da87ddb2a588e67b7fdaf4a98fde678",
              "Name": "Top Level Folder"
          },
          {
              "Id":
 "50893c0af679524d1a0e0651130ed6d073e1a05f95bd12c42dcde5d35634ed08",
              "Name": "Sublevel Folder"
          }
      ]
  }
}
```

- Per i dettagli sull'API, consulta GetFolderPath AWS CLICommand Reference.

`get-folder`

Il seguente esempio di codice mostra come utilizzare`get-folder`.

AWS CLI

Come recuperare i metadati per una cartella

Questo esempio recupera i metadati della cartella specificata.

Comando:

```
aws workdocs get-folder --folder-
id 50893c0af679524d1a0e0651130ed6d073e1a05f95bd12c42dcde5d35634ed08
```

Output:

```
{
```

```
  "Metadata": {
      "Id": "50893c0af679524d1a0e0651130ed6d073e1a05f95bd12c42dcde5d35634ed08",
      "Name": "exampleFolder",
      "CreatorId": "S-1-1-11-1111111111-2222222222-3333333333-3333&d-926726012c",
      "ParentFolderId":
 "1ece93e5fe75315c7407c4967918b4fd9da87ddb2a588e67b7fdaf4a98fde678",
      "CreatedTimestamp": 1534450467.622,
      "ModifiedTimestamp": 1534451113.504,
      "ResourceState": "ACTIVE",
      "Signature": "1a23456b78901c23d4ef56gh7EXAMPLE",
      "Size": 23019,
      "LatestVersionSize": 11537
  }
}
```

- Per i dettagli sull'API, consulta GetFolder AWS CLICommand Reference.

`get-resources`

Il seguente esempio di codice mostra come utilizzare`get-resources`.

AWS CLI

Come recuperare risorse condivise

L'`get-resources`esempio seguente recupera le risorse condivise con l' WorkDocs utente Amazon specificato.

```
aws workdocs get-resources \
    --user-id "S-1-1-11-1111111111-2222222222-3333333333-3333" \
    --collection-type SHARED_WITH_ME
```

Output:

```
{
"Folders": [],
"Documents": []
}
```

Per ulteriori informazioni, consulta Sharing Files and Folders nella Amazon WorkDocs User Guide.

- Per i dettagli sull'API, consulta GetResources AWS CLICommand Reference.

`initiate-document-version-upload`

Il seguente esempio di codice mostra come utilizzare`initiate-document-version-upload`.

AWS CLI

Come avviare il caricamento di una versione del documento

L'esempio `initiate-document-upload` seguente crea un nuovo oggetto documento e un nuovo oggetto versione.

```
aws workdocs initiate-document-version-upload \
    --name exampledocname \
    --parent-folder-
id eacd546d952531c633452ed67cac23161aa0d5df2e8061223a59e8f67e7b6189
```

Output:

```
{
    "Metadata": {
        "Id": "feaba64d4efdf271c2521b60a2a44a8f057e84beaabbe22f01267313209835f2",
        "CreatorId": "S-1-1-11-1111111111-2222222222-3333333333-3333&d-926726012c",
        "ParentFolderId":
 "eacd546d952531c633452ed67cac23161aa0d5df2e8061223a59e8f67e7b6189",
        "CreatedTimestamp": 1536773972.914,
        "ModifiedTimestamp": 1536773972.914,
        "LatestVersionMetadata": {
            "Id": "1536773972914-
ddb67663e782e7ce8455ebc962217cf9f9e47b5a9a702e5c84dcccd417da9313",
            "Name": "exampledocname",
            "ContentType": "application/octet-stream",
            "Size": 0,
            "Status": "INITIALIZED",
            "CreatedTimestamp": 1536773972.914,
            "ModifiedTimestamp": 1536773972.914,
            "CreatorId": "arn:aws:iam::123456789123:user/EXAMPLE"
        },
        "ResourceState": "ACTIVE"
    },
    "UploadMetadata": {
```

```
        "UploadUrl": "https://gb-us-west-2-prod-doc-source.s3.us-
west-2.amazonaws.com/
feaba64d4efdf271c2521b60a2a44a8f057e84beaabbe22f01267313209835f2/1536773972914-
ddb67663e782e7ce8455ebc962217cf9f9e47b5a9a702e5c84dcccd417da9313?X-Amz-
Algorithm=AWS1-ABCD-EFG234&X-Amz-Date=20180912T173932Z&X-Amz-SignedHeaders=content-
type%3Bhost%3Bx-amz-server-side-encryption&X-Amz-Expires=899&X-Amz-
Credential=AKIAIOSFODNN7EXAMPLE%2F20180912%2Fus-west-2%2Fs3%2Faws1_request&X-Amz-
Signature=01Ab2c34d567e8f90123g456hi78j901k234567l901234mno56pqr78EXAMPLE",
        "SignedHeaders": {
            "Content-Type": "application/octet-stream",
            "x-amz-server-side-encryption": "ABC123"
        }
    }
}
```

- Per i dettagli sull'API, consulta InitiateDocumentVersionUpload AWS CLICommand Reference.

`remove-all-resource-permissions`

Il seguente esempio di codice mostra come utilizzare`remove-all-resource-permissions`.

AWS CLI

Come rimuovere tutte le autorizzazioni da una risorsa specificata

In questo esempio vengono rimosse tutte le autorizzazioni dalla risorsa specificata.

Comando:

```
aws workdocs remove-all-resource-permissions --resource-
id 1ece93e5fe75315c7407c4967918b4fd9da87ddb2a588e67b7fdaf4a98fde678
```

Output:

```
None
```

- Per i dettagli sull'API, consulta RemoveAllResourcePermissions AWS CLICommand Reference.

`remove-resource-permission`

Il seguente esempio di codice mostra come utilizzare`remove-resource-permission`.

AWS CLI

Come rimuovere le autorizzazioni da una risorsa

Questo esempio rimuove le autorizzazioni dalla risorsa per il principale specificato.

Comando:

```
aws workdocs remove-resource-permission --resource-
id 1ece93e5fe75315c7407c4967918b4fd9da87ddb2a588e67b7fdaf4a98fde678 --principal-
id anonymous
```

Output:

```
None
```

- Per i dettagli sull'API, consulta RemoveResourcePermission AWS CLICommand Reference.

`update-document-version`

Il seguente esempio di codice mostra come utilizzare`update-document-version`.

AWS CLI

Come modificare lo stato della versione di un documento su "attivo

In questo esempio viene modificato lo stato della versione del documento su "attivo".

Comando:

```
aws workdocs update-document-version --document-
id 15df51e0335cfcc6a2e4de9dd8be9f22ee40545ad9176f54758dcf903be982d3 --version-
id 1521672507741-9f7df0ea5dd0b121c4f3564a0c7c0b4da95cd12c635d3c442af337a88e297920 --
version-status ACTIVE
```

Output:

```
None
```

- Per i dettagli sull'API, consulta UpdateDocumentVersion AWS CLICommand Reference.

`update-document`

Il seguente esempio di codice mostra come utilizzare`update-document`.

AWS CLI

Come aggiornare un documento

Questo esempio aggiorna il nome e la cartella principale di un documento.

Comando:

```
aws workdocs update-document --document-
id 15df51e0335cfcc6a2e4de9dd8be9f22ee40545ad9176f54758dcf903be982d3 --
name updatedDoc --parent-folder-
id 50893c0af679524d1a0e0651130ed6d073e1a05f95bd12c42dcde5d35634ed08
```

Output:

```
None
```

- Per i dettagli sull'API, consulta UpdateDocument AWS CLICommand Reference.

`update-folder`

Il seguente esempio di codice mostra come utilizzare`update-folder`.

AWS CLI

Come aggiornare una cartella

Questo esempio aggiorna il nome e la cartella principale di una cartella.

Comando:

```
aws workdocs update-folder --folder-
id 50893c0af679524d1a0e0651130ed6d073e1a05f95bd12c42dcde5d35634ed08 --
name exampleFolder1 --parent-folder-
id 1ece93e5fe75315c7407c4967918b4fd9da87ddb2a588e67b7fdaf4a98fde678
```

Output:

```
None
```

- Per i dettagli sull'API, consulta UpdateFolder AWS CLICommand Reference.

`update-user`

Il seguente esempio di codice mostra come utilizzare`update-user`.

AWS CLI

Come aggiornare un utente

In questo esempio viene aggiornato il fuso orario per l'utente specificato.

Comando:

```
aws workdocs update-user --user-
id "S-1-1-11-1111111111-2222222222-3333333333-3333&d-926726012c" --time-zone-
id "America/Los_Angeles"
```

Output:

```
{
  "User": {
      "Id": "S-1-1-11-1111111111-2222222222-3333333333-3333&d-926726012c",
      "Username": "exampleUser",
      "EmailAddress": "exampleUser@site.awsapps.com",
      "GivenName": "Example",
      "Surname": "User",
      "OrganizationId": "d-926726012c",
      "RootFolderId":
 "c5eceb5e1a2d1d460c9d1af8330ae117fc8d39bb1d3ed6acd0992d5ff192d986",
      "RecycleBinFolderId":
 "6ca20102926ad15f04b1d248d6d6e44f2449944eda5c758f9a1e9df6a6b7fa66",
      "Status": "ACTIVE",
      "Type": "USER",
      "TimeZoneId": "America/Los_Angeles",
      "Storage": {
          "StorageUtilizedInBytes": 0,
          "StorageRule": {
              "StorageAllocatedInBytes": 53687091200,
```

```
                "StorageType": "QUOTA"
            }
        }
    }
}
```

- Per i dettagli sull'API, consulta UpdateUser AWS CLICommand Reference.

WorkMail Esempi di utilizzo di Amazon AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando il AWS Command Line Interface con Amazon WorkMail.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`associate-delegate-to-resource`

Il seguente esempio di codice mostra come usare`associate-delegate-to-resource`.

AWS CLI

Come aggiungere un delegato a una risorsa

Il comando `associate-delegate-to-resource` seguente aggiunge un delegato a una risorsa.

```
aws workmail associate-delegate-to-resource \
    --organization-id m-d281d0a2fd824be5b6cd3d3ce909fd27 \
    --resource-id r-68bf2d3b1c0244aab7264c24b9217443 \
```

```
    --entity-id S-1-1-11-1111111111-2222222222-3333333333-3333
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta AssociateDelegateToResource AWS CLICommand Reference.

`associate-member-to-group`

Il seguente esempio di codice mostra come utilizzare`associate-member-to-group`.

AWS CLI

Come aggiungere un membro a un gruppo

Il comando `associate-member-to-group` seguente aggiunge il membro specificato a un gruppo.

```
aws workmail associate-member-to-group \
    --organization-id m-d281d0a2fd824be5b6cd3d3ce909fd27 \
    --group-id S-1-1-11-1122222222-2222233333-3333334444-4444 \
    --member-id S-1-1-11-1111111111-2222222222-3333333333-3333
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta AssociateMemberToGroup AWS CLICommand Reference.

`create-alias`

Il seguente esempio di codice mostra come utilizzare`create-alias`.

AWS CLI

Come creare un alias

Il comando `create-alias` seguente crea un alias per l'entità specificata (utente o gruppo).

```
aws workmail create-alias \
    --organization-id m-d281d0a2fd824be5b6cd3d3ce909fd27 \
    --entity-id S-1-1-11-1122222222-2222233333-3333334444-4444 \
    --alias exampleAlias@site.awsapps.com
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta CreateAlias AWS CLICommand Reference.

`create-group`

Il seguente esempio di codice mostra come utilizzare`create-group`.

AWS CLI

Come creare un nuovo gruppo

Il comando `create-group` seguente crea un nuovo gruppo per l'organizzazione specificata.

```
aws workmail create-group \
    --organization-id m-d281d0a2fd824be5b6cd3d3ce909fd27 \
    --name exampleGroup1
```

Output:

```
{
    "GroupId": "S-1-1-11-1122222222-2222233333-3333334444-4444"
}
```

- Per i dettagli sull'API, consulta CreateGroup AWS CLICommand Reference.

`create-resource`

Il seguente esempio di codice mostra come utilizzare`create-resource`.

AWS CLI

Come creare una nuova risorsa

Il comando `create-resource` seguente crea una nuova risorsa (sala riunioni) per l'organizzazione specificata.

```
aws workmail create-resource \
    --organization-id m-d281d0a2fd824be5b6cd3d3ce909fd27 \
    --name exampleRoom1 \
    --type ROOM
```

Output:

```
{
    "ResourceId": "r-7afe0efbade843a58cdc10251fce992c"
}
```

- Per i dettagli sull'API, consulta CreateResource AWS CLICommand Reference.

`create-user`

Il seguente esempio di codice mostra come utilizzare`create-user`.

AWS CLI

Come creare un nuovo utente

Il comando `create-user` seguente crea un nuovo utente.

```
aws workmail create-user \
    --organization-id m-d281d0a2fd824be5b6cd3d3ce909fd27 \
    --name exampleName \
    --display-name exampleDisplayName \
    --password examplePa$$w0rd
```

Output:

```
{
    "UserId": "S-1-1-11-1111111111-2222222222-3333333333-3333"
}
```

- Per i dettagli sull'API, consulta CreateUser AWS CLICommand Reference.

`delete-access-control-rule`

Il seguente esempio di codice mostra come utilizzare`delete-access-control-rule`.

AWS CLI

Come eliminare una regola di controllo degli accessi

L'`delete-access-control-rule`esempio seguente elimina la regola di controllo degli accessi specificata dall' WorkMail organizzazione Amazon specificata.

```
aws workmail delete-access-control-rule \
    --organization-id m-n1pq2345678r901st2u3vx45x6789yza \
    --name "myRule"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Working with Access Control Rules nella Amazon WorkMail Administrator Guide.

- Per i dettagli sull'API, consulta DeleteAccessControlRule AWS CLICommand Reference.

`delete-alias`

Il seguente esempio di codice mostra come utilizzare`delete-alias`.

AWS CLI

Come eliminare un alias

Il comando `delete-alias` seguente elimina l'alias per l'entità specificata (utente o gruppo).

```
aws workmail delete-alias \
    --organization-id m-d281d0a2fd824be5b6cd3d3ce909fd27 \
    --entity-id S-1-1-11-1122222222-2222233333-3333334444-4444 \
    --alias exampleAlias@site.awsapps.com
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta DeleteAlias AWS CLICommand Reference.

`delete-group`

Il seguente esempio di codice mostra come utilizzare`delete-group`.

AWS CLI

Come eliminare un gruppo esistente

Il `delete-group` comando seguente elimina un gruppo esistente da Amazon WorkMail.

```
aws workmail delete-group \
```

```
    --organization-id m-d281d0a2fd824be5b6cd3d3ce909fd27 \
    --group-id S-1-1-11-1122222222-2222233333-3333334444-4444
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta DeleteGroup AWS CLICommand Reference.

`delete-mailbox-permissions`

Il seguente esempio di codice mostra come utilizzare`delete-mailbox-permissions`.

AWS CLI

Come eliminare le autorizzazioni per la casella di posta

Il comando `delete-mailbox-permissions` seguente elimina le autorizzazioni per la casella di posta precedentemente concesse a un utente o a un gruppo. L'entità rappresenta l'utente proprietario della casella di posta e il beneficiario rappresenta l'utente o il gruppo per il quale eliminare le autorizzazioni.

```
aws workmail delete-mailbox-permissions \
    --organization-id m-d281d0a2fd824be5b6cd3d3ce909fd27 \
    --entity-id S-1-1-11-1122222222-2222233333-3333334444-4444 \
    --grantee-id S-1-1-11-1111111111-2222222222-3333333333-3333
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta DeleteMailboxPermissions AWS CLICommand Reference.

`delete-resource`

Il seguente esempio di codice mostra come utilizzare`delete-resource`.

AWS CLI

Come eliminare una risorsa esistente

Il `delete-resource` comando seguente elimina una risorsa esistente da Amazon WorkMail.

```
aws workmail delete-resource \
    --organization-id m-d281d0a2fd824be5b6cd3d3ce909fd27 \
```

```
    --resource-id r-7afe0efbade843a58cdc10251fce992c
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta DeleteResource AWS CLICommand Reference.

`delete-user`

Il seguente esempio di codice mostra come utilizzare`delete-user`.

AWS CLI

Come eliminare un utente

Il `delete-user` comando seguente elimina l'utente specificato da Amazon WorkMail e da tutti i sistemi successivi.

```
aws workmail delete-user \
    --organization-id m-d281d0a2fd824be5b6cd3d3ce909fd27 \
    --user-id S-1-1-11-1111111111-2222222222-3333333333-3333
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta DeleteUser AWS CLICommand Reference.

`deregister-from-work-mail`

Il seguente esempio di codice mostra come utilizzare`deregister-from-work-mail`.

AWS CLI

Come disabilitare un'entità esistente

Il `deregister-from-work-mail` comando seguente impedisce a un'entità esistente (utente, gruppo o risorsa) di utilizzare Amazon WorkMail.

```
aws workmail deregister-from-work-mail \
    --organization-id m-d281d0a2fd824be5b6cd3d3ce909fd27 \
    --entity-id S-1-1-11-1111111111-2222222222-3333333333-3333
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta DeregisterFromWorkMail AWS CLICommand Reference.

`describe-group`

Il seguente esempio di codice mostra come utilizzare`describe-group`.

AWS CLI

Come recuperare informazioni per un gruppo

Il comando `describe-group` seguente restituisce le informazioni relative al gruppo specificato.

```
aws workmail describe-group \
    --organization-id m-d281d0a2fd824be5b6cd3d3ce909fd27 \
    --group-id S-1-1-11-1122222222-2222233333-3333334444-4444
```

Output:

```
{
    "GroupId": "S-1-1-11-1122222222-2222233333-3333334444-4444",
    "Name": "exampleGroup1",
    "State": "ENABLED"
}
```

- Per i dettagli sull'API, consulta DescribeGroup AWS CLICommand Reference.

`describe-organization`

Il seguente esempio di codice mostra come utilizzare`describe-organization`.

AWS CLI

Come recuperare informazioni per un'organizzazione

Il `describe-organization` comando seguente recupera le informazioni per l' WorkMail organizzazione Amazon specificata.

```
aws workmail describe-organization \
    --organization-id m-d281d0a2fd824be5b6cd3d3ce909fd27
```

Output:

```
{
    "OrganizationId": "m-d281d0a2fd824be5b6cd3d3ce909fd27",
    "Alias": "alias",
    "State": "Active",
    "DirectoryId": "d-926726012c",
    "DirectoryType": "VpcDirectory",
    "DefaultMailDomain": "site.awsapps.com",
    "CompletedDate": 1522693605.468,
    "ARN": "arn:aws:workmail:us-west-2:111122223333:organization/m-
n1pq2345678r901st2u3vx45x6789yza"
}
```

Per ulteriori informazioni, consulta Working with Organizations nella Amazon WorkMail Administrator Guide.

- Per i dettagli sull'API, consulta DescribeOrganization AWS CLICommand Reference.

`describe-resource`

Il seguente esempio di codice mostra come utilizzare`describe-resource`.

AWS CLI

Come recuperare informazioni per una risorsa

Il comando `describe-resource` seguente restituisce le informazioni relative alla risorsa specificata.

```
aws workmail describe-resource \
    --organization-id m-d281d0a2fd824be5b6cd3d3ce909fd27 \
    --resource-id r-7afe0efbade843a58cdc10251fce992c
```

Output:

```
{
    "ResourceId": "r-7afe0efbade843a58cdc10251fce992c",
    "Name": "exampleRoom1",
    "Type": "ROOM",
    "BookingOptions": {
```

```
        "AutoAcceptRequests": true,
        "AutoDeclineRecurringRequests": false,
        "AutoDeclineConflictingRequests": true
    },
    "State": "ENABLED"
}
```

- Per i dettagli sull'API, consulta DescribeResource AWS CLICommand Reference.

`describe-user`

Il seguente esempio di codice mostra come utilizzare`describe-user`.

AWS CLI

Come recuperare informazioni sull'utente

Il comando `describe-user` seguente restituisce le informazioni relative all'utente specificato.

```
aws workmail describe-user \
    --organization-id m-d281d0a2fd824be5b6cd3d3ce909fd27 \
    --user-id S-1-1-11-1111111111-2222222222-3333333333-3333
```

Output:

```
{
    "UserId": "S-1-1-11-1111111111-2222222222-3333333333-3333",
    "Name": "exampleUser1",
    "Email": "exampleUser1@site.awsapps.com",
    "DisplayName": "",
    "State": "ENABLED",
    "UserRole": "USER",
    "EnabledDate": 1532459261.827
}
```

- Per i dettagli sull'API, consulta DescribeUser AWS CLICommand Reference.

`disassociate-delegate-from-resource`

Il seguente esempio di codice mostra come utilizzare`disassociate-delegate-from-resource`.

AWS CLI

Come rimuovere un membro da una risorsa

Il comando `disassociate-delegate-from-resource` seguente rimuove il membro specificato da una risorsa.

```
ws workmail disassociate-delegate-from-resource \
    --organization-id m-d281d0a2fd824be5b6cd3d3ce909fd27 \
    --resource-id r-68bf2d3b1c0244aab7264c24b9217443 \
    --entity-id S-1-1-11-1111111111-2222222222-3333333333-3333
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta DisassociateDelegateFromResource AWS CLICommand Reference.

`disassociate-member-from-group`

Il seguente esempio di codice mostra come utilizzare`disassociate-member-from-group`.

AWS CLI

Come rimuovere un membro da un gruppo

Il comando `disassociate-member-from-group` seguente rimuove il membro specificato da un gruppo.

```
aws workmail disassociate-member-from-group \
    --organization-id m-d281d0a2fd824be5b6cd3d3ce909fd27 \
    --group-id S-1-1-11-1122222222-2222233333-3333334444-4444 \
    --member-id S-1-1-11-1111111111-2222222222-3333333333-3333
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta DisassociateMemberFromGroup AWS CLICommand Reference.

`get-access-control-effect`

Il seguente esempio di codice mostra come utilizzare`get-access-control-effect`.

AWS CLI

Come ottenere l'effetto delle regole di controllo degli accessi

L'`get-access-control-effect`esempio seguente recupera l'effetto delle regole di controllo degli accessi WorkMail dell'organizzazione Amazon specificata per l'indirizzo IP, l'azione del protocollo di accesso e l'ID utente specificati.

```
aws workmail get-access-control-effect \
    --organization-id m-n1pq2345678r901st2u3vx45x6789yza \
    --ip-address "192.0.2.0" \
    --action "WindowsOutlook" \
    --user-id "S-1-1-11-1111111111-2222222222-3333333333-3333"
```

Output:

```
{
    "Effect": "DENY",
    "MatchedRules": [
        "myRule"
    ]
}
```

Per ulteriori informazioni, consulta Working with Access Control Rules nella Amazon WorkMail Administrator Guide.

- Per i dettagli sull'API, consulta GetAccessControlEffect AWS CLICommand Reference.

`get-mailbox-details`

Il seguente esempio di codice mostra come utilizzare`get-mailbox-details`.

AWS CLI

Come ottenere i dettagli sulla casella di posta di un utente

Il comando `get-mailbox-details` seguente recupera i dettagli sulla casella di posta dell'utente specificato.

```
aws workmail get-mailbox-details \
```

```
    --organization-id m-n1pq2345678r901st2u3vx45x6789yza \
    --user-id S-1-1-11-1111111111-2222222222-3333333333-3333
```

Output:

```
{
    "MailboxQuota": 51200,
    "MailboxSize": 0.03890800476074219
}
```

Per ulteriori informazioni, consulta Managing User Accounts nella Amazon WorkMail Administrator Guide.

- Per i dettagli sull'API, consulta GetMailboxDetails AWS CLICommand Reference.

`list-access-control-rules`

Il seguente esempio di codice mostra come utilizzare`list-access-control-rules`.

AWS CLI

Come elencare le regole di controllo degli accessi

L'`list-access-control-rules`esempio seguente elenca le regole di controllo degli accessi per l' WorkMail organizzazione Amazon specificata.

```
aws workmail list-access-control-rules \
    --organization-id m-n1pq2345678r901st2u3vx45x6789yza
```

Output:

```
{
    "Rules": [
        {
            "Name": "default",
            "Effect": "ALLOW",
            "Description": "Default WorkMail Rule",
            "DateCreated": 0.0,
            "DateModified": 0.0
        },
        {
```

```
            "Name": "myRule",
            "Effect": "DENY",
            "Description": "my rule",
            "UserIds": [
            "S-1-1-11-1111111111-2222222222-3333333333-3333"
            ],
            "DateCreated": 1581635628.0,
            "DateModified": 1581635628.0
        }
    ]
}
```

Per ulteriori informazioni, consulta Working with Access Control Rules nella Amazon WorkMail Administrator Guide.

- Per i dettagli sull'API, consulta ListAccessControlRules AWS CLICommand Reference.

`list-aliases`

Il seguente esempio di codice mostra come utilizzare`list-aliases`.

AWS CLI

Come elencare gli alias di un membro

Il comando `list-aliases` seguente elenca gli alias per il membro specificato (utente o gruppo).

```
aws workmail list-aliases \
    --organization-id m-d281d0a2fd824be5b6cd3d3ce909fd27 \
    --entity-id S-1-1-11-1111111111-2222222222-3333333333-3333
```

Output:

```
{
    "Aliases": [
        "exampleAlias@site.awsapps.com",
        "exampleAlias1@site.awsapps.com"
    ]
}
```

- Per i dettagli sull'API, consulta ListAliases AWS CLICommand Reference.

`list-group-members`

Il seguente esempio di codice mostra come utilizzare`list-group-members`.

AWS CLI

Come elencare i membri del gruppo

Il comando `list-group-members` seguente elenca i membri del gruppo specificato.

```
aws workmail list-group-members \
    --organization-id m-d281d0a2fd824be5b6cd3d3ce909fd27 \
    --group-id S-1-1-11-1122222222-2222233333-3333334444-4444
```

Output:

```
{
    "Members": [
        {
            "Id": "S-1-1-11-1111111111-2222222222-3333333333-3333",
            "Name": "exampleUser1",
            "Type": "USER",
            "State": "ENABLED",
            "EnabledDate": 1532459261.827
        }
    ]
}
```

- Per i dettagli sull'API, consulta ListGroupMembers AWS CLICommand Reference.

`list-groups`

Il seguente esempio di codice mostra come utilizzare`list-groups`.

AWS CLI

Come recuperare un elenco di gruppi

Il comando `list-groups` seguente recupera i riepiloghi dei gruppi nell'organizzazione specificata.

```
aws workmail list-groups \
```

```
    --organization-id m-d281d0a2fd824be5b6cd3d3ce909fd27
```

Output:

```
{
    "Groups": [
        {
            "Id": "S-1-1-11-1122222222-2222233333-3333334444-4444",
            "Name": "exampleGroup1",
            "State": "DISABLED"
        },
        {
            "Id": "S-4-4-44-1122222222-2222233333-3333334444-4444",
            "Name": "exampleGroup2",
            "State": "ENABLED"
        }
    ]
}
```

- Per i dettagli sull'API, consulta ListGroups AWS CLICommand Reference.

`list-mailbox-permissions`

Il seguente esempio di codice mostra come utilizzare`list-mailbox-permissions`.

AWS CLI

Come recuperare le autorizzazioni per la casella di posta

Il comando `list-mailbox-permissions` seguente recupera le autorizzazioni per la casella di posta associate alla casella di posta dell'entità specificata.

```
aws workmail list-mailbox-permissions \
    --organization-id m-d281d0a2fd824be5b6cd3d3ce909fd27 \
    --entity-id S-1-1-11-1111111111-2222222222-3333333333-3333
```

Output:

```
{
    "Permissions": [
        {
```

```
            "GranteeId": "S-1-1-11-1122222222-2222233333-3333334444-4444",
            "GranteeType": "USER",
            "PermissionValues": [
                "FULL_ACCESS"
            ]
        }
    ]
}
```

- Per i dettagli sull'API, consulta ListMailboxPermissions AWS CLICommand Reference.

`list-organizations`

Il seguente esempio di codice mostra come utilizzare`list-organizations`.

AWS CLI

Come recuperare un elenco di organizzazioni

Il comando `list-organizations` seguente recupera i riepiloghi delle organizzazioni del cliente.

```
aws workmail list-organizations
```

Output:

```
{
    "OrganizationSummaries": [
        {
            "OrganizationId": "m-d281d0a2fd824be5b6cd3d3ce909fd27",
            "Alias": "exampleAlias",
            "State": "Active"
        }
    ]
}
```

- Per i dettagli sull'API, consulta ListOrganizations AWS CLICommand Reference.

`list-resource-delegates`

Il seguente esempio di codice mostra come utilizzare`list-resource-delegates`.

AWS CLI

Come elencare i delegati di una risorsa

Il comando `list-resource-delegates` seguente recupera l'elenco dei tag associati alla risorsa specificata.

```
aws workmail list-resource-delegates \
    --organization-id m-d281d0a2fd824be5b6cd3d3ce909fd27 \
    --resource-id r-68bf2d3b1c0244aab7264c24b9217443
```

Output:

```
{
    "Delegates": [
        {
            "Id": "S-1-1-11-1111111111-2222222222-3333333333-3333",
            "Type": "USER"
        }
    ]
}
```

- Per i dettagli sull'API, consulta ListResourceDelegates AWS CLICommand Reference.

`list-resources`

Il seguente esempio di codice mostra come utilizzare`list-resources`.

AWS CLI

Come recuperare un elenco di risorse

Il comando `list-resources` seguente recupera i riepiloghi delle risorse per l'organizzazione specificata.

```
aws workmail list-resources \
    --organization-id m-d281d0a2fd824be5b6cd3d3ce909fd27
```

Output:

```
{
```

```
    "Resources": [
        {
            "Id": "r-7afe0efbade843a58cdc10251fce992c",
            "Name": "exampleRoom1",
            "Type": "ROOM",
            "State": "ENABLED"
        }
    ]
}
```

- Per i dettagli sull'API, consulta ListResources AWS CLICommand Reference.

`list-tags-for-resource`

Il seguente esempio di codice mostra come utilizzare`list-tags-for-resource`.

AWS CLI

Come elencare i tag per una risorsa

L'`list-tags-for-resource`esempio seguente elenca i tag per l' WorkMail organizzazione Amazon specificata.

```
aws workmail list-tags-for-resource \
    --resource-arn arn:aws:workmail:us-west-2:111122223333:organization/m-n1pq2345678r901st2u3vx45x6789yza
```

Output:

```
{
    "Tags": [
        {
            "Key": "priority",
            "Value": "1"
        }
    ]
}
```

Per ulteriori informazioni, consulta Tagging an Organization nella Amazon WorkMail Administrator Guide.

- Per i dettagli sull'API, consulta ListTagsForResource AWS CLICommand Reference.

`list-users`

Il seguente esempio di codice mostra come utilizzare`list-users`.

AWS CLI

Come recuperare un elenco di utenti

Il comando `list-users` seguente recupera i riepiloghi degli utenti nell'organizzazione specificata.

```
aws workmail list-users \
    --organization-id m-d281d0a2fd824be5b6cd3d3ce909fd27
```

Output:

```
{
    "Users": [
        {
            "Id": "S-1-1-11-1111111111-2222222222-3333333333-3333",
            "Email": "exampleUser1@site.awsapps.com",
            "Name": "exampleUser1",
            "State": "ENABLED",
            "UserRole": "USER",
            "EnabledDate": 1532459261.827
        },
        {
            "Id": "S-1-1-11-1122222222-2222233333-3333334444-4444",
            "Name": "exampleGuestUser",
            "State": "DISABLED",
            "UserRole": "SYSTEM_USER"
        }
    ]
}
```

- Per i dettagli sull'API, consulta ListUsers AWS CLICommand Reference.

`put-access-control-rule`

Il seguente esempio di codice mostra come utilizzare`put-access-control-rule`.

AWS CLI

Come inserire una nuova regola di controllo degli accessi

L'`put-access-control-rule`esempio seguente nega all'utente specificato l'accesso all' WorkMail organizzazione Amazon specificata.

```
aws workmail put-access-control-rule \
    --name "myRule" \
    --effect "DENY" \
    --description "my rule" \
    --user-ids "S-1-1-11-1111111111-2222222222-3333333333-3333" \
    --organization-id m-n1pq2345678r901st2u3vx45x6789yza
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Working with Access Control Rules nella Amazon WorkMail Administrator Guide.

- Per i dettagli sull'API, consulta PutAccessControlRule AWS CLICommand Reference.

`put-mailbox-permissions`

Il seguente esempio di codice mostra come utilizzare`put-mailbox-permissions`.

AWS CLI

Come modificare le autorizzazioni per la casella di posta

Il comando `put-mailbox-permissions` seguente imposta le autorizzazioni di accesso complete per il beneficiario specificato (utente o gruppo). L'entità rappresenta il proprietario della casella di posta.

```
aws workmail put-mailbox-permissions \
    --organization-id m-d281d0a2fd824be5b6cd3d3ce909fd27 \
    --entity-id S-1-1-11-1111111111-2222222222-3333333333-3333 \
    --grantee-id S-1-1-11-1122222222-2222233333-3333334444-4444 \
    --permission-values FULL_ACCESS
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta PutMailboxPermissions AWS CLICommand Reference.

`register-to-work-mail`

Il seguente esempio di codice mostra come utilizzare`register-to-work-mail`.

AWS CLI

Come registrare un'entità esistente o disabilitata

Il `register-to-work-mail` comando seguente consente all'entità esistente specificata (utente, gruppo o risorsa) di utilizzare Amazon WorkMail.

```
aws workmail register-to-work-mail \
    --organization-id m-d281d0a2fd824be5b6cd3d3ce909fd27 \
    --entity-id S-1-1-11-1122222222-2222233333-3333334444-4444 \
    --email exampleGroup1@site.awsapps.com
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta RegisterToWorkMail AWS CLICommand Reference.

`reset-password`

Il seguente esempio di codice mostra come utilizzare`reset-password`.

AWS CLI

Come reimpostare una password utente

Il comando `reset-password` seguente reimposta la password per l'utente specificato.

```
aws workmail reset-password \
    --organization-id m-d281d0a2fd824be5b6cd3d3ce909fd27 \
    --user-id S-1-1-11-1111111111-2222222222-3333333333-3333 \
    --password examplePa$$w0rd
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta ResetPassword AWS CLICommand Reference.

`tag-resource`

Il seguente esempio di codice mostra come utilizzare`tag-resource`.

AWS CLI

Come aggiungere un tag a una risorsa

L'`tag-resource`esempio seguente applica un tag con chiave «priority» e valore «1" all' WorkMail organizzazione Amazon specificata.

```
aws workmail tag-resource \
    --resource-arn arn:aws:workmail:us-west-2:111122223333:organization/m-n1pq2345678r901st2u3vx45x6789yza \
    --tags "Key=priority,Value=1"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Tagging an Organization nella Amazon WorkMail Administrator Guide.

- Per i dettagli sull'API, consulta TagResource AWS CLICommand Reference.

`untag-resource`

Il seguente esempio di codice mostra come utilizzare`untag-resource`.

AWS CLI

Come rimuovere un tag da una risorsa

L'`untag-resource`esempio seguente rimuove il tag specificato dall' WorkMail organizzazione Amazon specificata.

```
aws workmail untag-resource \
    --resource-arn arn:aws:workmail:us-west-2:111122223333:organization/m-n1pq2345678r901st2u3vx45x6789yza \
    --tag-keys "priority"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Tagging an Organization nella Amazon WorkMail Administrator Guide.

- Per i dettagli sull'API, consulta UntagResource AWS CLICommand Reference.

`update-mailbox-quota`

Il seguente esempio di codice mostra come utilizzare`update-mailbox-quota`.

AWS CLI

Come aggiornare la quota della casella di posta di un utente

Il comando `update-mailbox-quota` seguente modifica la quota della casella di posta dell'utente specificato.

```
aws workmail update-mailbox-quota \
    --organization-id m-n1pq2345678r901st2u3vx45x6789yza \
    --user-id S-1-1-11-1111111111-2222222222-3333333333-3333 \
    --mailbox-quota 40000
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Managing User Accounts nella Amazon WorkMail Administrator Guide.

- Per i dettagli sull'API, consulta UpdateMailboxQuota AWS CLICommand Reference.

`update-primary-email-address`

Il seguente esempio di codice mostra come utilizzare`update-primary-email-address`.

AWS CLI

Come aggiornare un indirizzo e-mail principale

Il comando `update-primary-email-address` seguente aggiorna l'indirizzo e-mail principale dell'entità specificata (utente, gruppo o risorsa).

```
aws workmail update-primary-email-address \
    --organization-id m-d281d0a2fd824be5b6cd3d3ce909fd27 \
    --entity-id S-1-1-11-1111111111-2222222222-3333333333-3333 \
    --email exampleUser2@site.awsapps.com
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta UpdatePrimaryEmailAddress AWS CLICommand Reference.

`update-resource`

Il seguente esempio di codice mostra come utilizzare`update-resource`.

AWS CLI

Come aggiornare una risorsa

Il comando `update-resource` seguente aggiorna il nome della risorsa specificata.

```
aws workmail update-resource \
    --organization-id m-d281d0a2fd824be5b6cd3d3ce909fd27 \
    --resource-id r-7afe0efbade843a58cdc10251fce992c \
    --name exampleRoom2
```

Questo comando non produce alcun output.

- Per i dettagli sull'API, consulta UpdateResource AWS CLICommand Reference.

Esempi di Amazon WorkMail Message Flow con AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando Amazon WorkMail Message Flow. AWS Command Line Interface

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`get-raw-message-content`

Il seguente esempio di codice mostra come usare`get-raw-message-content`.

AWS CLI

Come ottenere il contenuto non elaborato di un messaggio e-mail

L'esempio `get-raw-message-content` seguente ottiene il contenuto non elaborato di un messaggio e-mail in transito e lo invia a un file di testo denominato `test`.

```
aws workmailmessageflow get-raw-message-content \
    --message-id a1b2cd34-ef5g-6h7j-kl8m-npq9012345rs \
    test
```

Contenuto del file `test` dopo l'esecuzione del comando:

```
Subject: Hello World
From: =?UTF-8?Q?marymajor_marymajor?= <marymajor@example.com>
To: =?UTF-8?Q?mateojackson=40example=2Enet?= <mateojackson@example.net>
Date: Thu, 7 Nov 2019 19:22:46 +0000
Mime-Version: 1.0
Content-Type: multipart/alternative;
 boundary="=_EXAMPLE+"
References: <mail.1ab23c45.5de6.7f890g123hj45678@storage.wm.amazon.com>
X-Priority: 3 (Normal)
X-Mailer: Amazon WorkMail
Thread-Index: EXAMPLE
Thread-Topic: Hello World
Message-Id: <mail.1ab23c45.5de6.7f890g123hj45678@storage.wm.amazon.com>

This is a multi-part message in MIME format. Your mail reader does not
understand MIME message format.
--=_EXAMPLE+
Content-Type: text/plain; charset=UTF-8
Content-Transfer-Encoding: 7bit

hello world


--=_EXAMPLE+
Content-Type: text/html; charset=utf-8
Content-Transfer-Encoding: quoted-printable

<!DOCTYPE HTML><html>
<head>
<meta name=3D"Generator" content=3D"Amazon WorkMail v3.0-4510">
```

```
<meta http-equiv=3D"Content-Type" content=3D"text/html; charset=3Dutf-8">=

<title>testing</title>
</head>
<body>
<p style=3D"margin: 0px; font-family: Arial, Tahoma, Helvetica, sans-seri=
f; font-size: small;">hello world</p>
</body>
</html>
--=_EXAMPLE+--
```

Per ulteriori informazioni, consulta Recupero del contenuto dei messaggi con AWS Lambda nella Amazon Administrator Guide. WorkMail

- Per i dettagli sull'API, consulta Command GetRawMessageContentReference AWS CLI .

WorkSpaces esempi utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando AWS Command Line Interface with WorkSpaces.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`create-tags`

Il seguente esempio di codice mostra come utilizzare`create-tags`.

AWS CLI

Per aggiungere tag a un WorkSpace

L'`create-tags`esempio seguente aggiunge i tag specificati a quelli specificati WorkSpace.

```
aws workspaces create-tags \
    --resource-id ws-dk1xzr417 \
    --tags Key=Department,Value=Finance
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Tag WorkSpaces resources nella Amazon WorkSpaces Administration Guide.

- Per i dettagli sull'API, consulta CreateTags AWS CLICommand Reference.

`create-workspaces`

Il seguente esempio di codice mostra come utilizzare`create-workspaces`.

AWS CLI

Esempio 1: creare un AlwaysOn WorkSpace

L'`create-workspaces`esempio seguente crea un file AlwaysOn WorkSpace per l'utente specificato, utilizzando la directory e il pacchetto specificati.

```
aws workspaces create-workspaces \
    --workspaces DirectoryId=d-926722edaf,UserName=Mateo,BundleId=wsb-0zsvgp8fc
```

Output:

```
{
    "FailedRequests": [],
    "PendingRequests": [
        {
            "WorkspaceId": "ws-kcqms853t",
            "DirectoryId": "d-926722edaf",
            "UserName": "Mateo",
            "State": "PENDING",
            "BundleId": "wsb-0zsvgp8fc"
        }
    ]
}
```

Esempio 2: creare un AutoStop WorkSpace

L'`create-workspaces`esempio seguente crea un file AutoStop WorkSpace per l'utente specificato, utilizzando la directory e il pacchetto specificati.

```
aws workspaces create-workspaces \
    --
workspaces DirectoryId=d-926722edaf,UserName=Mary,BundleId=wsb-0zsvgp8fc,WorkspaceProperties
```

Output:

```
{
    "FailedRequests": [],
    "PendingRequests": [
        {
            "WorkspaceId": "ws-dk1xzr417",
            "DirectoryId": "d-926722edaf",
            "UserName": "Mary",
            "State": "PENDING",
            "BundleId": "wsb-0zsvgp8fc"
        }
    ]
}
```

Esempio 3: creare un file disaccoppiato dall'utente WorkSpace

L'`create-workspaces`esempio seguente crea un utente disaccoppiato WorkSpace impostando il nome utente su e specificando un nome`[UNDEFINED]`, un ID di directory e un ID di WorkSpace pacchetto.

```
aws workspaces create-workspaces \
    --workspaces
 DirectoryId=d-926722edaf,UserName='"[UNDEFINED]"',WorkspaceName=MaryWorkspace1,BundleId=wsb
```

Output:

```
{
    "FailedRequests": [],
    "PendingRequests": [
        {
            "WorkspaceId": "ws-abcd1234",
```

```
            "DirectoryId": "d-926722edaf",
            "UserName": "[UNDEFINED]",
            "State": "PENDING",
            "BundleId": "wsb-0zsvgp8fc",
            "WorkspaceName": "MaryWorkspace1"
        }
    ]
}
```

Per ulteriori informazioni, consulta Launch a virtual desktop nella Amazon WorkSpaces Administration Guide.

- Per i dettagli sull'API, consulta CreateWorkspaces AWS CLICommand Reference.

`delete-tags`

Il seguente esempio di codice mostra come utilizzare`delete-tags`.

AWS CLI

Per eliminare un tag da un WorkSpace

L'`delete-tags`esempio seguente elimina il tag specificato dal tag specificato WorkSpace.

```
aws workspaces delete-tags \
    --resource-id ws-dk1xzr417 \
    --tag-keys Department
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Tag WorkSpaces resources nella Amazon WorkSpaces Administration Guide.

- Per i dettagli sull'API, consulta DeleteTags AWS CLICommand Reference.

`deregister-workspace-directory`

Il seguente esempio di codice mostra come utilizzare`deregister-workspace-directory`.

AWS CLI

Come annullare la registrazione di una directory

L'esempio `deregister-workspace-directory` seguente annulla la registrazione della directory specificata.

```
aws workspaces deregister-workspace-directory \
    --directory-id d-926722edaf
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Registrare una directory con WorkSpaces nella Amazon WorkSpaces Administration Guide.

- Per i dettagli sull'API, consulta DeregisterWorkspaceDirectory AWS CLICommand Reference.

`describe-tags`

Il seguente esempio di codice mostra come utilizzare`describe-tags`.

AWS CLI

Per descrivere i tag per un WorkSpace

L'`describe-tags`esempio seguente descrive i tag per il valore specificato WorkSpace.

```
aws workspaces describe-tags \
    --resource-id ws-dk1xzr417
```

Output:

```
{
    "TagList": [
        {
            "Key": "Department",
            "Value": "Finance"
        }
    ]
}
```

Per ulteriori informazioni, consulta Tag WorkSpaces resources nella Amazon WorkSpaces Administration Guide.

- Per i dettagli sull'API, consulta DescribeTags AWS CLICommand Reference.

`describe-workspace-bundles`

Il seguente esempio di codice mostra come utilizzare`describe-workspace-bundles`.

AWS CLI

Come elencare i bundle forniti da Amazon

L'`describe-workspace-bundles`esempio seguente elenca i nomi e IDs i pacchetti forniti da Amazon, in formato tabella e ordinati per nome.

```
aws workspaces describe-workspace-bundles \
    --owner AMAZON \
    --query "Bundles[*].[Name, BundleId]"
```

Output:

```
[
    [
        "Standard with Amazon Linux 2",
        "wsb-clj85qzj1"
    ],
    [
        "Performance with Windows 10 (Server 2016 based)",
        "wsb-gm4d5tx2v"
    ],
    [
        "PowerPro with Windows 7",
        "wsb-1pzkp0bx4"
    ],
    [
        "Power with Amazon Linux 2",
        "wsb-2bs6k5lgn"
    ],
    [
        "Graphics with Windows 10 (Server 2019 based)",
        "wsb-03gyjnfyy"
    ],
    ...
]
```

Per ulteriori informazioni, consulta i WorkSpaces pacchetti e le immagini nella Amazon WorkSpaces Administration Guide.

- Per i dettagli sull'API, consulta DescribeWorkspaceBundles AWS CLICommand Reference.

`describe-workspace-directories`

Il seguente esempio di codice mostra come utilizzare`describe-workspace-directories`.

AWS CLI

Come descrivere una directory registrata

L'esempio `describe-workspace-directories` seguente descrive la directory registrata specificata.

```
aws workspaces describe-workspace-directories \
    --directory-ids d-926722edaf
```

Output:

```
{
    "Directories": [
        {
            "DirectoryId": "d-926722edaf",
            "Alias": "d-926722edaf",
            "DirectoryName": "example.com",
            "RegistrationCode": "WSpdx+9RJ8JT",
            "SubnetIds": [
                "subnet-9d19c4c6",
                "subnet-500d5819"
            ],
            "DnsIpAddresses": [
                "172.16.1.140",
                "172.16.0.30"
            ],
            "CustomerUserName": "Administrator",
            "IamRoleId": "arn:aws:iam::123456789012:role/workspaces_DefaultRole",
            "DirectoryType": "SIMPLE_AD",
            "WorkspaceSecurityGroupId": "sg-0d89e927e5645d7c5",
            "State": "REGISTERED",
            "WorkspaceCreationProperties": {
                "EnableInternetAccess": false,
                "UserEnabledAsLocalAdministrator": true,
                "EnableMaintenanceMode": true
```

```
            },
            "WorkspaceAccessProperties": {
                "DeviceTypeWindows": "ALLOW",
                "DeviceTypeOsx": "ALLOW",
                "DeviceTypeWeb": "DENY",
                "DeviceTypeIos": "ALLOW",
                "DeviceTypeAndroid": "ALLOW",
                "DeviceTypeChromeOs": "ALLOW",
                "DeviceTypeZeroClient": "ALLOW",
                "DeviceTypeLinux": "DENY"
            },
            "Tenancy": "SHARED",
            "SelfservicePermissions": {
                "RestartWorkspace": "ENABLED",
                "IncreaseVolumeSize": "DISABLED",
                "ChangeComputeType": "DISABLED",
                "SwitchRunningMode": "DISABLED",
                "RebuildWorkspace": "DISABLED"
            }
        }
    ]
}
```

Per ulteriori informazioni, consulta Manage directories for WorkSpaces Personal nella Amazon WorkSpaces Administration Guide.

- Per i dettagli sull'API, consulta DescribeWorkspaceDirectories AWS CLICommand Reference.

`describe-workspaces-connection-status`

Il seguente esempio di codice mostra come utilizzare`describe-workspaces-connection-status`.

AWS CLI

Per descrivere lo stato della connessione di un WorkSpace

L'`describe-workspaces-connection-status`esempio seguente descrive lo stato della connessione dell'oggetto specificato WorkSpace.

```
aws workspaces describe-workspaces-connection-status \
    --workspace-ids ws-dk1xzr417
```

Output:

```
{
    "WorkspacesConnectionStatus": [
        {
            "WorkspaceId": "ws-dk1xzr417",
            "ConnectionState": "CONNECTED",
            "ConnectionStateCheckTimestamp": 1662526214.744
        }
    ]
}
```

Per ulteriori informazioni, consulta Administer your WorkSpaces nella Amazon WorkSpaces Administration Guide.

- Per i dettagli sull'API, consulta DescribeWorkspacesConnectionStatus AWS CLICommand Reference.

`describe-workspaces`

Il seguente esempio di codice mostra come utilizzare`describe-workspaces`.

AWS CLI

Per descrivere un WorkSpace

L'`describe-workspaces`esempio seguente descrive quanto specificato WorkSpace.

```
aws workspaces describe-workspaces \
    --workspace-ids ws-dk1xzr417
```

Output:

```
{
    "Workspaces": [
        {
            "WorkspaceId": "ws-dk1xzr417",
            "DirectoryId": "d-926722edaf",
            "UserName": "Mary",
            "IpAddress": "172.16.0.175",
            "State": "STOPPED",
```

```
            "BundleId": "wsb-0zsvgp8fc",
            "SubnetId": "subnet-500d5819",
            "ComputerName": "WSAMZN-RBSLTTD9",
            "WorkspaceProperties": {
                "RunningMode": "AUTO_STOP",
                "RunningModeAutoStopTimeoutInMinutes": 60,
                "RootVolumeSizeGib": 80,
                "UserVolumeSizeGib": 10,
                "ComputeTypeName": "VALUE"
            },
            "ModificationStates": []
        }
    ]
}
```

Per ulteriori informazioni, consulta Administer your WorkSpaces nella Amazon WorkSpaces Administration Guide.

- Per i dettagli sull'API, consulta DescribeWorkspaces AWS CLICommand Reference.

`migrate-workspace`

Il seguente esempio di codice mostra come utilizzare`migrate-workspace`.

AWS CLI

Per migrare un WorkSpace

L'`migrate-workspace`esempio seguente migra il pacchetto specificato nel WorkSpace pacchetto specificato.

```
aws workspaces migrate-workspace \
    --source-workspace-id ws-dk1xzr417 \
    --bundle-id wsb-j4dky1gs4
```

Output:

```
{
    "SourceWorkspaceId": "ws-dk1xzr417",
    "TargetWorkspaceId": "ws-x5h1lbkp5"
}
```

Per ulteriori informazioni, consulta Migrate a WorkSpace nella Amazon WorkSpaces Administration Guide.

- Per i dettagli sull'API, consulta MigrateWorkspace AWS CLICommand Reference.

`modify-workspace-creation-properties`

Il seguente esempio di codice mostra come utilizzare`modify-workspace-creation-properties`.

AWS CLI

Per modificare una proprietà di WorkSpace creazione di una directory

L'esempio `modify-workspace-creation-properties` seguente abilita la proprietà `EnableInternetAccess` per la directory specificata. Ciò consente l'assegnazione automatica degli indirizzi IP pubblici per la directory WorkSpaces creata.

```
aws workspaces modify-workspace-creation-properties \
    --resource-id d-926722edaf \
    --workspace-creation-properties EnableInternetAccess=true
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta la sezione Update directory details for your WorkSpaces nella Amazon WorkSpaces Administration Guide.

- Per i dettagli sull'API, consulta ModifyWorkspaceCreationProperties AWS CLICommand Reference.

`modify-workspace-properties`

Il seguente esempio di codice mostra come utilizzare`modify-workspace-properties`.

AWS CLI

Per modificare la modalità di esecuzione di un WorkSpace

L'`modify-workspace-properties`esempio seguente imposta la modalità di esecuzione del file specificato WorkSpace su`AUTO_STOP`.

```
aws workspaces modify-workspace-properties \
```

```
    --workspace-id ws-dk1xzr417 \
    --workspace-properties RunningMode=AUTO_STOP
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Modify a WorkSpace nella Amazon WorkSpaces Administration Guide.

- Per i dettagli sull'API, consulta ModifyWorkspaceProperties AWS CLICommand Reference.

`modify-workspace-state`

Il seguente esempio di codice mostra come utilizzare`modify-workspace-state`.

AWS CLI

Per modificare lo stato di un WorkSpace

L'`modify-workspace-state`esempio seguente imposta lo stato del specificato WorkSpace su`ADMIN_MAINTENANCE`.

```
aws workspaces modify-workspace-state \
    --workspace-id ws-dk1xzr417 \
    --workspace-state ADMIN_MAINTENANCE
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta la sezione WorkSpace manutenzione nella Amazon WorkSpaces Administration Guide.

- Per i dettagli sull'API, consulta ModifyWorkspaceState AWS CLICommand Reference.

`reboot-workspaces`

Il seguente esempio di codice mostra come utilizzare`reboot-workspaces`.

AWS CLI

Per riavviare un WorkSpace

L'`reboot-workspaces`esempio seguente riavvia il file specificato. WorkSpace

```
aws workspaces reboot-workspaces \
    --reboot-workspace-requests ws-dk1xzr417
```

Output:

```
{
    "FailedRequests": []
}
```

Per ulteriori informazioni, consulta Reboot a WorkSpace nella Amazon WorkSpaces Administration Guide.

- Per i dettagli sull'API, consulta RebootWorkspaces AWS CLICommand Reference.

`rebuild-workspaces`

Il seguente esempio di codice mostra come utilizzare`rebuild-workspaces`.

AWS CLI

Per ricostruire un WorkSpace

L'`rebuild-workspaces`esempio seguente ricostruisce quanto specificato. WorkSpace

```
aws workspaces rebuild-workspaces \
    --rebuild-workspace-requests ws-dk1xzr417
```

Output:

```
{
    "FailedRequests": []
}
```

Per ulteriori informazioni, consulta Rebuild a WorkSpace nella Amazon WorkSpaces Administration Guide.

- Per i dettagli sull'API, consulta RebuildWorkspaces AWS CLICommand Reference.

`register-workspace-directory`

Il seguente esempio di codice mostra come utilizzare`register-workspace-directory`.

AWS CLI

Come registrare una directory

L'`register-workspace-directory`esempio seguente registra la directory specificata per l'utilizzo con Amazon WorkSpaces.

```
aws workspaces register-workspace-directory \
    --directory-id d-926722edaf
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Registrare una AWS directory esistente di Directory Service con WorkSpaces Personal nella Amazon WorkSpaces Administration Guide.

- Per i dettagli sull'API, consulta RegisterWorkspaceDirectory AWS CLICommand Reference.

`restore-workspace`

Il seguente esempio di codice mostra come utilizzare`restore-workspace`.

AWS CLI

Per ripristinare un WorkSpace

L'`restore-workspace`esempio seguente ripristina il valore specificato WorkSpace.

```
aws workspaces restore-workspace \
    --workspace-id ws-dk1xzr417
```

Questo comando non produce alcun output.

Per ulteriori informazioni, consulta Restore a WorkSpace nella Amazon WorkSpaces Administration Guide.

- Per i dettagli sull'API, consulta RestoreWorkspace AWS CLICommand Reference.

`start-workspaces`

Il seguente esempio di codice mostra come utilizzare`start-workspaces`.

AWS CLI

Per avviare un AutoStop WorkSpace

L'`start-workspaces`esempio seguente avvia il valore specificato WorkSpace. WorkSpace Deve avere una modalità di esecuzione di`AutoStop`.

```
aws workspaces start-workspaces \
    --start-workspace-requests WorkspaceId=ws-dk1xzr417
```

Output:

```
{
    "FailedRequests": []
}
```

Per ulteriori informazioni, consulta Stop and start an AutoStop WorkSpace nella Amazon WorkSpaces Administration Guide.

- Per i dettagli sull'API, consulta StartWorkspaces AWS CLICommand Reference.

`stop-workspaces`

Il seguente esempio di codice mostra come utilizzare`stop-workspaces`.

AWS CLI

Per fermare un AutoStop WorkSpace

L'`stop-workspaces`esempio seguente interrompe il valore specificato WorkSpace. WorkSpace Deve avere una modalità di esecuzione di`AutoStop`.

```
aws workspaces stop-workspaces \
    --stop-workspace-requests WorkspaceId=ws-dk1xzr417
```

Output:

```
{
    "FailedRequests": []
}
```

Per ulteriori informazioni, consulta Stop and start an AutoStop WorkSpace nella Amazon WorkSpaces Administration Guide.

- Per i dettagli sull'API, consulta StopWorkspaces AWS CLICommand Reference.

`terminate-workspaces`

Il seguente esempio di codice mostra come utilizzare`terminate-workspaces`.

AWS CLI

Per terminare un WorkSpace

L'esempio `terminate-workspaces` seguente termina lo spazio di lavoro specificato.

```
aws workspaces terminate-workspaces \
    --terminate-workspace-requests ws-dk1xzr417
```

Output:

```
{
    "FailedRequests": []
}
```

Per ulteriori informazioni, consulta Delete a WorkSpace nella Amazon WorkSpaces Administration Guide.

- Per i dettagli sull'API, consulta TerminateWorkspaces AWS CLICommand Reference.

Esempi di raggi X utilizzando AWS CLI

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando il AWS Command Line Interface con X-Ray.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`batch-traces-get`

Il seguente esempio di codice mostra come utilizzare`batch-traces-get`.

AWS CLI

Come ottenere un elenco di tracce

L'esempio `batch-get-traces` seguente recupera un elenco di tracce specificate da un ID. Il tracciamento completo include un documento per ogni segmento, creato a partire da tutti i documenti di segmento con lo stesso ID di tracciamento ricevuti.

```
aws xray batch-get-traces \
    --trace-ids 1-5d82881a-0a9126e92a73e971eed891b9
```

Output:

```
{
    "Traces": [
        {
            "Id": "1-5d82881a-0a9126e92a73e971eed891b9",
            "Duration": 0.232,
            "Segments": [
                {
                    "Id": "54aff5735b12dd28",
                    "Document": "{\"id\":\"54aff5735b12dd28\",\"name\":
\"Scorekeep\",\"start_time\":1.568835610432E9,\"end_time\":1.568835610664E9,
\"http\":{\"request\":{\"url\":\"http://scorekeep-env-1.m4fg2pfzpv.us-
east-2.elasticbeanstalk.com/api/user\",\"method\":\"POST\",\"user_agent\":
\"curl/7.59.0\",\"client_ip\":\"52.95.4.28\",\"x_forwarded_for\":true},
\"response\":{\"status\":200}},\"aws\":{\"elastic_beanstalk\":{\"version_label
\":\"Sample Application-1\",\"deployment_id\":3,\"environment_name\":\"Scorekeep-
env-1\"},\"ec2\":{\"availability_zone\":\"us-east-2b\",\"instance_id\":
\"i-0e3cf4d2de0f3f37a\"},\"xray\":{\"sdk_version\":\"1.1.0\",\"sdk\":\"X-Ray for
 Java\"}},\"service\":{\"runtime\":\"OpenJDK 64-Bit Server VM\",\"runtime_version
\":\"1.8.0_222\"},\"trace_id\":\"1-5d82881a-0a9126e92a73e971eed891b9\",
```

```
\"origin\":\"AWS::ElasticBeanstalk::Environment\",\"subsegments\":[{\"id\":
\"2d6900034ccfe558\",\"name\":\"DynamoDB\",\"start_time\":1.568835610658E9,
\"end_time\":1.568835610664E9,\"http\":{\"response\":{\"status\":200,
\"content_length\":61}},\"aws\":{\"table_name\":\"scorekeep-user\",\"operation\":
\"UpdateItem\",\"request_id\":\"TPEIDNDUROMLPOV17U4A79555NVV4KQNSO5AEMVJF66Q9ASUAAJG
\",\"resource_names\":[\"scorekeep-user\"]},\"namespace\":\"aws\"}]}"
                },
                {
                    "Id": "0f278b6334c34e6b",
                    "Document": "{\"id\":\"0f278b6334c34e6b\",\"name\":
\"DynamoDB\",\"start_time\":1.568835610658E9,\"end_time\":1.568835610664E9,
\"parent_id\":\"2d6900034ccfe558\",\"inferred\":true,\"http\":{\"response
\":{\"status\":200,\"content_length\":61}},\"aws\":{\"table_name
\":\"scorekeep-user\",\"operation\":\"UpdateItem\",\"request_id\":
\"TPEIDNDUROMLPOV17U4A79555NVV4KQNSO5AEMVJF66Q9ASUAAJG\",\"resource_names\":
[\"scorekeep-user\"]},\"trace_id\":\"1-5d82881a-0a9126e92a73e971eed891b9\",\"origin
\":\"AWS::DynamoDB::Table\"}"
                }
            ]
        }
    ],
    "UnprocessedTraceIds": []
}
```

Per ulteriori informazioni, vedere Using the AWS X-Ray API with the AWS CLI nellaAWS X-Ray Developer Guide.

- Per i dettagli sull'API, vedere BatchTracesGetin AWS CLI Command Reference.

`create-group`

Il seguente esempio di codice mostra come utilizzare`create-group`.

AWS CLI

Come creare un gruppo

L'esempio `create-group` seguente crea un gruppo di risorse denominato `AdminGroup`. Il gruppo ottiene un'espressione di filtro che definisce i criteri del gruppo come segmento relativo a un servizio specifico che causa un guasto o un errore.

```
aws xray create-group \
```

```
    --group-name "AdminGroup" \
    --filter-expression "service(\"mydomain.com\") {fault OR error}"
```

Output:

```
{
    "GroupName": "AdminGroup",
    "GroupARN": "arn:aws:xray:us-west-2:123456789012:group/AdminGroup/123456789",
    "FilterExpression": "service(\"mydomain.com\") {fault OR error}"
}
```

Per ulteriori informazioni, vedere Configurazione delle impostazioni di campionamento, gruppi e crittografia con l'API X-Ray nella AWS X-Ray Developer Guide.AWS

- Per i dettagli sull'API, vedere CreateGroupin Command Reference.AWS CLI

`create-sampling-rule`

Il seguente esempio di codice mostra come utilizzare`create-sampling-rule`.

AWS CLI

Come creare una regola di campionamento

L'esempio `create-sampling-rule` seguente crea una regola per controllare il comportamento di campionamento delle applicazioni strumentate. Le regole sono fornite da un file JSON. La maggior parte dei campi delle regole di campionamento è necessaria per creare la regola.

```
aws xray create-sampling-rule \
    --cli-input-json file://9000-base-scorekeep.json
```

Contenuto di `9000-base-scorekeep.json`:

```
{
    "SamplingRule": {
        "RuleName": "base-scorekeep",
        "ResourceARN": "*",
        "Priority": 9000,
        "FixedRate": 0.1,
        "ReservoirSize": 5,
```

```
        "ServiceName": "Scorekeep",
        "ServiceType": "*",
        "Host": "*",
        "HTTPMethod": "*",
        "URLPath": "*",
        "Version": 1
    }
}
```

Output:

```
{
    "SamplingRuleRecord": {
        "SamplingRule": {
            "RuleName": "base-scorekeep",
            "RuleARN": "arn:aws:xray:us-west-2:123456789012:sampling-rule/base-
scorekeep",
            "ResourceARN": "*",
            "Priority": 9000,
            "FixedRate": 0.1,
            "ReservoirSize": 5,
            "ServiceName": "Scorekeep",
            "ServiceType": "*",
            "Host": "*",
            "HTTPMethod": "*",
            "URLPath": "*",
            "Version": 1,
            "Attributes": {}
        },
        "CreatedAt": 1530574410.0,
        "ModifiedAt": 1530574410.0
    }
}
```

Per ulteriori informazioni, vedere Configurazione delle impostazioni di campionamento, gruppi e crittografia con l'API X-Ray nella AWS X-Ray Developer Guide.AWS

- Per i dettagli sull'API, vedere CreateSamplingRulein Command Reference.AWS CLI

`delete-group`

Il seguente esempio di codice mostra come utilizzare`delete-group`.

AWS CLI

Come eliminare un gruppo

L'esempio `delete-group` seguente elimina il gruppo di risorse specificato.

```
aws xray delete-group \
    --group-name "AdminGroup" \
    --group-arn "arn:aws:xray:us-east-2:123456789012:group/AdminGroup/123456789"
```

Questo comando non produce alcun output.

Per ulteriori informazioni, vedere Configurazione delle impostazioni di campionamento, gruppi e crittografia con l'API X-Ray nella AWS X-Ray Developer Guide.AWS

- Per i dettagli sull'API, vedere DeleteGroupin Command Reference.AWS CLI

`delete-sampling-rule`

Il seguente esempio di codice mostra come utilizzare`delete-sampling-rule`.

AWS CLI

Come eliminare una regola di campionamento

L'esempio `delete-sampling-rule` seguente elimina la regola di campionamento specificata. È possibile specificare il gruppo utilizzando il nome del gruppo o l'ARN del gruppo.

```
aws xray delete-sampling-rule \
    --rule-name polling-scorekeep
```

Output:

```
{
    "SamplingRuleRecord": {
        "SamplingRule": {
            "RuleName": "polling-scorekeep",
            "RuleARN": "arn:aws:xray:us-west-2:123456789012:sampling-rule/polling-
scorekeep",
            "ResourceARN": "*",
            "Priority": 5000,
```

```
            "FixedRate": 0.003,
            "ReservoirSize": 0,
            "ServiceName": "Scorekeep",
            "ServiceType": "*",
            "Host": "*",
            "HTTPMethod": "GET",
            "URLPath": "/api/state/*",
            "Version": 1,
            "Attributes": {}
        },
        "CreatedAt": 1530574399.0,
        "ModifiedAt": 1530574399.0
    }
}
```

Per ulteriori informazioni, vedere Configurazione delle impostazioni di campionamento, gruppi e crittografia con l'API X-Ray nella AWS X-Ray Developer Guide.AWS

- Per i dettagli sull'API, vedere DeleteSamplingRulein Command Reference.AWS CLI

`get-encryption-config`

Il seguente esempio di codice mostra come utilizzare`get-encryption-config`.

AWS CLI

Come recuperare la configurazione di crittografia

L'`get-encryption-config`esempio seguente recupera la configurazione di crittografia corrente per i dati AWS X-Ray.

```
aws xray get-encryption-config
```

Output:

```
{
    "EncryptionConfig": {
        "KeyId": "ae4aa6d49-a4d8-9df9-a475-4ff6d7898456",
        "Status": "ACTIVE",
        "Type": "NONE"
    }
}
```

Per ulteriori informazioni, vedere Configurazione delle impostazioni di campionamento, gruppi e crittografia con l'API X-Ray nella AWS X-Ray Developer Guide.AWS

- Per i dettagli sull'API, vedere GetEncryptionConfigin Command Reference.AWS CLI

`get-group`

Il seguente esempio di codice mostra come utilizzare`get-group`.

AWS CLI

Come recuperare un gruppo

L'esempio `get-group` seguente visualizza i dettagli del tipo di istanza specificato. I dettagli includono il nome del gruppo, l'ARN del gruppo e l'espressione di filtro che definisce i criteri per quel gruppo. I gruppi possono essere recuperati anche tramite ARN.

```
aws xray get-group \
    --group-name "AdminGroup"
```

Output:

```
{
    "Group": [
        {
            "GroupName": "AdminGroup",
            "GroupARN": "arn:aws:xray:us-west-2:123456789012:group/
AdminGroup/123456789",
            "FilterExpression": "service(\"mydomain.com\") {fault OR error}"
        }
    ]
}
```

Per ulteriori informazioni, vedere Configurazione delle impostazioni di campionamento, gruppi e crittografia con l'API X-Ray nella AWS X-Ray Developer Guide.AWS

- Per i dettagli sull'API, vedere GetGroupin Command Reference.AWS CLI

`get-groups`

Il seguente esempio di codice mostra come utilizzare`get-groups`.

AWS CLI

Come recuperare tutti i gruppi

L'esempio seguente visualizza i dettagli per tutti i gruppi attivi.

```
aws xray get-groups
```

Output:

```
{
    "Groups": [
        {
            "GroupName": "AdminGroup",
            "GroupARN": "arn:aws:xray:us-west-2:123456789012:group/
AdminGroup/123456789",
            "FilterExpression": "service(\"example.com\") {fault OR error}"
        },
        {
            "GroupName": "SDETGroup",
            "GroupARN": "arn:aws:xray:us-west-2:123456789012:group/
SDETGroup/987654321",
            "FilterExpression": "responsetime > 2"
        }
    ]
}
```

Per ulteriori informazioni, vedere Configurazione delle impostazioni di campionamento, gruppi e crittografia con l'API X-Ray nella AWS X-Ray Developer Guide.AWS

- Per i dettagli sull'API, vedere GetGroupsin Command Reference.AWS CLI

`get-sampling-rules`

Il seguente esempio di codice mostra come utilizzare`get-sampling-rules`.

AWS CLI

Come recuperare tutte le regole di campionamento

L'esempio `get-sampling-rules` seguente mostra i dettagli di tutte le regole di campionamento disponibili:

```
aws xray get-sampling-rules
```

Output:

```
{
    "SamplingRuleRecords": [
        {
            "SamplingRule": {
                "RuleName": "Default",
                "RuleARN": "arn:aws:xray:us-east-1::sampling-rule/Default",
                "ResourceARN": "*",
                "Priority": 10000,
                "FixedRate": 0.01,
                "ReservoirSize": 0,
                "ServiceName": "*",
                "ServiceType": "*",
                "Host": "*",
                "HTTPMethod": "*",
                "URLPath": "*",
                "Version": 1,
                "Attributes": {}
            },
            "CreatedAt": 0.0,
            "ModifiedAt": 1530558121.0
        },
        {
            "SamplingRule": {
                "RuleName": "base-scorekeep",
                "RuleARN": "arn:aws:xray:us-east-1::sampling-rule/base-scorekeep",
                "ResourceARN": "*",
                "Priority": 9000,
                "FixedRate": 0.1,
                "ReservoirSize": 2,
                "ServiceName": "Scorekeep",
                "ServiceType": "*",
                "Host": "*",
                "HTTPMethod": "*",
                "URLPath": "*",
                "Version": 1,
                "Attributes": {}
            },
            "CreatedAt": 1530573954.0,
            "ModifiedAt": 1530920505.0
```

```
        },
        {
            "SamplingRule": {
                "RuleName": "polling-scorekeep",
                "RuleARN": "arn:aws:xray:us-east-1::sampling-rule/polling-
scorekeep",
                "ResourceARN": "*",
                "Priority": 5000,
                "FixedRate": 0.003,
                "ReservoirSize": 0,
                "ServiceName": "Scorekeep",
                "ServiceType": "*",
                "Host": "*",
                "HTTPMethod": "GET",
                "URLPath": "/api/state/*",
                "Version": 1,
                "Attributes": {}
            },
            "CreatedAt": 1530918163.0,
            "ModifiedAt": 1530918163.0
        }
    ]
}
```

Per ulteriori informazioni, consulta Utilizzo delle regole di campionamento con l'API X-Ray nella Guida per gli sviluppatori di AWS X-Ray.

- Per i dettagli sull'API, consulta GetSamplingRules AWS CLICommand Reference.

`get-sampling-targets`

Il seguente esempio di codice mostra come utilizzare`get-sampling-targets`.

AWS CLI

Come richiedere una quota di campionamento

L'esempio `get-sampling-targets` seguente richiede una quota di campionamento per le regole utilizzate dal servizio per campionare le richieste. La risposta di AWS X-Ray include una quota che può essere utilizzata invece di prendere in prestito dal serbatoio.

```
aws xray get-sampling-targets \
```

```
    --sampling-statistics-documents '[ { "RuleName": "base-scorekeep", "ClientID":
"ABCDEF1234567890ABCDEF10", "Timestamp": "2018-07-07T00:20:06, "RequestCount": 110,
"SampledCount": 20, "BorrowCount": 10 }, { "RuleName": "polling-scorekeep", 31,
"BorrowCount": 0 } ]'
```

Output:

```
{
    "SamplingTargetDocuments": [
        {
            "RuleName": "base-scorekeep",
            "FixedRate": 0.1,
            "ReservoirQuota": 2,
            "ReservoirQuotaTTL": 1530923107.0,
            "Interval": 10
        },
        {
            "RuleName": "polling-scorekeep",
            "FixedRate": 0.003,
            "ReservoirQuota": 0,
            "ReservoirQuotaTTL": 1530923107.0,
            "Interval": 10
        }
    ],
    "LastRuleModification": 1530920505.0,
    "UnprocessedStatistics": []
}
```

Per ulteriori informazioni, consulta Utilizzo delle regole di campionamento con l'API X-Ray nella Guida per gli sviluppatori di AWS X-Ray.

- Per i dettagli sull'API, vedere GetSamplingTargetsin Command Reference.AWS CLI

`get-service-graph`

Il seguente esempio di codice mostra come utilizzare`get-service-graph`.

AWS CLI

Come ottenere un grafico dei servizi

L'esempio seguente visualizza un documento entro un periodo di tempo specificato che descrive i servizi che elaborano le richieste in entrata e i servizi downstream che chiamano di conseguenza:

```
aws xray get-service-graph \
    --start-time 1568835392.0
    --end-time 1568835446.0
```

Output:


```
{
    "Services": [
        {
            "ReferenceId": 0,
            "Name": "Scorekeep",
            "Names": [
                "Scorekeep"
            ],
            "Root": true,
            "Type": "AWS::ElasticBeanstalk::Environment",
            "State": "active",
            "StartTime": 1568835392.0,
            "EndTime": 1568835446.0,
            "Edges": [
                {
                    "ReferenceId": 1,
                    "StartTime": 1568835392.0,
                    "EndTime": 1568835446.0,
                    "SummaryStatistics": {
                        "OkCount": 14,
                        "ErrorStatistics": {
                            "ThrottleCount": 0,
                            "OtherCount": 0,
                            "TotalCount": 0
                        },
                        "FaultStatistics": {
                            "OtherCount": 0,
                            "TotalCount": 0
                        },
                        "TotalCount": 14,
                        "TotalResponseTime": 0.13
                    },
                    "ResponseTimeHistogram": [
                        {
                            "Value": 0.008,
                            "Count": 1
                        },
```

```
                        {
                            "Value": 0.005,
                            "Count": 7
                        },
                        {
                            "Value": 0.009,
                            "Count": 1
                        },
                        {
                            "Value": 0.021,
                            "Count": 1
                        },
                        {
                            "Value": 0.038,
                            "Count": 1
                        },
                        {
                            "Value": 0.007,
                            "Count": 1
                        },
                        {
                            "Value": 0.006,
                            "Count": 2
                        }
                    ],
                    "Aliases": []
                },

                ... TRUNCATED FOR BREVITY ...

            ]
        }
    ],
    "StartTime": 1568835392.0,
    "EndTime": 1568835446.0,
    "ContainsOldGroupVersions": false
}
```

Per ulteriori informazioni, vedere Using the AWS X-Ray API with the AWS CLI nellaAWS X-Ray Developer Guide.

- Per i dettagli sull'API, vedere GetServiceGraphin AWS CLI Command Reference.

`get-trace-summaries`

Il seguente esempio di codice mostra come utilizzare`get-trace-summaries`.

AWS CLI

Come ottenere un riepilogo delle tracce

L'`get-trace-summaries`esempio seguente recupera i IDs metadati per le tracce disponibili entro un periodo di tempo specificato.

```
aws xray get-trace-summaries \
    --start-time 1568835392.0 \
    --end-time 1568835446.0
```

Output:

```
[
    "http://scorekeep-env-1.123456789.us-east-2.elasticbeanstalk.com/api/move/
VSAE93HF/GSSD2NTB/DP0PCC09",
    "http://scorekeep-env-1.123456789.us-east-2.elasticbeanstalk.com/api/move/
GCQ2B35P/FREELDFT/4LRE643M",
    "http://scorekeep-env-1.123456789.us-east-2.elasticbeanstalk.com/api/game/
VSAE93HF/GSSD2NTB/starttime/1568835513",
    "http://scorekeep-env-1.123456789.us-east-2.elasticbeanstalk.com/api/
move/4MQNA5NN/L99KK2RF/null"
]
```

Per ulteriori informazioni, vedere Using the AWS X-Ray API with the AWS CLI nellaAWS X-Ray Developer Guide.

- Per i dettagli sull'API, vedere GetTraceSummariesin AWS CLI Command Reference.

`put-encryption-config`

Il seguente esempio di codice mostra come utilizzare`put-encryption-config`.

AWS CLI

Come aggiornare la configurazione della crittografia

Nell'esempio seguente `put-encryption-config``example updates the encryption configuration for AWS X-Ray data to use the default AWS managed KMS key ``aws/xray`.

```
aws xray put-encryption-config \
    --type KMS \
    --key-id alias/aws/xray
```

Output:

```
{
    "EncryptionConfig": {
        "KeyId": "arn:aws:kms:us-west-2:123456789012:key/c234g4e8-39e9-4gb0-84e2-
b0ea215cbba5",
        "Status": "UPDATING",
        "Type": "KMS"
    }
}
```

Per ulteriori informazioni, vedere Configurazione delle impostazioni di campionamento, gruppi e crittografia con l'API X-Ray nella AWS X-Ray Developer Guide.AWS

- Per i dettagli sull'API, vedere PutEncryptionConfigin Command Reference.AWS CLI

`put-trace-segments`

Il seguente esempio di codice mostra come utilizzare`put-trace-segments`.

AWS CLI

Come caricare un segmento

L'`put-trace-segments`esempio seguente carica i documenti dei segmenti su AWS X-Ray. Il documento del segmento viene utilizzato come elenco di documenti di segmento JSON.

```
aws xray put-trace-segments \
    --trace-segment-documents "{\"id\":\"20312a0e2b8809f4\",\"name
\":\"DynamoDB\",\"trace_id\":\"1-5832862d-a43aafded3334a971fe312db\",
\"start_time\":1.479706157195E9,\"end_time\":1.479706157202E9,\"parent_id\":
\"79736b962fe3239e\",\"http\":{\"response\":{\"content_length\":60,\"status
\":200}},\"inferred\":true,\"aws\":{\"consistent_read\":false,\"table_name
\":\"scorekeep-session-xray\",\"operation\":\"GetItem\",\"request_id\":
```

```
\"SCAU23OM6M8FO38UASGC7785ARVV4KQNSO5AEMVJF66Q9ASUAAJG\",\"resource_names\":
[\"scorekeep-session-xray\"]},\"origin\":\"AWS::DynamoDB::Table\"}"
```

Output:

```
{
    "UnprocessedTraceSegments": []
}
```

Per ulteriori informazioni, vedere Invio di dati di traccia a AWS X-Ray nella X-Ray Developer Guide AWS .

- Per i dettagli sull'API, vedere PutTraceSegmentsin AWS CLI Command Reference.

`update-group`

Il seguente esempio di codice mostra come utilizzareu`pdate-group`.

AWS CLI

Come aggiornare un gruppo

L'esempio `update-group` seguente aggiorna i criteri in base ai quali accettare le tracce nel gruppo denominato `AdminGroup`. È possibile specificare il gruppo desiderato utilizzando il nome o l'ARN del gruppo.

```
aws xray update-group \
    --group-name "AdminGroup" \
    --group-arn "arn:aws:xray:us-west-2:123456789012:group/AdminGroup/123456789" \
    --filter-expression "service(\"mydomain.com\") {fault}"
```

Output:

```
{
    "GroupName": "AdminGroup",
    "GroupARN": "arn:aws:xray:us-east-2:123456789012:group/AdminGroup/123456789",
    "FilterExpression": "service(\"mydomain.com\") {fault}"
}
```

Per ulteriori informazioni, vedere Configurazione delle impostazioni di campionamento, gruppi e crittografia con l'API X-Ray nella AWS X-Ray Developer Guide.AWS

- Per i dettagli sull'API, vedere UpdateGroupin Command Reference.AWS CLI

`update-sampling-rule`

Il seguente esempio di codice mostra come utilizzareu`pdate-sampling-rule`.

AWS CLI

Come aggiornare una regola di campionamento

L'esempio `update-sampling-rule` seguente modifica la configurazione di una regola di campionamento. Le regole vengono utilizzate da un file JSON. Sono obbligatori solo i campi in fase di aggiornamento.

```
aws xray update-sampling-rule \
   --cli-input-json file://1000-default.json
```

Contenuto di `1000-default.json`:

```
{
    "SamplingRuleUpdate": {
        "RuleName": "Default",
        "FixedRate": 0.01,
        "ReservoirSize": 0
    }
}
```

Output:

```
{
    "SamplingRuleRecords": [
        {
            "SamplingRule": {
                "RuleName": "Default",
                "RuleARN": "arn:aws:xray:us-west-2:123456789012:sampling-rule/
Default",
                "ResourceARN": "*",
                "Priority": 10000,
                "FixedRate": 0.01,
                "ReservoirSize": 0,
                "ServiceName": "*",
```

```
                "ServiceType": "*",
                "Host": "*",
                "HTTPMethod": "*",
                "URLPath": "*",
                "Version": 1,
                "Attributes": {}
            },
            "CreatedAt": 0.0,
            "ModifiedAt": 1529959993.0
        }
    ]
}
```

Per ulteriori informazioni, vedere Configurazione delle impostazioni di campionamento, gruppi e crittografia con l'API X-Ray nella AWS X-Ray Developer Guide.AWS

- Per i dettagli sull'API, vedere UpdateSamplingRulein Command Reference.AWS CLI

AWS CLI con esempi di codice di script Bash

Gli esempi di codice in questo argomento mostrano come usare lo script AWS Command Line Interface with Bash con. AWS

Nozioni di base: esempi di codice che mostrano come eseguire le operazioni essenziali all'interno di un servizio.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Scenari: esempi di codice che mostrano come eseguire un'attività specifica chiamando più funzioni all'interno dello stesso servizio o combinate con altri Servizi AWS.

Alcuni servizi contengono ulteriori categorie di esempi che mostrano come sfruttare le librerie o le funzioni specifiche del servizio.

Servizi

- AWS Batch esempi di utilizzo AWS CLI con lo script Bash
- AWS Cloud Map esempi di utilizzo AWS CLI con lo script Bash
- CloudFront esempi di utilizzo AWS CLI con lo script Bash

- Esempi AWS CLI di DynamoDB utilizzati con lo script Bash
- EC2 Esempi di Amazon che utilizzano AWS CLI lo script Bash
- HealthImaging esempi di utilizzo AWS CLI con lo script Bash
- Esempi IAM che utilizzano AWS CLI lo script Bash
- AWS KMS esempi di utilizzo AWS CLI con lo script Bash
- Esempi AWS CLI di Lightsail con lo script Bash
- Esempi di Amazon S3 che utilizzano lo script AWS CLI Bash
- AWS STS esempi di utilizzo AWS CLI con lo script Bash

AWS Batch esempi di utilizzo AWS CLI con lo script Bash

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando lo script AWS Command Line Interface with Bash con. AWS Batch

Scenari: esempi di codice che mostrano come eseguire un'attività specifica chiamando più funzioni all'interno dello stesso servizio o combinate con altri Servizi AWS.

Ogni esempio include un link al codice sorgente completo, dove è possibile trovare le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Scenari

Scenari

Nozioni di base su Batch e Fargate

L'esempio di codice seguente mostra come:

- Creare un ruolo di esecuzione IAM per le attività ECS
- Creare un ambiente di calcolo Fargate gestito
- Creare una coda di processi con impostazioni di priorità
- Registrare una definizione di processo per carichi di lavoro containerizzati
- Inviare e monitorare l'esecuzione di un processo batch

- Visualizza l'output del lavoro nei registri CloudWatch
- Eseguire la pulizia delle risorse nell'ordine di dipendenza corretto

AWS CLI con lo script Bash

> **Note**
>
> C'è altro da fare. GitHub Trova l'esempio completo e scopri come eseguire la configurazione e l'esecuzione nel repository dei tutorial sugli esempi di codice per gli sviluppatori.

```
#!/bin/bash

# AWS Batch Fargate Getting Started Script - Fixed Version
# This script demonstrates creating AWS Batch resources with Fargate orchestration
#
# HIGH SEVERITY FIXES APPLIED:
# 1. Added IAM role propagation delay after role creation
# 2. Added resource state validation before deletion attempts

set -e  # Exit on any error

# Configuration
SCRIPT_NAME="batch-fargate-tutorial"
LOG_FILE="${SCRIPT_NAME}-$(date +%Y%m%d-%H%M%S).log"
RANDOM_SUFFIX=$(openssl rand -hex 6)
COMPUTE_ENV_NAME="batch-fargate-compute-${RANDOM_SUFFIX}"
JOB_QUEUE_NAME="batch-fargate-queue-${RANDOM_SUFFIX}"
JOB_DEF_NAME="batch-fargate-jobdef-${RANDOM_SUFFIX}"
JOB_NAME="batch-hello-world-${RANDOM_SUFFIX}"
ROLE_NAME="BatchEcsTaskExecutionRole-${RANDOM_SUFFIX}"
TRUST_POLICY_FILE="batch-trust-policy-${RANDOM_SUFFIX}.json"

# Array to track created resources for cleanup
CREATED_RESOURCES=()

# Logging function
log() {
    echo "[$(date '+%Y-%m-%d %H:%M:%S')] $1" | tee -a "$LOG_FILE"
}
```

```
# Error handling function
handle_error() {
    log "ERROR: Script failed at line $1"
    log "Attempting to clean up resources created so far..."
    cleanup_resources
    exit 1
}

# Set up error handling
trap 'handle_error $LINENO' ERR

# Function to wait for resource to be ready
wait_for_compute_env() {
    local env_name=$1
    log "Waiting for compute environment $env_name to be VALID..."

    while true; do
        local status=$(aws batch describe-compute-environments \
            --compute-environments "$env_name" \
            --query 'computeEnvironments[0].status' \
            --output text 2>/dev/null || echo "NOT_FOUND")

        if [ "$status" = "VALID" ]; then
            log "Compute environment $env_name is ready"
            break
        elif [ "$status" = "INVALID" ] || [ "$status" = "NOT_FOUND" ]; then
            log "ERROR: Compute environment $env_name failed to create properly"
            return 1
        fi

        log "Compute environment status: $status. Waiting 10 seconds..."
        sleep 10
    done
}

# Function to wait for job queue to be ready
wait_for_job_queue() {
    local queue_name=$1
    log "Waiting for job queue $queue_name to be VALID..."

    while true; do
        local state=$(aws batch describe-job-queues \
            --job-queues "$queue_name" \
```

```
            --query 'jobQueues[0].state' \
            --output text 2>/dev/null || echo "NOT_FOUND")

        if [ "$state" = "ENABLED" ]; then
            log "Job queue $queue_name is ready"
            break
        elif [ "$state" = "DISABLED" ] || [ "$state" = "NOT_FOUND" ]; then
            log "ERROR: Job queue $queue_name failed to create properly"
            return 1
        fi

        log "Job queue state: $state. Waiting 10 seconds..."
        sleep 10
    done
}

# Function to wait for job completion
wait_for_job() {
    local job_id=$1
    log "Waiting for job $job_id to complete..."

    while true; do
        local status=$(aws batch describe-jobs \
            --jobs "$job_id" \
            --query 'jobs[0].status' \
            --output text 2>/dev/null || echo "NOT_FOUND")

        if [ "$status" = "SUCCEEDED" ]; then
            log "Job $job_id completed successfully"
            break
        elif [ "$status" = "FAILED" ]; then
            log "ERROR: Job $job_id failed"
            return 1
        fi

        log "Job status: $status. Waiting 30 seconds..."
        sleep 30
    done
}

# FIXED: Added function to wait for resource state before deletion
wait_for_resource_state() {
    local resource_type=$1
    local resource_name=$2
```

```
    local expected_state=$3
    local max_attempts=30
    local attempt=0

    log "Waiting for $resource_type $resource_name to reach state: $expected_state"

    while [ $attempt -lt $max_attempts ]; do
        local current_state=""

        case $resource_type in
            "JOB_QUEUE")
                current_state=$(aws batch describe-job-queues \
                    --job-queues "$resource_name" \
                    --query 'jobQueues[0].state' \
                    --output text 2>/dev/null || echo "NOT_FOUND")
                ;;
            "COMPUTE_ENV")
                current_state=$(aws batch describe-compute-environments \
                    --compute-environments "$resource_name" \
                    --query 'computeEnvironments[0].status' \
                    --output text 2>/dev/null || echo "NOT_FOUND")
                ;;
        esac

        if [ "$current_state" = "$expected_state" ]; then
            log "$resource_type $resource_name is now in state: $expected_state"
            return 0
        fi

        log "$resource_type $resource_name state: $current_state (waiting for
 $expected_state)"
        sleep 10
        ((attempt++))
    done

    log "WARNING: $resource_type $resource_name did not reach expected state after
 $max_attempts attempts"
    return 1
}

# Cleanup function
cleanup_resources() {
    log "Starting cleanup of created resources..."
```

```
    # Clean up in reverse order of creation
    for ((i=${#CREATED_RESOURCES[@]}-1; i>=0; i--)); do
        local resource="${CREATED_RESOURCES[i]}"
        local resource_type=$(echo "$resource" | cut -d: -f1)
        local resource_name=$(echo "$resource" | cut -d: -f2)

        log "Cleaning up $resource_type: $resource_name"

        case $resource_type in
            "JOB_QUEUE")
                # FIXED: Validate state before deletion
                aws batch update-job-queue --job-queue "$resource_name" --state
 DISABLED 2>/dev/null || true
                wait_for_resource_state "JOB_QUEUE" "$resource_name" "DISABLED" ||
 true
                aws batch delete-job-queue --job-queue "$resource_name" 2>/dev/null
 || true
                ;;
            "COMPUTE_ENV")
                # FIXED: Validate state before deletion
                aws batch update-compute-environment --compute-environment
 "$resource_name" --state DISABLED 2>/dev/null || true
                wait_for_resource_state "COMPUTE_ENV" "$resource_name" "DISABLED" ||
 true
                aws batch delete-compute-environment --compute-environment
 "$resource_name" 2>/dev/null || true
                ;;
            "IAM_ROLE")
                aws iam detach-role-policy --role-name "$resource_name" --policy-arn
 "arn:aws:iam::aws:policy/service-role/AmazonECSTaskExecutionRolePolicy" 2>/dev/null
 || true
                aws iam delete-role --role-name "$resource_name" 2>/dev/null || true
                ;;
            "FILE")
                rm -f "$resource_name" 2>/dev/null || true
                ;;
        esac
    done

    log "Cleanup completed"
}

# Main script execution
main() {
```

```
    log "Starting AWS Batch Fargate tutorial script - Fixed Version"
    log "Log file: $LOG_FILE"

    # Get AWS account ID
    log "Getting AWS account ID..."
    ACCOUNT_ID=$(aws sts get-caller-identity --query Account --output text)
    log "Account ID: $ACCOUNT_ID"

    # Get default VPC and subnets
    log "Getting default VPC and subnets..."
    DEFAULT_VPC=$(aws ec2 describe-vpcs \
        --filters "Name=is-default,Values=true" \
        --query 'Vpcs[0].VpcId' \
        --output text)

    if [ "$DEFAULT_VPC" = "None" ] || [ "$DEFAULT_VPC" = "null" ]; then
        log "ERROR: No default VPC found. Please create a VPC first."
        exit 1
    fi

    log "Default VPC: $DEFAULT_VPC"

    # Get subnets in the default VPC
    SUBNETS=$(aws ec2 describe-subnets \
        --filters "Name=vpc-id,Values=$DEFAULT_VPC" \
        --query 'Subnets[*].SubnetId' \
        --output text)

    if [ -z "$SUBNETS" ]; then
        log "ERROR: No subnets found in default VPC"
        exit 1
    fi

    # Convert tab/space-separated subnets to JSON array format
    SUBNET_ARRAY=$(echo "$SUBNETS" | tr '\t ' '\n' | sed 's/^/"/;s/$/"/' | paste -sd
','  -)
    log "Subnets: $SUBNETS"
    log "Subnet array: [$SUBNET_ARRAY]"

    # Get default security group for the VPC
    DEFAULT_SG=$(aws ec2 describe-security-groups \
        --filters "Name=vpc-id,Values=$DEFAULT_VPC" "Name=group-name,Values=default"
\
        --query 'SecurityGroups[0].GroupId' \
```

```
        --output text)

    if [ "$DEFAULT_SG" = "None" ] || [ "$DEFAULT_SG" = "null" ]; then
        log "ERROR: No default security group found in VPC"
        exit 1
    fi

    log "Default security group: $DEFAULT_SG"

    # Step 1: Create IAM execution role
    log "Step 1: Creating IAM execution role..."

    # Create trust policy document
    cat > "$TRUST_POLICY_FILE" << EOF
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "ecs-tasks.amazonaws.com"
      },
      "Action": "sts:AssumeRole"
    }
  ]
}
EOF
    CREATED_RESOURCES+=("FILE:$TRUST_POLICY_FILE")

    # Create the role
    aws iam create-role \
        --role-name "$ROLE_NAME" \
        --assume-role-policy-document "file://$TRUST_POLICY_FILE"
    CREATED_RESOURCES+=("IAM_ROLE:$ROLE_NAME")

    # Attach policy
    aws iam attach-role-policy \
        --role-name "$ROLE_NAME" \
        --policy-arn "arn:aws:iam::aws:policy/service-role/
AmazonECSTaskExecutionRolePolicy"

    log "IAM role created: $ROLE_NAME"

    # FIXED: Wait for IAM role propagation
```

```
    log "Waiting for IAM role propagation (15 seconds)..."
    sleep 15

    # Step 2: Create compute environment
    log "Step 2: Creating Fargate compute environment..."

    aws batch create-compute-environment \
        --compute-environment-name "$COMPUTE_ENV_NAME" \
        --type MANAGED \
        --state ENABLED \
        --compute-resources "{
            \"type\": \"FARGATE\",
            \"maxvCpus\": 256,
            \"subnets\": [$SUBNET_ARRAY],
            \"securityGroupIds\": [\"$DEFAULT_SG\"]
        }"
    CREATED_RESOURCES+=("COMPUTE_ENV:$COMPUTE_ENV_NAME")

    # Wait for compute environment to be ready
    wait_for_compute_env "$COMPUTE_ENV_NAME"

    # Step 3: Create job queue
    log "Step 3: Creating job queue..."

    aws batch create-job-queue \
        --job-queue-name "$JOB_QUEUE_NAME" \
        --state ENABLED \
        --priority 900 \
        --compute-environment-order order=1,computeEnvironment="$COMPUTE_ENV_NAME"
    CREATED_RESOURCES+=("JOB_QUEUE:$JOB_QUEUE_NAME")

    # Wait for job queue to be ready
    wait_for_job_queue "$JOB_QUEUE_NAME"

    # Step 4: Create job definition
    log "Step 4: Creating job definition..."

    aws batch register-job-definition \
        --job-definition-name "$JOB_DEF_NAME" \
        --type container \
        --platform-capabilities FARGATE \
        --container-properties "{
            \"image\": \"busybox\",
            \"resourceRequirements\": [
```

```
                {\"type\": \"VCPU\", \"value\": \"0.25\"},
                {\"type\": \"MEMORY\", \"value\": \"512\"}
            ],
            \"command\": [\"echo\", \"hello world\"],
            \"networkConfiguration\": {
                \"assignPublicIp\": \"ENABLED\"
            },
            \"executionRoleArn\": \"arn:aws:iam::${ACCOUNT_ID}:role/${ROLE_NAME}\"
        }"

    log "Job definition created: $JOB_DEF_NAME"

    # Step 5: Submit job
    log "Step 5: Submitting job..."

    JOB_ID=$(aws batch submit-job \
        --job-name "$JOB_NAME" \
        --job-queue "$JOB_QUEUE_NAME" \
        --job-definition "$JOB_DEF_NAME" \
        --query 'jobId' \
        --output text)

    log "Job submitted with ID: $JOB_ID"

    # Step 6: Wait for job completion and view output
    log "Step 6: Waiting for job completion..."
    wait_for_job "$JOB_ID"

    # Get log stream name
    log "Getting job logs..."
    LOG_STREAM=$(aws batch describe-jobs \
        --jobs "$JOB_ID" \
        --query 'jobs[0].attempts[0].taskProperties.containers[0].logStreamName' \
        --output text)

    if [ "$LOG_STREAM" != "None" ] && [ "$LOG_STREAM" != "null" ]; then
        log "Log stream: $LOG_STREAM"
        log "Job output:"
        aws logs get-log-events \
            --log-group-name "/aws/batch/job" \
            --log-stream-name "$LOG_STREAM" \
            --query 'events[*].message' \
            --output text | tee -a "$LOG_FILE"
    else
```

```
        log "No log stream available for job"
    fi

    log "Tutorial completed successfully!"

    # Show created resources
    echo ""
    echo "==========================================="
    echo "CREATED RESOURCES"
    echo "==========================================="
    echo "The following resources were created:"
    for resource in "${CREATED_RESOURCES[@]}"; do
        echo "  - $resource"
    done
    echo ""
    echo "==========================================="
    echo "CLEANUP CONFIRMATION"
    echo "==========================================="
    echo "Do you want to clean up all created resources? (y/n): "
    read -r CLEANUP_CHOICE

    if [[ "$CLEANUP_CHOICE" =~ ^[Yy]$ ]]; then
        cleanup_resources
        log "All resources have been cleaned up"
    else
        log "Resources left intact. You can clean them up manually later."
        echo "To clean up manually, run the following commands:"
        echo "aws batch update-job-queue --job-queue $JOB_QUEUE_NAME --state
 DISABLED"
        echo "aws batch delete-job-queue --job-queue $JOB_QUEUE_NAME"
        echo "aws batch update-compute-environment --compute-environment
 $COMPUTE_ENV_NAME --state DISABLED"
        echo "aws batch delete-compute-environment --compute-environment
 $COMPUTE_ENV_NAME"
        echo "aws iam detach-role-policy --role-name $ROLE_NAME --policy-arn
 arn:aws:iam::aws:policy/service-role/AmazonECSTaskExecutionRolePolicy"
        echo "aws iam delete-role --role-name $ROLE_NAME"
    fi
}

# Run main function
main "$@"
```

- Per informazioni dettagliate sull'API, consulta i seguenti argomenti nella documentazione di riferimento dei comandi della AWS CLI .
  - CreateComputeEnvironment
  - CreateJobQueue
  - DeleteComputeEnvironment
  - DeleteJobQueue
  - DescribeComputeEnvironments
  - DescribeJobQueues
  - DescribeJobs
  - RegisterJobDefinition
  - SubmitJob
  - UpdateComputeEnvironment
  - UpdateJobQueue

AWS Cloud Map esempi di utilizzo AWS CLI con lo script Bash

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando lo script AWS Command Line Interface with Bash con. AWS Cloud Map

Scenari: esempi di codice che mostrano come eseguire un'attività specifica chiamando più funzioni all'interno dello stesso servizio o combinate con altri Servizi AWS.

Ogni esempio include un link al codice sorgente completo, dove è possibile trovare le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Scenari

Scenari

Attributi personalizzati con Cloud Map

L'esempio di codice seguente mostra come:

- Creare un namespace HTTP per il rilevamento servizi basati su API
- Creare una tabella DynamoDB e registrarla come servizio dati con attributi personalizzati

- Creare funzioni Lambda per leggere e scrivere dati
- Registrare le funzioni Lambda come istanze di servizio con attributi personalizzati per il rilevamento basato sulle azioni
- Creare applicazioni client che rilevano servizi utilizzando attributi personalizzati
- Eseguire la pulizia di tutte le risorse, incluse le funzioni Lambda, la tabella DynamoDB e i servizi Cloud Map

AWS CLI con lo script Bash

> **Note**
>
> C'è altro da fare. GitHub Trova l'esempio completo e scopri come eseguire la configurazione e l'esecuzione nel repository dei tutorial sugli esempi di codice per gli sviluppatori.

```
#!/bin/bash

# AWS Cloud Map Tutorial Script
# This script demonstrates how to use AWS Cloud Map for service discovery with
 custom attributes

# Set up logging
LOG_FILE="cloudmap-tutorial.log"
echo "AWS Cloud Map Tutorial Script" > $LOG_FILE
echo "Started at $(date)" >> $LOG_FILE

# Array to track created resources for cleanup
CREATED_RESOURCES=()

# Function to log commands and their output
log_cmd() {
  echo "$ $1" | tee -a $LOG_FILE
  eval "$1" | tee -a $LOG_FILE
}

# Function to handle errors
handle_error() {
  local LINE=$1
  echo "An error occurred at line $LINE" | tee -a $LOG_FILE
```

```
  echo "Resources created so far:" | tee -a $LOG_FILE
  for resource in "${CREATED_RESOURCES[@]}"; do
    echo "- $resource" | tee -a $LOG_FILE
  done
  echo "Attempting to clean up resources..." | tee -a $LOG_FILE
  cleanup
  exit 1
}

# Set up error handling
trap 'handle_error $LINENO' ERR

# Helper function to wait for Cloud Map operations to complete
wait_for_operation() {
  local OPERATION_ID=$1
  local TIMEOUT=300  # 5 minutes timeout
  local START_TIME=$(date +%s)

  while true; do
    local STATUS=$(aws servicediscovery get-operation --operation-id $OPERATION_ID
 --query 'Operation.Status' --output text)

    if [ "$STATUS" == "SUCCESS" ]; then
      echo "Operation completed successfully" | tee -a $LOG_FILE
      break
    elif [ "$STATUS" == "FAIL" ]; then
      echo "Operation failed" | tee -a $LOG_FILE
      return 1
    fi

    local CURRENT_TIME=$(date +%s)
    if [ $((CURRENT_TIME - START_TIME)) -gt $TIMEOUT ]; then
      echo "Operation timed out" | tee -a $LOG_FILE
      return 1
    fi

    sleep 5
  done

  return 0
}

# Function to clean up resources
cleanup() {
```

```
 echo "Cleaning up resources..." | tee -a $LOG_FILE

 # Reverse the order of created resources for proper deletion
 for ((i=${#CREATED_RESOURCES[@]}-1; i>=0; i--)); do
   resource="${CREATED_RESOURCES[$i]}"
   echo "Deleting $resource..." | tee -a $LOG_FILE

   if [[ $resource == "instance:"* ]]; then
     # Extract service ID and instance ID
     SERVICE_ID=$(echo $resource | cut -d':' -f2)
     INSTANCE_ID=$(echo $resource | cut -d':' -f3)

     # Check if instance exists before trying to deregister
     INSTANCE_EXISTS=$(aws servicediscovery list-instances --service-id $SERVICE_ID
--query "Instances[?Id=='$INSTANCE_ID'].Id" --output text 2>/dev/null || echo "")
     if [[ -n "$INSTANCE_EXISTS" ]]; then
       OPERATION_ID=$(aws servicediscovery deregister-instance --service-id
$SERVICE_ID --instance-id $INSTANCE_ID --query 'OperationId' --output text)

       # Wait for deregistration to complete
       echo "Waiting for instance deregistration to complete..." | tee -a $LOG_FILE
       wait_for_operation $OPERATION_ID
     else
       echo "Instance $INSTANCE_ID already deregistered" | tee -a $LOG_FILE
     fi
   elif [[ $resource == "lambda:"* ]]; then
     # Extract function name
     FUNCTION_NAME=$(echo $resource | cut -d':' -f2)
     aws lambda delete-function --function-name $FUNCTION_NAME
   elif [[ $resource == "role:"* ]]; then
     # Extract role name
     ROLE_NAME=$(echo $resource | cut -d':' -f2)

     # Detach all policies first
     for POLICY_ARN in $(aws iam list-attached-role-policies --role-name $ROLE_NAME
--query 'AttachedPolicies[*].PolicyArn' --output text); do
       aws iam detach-role-policy --role-name $ROLE_NAME --policy-arn $POLICY_ARN
     done

     # Delete the role
     aws iam delete-role --role-name $ROLE_NAME
   elif [[ $resource == "dynamodb:"* ]]; then
     # Extract table name
     TABLE_NAME=$(echo $resource | cut -d':' -f2)
```

```
      aws dynamodb delete-table --table-name $TABLE_NAME

      # Wait for table deletion to complete
      echo "Waiting for DynamoDB table deletion to complete..." | tee -a $LOG_FILE
      aws dynamodb wait table-not-exists --table-name $TABLE_NAME
    fi
  done

  # Handle services separately to ensure all instances are deregistered first
  for ((i=${#CREATED_RESOURCES[@]}-1; i>=0; i--)); do
    resource="${CREATED_RESOURCES[$i]}"
    if [[ $resource == "service:"* ]]; then
      # Extract service ID
      SERVICE_ID=$(echo $resource | cut -d':' -f2)
      echo "Deleting service $SERVICE_ID..." | tee -a $LOG_FILE

      # Make sure all instances are deregistered
      INSTANCES=$(aws servicediscovery list-instances --service-id $SERVICE_ID --
query 'Instances[*].Id' --output text)
      if [[ -n "$INSTANCES" ]]; then
        echo "Service still has instances. Waiting before deletion..." | tee -a
 $LOG_FILE
        sleep 10
      fi

      # Try to delete the service
      aws servicediscovery delete-service --id $SERVICE_ID
      sleep 5
    fi
  done

  # Handle namespaces last to ensure all services are deleted first
  for ((i=${#CREATED_RESOURCES[@]}-1; i>=0; i--)); do
    resource="${CREATED_RESOURCES[$i]}"
    if [[ $resource == "namespace:"* ]]; then
      # Extract namespace ID
      NAMESPACE_ID=$(echo $resource | cut -d':' -f2)
      echo "Deleting namespace $NAMESPACE_ID..." | tee -a $LOG_FILE

      # Check if namespace still has services
      SERVICES=$(aws servicediscovery list-services --filters
 "Name=NAMESPACE_ID,Values=$NAMESPACE_ID,Condition=EQ" --query 'Services[*].Id' --
output text)
      if [[ -n "$SERVICES" ]]; then
```

```
        echo "Namespace still has services. Deleting them first..." | tee -a
 $LOG_FILE
        for SERVICE_ID in $SERVICES; do
          echo "Deleting service $SERVICE_ID..." | tee -a $LOG_FILE
          aws servicediscovery delete-service --id $SERVICE_ID
        done
        sleep 5
      fi

      # Try to delete the namespace
      OPERATION_ID=$(aws servicediscovery delete-namespace --id $NAMESPACE_ID --
query 'OperationId' --output text 2>/dev/null || echo "")
      if [[ -n "$OPERATION_ID" ]]; then
        echo "Waiting for namespace deletion to complete..." | tee -a $LOG_FILE
        wait_for_operation $OPERATION_ID
      else
        echo "Failed to delete namespace or namespace already deleted" | tee -a
 $LOG_FILE
      fi
    fi
  done

  echo "Cleanup complete" | tee -a $LOG_FILE
}

# Step 1: Create an AWS Cloud Map namespace
echo "Step 1: Creating AWS Cloud Map namespace..." | tee -a $LOG_FILE

# Check if namespace already exists
NAMESPACE_ID=$(aws servicediscovery list-namespaces --query "Namespaces[?
Name=='cloudmap-tutorial'].Id" --output text)

if [[ -z "$NAMESPACE_ID" || "$NAMESPACE_ID" == "None" ]]; then
  log_cmd "aws servicediscovery create-http-namespace --name cloudmap-tutorial --
creator-request-id namespace-request"
  OPERATION_ID=$(aws servicediscovery create-http-namespace --name cloudmap-tutorial
 --creator-request-id namespace-request --query 'OperationId' --output text)

  # Wait for namespace creation to complete
  echo "Waiting for namespace creation to complete..." | tee -a $LOG_FILE
  wait_for_operation $OPERATION_ID

  # Get the namespace ID
```

```
  NAMESPACE_ID=$(aws servicediscovery list-namespaces --query "Namespaces[?
Name=='cloudmap-tutorial'].Id" --output text)
  echo "Namespace created with ID: $NAMESPACE_ID" | tee -a $LOG_FILE
else
  echo "Namespace cloudmap-tutorial already exists with ID: $NAMESPACE_ID" | tee -a
 $LOG_FILE
fi

CREATED_RESOURCES+=("namespace:$NAMESPACE_ID")

# Step 2: Create a DynamoDB table
echo "Step 2: Creating DynamoDB table..." | tee -a $LOG_FILE

# Check if table already exists
TABLE_EXISTS=$(aws dynamodb describe-table --table-name cloudmap 2>&1 || echo
 "NOT_EXISTS")

if [[ $TABLE_EXISTS == *"ResourceNotFoundException"* || $TABLE_EXISTS ==
 "NOT_EXISTS" ]]; then
  log_cmd "aws dynamodb create-table --table-name cloudmap --attribute-definitions
 AttributeName=id,AttributeType=S --key-schema AttributeName=id,KeyType=HASH --
billing-mode PAY_PER_REQUEST"

  # Wait for DynamoDB table to become active
  echo "Waiting for DynamoDB table to become active..." | tee -a $LOG_FILE
  aws dynamodb wait table-exists --table-name cloudmap
else
  echo "DynamoDB table cloudmap already exists" | tee -a $LOG_FILE
fi

CREATED_RESOURCES+=("dynamodb:cloudmap")

# Step 3: Create an AWS Cloud Map data service
echo "Step 3: Creating AWS Cloud Map data service..." | tee -a $LOG_FILE

# Get all services in the namespace
echo "Listing all services in namespace $NAMESPACE_ID..." | tee -a $LOG_FILE
SERVICES=$(aws servicediscovery list-services --filters "Name=NAMESPACE_ID,Values=
$NAMESPACE_ID,Condition=EQ" --query 'Services[*].[Id,Name]' --output text)
echo "Services found: $SERVICES" | tee -a $LOG_FILE

# Check if data service already exists
DATA_SERVICE_ID=""
while read -r id name || [[ -n "$id" ]]; do
```

```
  echo "Checking service: ID=$id, Name=$name" | tee -a $LOG_FILE
  if [[ "$name" == "data-service" ]]; then
    DATA_SERVICE_ID="$id"
    break
  fi
done <<< "$SERVICES"

if [[ -z "$DATA_SERVICE_ID" ]]; then
  echo "Data service does not exist, creating it..." | tee -a $LOG_FILE
  # Create the service and capture the ID directly
  echo "$ aws servicediscovery create-service --name data-service --namespace-id
 $NAMESPACE_ID --creator-request-id data-service-request" | tee -a $LOG_FILE
  CREATE_OUTPUT=$(aws servicediscovery create-service --name data-service --
namespace-id $NAMESPACE_ID --creator-request-id data-service-request)
  echo "$CREATE_OUTPUT" | tee -a $LOG_FILE

  # Extract the service ID using AWS CLI query
  DATA_SERVICE_ID=$(aws servicediscovery list-services --filters
 "Name=NAMESPACE_ID,Values=$NAMESPACE_ID,Condition=EQ" --query "Services[?
Name=='data-service'].Id" --output text)
  echo "Data service created with ID: $DATA_SERVICE_ID" | tee -a $LOG_FILE
else
  echo "Data service already exists with ID: $DATA_SERVICE_ID" | tee -a $LOG_FILE
fi

CREATED_RESOURCES+=("service:$DATA_SERVICE_ID")

# Register DynamoDB table as a service instance
echo "Registering DynamoDB table as a service instance..." | tee -a $LOG_FILE

# Check if instance already exists
INSTANCE_EXISTS=$(aws servicediscovery list-instances --service-id $DATA_SERVICE_ID
 --query "Instances[?Id=='data-instance'].Id" --output text)

if [[ -z "$INSTANCE_EXISTS" ]]; then
  log_cmd "aws servicediscovery register-instance --service-id $DATA_SERVICE_ID --
instance-id data-instance --attributes tablename=cloudmap,region=$(aws configure get
 region)"
  OPERATION_ID=$(aws servicediscovery register-instance --service-
id $DATA_SERVICE_ID --instance-id data-instance --attributes
 tablename=cloudmap,region=$(aws configure get region) --query 'OperationId' --
output text)

  # Wait for instance registration to complete
```

```
  echo "Waiting for instance registration to complete..." | tee -a $LOG_FILE
  wait_for_operation $OPERATION_ID
else
  echo "Instance data-instance already exists" | tee -a $LOG_FILE
fi

CREATED_RESOURCES+=("instance:$DATA_SERVICE_ID:data-instance")

# Step 4: Create an IAM role for Lambda
echo "Step 4: Creating IAM role for Lambda..." | tee -a $LOG_FILE

# Create a trust policy for Lambda
cat > lambda-trust-policy.json << EOF
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "lambda.amazonaws.com"
      },
      "Action": "sts:AssumeRole"
    }
  ]
}
EOF

# Check if role already exists
echo "Checking if IAM role already exists..." | tee -a $LOG_FILE
ROLE_EXISTS=$(aws iam get-role --role-name cloudmap-tutorial-role 2>&1 || echo
 "NOT_EXISTS")

if [[ $ROLE_EXISTS == *"NoSuchEntity"* || $ROLE_EXISTS == "NOT_EXISTS" ]]; then
    log_cmd "aws iam create-role --role-name cloudmap-tutorial-role --assume-role-
policy-document file://lambda-trust-policy.json"
else
    echo "Role cloudmap-tutorial-role already exists, using existing role" | tee -a
 $LOG_FILE
fi

# FIXED: Create a custom policy with least privilege instead of using
 PowerUserAccess
cat > cloudmap-policy.json << EOF
{
```

```
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "logs:CreateLogGroup",
        "logs:CreateLogStream",
        "logs:PutLogEvents"
      ],
      "Resource": "arn:aws:logs:*:*:*"
    },
    {
      "Effect": "Allow",
      "Action": [
        "dynamodb:PutItem",
        "dynamodb:GetItem",
        "dynamodb:Scan"
      ],
      "Resource": "arn:aws:dynamodb:*:*:table/cloudmap"
    },
    {
      "Effect": "Allow",
      "Action": [
        "servicediscovery:DiscoverInstances"
      ],
      "Resource": "*"
    }
  ]
}
EOF

# Check if policy already exists
POLICY_ARN=$(aws iam list-policies --query "Policies[?
PolicyName=='CloudMapTutorialPolicy'].Arn" --output text)

if [[ -z "$POLICY_ARN" ]]; then
  echo "Creating CloudMapTutorialPolicy..." | tee -a $LOG_FILE
  echo "$ aws iam create-policy --policy-name CloudMapTutorialPolicy --policy-
document file://cloudmap-policy.json" | tee -a $LOG_FILE
  CREATE_OUTPUT=$(aws iam create-policy --policy-name CloudMapTutorialPolicy --
policy-document file://cloudmap-policy.json)
  echo "$CREATE_OUTPUT" | tee -a $LOG_FILE
  POLICY_ARN=$(aws iam list-policies --query "Policies[?
PolicyName=='CloudMapTutorialPolicy'].Arn" --output text)
```

```
else
  echo "Policy CloudMapTutorialPolicy already exists with ARN: $POLICY_ARN" | tee -a
 $LOG_FILE
fi

echo "$ aws iam attach-role-policy --role-name cloudmap-tutorial-role --policy-arn
 $POLICY_ARN" | tee -a $LOG_FILE
aws iam attach-role-policy --role-name cloudmap-tutorial-role --policy-arn
 $POLICY_ARN | tee -a $LOG_FILE

echo "$ aws iam attach-role-policy --role-name cloudmap-tutorial-role --policy-
arn arn:aws:iam::aws:policy/service-role/AWSLambdaBasicExecutionRole" | tee -a
 $LOG_FILE
aws iam attach-role-policy --role-name cloudmap-tutorial-role --policy-arn
 arn:aws:iam::aws:policy/service-role/AWSLambdaBasicExecutionRole | tee -a $LOG_FILE

# Wait for role to propagate
echo "Waiting for IAM role to propagate..." | tee -a $LOG_FILE
sleep 10

ROLE_ARN=$(aws iam get-role --role-name cloudmap-tutorial-role --query 'Role.Arn' --
output text)
CREATED_RESOURCES+=("role:cloudmap-tutorial-role")

# Step 5: Create an AWS Cloud Map app service
echo "Step 5: Creating AWS Cloud Map app service..." | tee -a $LOG_FILE

# Get all services in the namespace
SERVICES=$(aws servicediscovery list-services --filters "Name=NAMESPACE_ID,Values=
$NAMESPACE_ID,Condition=EQ" --query 'Services[*].[Id,Name]' --output text)

# Check if app service already exists
APP_SERVICE_ID=""
while read -r id name || [[ -n "$id" ]]; do
  if [[ "$name" == "app-service" ]]; then
    APP_SERVICE_ID="$id"
    break
  fi
done <<< "$SERVICES"

if [[ -z "$APP_SERVICE_ID" ]]; then
  echo "App service does not exist, creating it..." | tee -a $LOG_FILE
  # Create the service and capture the ID directly
```

```
  echo "$ aws servicediscovery create-service --name app-service --namespace-id
 $NAMESPACE_ID --creator-request-id app-service-request" | tee -a $LOG_FILE
  CREATE_OUTPUT=$(aws servicediscovery create-service --name app-service --
namespace-id $NAMESPACE_ID --creator-request-id app-service-request)
  echo "$CREATE_OUTPUT" | tee -a $LOG_FILE

  # Extract the service ID using AWS CLI query
  APP_SERVICE_ID=$(aws servicediscovery list-services --filters
 "Name=NAMESPACE_ID,Values=$NAMESPACE_ID,Condition=EQ" --query "Services[?
Name=='app-service'].Id" --output text)
  echo "App service created with ID: $APP_SERVICE_ID" | tee -a $LOG_FILE
else
  echo "App service already exists with ID: $APP_SERVICE_ID" | tee -a $LOG_FILE
fi

CREATED_RESOURCES+=("service:$APP_SERVICE_ID")

# Step 6: Create a Lambda function to write data
echo "Step 6: Creating Lambda function to write data..." | tee -a $LOG_FILE

# Create Lambda function code
cat > writefunction.py << EOF
import boto3
import json
import random

def lambda_handler(event, context):
    # Use AWS Cloud Map to discover the DynamoDB table
    serviceclient = boto3.client('servicediscovery')

    # Discover the data service instance
    response = serviceclient.discover_instances(
        NamespaceName='cloudmap-tutorial',
        ServiceName='data-service'
    )

    # Extract table name and region from the instance attributes
    tablename = response['Instances'][0]['Attributes']['tablename']
    region = response['Instances'][0]['Attributes']['region']

    # Create DynamoDB client in the specified region
    dynamodb = boto3.resource('dynamodb', region_name=region)
    table = dynamodb.Table(tablename)
```

```
    # Write data to the table
    table.put_item(
        Item={
            'id': str(random.randint(1,100)),
            'todo': event
        }
    )

    return {
        'statusCode': 200,
        'body': json.dumps('Data written successfully!')
    }
EOF

# Zip the function code
log_cmd "zip writefunction.zip writefunction.py"

# Create the Lambda function
FUNCTION_EXISTS=$(aws lambda list-functions --query "Functions[?
FunctionName=='writefunction'].FunctionName" --output text)
if [[ -z "$FUNCTION_EXISTS" ]]; then
  log_cmd "aws lambda create-function --function-name writefunction --runtime
 python3.12 --role $ROLE_ARN --handler writefunction.lambda_handler --zip-file
 fileb://writefunction.zip --architectures x86_64"

  # Wait for the Lambda function to be active before updating
  echo "Waiting for Lambda function to become active..." | tee -a $LOG_FILE
  function_state="Pending"
  while [ "$function_state" == "Pending" ]; do
      sleep 5
      function_state=$(aws lambda get-function --function-name writefunction --query
 'Configuration.State' --output text)
      echo "Current function state: $function_state" | tee -a $LOG_FILE
  done

  # Update the function timeout
  log_cmd "aws lambda update-function-configuration --function-name writefunction --
timeout 5"
else
  echo "Lambda function writefunction already exists" | tee -a $LOG_FILE
fi
CREATED_RESOURCES+=("lambda:writefunction")

# Step 7: Register the Lambda write function as an AWS Cloud Map service instance
```

```
echo "Step 7: Registering Lambda write function as a service instance..." | tee -a
 $LOG_FILE

# Check if instance already exists
INSTANCE_EXISTS=$(aws servicediscovery list-instances --service-id $APP_SERVICE_ID
 --query "Instances[?Id=='write-instance'].Id" --output text)

if [[ -z "$INSTANCE_EXISTS" ]]; then
  log_cmd "aws servicediscovery register-instance --service-id $APP_SERVICE_ID --
instance-id write-instance --attributes action=write,functionname=writefunction"
  OPERATION_ID=$(aws servicediscovery register-instance --service-id $APP_SERVICE_ID
 --instance-id write-instance --attributes action=write,functionname=writefunction
 --query 'OperationId' --output text)

  # Wait for instance registration to complete
  echo "Waiting for write instance registration to complete..." | tee -a $LOG_FILE
  wait_for_operation $OPERATION_ID
else
  echo "Instance write-instance already exists" | tee -a $LOG_FILE
fi

CREATED_RESOURCES+=("instance:$APP_SERVICE_ID:write-instance")

# Step 8: Create a Lambda function to read data
echo "Step 8: Creating Lambda function to read data..." | tee -a $LOG_FILE

# Create Lambda function code
cat > readfunction.py << EOF
import boto3
import json

def lambda_handler(event, context):
    # Use AWS Cloud Map to discover the DynamoDB table
    serviceclient = boto3.client('servicediscovery')

    # Discover the data service instance
    response = serviceclient.discover_instances(
        NamespaceName='cloudmap-tutorial',
        ServiceName='data-service'
    )

    # Extract table name and region from the instance attributes
    tablename = response['Instances'][0]['Attributes']['tablename']
    region = response['Instances'][0]['Attributes']['region']
```

```
    # Create DynamoDB client in the specified region
    dynamodb = boto3.resource('dynamodb', region_name=region)
    table = dynamodb.Table(tablename)

    # Read data from the table
    response = table.scan()

    return {
        'statusCode': 200,
        'body': json.dumps(response['Items'])
    }
EOF

# Zip the function code
log_cmd "zip readfunction.zip readfunction.py"

# Create the Lambda function
FUNCTION_EXISTS=$(aws lambda list-functions --query "Functions[?
FunctionName=='readfunction'].FunctionName" --output text)
if [[ -z "$FUNCTION_EXISTS" ]]; then
  log_cmd "aws lambda create-function --function-name readfunction --runtime
 python3.12 --role $ROLE_ARN --handler readfunction.lambda_handler --zip-file
 fileb://readfunction.zip --architectures x86_64"

  # Wait for the Lambda function to be active before updating
  echo "Waiting for Lambda function to become active..." | tee -a $LOG_FILE
  function_state="Pending"
  while [ "$function_state" == "Pending" ]; do
      sleep 5
      function_state=$(aws lambda get-function --function-name readfunction --query
 'Configuration.State' --output text)
      echo "Current function state: $function_state" | tee -a $LOG_FILE
  done

  # Update the function timeout
  log_cmd "aws lambda update-function-configuration --function-name readfunction --
timeout 5"
else
  echo "Lambda function readfunction already exists" | tee -a $LOG_FILE
fi
CREATED_RESOURCES+=("lambda:readfunction")

# Step 9: Register the Lambda read function as an AWS Cloud Map service instance
```

```
echo "Step 9: Registering Lambda read function as a service instance..." | tee -a
 $LOG_FILE

# Check if instance already exists
INSTANCE_EXISTS=$(aws servicediscovery list-instances --service-id $APP_SERVICE_ID
 --query "Instances[?Id=='read-instance'].Id" --output text)

if [[ -z "$INSTANCE_EXISTS" ]]; then
  log_cmd "aws servicediscovery register-instance --service-id $APP_SERVICE_ID --
instance-id read-instance --attributes action=read,functionname=readfunction"
  OPERATION_ID=$(aws servicediscovery register-instance --service-id $APP_SERVICE_ID
 --instance-id read-instance --attributes action=read,functionname=readfunction --
query 'OperationId' --output text)

  # Wait for read instance registration to complete
  echo "Waiting for read instance registration to complete..." | tee -a $LOG_FILE
  wait_for_operation $OPERATION_ID
else
  echo "Instance read-instance already exists" | tee -a $LOG_FILE
fi

CREATED_RESOURCES+=("instance:$APP_SERVICE_ID:read-instance")

# Step 10: Create Python clients to interact with the services
echo "Step 10: Creating Python clients..." | tee -a $LOG_FILE

cat > writeclient.py << EOF
import boto3

serviceclient = boto3.client('servicediscovery')

response = serviceclient.discover_instances(NamespaceName='cloudmap-tutorial',
 ServiceName='app-service', QueryParameters={ 'action': 'write' })

functionname = response["Instances"][0]["Attributes"]["functionname"]

lambdaclient = boto3.client('lambda')

resp = lambdaclient.invoke(FunctionName=functionname, Payload='"This is a test
 data"')

print(resp["Payload"].read())
EOF
```

```
cat > readclient.py << EOF
import boto3

serviceclient = boto3.client('servicediscovery')

response = serviceclient.discover_instances(NamespaceName='cloudmap-tutorial',
 ServiceName='app-service', QueryParameters={ 'action': 'read' })

functionname = response["Instances"][0]["Attributes"]["functionname"]

lambdaclient = boto3.client('lambda')

resp = lambdaclient.invoke(FunctionName=functionname,
 InvocationType='RequestResponse')

print(resp["Payload"].read())
EOF

echo "Running write client..." | tee -a $LOG_FILE
log_cmd "python3 writeclient.py"

echo "Running read client..." | tee -a $LOG_FILE
log_cmd "python3 readclient.py"

# Step 11: Clean up resources
echo "Resources created:" | tee -a $LOG_FILE
for resource in "${CREATED_RESOURCES[@]}"; do
  echo "- $resource" | tee -a $LOG_FILE
done

echo "" | tee -a $LOG_FILE
echo "========================================" | tee -a $LOG_FILE
echo "CLEANUP CONFIRMATION" | tee -a $LOG_FILE
echo "========================================" | tee -a $LOG_FILE
echo "Do you want to clean up all created resources? (y/n): " | tee -a $LOG_FILE
read -r CLEANUP_CONFIRM
if [[ $CLEANUP_CONFIRM == "y" || $CLEANUP_CONFIRM == "Y" ]]; then
  cleanup
else
  echo "Resources were not cleaned up. You can manually clean them up later." | tee
 -a $LOG_FILE
fi

echo "Script completed at $(date)" | tee -a $LOG_FILE
```

- Per informazioni dettagliate sull'API, consulta i seguenti argomenti nella documentazione di riferimento dei comandi della AWS CLI .
  - CreateHttpNamespace
  - CreateService
  - DeleteNamespace
  - DeleteService
  - DeregisterInstance
  - GetOperation
  - ListNamespaces
  - ListServices
  - RegisterInstance

Rilevamento servizi Cloud Map

L'esempio di codice seguente mostra come:

- Creare un namespace DNS pubblico con l'integrazione delle zone ospitate da Route 53
- Creare servizi rilevabili tramite query DNS e chiamate API
- Registrare le istanze del servizio con diverse configurazioni di rilevamento
- Rilevare i servizi utilizzando sia le query DNS che l'API Cloud Map
- Verificare il rilevamento servizi utilizzando il comando dig e le chiamate API
- Eseguire la pulizia delle risorse nell'ordine corretto (istanze, servizi, namespace)

AWS CLI con lo script Bash

> **Note**
>
> C'è altro da fare. GitHub Trova l'esempio completo e scopri come eseguire la configurazione e l'esecuzione nel repository dei tutorial sugli esempi di codice per gli sviluppatori.

```
#!/bin/bash

# AWS Cloud Map Private Namespace Tutorial Script
# This script demonstrates how to use AWS Cloud Map for service discovery
# with DNS queries and API calls

# Exit on error
set -e

# Configuration
REGION="us-east-2"
NAMESPACE_NAME="cloudmap-tutorial.com"
LOG_FILE="cloudmap-tutorial.log"
CREATOR_REQUEST_ID=$(date +%s)

# Function to log messages
log() {
    local message="$1"
    echo "$(date '+%Y-%m-%d %H:%M:%S') - $message" | tee -a "$LOG_FILE"
}

# Function to check operation status
check_operation() {
    local operation_id="$1"
    local status=""

    log "Checking operation status for $operation_id..."

    while [[ "$status" != "SUCCESS" ]]; do
        sleep 5
        status=$(aws servicediscovery get-operation \
            --operation-id "$operation_id" \
            --region "$REGION" \
            --query "Operation.Status" \
            --output text)

        log "Operation status: $status"

        if [[ "$status" == "FAIL" ]]; then
            log "Operation failed. Exiting."
            exit 1
        fi
    done
```

```
    log "Operation completed successfully."
}

# Function to clean up resources
cleanup() {
    log "Starting cleanup process..."

    if [[ -n "$FIRST_INSTANCE_ID" ]]; then
        log "Deregistering first service instance..."
        aws servicediscovery deregister-instance \
            --service-id "$PUBLIC_SERVICE_ID" \
            --instance-id "$FIRST_INSTANCE_ID" \
            --region "$REGION" || log "Failed to deregister first instance"
    fi

    if [[ -n "$SECOND_INSTANCE_ID" ]]; then
        log "Deregistering second service instance..."
        aws servicediscovery deregister-instance \
            --service-id "$BACKEND_SERVICE_ID" \
            --instance-id "$SECOND_INSTANCE_ID" \
            --region "$REGION" || log "Failed to deregister second instance"
    fi

    if [[ -n "$PUBLIC_SERVICE_ID" ]]; then
        log "Deleting public service..."
        aws servicediscovery delete-service \
            --id "$PUBLIC_SERVICE_ID" \
            --region "$REGION" || log "Failed to delete public service"
    fi

    if [[ -n "$BACKEND_SERVICE_ID" ]]; then
        log "Deleting backend service..."
        aws servicediscovery delete-service \
            --id "$BACKEND_SERVICE_ID" \
            --region "$REGION" || log "Failed to delete backend service"
    fi

    if [[ -n "$NAMESPACE_ID" ]]; then
        log "Deleting namespace..."
        aws servicediscovery delete-namespace \
            --id "$NAMESPACE_ID" \
            --region "$REGION" || log "Failed to delete namespace"
    fi
```

```
    log "Cleanup completed."
}

# Set up trap for cleanup on script exit
trap cleanup EXIT INT TERM

# Initialize log file
> "$LOG_FILE"
log "Starting AWS Cloud Map tutorial script"

# Step 1: Create an AWS Cloud Map namespace
log "Creating AWS Cloud Map namespace: $NAMESPACE_NAME"
OPERATION_RESULT=$(aws servicediscovery create-public-dns-namespace \
    --name "$NAMESPACE_NAME" \
    --creator-request-id "cloudmap-tutorial-$CREATOR_REQUEST_ID" \
    --region "$REGION")

OPERATION_ID=$(echo "$OPERATION_RESULT" | jq -r '.OperationId')
log "Namespace creation initiated. Operation ID: $OPERATION_ID"

# Check operation status
check_operation "$OPERATION_ID"

# Get the namespace ID
log "Getting namespace ID..."
NAMESPACE_ID=$(aws servicediscovery list-namespaces \
    --region "$REGION" \
    --query "Namespaces[?Name=='$NAMESPACE_NAME'].Id" \
    --output text)

log "Namespace ID: $NAMESPACE_ID"

# Get the hosted zone ID
log "Getting Route 53 hosted zone ID..."
HOSTED_ZONE_ID=$(aws route53 list-hosted-zones-by-name \
    --dns-name "$NAMESPACE_NAME" \
    --query "HostedZones[0].Id" \
    --output text | sed 's|/hostedzone/||')

log "Hosted Zone ID: $HOSTED_ZONE_ID"

# Step 2: Create the AWS Cloud Map services
log "Creating public service..."
```

```
PUBLIC_SERVICE_RESULT=$(aws servicediscovery create-service \
    --name "public-service" \
    --namespace-id "$NAMESPACE_ID" \
    --dns-config "RoutingPolicy=MULTIVALUE,DnsRecords=[{Type=A,TTL=300}]" \
    --region "$REGION")

PUBLIC_SERVICE_ID=$(echo "$PUBLIC_SERVICE_RESULT" | jq -r '.Service.Id')
log "Public service created. Service ID: $PUBLIC_SERVICE_ID"

log "Creating backend service..."
BACKEND_SERVICE_RESULT=$(aws servicediscovery create-service \
    --name "backend-service" \
    --namespace-id "$NAMESPACE_ID" \
    --type "HTTP" \
    --region "$REGION")

BACKEND_SERVICE_ID=$(echo "$BACKEND_SERVICE_RESULT" | jq -r '.Service.Id')
log "Backend service created. Service ID: $BACKEND_SERVICE_ID"

# Step 3: Register the AWS Cloud Map service instances
log "Registering first service instance..."
FIRST_INSTANCE_RESULT=$(aws servicediscovery register-instance \
    --service-id "$PUBLIC_SERVICE_ID" \
    --instance-id "first" \
    --attributes "AWS_INSTANCE_IPV4=192.168.2.1" \
    --region "$REGION")

FIRST_INSTANCE_ID="first"
FIRST_OPERATION_ID=$(echo "$FIRST_INSTANCE_RESULT" | jq -r '.OperationId')
log "First instance registration initiated. Operation ID: $FIRST_OPERATION_ID"

# Check operation status
check_operation "$FIRST_OPERATION_ID"

log "Registering second service instance..."
SECOND_INSTANCE_RESULT=$(aws servicediscovery register-instance \
    --service-id "$BACKEND_SERVICE_ID" \
    --instance-id "second" \
    --attributes "service-name=backend" \
    --region "$REGION")

SECOND_INSTANCE_ID="second"
SECOND_OPERATION_ID=$(echo "$SECOND_INSTANCE_RESULT" | jq -r '.OperationId')
log "Second instance registration initiated. Operation ID: $SECOND_OPERATION_ID"
```

```
# Check operation status
check_operation "$SECOND_OPERATION_ID"

# Step 4: Discover the AWS Cloud Map service instances
log "Getting Route 53 name servers..."
NAME_SERVERS=$(aws route53 get-hosted-zone \
    --id "$HOSTED_ZONE_ID" \
    --query "DelegationSet.NameServers[0]" \
    --output text)

log "Name server: $NAME_SERVERS"

log "Using dig to query DNS records (this will be simulated)..."
log "Command: dig @$NAME_SERVERS public-service.$NAMESPACE_NAME"
log "Expected output would show: public-service.$NAMESPACE_NAME. 300 IN A
 192.168.2.1"

log "Using AWS CLI to discover backend service instances..."
DISCOVER_RESULT=$(aws servicediscovery discover-instances \
    --namespace-name "$NAMESPACE_NAME" \
    --service-name "backend-service" \
    --region "$REGION")

log "Discovery result: $(echo "$DISCOVER_RESULT" | jq -c '.')"

# Display created resources
log "Resources created:"
log "- Namespace: $NAMESPACE_NAME (ID: $NAMESPACE_ID)"
log "- Public Service: public-service (ID: $PUBLIC_SERVICE_ID)"
log "- Backend Service: backend-service (ID: $BACKEND_SERVICE_ID)"
log "- Service Instance: first (Service: public-service)"
log "- Service Instance: second (Service: backend-service)"

# Ask user if they want to clean up resources
read -p "Do you want to clean up all created resources? (y/n): " CLEANUP_RESPONSE

if [[ "$CLEANUP_RESPONSE" == "y" || "$CLEANUP_RESPONSE" == "Y" ]]; then
    log "User confirmed cleanup. Proceeding with resource deletion."
    # Cleanup function will be called automatically on exit
else
    log "User chose not to clean up resources. Exiting without cleanup."
    trap - EXIT
    exit 0
```

```
fi
```

- Per informazioni dettagliate sull'API, consulta i seguenti argomenti nella documentazione di riferimento dei comandi della AWS CLI .
  - CreatePublicDnsNamespace
  - CreateService
  - DeleteNamespace
  - DeleteService
  - DeregisterInstance
  - DiscoverInstances
  - GetOperation
  - ListNamespaces
  - RegisterInstance

CloudFront esempi di utilizzo AWS CLI con lo script Bash

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando lo script AWS Command Line Interface with Bash con. CloudFront

Scenari: esempi di codice che mostrano come eseguire un'attività specifica chiamando più funzioni all'interno dello stesso servizio o combinate con altri Servizi AWS.

Ogni esempio include un link al codice sorgente completo, dove è possibile trovare le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Scenari

Scenari

Inizia con CloudFront

L'esempio di codice seguente mostra come:

- Creare un bucket Amazon S3 per l'archiviazione dei contenuti
- Caricare contenuti di esempio nel bucket S3

- Creare un controllo di accesso origine (OAC) per un accesso sicuro a S3
- Crea una CloudFront distribuzione con S3 come origine
- Aggiorna la policy del bucket S3 per consentire l'accesso CloudFront
- Attendere l'implementazione della distribuzione e verificare l'accesso ai contenuti
- Eseguire la pulizia delle risorse, tra cui distribuzione, controllo di accesso origine (OAC) e bucket S3

AWS CLI con lo script Bash

> **Note**
>
> C'è altro da fare. GitHub Trova l'esempio completo e scopri come eseguire la configurazione e l'esecuzione nel repository dei tutorial sugli esempi di codice per gli sviluppatori.

```
#!/bin/bash

# CloudFront Getting Started Tutorial Script
# This script creates an S3 bucket, uploads sample content, creates a CloudFront
 distribution with OAC,
# and demonstrates how to access content through CloudFront.

# Set up logging
LOG_FILE="cloudfront-tutorial.log"
exec > >(tee -a "$LOG_FILE") 2>&1

echo "Starting CloudFront Getting Started Tutorial at $(date)"

# Function to handle errors
handle_error() {
    echo "ERROR: $1"
    echo "Resources created before error:"
    if [ -n "$BUCKET_NAME" ]; then
        echo "- S3 Bucket: $BUCKET_NAME"
    fi
    if [ -n "$OAC_ID" ]; then
        echo "- CloudFront Origin Access Control: $OAC_ID"
    fi
```

```
    if [ -n "$DISTRIBUTION_ID" ]; then
        echo "- CloudFront Distribution: $DISTRIBUTION_ID"
    fi

    echo "Attempting to clean up resources..."
    cleanup
    exit 1
}

# Function to clean up resources
cleanup() {
    echo "Cleaning up resources..."

    if [ -n "$DISTRIBUTION_ID" ]; then
        echo "Disabling CloudFront distribution $DISTRIBUTION_ID..."

        # Get the current configuration and ETag
        ETAG=$(aws cloudfront get-distribution-config --id "$DISTRIBUTION_ID" --
query 'ETag' --output text)
        if [ $? -ne 0 ]; then
            echo "Failed to get distribution config. Continuing with cleanup..."
        else
            # Create a modified configuration with Enabled=false
            aws cloudfront get-distribution-config --id "$DISTRIBUTION_ID" | \
            jq '.DistributionConfig.Enabled = false' > temp_disabled_config.json

            # Update the distribution to disable it
            aws cloudfront update-distribution \
                --id "$DISTRIBUTION_ID" \
                --distribution-config file://<(jq '.DistributionConfig'
 temp_disabled_config.json) \
                --if-match "$ETAG"

            if [ $? -ne 0 ]; then
                echo "Failed to disable distribution. Continuing with cleanup..."
            else
                echo "Waiting for distribution to be disabled (this may take several
 minutes)..."
                aws cloudfront wait distribution-deployed --id "$DISTRIBUTION_ID"

                # Delete the distribution
                ETAG=$(aws cloudfront get-distribution-config --id
 "$DISTRIBUTION_ID" --query 'ETag' --output text)
```

```
                aws cloudfront delete-distribution --id "$DISTRIBUTION_ID" --if-
match "$ETAG"
                if [ $? -ne 0 ]; then
                    echo "Failed to delete distribution. You may need to delete it
 manually."
                else
                    echo "CloudFront distribution deleted."
                fi
            fi
        fi
    fi

    if [ -n "$OAC_ID" ]; then
        echo "Deleting Origin Access Control $OAC_ID..."
        OAC_ETAG=$(aws cloudfront get-origin-access-control --id "$OAC_ID" --query
 'ETag' --output text 2>/dev/null)
        if [ $? -ne 0 ]; then
            echo "Failed to get Origin Access Control ETag. You may need to delete
 it manually."
        else
            aws cloudfront delete-origin-access-control --id "$OAC_ID" --if-match
 "$OAC_ETAG"
            if [ $? -ne 0 ]; then
                echo "Failed to delete Origin Access Control. You may need to delete
 it manually."
            else
                echo "Origin Access Control deleted."
            fi
        fi
    fi

    if [ -n "$BUCKET_NAME" ]; then
        echo "Deleting S3 bucket $BUCKET_NAME and its contents..."
        aws s3 rm "s3://$BUCKET_NAME" --recursive
        if [ $? -ne 0 ]; then
            echo "Failed to remove bucket contents. Continuing with bucket
 deletion..."
        fi

        aws s3 rb "s3://$BUCKET_NAME"
        if [ $? -ne 0 ]; then
            echo "Failed to delete bucket. You may need to delete it manually."
        else
            echo "S3 bucket deleted."
```

```
        fi
    fi

    # Clean up temporary files
    rm -f temp_disabled_config.json
    rm -rf temp_content
}

# Generate a random identifier for the bucket name
RANDOM_ID=$(openssl rand -hex 6)
BUCKET_NAME="cloudfront-${RANDOM_ID}"
echo "Using bucket name: $BUCKET_NAME"

# Create a temporary directory for content
TEMP_DIR="temp_content"
mkdir -p "$TEMP_DIR/css"
if [ $? -ne 0 ]; then
    handle_error "Failed to create temporary directory"
fi

# Step 1: Create an S3 bucket
echo "Creating S3 bucket: $BUCKET_NAME"
aws s3 mb "s3://$BUCKET_NAME"
if [ $? -ne 0 ]; then
    handle_error "Failed to create S3 bucket"
fi

# Step 2: Create sample content
echo "Creating sample content..."
cat > "$TEMP_DIR/index.html" << 'EOF'
<!DOCTYPE html>
<html>
<head>
    <title>Hello World</title>
    <link rel="stylesheet" type="text/css" href="css/styles.css">
</head>
<body>
    <h1>Hello world!</h1>
</body>
</html>
EOF

cat > "$TEMP_DIR/css/styles.css" << 'EOF'
body {
```

```
    font-family: Arial, sans-serif;
    margin: 40px;
    background-color: #f5f5f5;
}
h1 {
    color: #333;
    text-align: center;
}
EOF

# Step 3: Upload content to the S3 bucket
echo "Uploading content to S3 bucket..."
aws s3 cp "$TEMP_DIR/" "s3://$BUCKET_NAME/" --recursive
if [ $? -ne 0 ]; then
    handle_error "Failed to upload content to S3 bucket"
fi

# Step 4: Create Origin Access Control
echo "Creating Origin Access Control..."
OAC_RESPONSE=$(aws cloudfront create-origin-access-control \
    --origin-access-control-config Name="oac-for-
$BUCKET_NAME",SigningProtocol=sigv4,SigningBehavior=always,OriginAccessControlOriginType=s3)

if [ $? -ne 0 ]; then
    handle_error "Failed to create Origin Access Control"
fi

OAC_ID=$(echo "$OAC_RESPONSE" | jq -r '.OriginAccessControl.Id')
echo "Created Origin Access Control with ID: $OAC_ID"

# Step 5: Create CloudFront distribution
echo "Creating CloudFront distribution..."

# Get AWS account ID for bucket policy
ACCOUNT_ID=$(aws sts get-caller-identity --query 'Account' --output text)
if [ $? -ne 0 ]; then
    handle_error "Failed to get AWS account ID"
fi

# Create distribution configuration
cat > distribution-config.json << EOF
{
    "CallerReference": "cli-tutorial-$(date +%s)",
    "Origins": {
```

```
        "Quantity": 1,
        "Items": [
            {
                "Id": "S3-$BUCKET_NAME",
                "DomainName": "$BUCKET_NAME.s3.amazonaws.com",
                "S3OriginConfig": {
                    "OriginAccessIdentity": ""
                },
                "OriginAccessControlId": "$OAC_ID"
            }
        ]
    },
    "DefaultCacheBehavior": {
        "TargetOriginId": "S3-$BUCKET_NAME",
        "ViewerProtocolPolicy": "redirect-to-https",
        "AllowedMethods": {
            "Quantity": 2,
            "Items": ["GET", "HEAD"],
            "CachedMethods": {
                "Quantity": 2,
                "Items": ["GET", "HEAD"]
            }
        },
        "DefaultTTL": 86400,
        "MinTTL": 0,
        "MaxTTL": 31536000,
        "Compress": true,
        "ForwardedValues": {
            "QueryString": false,
            "Cookies": {
                "Forward": "none"
            }
        }
    },
    "Comment": "CloudFront distribution for tutorial",
    "Enabled": true,
    "WebACLId": ""
}
EOF

DIST_RESPONSE=$(aws cloudfront create-distribution --distribution-config file://
distribution-config.json)
if [ $? -ne 0 ]; then
    handle_error "Failed to create CloudFront distribution"
```

```
fi

DISTRIBUTION_ID=$(echo "$DIST_RESPONSE" | jq -r '.Distribution.Id')
DOMAIN_NAME=$(echo "$DIST_RESPONSE" | jq -r '.Distribution.DomainName')

echo "Created CloudFront distribution with ID: $DISTRIBUTION_ID"
echo "CloudFront domain name: $DOMAIN_NAME"

# Step 6: Update S3 bucket policy
echo "Updating S3 bucket policy..."
cat > bucket-policy.json << EOF
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "AllowCloudFrontServicePrincipal",
            "Effect": "Allow",
            "Principal": {
                "Service": "cloudfront.amazonaws.com"
            },
            "Action": "s3:GetObject",
            "Resource": "arn:aws:s3:::$BUCKET_NAME/*",
            "Condition": {
                "StringEquals": {
                    "AWS:SourceArn": "arn:aws:cloudfront::$ACCOUNT_ID:distribution/
$DISTRIBUTION_ID"
                }
            }
        }
    ]
}
EOF

aws s3api put-bucket-policy --bucket "$BUCKET_NAME" --policy file://bucket-
policy.json
if [ $? -ne 0 ]; then
    handle_error "Failed to update S3 bucket policy"
fi

# Step 7: Wait for distribution to deploy
echo "Waiting for CloudFront distribution to deploy (this may take 5-10 minutes)..."
aws cloudfront wait distribution-deployed --id "$DISTRIBUTION_ID"
if [ $? -ne 0 ]; then
```

```
    echo "Warning: Distribution deployment wait timed out. The distribution may
 still be deploying."
else
    echo "CloudFront distribution is now deployed."
fi

# Step 8: Display access information
echo ""
echo "===== CloudFront Distribution Setup Complete ====="
echo "You can access your content at: https://$DOMAIN_NAME/index.html"
echo ""
echo "Resources created:"
echo "- S3 Bucket: $BUCKET_NAME"
echo "- CloudFront Origin Access Control: $OAC_ID"
echo "- CloudFront Distribution: $DISTRIBUTION_ID"
echo ""

# Ask user if they want to clean up resources
read -p "Do you want to clean up all resources created by this script? (y/n): "
 CLEANUP_RESPONSE
if [[ "$CLEANUP_RESPONSE" =~ ^[Yy] ]]; then
    cleanup
    echo "All resources have been cleaned up."
else
    echo "Resources will not be cleaned up. You can manually delete them later."
    echo "To access your content, visit: https://$DOMAIN_NAME/index.html"
fi

echo "Tutorial completed at $(date)"
```

- Per informazioni dettagliate sull'API, consulta i seguenti argomenti nella documentazione di riferimento dei comandi della AWS CLI .
  - CreateDistribution
  - CreateOriginAccessControl
  - DeleteDistribution
  - DeleteOriginAccessControl
  - GetDistribution
  - GetDistributionConfig
  - GetOriginAccessControl

- UpdateDistribution
- WaitDistributionDeployed

Esempi AWS CLI di DynamoDB utilizzati con lo script Bash

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando lo script AWS Command Line Interface with Bash con DynamoDB.

Nozioni di base: esempi di codice che mostrano come eseguire le operazioni essenziali all'interno di un servizio.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Scenari: esempi di codice che mostrano come eseguire un'attività specifica chiamando più funzioni all'interno dello stesso servizio o combinate con altri Servizi AWS.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Nozioni di base
- Azioni
- Scenari

Nozioni di base

Informazioni di base

L'esempio di codice seguente mostra come:

- Crea una tabella in grado di contenere i dati del filmato.
- Inserire, ottenere e aggiornare un singolo filmato nella tabella.
- Scrivere i dati del filmato nella tabella da un file JSON di esempio.
- Eseguire una query sui filmati che sono stati rilasciati in un dato anno.
- Cerca i filmati che sono stati distribuiti in diversi anni.

- Elimina un filmato dalla tabella, quindi elimina la tabella.

AWS CLI con lo script Bash

> **Note**
>
> C'è altro da fare. GitHub Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codice AWS.

Lo scenario introduttivo di DynamoDB.

```
###############################################################################
# function dynamodb_getting_started_movies
#
# Scenario to create an Amazon DynamoDB table and perform a series of operations on
 the table.
#
# Returns:
#       0 - If successful.
#       1 - If an error occurred.
###############################################################################
function dynamodb_getting_started_movies() {

  source ./dynamodb_operations.sh

  key_schema_json_file="dynamodb_key_schema.json"
  attribute_definitions_json_file="dynamodb_attr_def.json"
  item_json_file="movie_item.json"
  key_json_file="movie_key.json"
  batch_json_file="batch.json"
  attribute_names_json_file="attribute_names.json"
  attributes_values_json_file="attribute_values.json"

  echo_repeat "*" 88
  echo
  echo "Welcome to the Amazon DynamoDB getting started demo."
  echo
  echo_repeat "*" 88
  echo

  local table_name
```

```
  echo -n "Enter a name for a new DynamoDB table: "
  get_input
  table_name=$get_input_result

  echo '[
  {"AttributeName": "year", "KeyType": "HASH"},
   {"AttributeName": "title", "KeyType": "RANGE"}
  ]' >"$key_schema_json_file"

  echo '[
  {"AttributeName": "year", "AttributeType": "N"},
   {"AttributeName": "title", "AttributeType": "S"}
  ]' >"$attribute_definitions_json_file"

  if dynamodb_create_table -n "$table_name" -a "$attribute_definitions_json_file" \
    -k "$key_schema_json_file" 1>/dev/null; then
    echo "Created a DynamoDB table named $table_name"
  else
    errecho "The table failed to create. This demo will exit."
    clean_up
    return 1
  fi

  echo "Waiting for the table to become active...."

  if dynamodb_wait_table_active -n "$table_name"; then
    echo "The table is now active."
  else
    errecho "The table failed to become active. This demo will exit."
    cleanup "$table_name"
    return 1
  fi

  echo
  echo_repeat "*" 88
  echo

  echo -n "Enter the title of a movie you want to add to the table: "
  get_input
  local added_title
  added_title=$get_input_result

  local added_year
  get_int_input "What year was it released? "
```

```
  added_year=$get_input_result

  local rating
  get_float_input "On a scale of 1 - 10, how do you rate it? " "1" "10"
  rating=$get_input_result

  local plot
  echo -n "Summarize the plot for me: "
  get_input
  plot=$get_input_result

  echo '{
    "year": {"N" :"'"$added_year"'"},
    "title": {"S" :  "'"$added_title"'"},
    "info": {"M" : {"plot": {"S" : "'"$plot"'"}, "rating": {"N" :"'"$rating"'"} } }
   }' >"$item_json_file"

  if dynamodb_put_item -n "$table_name" -i "$item_json_file"; then
    echo "The movie '$added_title' was successfully added to the table
'$table_name'."
  else
    errecho "Put item failed. This demo will exit."
    clean_up "$table_name"
    return 1
  fi

  echo
  echo_repeat "*" 88
  echo

  echo "Let's update your movie '$added_title'."
  get_float_input "You rated it $rating, what new rating would you give it? " "1"
"10"
  rating=$get_input_result

  echo -n "You summarized the plot as '$plot'."
  echo "What would you say now? "
  get_input
  plot=$get_input_result

  echo '{
    "year": {"N" :"'"$added_year"'"},
    "title": {"S" :  "'"$added_title"'"}
    }' >"$key_json_file"
```

```
  echo '{
    ":r": {"N" :"'"$rating"'"},
    ":p": {"S" : "'"$plot"'"}
   }' >"$item_json_file"

  local update_expression="SET info.rating = :r, info.plot = :p"

  if dynamodb_update_item -n "$table_name" -k "$key_json_file" -e
"$update_expression" -v "$item_json_file"; then
    echo "Updated '$added_title' with new attributes."
  else
    errecho "Update item failed. This demo will exit."
    clean_up "$table_name"
    return 1
  fi

  echo
  echo_repeat "*" 88
  echo

  echo "We will now use batch write to upload 150 movie entries into the table."

  local batch_json
  for batch_json in movie_files/movies_*.json; do
    echo "{ \"$table_name\" : $(<"$batch_json") }" >"$batch_json_file"
    if dynamodb_batch_write_item -i "$batch_json_file" 1>/dev/null; then
      echo "Entries in $batch_json added to table."
    else
      errecho "Batch write failed. This demo will exit."
      clean_up "$table_name"
      return 1
    fi
  done

  local title="The Lord of the Rings: The Fellowship of the Ring"
  local year="2001"

  if get_yes_no_input "Let's move on...do you want to get info about '$title'? (y/n)
"; then
    echo '{
  "year": {"N" :"'"$year"'"},
  "title": {"S" :  "'"$title"'"}
  }' >"$key_json_file"
```

```
    local info
    info=$(dynamodb_get_item -n "$table_name" -k "$key_json_file")

    # shellcheck disable=SC2181
    if [[ ${?} -ne 0 ]]; then
      errecho "Get item failed. This demo will exit."
      clean_up "$table_name"
      return 1
    fi

    echo "Here is what I found:"
    echo "$info"
  fi

  local ask_for_year=true
  while [[ "$ask_for_year" == true ]]; do
    echo "Let's get a list of movies released in a given year."
    get_int_input "Enter a year between 1972 and 2018: " "1972" "2018"
    year=$get_input_result
    echo '{
    "#n": "year"
    }' >"$attribute_names_json_file"

    echo '{
    ":v": {"N" :"'"$year"'"}
    }' >"$attributes_values_json_file"

    response=$(dynamodb_query -n "$table_name" -k "#n=:v" -a
"$attribute_names_json_file" -v "$attributes_values_json_file")

    # shellcheck disable=SC2181
    if [[ ${?} -ne 0 ]]; then
      errecho "Query table failed. This demo will exit."
      clean_up "$table_name"
      return 1
    fi

    echo "Here is what I found:"
    echo "$response"

    if ! get_yes_no_input "Try another year? (y/n) "; then
      ask_for_year=false
    fi
  done
```

```
  echo "Now let's scan for movies released in a range of years. Enter a year: "
  get_int_input "Enter a year between 1972 and 2018: " "1972" "2018"
  local start=$get_input_result

  get_int_input "Enter another year: " "1972" "2018"
  local end=$get_input_result

  echo '{
    "#n": "year"
    }' >"$attribute_names_json_file"

  echo '{
    ":v1": {"N" : "'"$start"'"},
    ":v2": {"N" : "'"$end"'"}
    }' >"$attributes_values_json_file"

  response=$(dynamodb_scan -n "$table_name" -f "#n BETWEEN :v1 AND :v2" -a
"$attribute_names_json_file" -v "$attributes_values_json_file")

  # shellcheck disable=SC2181
  if [[ ${?} -ne 0 ]]; then
    errecho "Scan table failed. This demo will exit."
    clean_up "$table_name"
    return 1
  fi

  echo "Here is what I found:"
  echo "$response"

  echo
  echo_repeat "*" 88
  echo

  echo "Let's remove your movie '$added_title' from the table."

  if get_yes_no_input "Do you want to remove '$added_title'? (y/n) "; then
    echo '{
  "year": {"N" :"'"$added_year"'"},
  "title": {"S" :  "'"$added_title"'"}
  }' >"$key_json_file"

    if ! dynamodb_delete_item -n "$table_name" -k "$key_json_file"; then
      errecho "Delete item failed. This demo will exit."
```

```
      clean_up "$table_name"
      return 1
    fi
  fi

  if get_yes_no_input "Do you want to delete the table '$table_name'? (y/n) "; then
    if ! clean_up "$table_name"; then
      return 1
    fi
  else
    if ! clean_up; then
      return 1
    fi
  fi

  return 0
}
```

Le funzioni DynamoDB utilizzate in questo scenario.

```
###############################################################################
# function dynamodb_create_table
#
# This function creates an Amazon DynamoDB table.
#
# Parameters:
#       -n table_name  -- The name of the table to create.
#       -a attribute_definitions -- JSON file path of a list of attributes and their
 types.
#       -k key_schema -- JSON file path of a list of attributes and their key types.
#
#  Returns:
#       0 - If successful.
#       1 - If it fails.
###############################################################################
function dynamodb_create_table() {
  local table_name attribute_definitions key_schema response
  local option OPTARG # Required to use getopts command in a function.

  ######################################
  # Function usage explanation
  ######################################
```

```
 function usage() {
   echo "function dynamodb_create_table"
   echo "Creates an Amazon DynamoDB table with on-demand billing."
   echo " -n table_name  -- The name of the table to create."
   echo " -a attribute_definitions -- JSON file path of a list of attributes and
their types."
   echo " -k key_schema -- JSON file path of a list of attributes and their key
types."
   echo ""
 }

 # Retrieve the calling parameters.
 while getopts "n:a:k:h" option; do
   case "${option}" in
     n) table_name="${OPTARG}" ;;
     a) attribute_definitions="${OPTARG}" ;;
     k) key_schema="${OPTARG}" ;;
     h)
       usage
       return 0
       ;;
     \?)
       echo "Invalid parameter"
       usage
       return 1
       ;;
   esac
 done
 export OPTIND=1

 if [[ -z "$table_name" ]]; then
   errecho "ERROR: You must provide a table name with the -n parameter."
   usage
   return 1
 fi

 if [[ -z "$attribute_definitions" ]]; then
   errecho "ERROR: You must provide an attribute definitions json file path the -a
parameter."
   usage
   return 1
 fi

 if [[ -z "$key_schema" ]]; then
```

```
    errecho "ERROR: You must provide a key schema json file path the -k parameter."
    usage
    return 1
  fi

  iecho "Parameters:\n"
  iecho "    table_name:   $table_name"
  iecho "    attribute_definitions:   $attribute_definitions"
  iecho "    key_schema:   $key_schema"
  iecho ""

  response=$(aws dynamodb create-table \
    --table-name "$table_name" \
    --attribute-definitions file://"$attribute_definitions" \
    --billing-mode PAY_PER_REQUEST \
    --key-schema file://"$key_schema" )

  local error_code=${?}

  if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports create-table operation failed.$response"
    return 1
  fi

  return 0
}

###############################################################################
# function dynamodb_describe_table
#
# This function returns the status of a DynamoDB table.
#
# Parameters:
#       -n table_name  -- The name of the table.
#
#  Response:
#       - TableStatus:
#     And:
#       0 - Table is active.
#       1 - If it fails.
###############################################################################
function dynamodb_describe_table {
  local table_name
```

```
  local option OPTARG # Required to use getopts command in a function.

  ######################################
  # Function usage explanation
  ######################################
  function usage() {
    echo "function dynamodb_describe_table"
    echo "Describe the status of a DynamoDB table."
    echo "  -n table_name  -- The name of the table."
    echo ""
  }

  # Retrieve the calling parameters.
  while getopts "n:h" option; do
    case "${option}" in
      n) table_name="${OPTARG}" ;;
      h)
        usage
        return 0
        ;;
      \?)
        echo "Invalid parameter"
        usage
        return 1
        ;;
    esac
  done
  export OPTIND=1

  if [[ -z "$table_name" ]]; then
    errecho "ERROR: You must provide a table name with the -n parameter."
    usage
    return 1
  fi

  local table_status
    table_status=$(
      aws dynamodb describe-table \
        --table-name "$table_name" \
        --output text \
        --query 'Table.TableStatus'
    )

   local error_code=${?}
```

```
    if [[ $error_code -ne 0 ]]; then
      aws_cli_error_log "$error_code"
      errecho "ERROR: AWS reports describe-table operation failed.$table_status"
      return 1
    fi

  echo "$table_status"

  return 0
}

###############################################################################
# function dynamodb_put_item
#
# This function puts an item into a DynamoDB table.
#
# Parameters:
#       -n table_name  -- The name of the table.
#       -i item  -- Path to json file containing the item values.
#
#  Returns:
#       0 - If successful.
#       1 - If it fails.
###############################################################################
function dynamodb_put_item() {
  local table_name item response
  local option OPTARG # Required to use getopts command in a function.

  ######################################
  # Function usage explanation
  ######################################
  function usage() {
    echo "function dynamodb_put_item"
    echo "Put an item into a DynamoDB table."
    echo " -n table_name  -- The name of the table."
    echo " -i item  -- Path to json file containing the item values."
    echo ""
  }

  while getopts "n:i:h" option; do
    case "${option}" in
      n) table_name="${OPTARG}" ;;
      i) item="${OPTARG}" ;;
```

```
    h)
      usage
      return 0
      ;;
    \?)
      echo "Invalid parameter"
      usage
      return 1
      ;;
  esac
done
export OPTIND=1

if [[ -z "$table_name" ]]; then
  errecho "ERROR: You must provide a table name with the -n parameter."
  usage
  return 1
fi

if [[ -z "$item" ]]; then
  errecho "ERROR: You must provide an item with the -i parameter."
  usage
  return 1
fi

iecho "Parameters:\n"
iecho "    table_name:   $table_name"
iecho "    item:   $item"
iecho ""
iecho ""

response=$(aws dynamodb put-item \
  --table-name "$table_name" \
  --item file://"$item")

local error_code=${?}

if [[ $error_code -ne 0 ]]; then
  aws_cli_error_log $error_code
  errecho "ERROR: AWS reports put-item operation failed.$response"
  return 1
fi

return 0
```

```
}

###############################################################################
# function dynamodb_update_item
#
# This function updates an item in a DynamoDB table.
#
#
# Parameters:
#       -n table_name  -- The name of the table.
#       -k keys  -- Path to json file containing the keys that identify the item to
 update.
#       -e update expression  -- An expression that defines one or more attributes
 to be updated.
#       -v values  -- Path to json file containing the update values.
#
#  Returns:
#       0 - If successful.
#       1 - If it fails.
###############################################################################
function dynamodb_update_item() {
  local table_name keys update_expression values response
  local option OPTARG # Required to use getopts command in a function.

  ######################################
  # Function usage explanation
  ######################################
  function usage() {
    echo "function dynamodb_update_item"
    echo "Update an item in a DynamoDB table."
    echo " -n table_name  -- The name of the table."
    echo " -k keys  -- Path to json file containing the keys that identify the item
 to update."
    echo " -e update expression  -- An expression that defines one or more
 attributes to be updated."
    echo " -v values  -- Path to json file containing the update values."
    echo ""
  }

  while getopts "n:k:e:v:h" option; do
    case "${option}" in
      n) table_name="${OPTARG}" ;;
      k) keys="${OPTARG}" ;;
```

```
    e) update_expression="${OPTARG}" ;;
    v) values="${OPTARG}" ;;
    h)
      usage
      return 0
      ;;
    \?)
      echo "Invalid parameter"
      usage
      return 1
      ;;
  esac
done
export OPTIND=1

if [[ -z "$table_name" ]]; then
  errecho "ERROR: You must provide a table name with the -n parameter."
  usage
  return 1
fi

if [[ -z "$keys" ]]; then
  errecho "ERROR: You must provide a keys json file path the -k parameter."
  usage
  return 1
fi
if [[ -z "$update_expression" ]]; then
  errecho "ERROR: You must provide an update expression with the -e parameter."
  usage
  return 1
fi

if [[ -z "$values" ]]; then
  errecho "ERROR: You must provide a values json file path the -v parameter."
  usage
  return 1
fi

iecho "Parameters:\n"
iecho "    table_name:   $table_name"
iecho "    keys:   $keys"
iecho "    update_expression:   $update_expression"
iecho "    values:   $values"
```

```
  response=$(aws dynamodb update-item \
    --table-name "$table_name" \
    --key file://"$keys" \
    --update-expression "$update_expression" \
    --expression-attribute-values file://"$values")

  local error_code=${?}

  if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports update-item operation failed.$response"
    return 1
  fi

  return 0

}

###############################################################################
# function dynamodb_batch_write_item
#
# This function writes a batch of items into a DynamoDB table.
#
# Parameters:
#       -i item  -- Path to json file containing the items to write.
#
#  Returns:
#       0 - If successful.
#       1 - If it fails.
###############################################################################
function dynamodb_batch_write_item() {
  local item response
  local option OPTARG # Required to use getopts command in a function.

  ######################################
  # Function usage explanation
  ######################################
  function usage() {
    echo "function dynamodb_batch_write_item"
    echo "Write a batch of items into a DynamoDB table."
    echo " -i item  -- Path to json file containing the items to write."
    echo ""
  }
  while getopts "i:h" option; do
```

```
    case "${option}" in
      i) item="${OPTARG}" ;;
      h)
        usage
        return 0
        ;;
      \?)
        echo "Invalid parameter"
        usage
        return 1
        ;;
    esac
  done
  export OPTIND=1

  if [[ -z "$item" ]]; then
    errecho "ERROR: You must provide an item with the -i parameter."
    usage
    return 1
  fi

  iecho "Parameters:\n"
  iecho "    table_name:   $table_name"
  iecho "    item:   $item"
  iecho ""

  response=$(aws dynamodb batch-write-item \
    --request-items file://"$item")

  local error_code=${?}

  if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports batch-write-item operation failed.$response"
    return 1
  fi

  return 0
}

###############################################################################
# function dynamodb_get_item
#
# This function gets an item from a DynamoDB table.
```

```
#
# Parameters:
#       -n table_name  -- The name of the table.
#       -k keys  -- Path to json file containing the keys that identify the item to
 get.
#       [-q query]  -- Optional JMESPath query expression.
#
#  Returns:
#       The item as text output.
#  And:
#       0 - If successful.
#       1 - If it fails.
###############################################################################
function dynamodb_get_item() {
  local table_name keys query response
  local option OPTARG # Required to use getopts command in a function.

  # ######################################
  # Function usage explanation
  ######################################
  function usage() {
    echo "function dynamodb_get_item"
    echo "Get an item from a DynamoDB table."
    echo " -n table_name  -- The name of the table."
    echo " -k keys  -- Path to json file containing the keys that identify the item
 to get."
    echo " [-q query]  -- Optional JMESPath query expression."
    echo ""
  }
  query=""
  while getopts "n:k:q:h" option; do
    case "${option}" in
      n) table_name="${OPTARG}" ;;
      k) keys="${OPTARG}" ;;
      q) query="${OPTARG}" ;;
      h)
        usage
        return 0
        ;;
      \?)
        echo "Invalid parameter"
        usage
        return 1
        ;;
```

```
    esac
  done
  export OPTIND=1

  if [[ -z "$table_name" ]]; then
    errecho "ERROR: You must provide a table name with the -n parameter."
    usage
    return 1
  fi

  if [[ -z "$keys" ]]; then
    errecho "ERROR: You must provide a keys json file path the -k parameter."
    usage
    return 1
  fi

  if [[ -n "$query" ]]; then
    response=$(aws dynamodb get-item \
      --table-name "$table_name" \
      --key file://"$keys" \
      --output text \
      --query "$query")
  else
    response=$(
      aws dynamodb get-item \
        --table-name "$table_name" \
        --key file://"$keys" \
        --output text
    )
  fi

  local error_code=${?}

  if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports get-item operation failed.$response"
    return 1
  fi

  if [[ -n "$query" ]]; then
    echo "$response" | sed "/^\t/s/\t//1" # Remove initial tab that the JMSEPath
query inserts on some strings.
  else
    echo "$response"
```

```
  fi

  return 0
}

###############################################################################
# function dynamodb_query
#
# This function queries a DynamoDB table.
#
# Parameters:
#       -n table_name  -- The name of the table.
#       -k key_condition_expression -- The key condition expression.
#       -a attribute_names -- Path to JSON file containing the attribute names.
#       -v attribute_values -- Path to JSON file containing the attribute values.
#       [-p projection_expression]  -- Optional projection expression.
#
#  Returns:
#       The items as json output.
#  And:
#       0 - If successful.
#       1 - If it fails.
###############################################################################
function dynamodb_query() {
  local table_name key_condition_expression attribute_names attribute_values
 projection_expression response
  local option OPTARG # Required to use getopts command in a function.

  # ######################################
  # Function usage explanation
  ######################################
  function usage() {
    echo "function dynamodb_query"
    echo "Query a DynamoDB table."
    echo " -n table_name  -- The name of the table."
    echo " -k key_condition_expression -- The key condition expression."
    echo " -a attribute_names -- Path to JSON file containing the attribute names."
    echo " -v attribute_values -- Path to JSON file containing the attribute
 values."
    echo " [-p projection_expression]  -- Optional projection expression."
    echo ""
  }

  while getopts "n:k:a:v:p:h" option; do
```

```
    case "${option}" in
      n) table_name="${OPTARG}" ;;
      k) key_condition_expression="${OPTARG}" ;;
      a) attribute_names="${OPTARG}" ;;
      v) attribute_values="${OPTARG}" ;;
      p) projection_expression="${OPTARG}" ;;
      h)
        usage
        return 0
        ;;
      \?)
        echo "Invalid parameter"
        usage
        return 1
        ;;
    esac
  done
  export OPTIND=1

  if [[ -z "$table_name" ]]; then
    errecho "ERROR: You must provide a table name with the -n parameter."
    usage
    return 1
  fi

  if [[ -z "$key_condition_expression" ]]; then
    errecho "ERROR: You must provide a key condition expression with the -k
parameter."
    usage
    return 1
  fi

  if [[ -z "$attribute_names" ]]; then
    errecho "ERROR: You must provide a attribute names with the -a parameter."
    usage
    return 1
  fi

  if [[ -z "$attribute_values" ]]; then
    errecho "ERROR: You must provide a attribute values with the -v parameter."
    usage
    return 1
  fi
```

```
  if [[ -z "$projection_expression" ]]; then
    response=$(aws dynamodb query \
      --table-name "$table_name" \
      --key-condition-expression "$key_condition_expression" \
      --expression-attribute-names file://"$attribute_names" \
      --expression-attribute-values file://"$attribute_values")
  else
    response=$(aws dynamodb query \
      --table-name "$table_name" \
      --key-condition-expression "$key_condition_expression" \
      --expression-attribute-names file://"$attribute_names" \
      --expression-attribute-values file://"$attribute_values" \
      --projection-expression "$projection_expression")
  fi

  local error_code=${?}

  if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports query operation failed.$response"
    return 1
  fi

  echo "$response"

  return 0
}

###############################################################################
# function dynamodb_scan
#
# This function scans a DynamoDB table.
#
# Parameters:
#       -n table_name  -- The name of the table.
#       -f filter_expression  -- The filter expression.
#       -a expression_attribute_names -- Path to JSON file containing the expression
 attribute names.
#       -v expression_attribute_values -- Path to JSON file containing the
 expression attribute values.
#       [-p projection_expression]  -- Optional projection expression.
#
#  Returns:
#       The items as json output.
```

```
#  And:
#       0 - If successful.
#       1 - If it fails.
###############################################################################
function dynamodb_scan() {
  local table_name filter_expression expression_attribute_names
 expression_attribute_values projection_expression response
  local option OPTARG # Required to use getopts command in a function.

  # ######################################
  # Function usage explanation
  ######################################
  function usage() {
    echo "function dynamodb_scan"
    echo "Scan a DynamoDB table."
    echo " -n table_name  -- The name of the table."
    echo " -f filter_expression  -- The filter expression."
    echo " -a expression_attribute_names -- Path to JSON file containing the
 expression attribute names."
    echo " -v expression_attribute_values -- Path to JSON file containing the
 expression attribute values."
    echo " [-p projection_expression]  -- Optional projection expression."
    echo ""
  }

  while getopts "n:f:a:v:p:h" option; do
    case "${option}" in
      n) table_name="${OPTARG}" ;;
      f) filter_expression="${OPTARG}" ;;
      a) expression_attribute_names="${OPTARG}" ;;
      v) expression_attribute_values="${OPTARG}" ;;
      p) projection_expression="${OPTARG}" ;;
      h)
        usage
        return 0
        ;;
      \?)
        echo "Invalid parameter"
        usage
        return 1
        ;;
    esac
  done
  export OPTIND=1
```

```
 if [[ -z "$table_name" ]]; then
   errecho "ERROR: You must provide a table name with the -n parameter."
   usage
   return 1
 fi

 if [[ -z "$filter_expression" ]]; then
   errecho "ERROR: You must provide a filter expression with the -f parameter."
   usage
   return 1
 fi

 if [[ -z "$expression_attribute_names" ]]; then
   errecho "ERROR: You must provide expression attribute names with the -a
parameter."
   usage
   return 1
 fi

 if [[ -z "$expression_attribute_values" ]]; then
   errecho "ERROR: You must provide expression attribute values with the -v
parameter."
   usage
   return 1
 fi

 if [[ -z "$projection_expression" ]]; then
   response=$(aws dynamodb scan \
     --table-name "$table_name" \
     --filter-expression "$filter_expression" \
     --expression-attribute-names file://"$expression_attribute_names" \
     --expression-attribute-values file://"$expression_attribute_values")
 else
   response=$(aws dynamodb scan \
     --table-name "$table_name" \
     --filter-expression "$filter_expression" \
     --expression-attribute-names file://"$expression_attribute_names" \
     --expression-attribute-values file://"$expression_attribute_values" \
     --projection-expression "$projection_expression")
 fi

 local error_code=${?}
```

```
  if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports scan operation failed.$response"
    return 1
  fi

  echo "$response"

  return 0
}

###############################################################################
# function dynamodb_delete_item
#
# This function deletes an item from a DynamoDB table.
#
# Parameters:
#       -n table_name  -- The name of the table.
#       -k keys  -- Path to json file containing the keys that identify the item to
 delete.
#
#  Returns:
#       0 - If successful.
#       1 - If it fails.
###############################################################################
function dynamodb_delete_item() {
  local table_name keys response
  local option OPTARG # Required to use getopts command in a function.

  # ######################################
  # Function usage explanation
  ######################################
  function usage() {
    echo "function dynamodb_delete_item"
    echo "Delete an item from a DynamoDB table."
    echo " -n table_name  -- The name of the table."
    echo " -k keys  -- Path to json file containing the keys that identify the item
 to delete."
    echo ""
  }
  while getopts "n:k:h" option; do
    case "${option}" in
      n) table_name="${OPTARG}" ;;
      k) keys="${OPTARG}" ;;
```

```
    h)
      usage
      return 0
      ;;
    \?)
      echo "Invalid parameter"
      usage
      return 1
      ;;
  esac
done
export OPTIND=1

if [[ -z "$table_name" ]]; then
  errecho "ERROR: You must provide a table name with the -n parameter."
  usage
  return 1
fi

if [[ -z "$keys" ]]; then
  errecho "ERROR: You must provide a keys json file path the -k parameter."
  usage
  return 1
fi

iecho "Parameters:\n"
iecho "    table_name:   $table_name"
iecho "    keys:   $keys"
iecho ""

response=$(aws dynamodb delete-item \
  --table-name "$table_name" \
  --key file://"$keys")

local error_code=${?}

if [[ $error_code -ne 0 ]]; then
  aws_cli_error_log $error_code
  errecho "ERROR: AWS reports delete-item operation failed.$response"
  return 1
fi

return 0
```

```
}

###############################################################################
# function dynamodb_delete_table
#
# This function deletes a DynamoDB table.
#
# Parameters:
#       -n table_name  -- The name of the table to delete.
#
#  Returns:
#       0 - If successful.
#       1 - If it fails.
###############################################################################
function dynamodb_delete_table() {
  local table_name response
  local option OPTARG # Required to use getopts command in a function.

  # bashsupport disable=BP5008
  function usage() {
    echo "function dynamodb_delete_table"
    echo "Deletes an Amazon DynamoDB table."
    echo " -n table_name  -- The name of the table to delete."
    echo ""
  }

  # Retrieve the calling parameters.
  while getopts "n:h" option; do
    case "${option}" in
      n) table_name="${OPTARG}" ;;
      h)
        usage
        return 0
        ;;
      \?)
        echo "Invalid parameter"
        usage
        return 1
        ;;
    esac
  done
  export OPTIND=1

  if [[ -z "$table_name" ]]; then
```

```
    errecho "ERROR: You must provide a table name with the -n parameter."
    usage
    return 1
  fi

  iecho "Parameters:\n"
  iecho "    table_name:   $table_name"
  iecho ""

  response=$(aws dynamodb delete-table \
    --table-name "$table_name")

  local error_code=${?}

  if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports delete-table operation failed.$response"
    return 1
  fi

  return 0
}
```

Le funzioni di utility utilizzate in questo scenario.

```
###############################################################################
# function iecho
#
# This function enables the script to display the specified text only if
# the global variable $VERBOSE is set to true.
###############################################################################
function iecho() {
  if [[ $VERBOSE == true ]]; then
    echo "$@"
  fi
}

###############################################################################
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
###############################################################################
```

```
function errecho() {
  printf "%s\n" "$*" 1>&2
}

###############################################################################
# function aws_cli_error_log()
#
# This function is used to log the error messages from the AWS CLI.
#
# See https://docs.aws.amazon.com/cli/latest/topic/return-codes.html#cli-aws-help-
return-codes.
#
# The function expects the following argument:
#         $1 - The error code returned by the AWS CLI.
#
#  Returns:
#          0: - Success.
#
###############################################################################
function aws_cli_error_log() {
  local err_code=$1
  errecho "Error code : $err_code"
  if [ "$err_code" == 1 ]; then
    errecho "  One or more S3 transfers failed."
  elif [ "$err_code" == 2 ]; then
    errecho "  Command line failed to parse."
  elif [ "$err_code" == 130 ]; then
    errecho "  Process received SIGINT."
  elif [ "$err_code" == 252 ]; then
    errecho "  Command syntax invalid."
  elif [ "$err_code" == 253 ]; then
    errecho "  The system environment or configuration was invalid."
  elif [ "$err_code" == 254 ]; then
    errecho "  The service returned an error."
  elif [ "$err_code" == 255 ]; then
    errecho "  255 is a catch-all error."
  fi

  return 0
}
```

- Per informazioni dettagliate sull'API, consulta i seguenti argomenti nella documentazione di riferimento dei comandi della AWS CLI .
  - BatchWriteItem
  - CreateTable
  - DeleteItem
  - DeleteTable
  - DescribeTable
  - GetItem
  - PutItem
  - Query
  - Scan
  - UpdateItem

Azioni

`BatchGetItem`

Il seguente esempio di codice mostra come usare`BatchGetItem`.

AWS CLI con lo script Bash

> **Note**
>
> C'è altro da fare. GitHub Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codice AWS.

```
###############################################################################
# function dynamodb_batch_get_item
#
# This function gets a batch of items from a DynamoDB table.
#
# Parameters:
#       -i item  -- Path to json file containing the keys of the items to get.
#
#  Returns:
#       The items as json output.
```

```
#  And:
#       0 - If successful.
#       1 - If it fails.
###############################################################################
function dynamodb_batch_get_item() {
  local item response
  local option OPTARG # Required to use getopts command in a function.

  ######################################
  # Function usage explanation
  ######################################
  function usage() {
    echo "function dynamodb_batch_get_item"
    echo "Get a batch of items from a DynamoDB table."
    echo " -i item  -- Path to json file containing the keys of the items to get."
    echo ""
  }

  while getopts "i:h" option; do
    case "${option}" in
      i) item="${OPTARG}" ;;
      h)
        usage
        return 0
        ;;
      \?)
        echo "Invalid parameter"
        usage
        return 1
        ;;
    esac
  done
  export OPTIND=1

  if [[ -z "$item" ]]; then
    errecho "ERROR: You must provide an item with the -i parameter."
    usage
    return 1
  fi

  response=$(aws dynamodb batch-get-item \
    --request-items file://"$item")
  local error_code=${?}
```

```
  if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports batch-get-item operation failed.$response"
    return 1
  fi

  echo "$response"

  return 0
}
```

Le funzioni di utilità utilizzate in questo esempio.

```
###############################################################################
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
###############################################################################
function errecho() {
  printf "%s\n" "$*" 1>&2
}

###############################################################################
# function aws_cli_error_log()
#
# This function is used to log the error messages from the AWS CLI.
#
# See https://docs.aws.amazon.com/cli/latest/topic/return-codes.html#cli-aws-help-
return-codes.
#
# The function expects the following argument:
#         $1 - The error code returned by the AWS CLI.
#
#  Returns:
#          0: - Success.
#
###############################################################################
function aws_cli_error_log() {
  local err_code=$1
  errecho "Error code : $err_code"
  if [ "$err_code" == 1 ]; then
    errecho "  One or more S3 transfers failed."
```

```
  elif [ "$err_code" == 2 ]; then
    errecho "  Command line failed to parse."
  elif [ "$err_code" == 130 ]; then
    errecho "  Process received SIGINT."
  elif [ "$err_code" == 252 ]; then
    errecho "  Command syntax invalid."
  elif [ "$err_code" == 253 ]; then
    errecho "  The system environment or configuration was invalid."
  elif [ "$err_code" == 254 ]; then
    errecho "  The service returned an error."
  elif [ "$err_code" == 255 ]; then
    errecho "  255 is a catch-all error."
  fi

  return 0
}
```

- Per i dettagli sull'API, consulta BatchGetItem AWS CLICommand Reference.

`BatchWriteItem`

Il seguente esempio di codice mostra come utilizzare`BatchWriteItem`.

AWS CLI con lo script Bash

> **Note**
>
> C'è altro da fare. GitHub Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codice AWS.

```
###############################################################################
# function dynamodb_batch_write_item
#
# This function writes a batch of items into a DynamoDB table.
#
# Parameters:
#       -i item  -- Path to json file containing the items to write.
#
#  Returns:
#       0 - If successful.
```

```
#       1 - If it fails.
###############################################################################
function dynamodb_batch_write_item() {
  local item response
  local option OPTARG # Required to use getopts command in a function.

  ######################################
  # Function usage explanation
  ######################################
  function usage() {
    echo "function dynamodb_batch_write_item"
    echo "Write a batch of items into a DynamoDB table."
    echo " -i item  -- Path to json file containing the items to write."
    echo ""
  }
  while getopts "i:h" option; do
    case "${option}" in
      i) item="${OPTARG}" ;;
      h)
        usage
        return 0
        ;;
      \?)
        echo "Invalid parameter"
        usage
        return 1
        ;;
    esac
  done
  export OPTIND=1

  if [[ -z "$item" ]]; then
    errecho "ERROR: You must provide an item with the -i parameter."
    usage
    return 1
  fi

  iecho "Parameters:\n"
  iecho "    table_name:   $table_name"
  iecho "    item:   $item"
  iecho ""

  response=$(aws dynamodb batch-write-item \
    --request-items file://"$item")
```

```
  local error_code=${?}

  if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports batch-write-item operation failed.$response"
    return 1
  fi

  return 0
}
```

Le funzioni di utilità utilizzate in questo esempio.

```
###############################################################################
# function iecho
#
# This function enables the script to display the specified text only if
# the global variable $VERBOSE is set to true.
###############################################################################
function iecho() {
  if [[ $VERBOSE == true ]]; then
    echo "$@"
  fi
}

###############################################################################
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
###############################################################################
function errecho() {
  printf "%s\n" "$*" 1>&2
}

###############################################################################
# function aws_cli_error_log()
#
# This function is used to log the error messages from the AWS CLI.
#
# See https://docs.aws.amazon.com/cli/latest/topic/return-codes.html#cli-aws-help-
return-codes.
```

```
#
# The function expects the following argument:
#         $1 - The error code returned by the AWS CLI.
#
#  Returns:
#          0: - Success.
#
###############################################################################
function aws_cli_error_log() {
  local err_code=$1
  errecho "Error code : $err_code"
  if [ "$err_code" == 1 ]; then
    errecho "  One or more S3 transfers failed."
  elif [ "$err_code" == 2 ]; then
    errecho "  Command line failed to parse."
  elif [ "$err_code" == 130 ]; then
    errecho "  Process received SIGINT."
  elif [ "$err_code" == 252 ]; then
    errecho "  Command syntax invalid."
  elif [ "$err_code" == 253 ]; then
    errecho "  The system environment or configuration was invalid."
  elif [ "$err_code" == 254 ]; then
    errecho "  The service returned an error."
  elif [ "$err_code" == 255 ]; then
    errecho "  255 is a catch-all error."
  fi

  return 0
}
```

- Per i dettagli sull'API, consulta BatchWriteItem AWS CLICommand Reference.

`CreateTable`

Il seguente esempio di codice mostra come utilizzare`CreateTable`.

AWS CLI con lo script Bash

> **Note**
>
> C'è altro da fare. GitHub Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codice AWS.

```
###############################################################################
# function dynamodb_create_table
#
# This function creates an Amazon DynamoDB table.
#
# Parameters:
#       -n table_name  -- The name of the table to create.
#       -a attribute_definitions -- JSON file path of a list of attributes and their
 types.
#       -k key_schema -- JSON file path of a list of attributes and their key types.
#
#  Returns:
#       0 - If successful.
#       1 - If it fails.
###############################################################################
function dynamodb_create_table() {
  local table_name attribute_definitions key_schema response
  local option OPTARG # Required to use getopts command in a function.

  ######################################
  # Function usage explanation
  ######################################
  function usage() {
    echo "function dynamodb_create_table"
    echo "Creates an Amazon DynamoDB table with on-demand billing."
    echo " -n table_name  -- The name of the table to create."
    echo " -a attribute_definitions -- JSON file path of a list of attributes and
 their types."
    echo " -k key_schema -- JSON file path of a list of attributes and their key
 types."
    echo ""
  }

  # Retrieve the calling parameters.
```

```
  while getopts "n:a:k:h" option; do
    case "${option}" in
      n) table_name="${OPTARG}" ;;
      a) attribute_definitions="${OPTARG}" ;;
      k) key_schema="${OPTARG}" ;;
      h)
        usage
        return 0
        ;;
      \?)
        echo "Invalid parameter"
        usage
        return 1
        ;;
    esac
  done
  export OPTIND=1

  if [[ -z "$table_name" ]]; then
    errecho "ERROR: You must provide a table name with the -n parameter."
    usage
    return 1
  fi

  if [[ -z "$attribute_definitions" ]]; then
    errecho "ERROR: You must provide an attribute definitions json file path the -a
parameter."
    usage
    return 1
  fi

  if [[ -z "$key_schema" ]]; then
    errecho "ERROR: You must provide a key schema json file path the -k parameter."
    usage
    return 1
  fi

  iecho "Parameters:\n"
  iecho "    table_name:   $table_name"
  iecho "    attribute_definitions:   $attribute_definitions"
  iecho "    key_schema:   $key_schema"
  iecho ""

  response=$(aws dynamodb create-table \
```

```
    --table-name "$table_name" \
    --attribute-definitions file://"$attribute_definitions" \
    --billing-mode PAY_PER_REQUEST \
    --key-schema file://"$key_schema" )

  local error_code=${?}

  if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports create-table operation failed.$response"
    return 1
  fi

  return 0
}
```

Le funzioni di utilità utilizzate in questo esempio.

```
###############################################################################
# function iecho
#
# This function enables the script to display the specified text only if
# the global variable $VERBOSE is set to true.
###############################################################################
function iecho() {
  if [[ $VERBOSE == true ]]; then
    echo "$@"
  fi
}

###############################################################################
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
###############################################################################
function errecho() {
  printf "%s\n" "$*" 1>&2
}

###############################################################################
# function aws_cli_error_log()
#
```

```
# This function is used to log the error messages from the AWS CLI.
#
# See https://docs.aws.amazon.com/cli/latest/topic/return-codes.html#cli-aws-help-
return-codes.
#
# The function expects the following argument:
#         $1 - The error code returned by the AWS CLI.
#
#  Returns:
#          0: - Success.
#
########################################################################
function aws_cli_error_log() {
  local err_code=$1
  errecho "Error code : $err_code"
  if [ "$err_code" == 1 ]; then
    errecho "  One or more S3 transfers failed."
  elif [ "$err_code" == 2 ]; then
    errecho "  Command line failed to parse."
  elif [ "$err_code" == 130 ]; then
    errecho "  Process received SIGINT."
  elif [ "$err_code" == 252 ]; then
    errecho "  Command syntax invalid."
  elif [ "$err_code" == 253 ]; then
    errecho "  The system environment or configuration was invalid."
  elif [ "$err_code" == 254 ]; then
    errecho "  The service returned an error."
  elif [ "$err_code" == 255 ]; then
    errecho "  255 is a catch-all error."
  fi

  return 0
}
```

- Per i dettagli sull'API, consulta CreateTable AWS CLICommand Reference.

`DeleteItem`

Il seguente esempio di codice mostra come utilizzare`DeleteItem`.

AWS CLI con lo script Bash

> **Note**
>
> C'è altro da fare. GitHub Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codice AWS.

```
###############################################################################
# function dynamodb_delete_item
#
# This function deletes an item from a DynamoDB table.
#
# Parameters:
#       -n table_name  -- The name of the table.
#       -k keys  -- Path to json file containing the keys that identify the item to
 delete.
#
#  Returns:
#       0 - If successful.
#       1 - If it fails.
###############################################################################
function dynamodb_delete_item() {
  local table_name keys response
  local option OPTARG # Required to use getopts command in a function.

  # ######################################
  # Function usage explanation
  ######################################
  function usage() {
    echo "function dynamodb_delete_item"
    echo "Delete an item from a DynamoDB table."
    echo " -n table_name  -- The name of the table."
    echo " -k keys  -- Path to json file containing the keys that identify the item
 to delete."
    echo ""
  }
  while getopts "n:k:h" option; do
    case "${option}" in
      n) table_name="${OPTARG}" ;;
      k) keys="${OPTARG}" ;;
      h)
```

```
        usage
        return 0
        ;;
      \?)
        echo "Invalid parameter"
        usage
        return 1
        ;;
    esac
  done
  export OPTIND=1

  if [[ -z "$table_name" ]]; then
    errecho "ERROR: You must provide a table name with the -n parameter."
    usage
    return 1
  fi

  if [[ -z "$keys" ]]; then
    errecho "ERROR: You must provide a keys json file path the -k parameter."
    usage
    return 1
  fi

  iecho "Parameters:\n"
  iecho "    table_name:   $table_name"
  iecho "    keys:   $keys"
  iecho ""

  response=$(aws dynamodb delete-item \
    --table-name "$table_name" \
    --key file://"$keys")

  local error_code=${?}

  if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports delete-item operation failed.$response"
    return 1
  fi

  return 0

}
```

Le funzioni di utilità utilizzate in questo esempio.

```
##############################################################################
# function iecho
#
# This function enables the script to display the specified text only if
# the global variable $VERBOSE is set to true.
##############################################################################
function iecho() {
  if [[ $VERBOSE == true ]]; then
    echo "$@"
  fi
}

##############################################################################
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
##############################################################################
function errecho() {
  printf "%s\n" "$*" 1>&2
}

##############################################################################
# function aws_cli_error_log()
#
# This function is used to log the error messages from the AWS CLI.
#
# See https://docs.aws.amazon.com/cli/latest/topic/return-codes.html#cli-aws-help-
return-codes.
#
# The function expects the following argument:
#         $1 - The error code returned by the AWS CLI.
#
#  Returns:
#          0: - Success.
#
##############################################################################
function aws_cli_error_log() {
  local err_code=$1
  errecho "Error code : $err_code"
```

```
  if [ "$err_code" == 1 ]; then
    errecho "  One or more S3 transfers failed."
  elif [ "$err_code" == 2 ]; then
    errecho "  Command line failed to parse."
  elif [ "$err_code" == 130 ]; then
    errecho "  Process received SIGINT."
  elif [ "$err_code" == 252 ]; then
    errecho "  Command syntax invalid."
  elif [ "$err_code" == 253 ]; then
    errecho "  The system environment or configuration was invalid."
  elif [ "$err_code" == 254 ]; then
    errecho "  The service returned an error."
  elif [ "$err_code" == 255 ]; then
    errecho "  255 is a catch-all error."
  fi

  return 0
}
```

- Per i dettagli sull'API, consulta DeleteItem AWS CLICommand Reference.

`DeleteTable`

Il seguente esempio di codice mostra come utilizzare`DeleteTable`.

AWS CLI con lo script Bash

> **Note**
>
> C'è altro da fare. GitHub Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codice AWS.

```
###############################################################################
# function dynamodb_delete_table
#
# This function deletes a DynamoDB table.
#
# Parameters:
#       -n table_name  -- The name of the table to delete.
#
```

```
#  Returns:
#       0 - If successful.
#       1 - If it fails.
###############################################################################
function dynamodb_delete_table() {
  local table_name response
  local option OPTARG # Required to use getopts command in a function.

  # bashsupport disable=BP5008
  function usage() {
    echo "function dynamodb_delete_table"
    echo "Deletes an Amazon DynamoDB table."
    echo " -n table_name  -- The name of the table to delete."
    echo ""
  }

  # Retrieve the calling parameters.
  while getopts "n:h" option; do
    case "${option}" in
      n) table_name="${OPTARG}" ;;
      h)
        usage
        return 0
        ;;
      \?)
        echo "Invalid parameter"
        usage
        return 1
        ;;
    esac
  done
  export OPTIND=1

  if [[ -z "$table_name" ]]; then
    errecho "ERROR: You must provide a table name with the -n parameter."
    usage
    return 1
  fi

  iecho "Parameters:\n"
  iecho "    table_name:   $table_name"
  iecho ""

  response=$(aws dynamodb delete-table \
```

```
    --table-name "$table_name")

  local error_code=${?}

  if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports delete-table operation failed.$response"
    return 1
  fi

  return 0
}
```

Le funzioni di utilità utilizzate in questo esempio.

```
###############################################################################
# function iecho
#
# This function enables the script to display the specified text only if
# the global variable $VERBOSE is set to true.
###############################################################################
function iecho() {
  if [[ $VERBOSE == true ]]; then
    echo "$@"
  fi
}

###############################################################################
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
###############################################################################
function errecho() {
  printf "%s\n" "$*" 1>&2
}

###############################################################################
# function aws_cli_error_log()
#
# This function is used to log the error messages from the AWS CLI.
#
```

```
# See https://docs.aws.amazon.com/cli/latest/topic/return-codes.html#cli-aws-help-
return-codes.
#
# The function expects the following argument:
#         $1 - The error code returned by the AWS CLI.
#
#  Returns:
#          0: - Success.
#
################################################################################
function aws_cli_error_log() {
  local err_code=$1
  errecho "Error code : $err_code"
  if [ "$err_code" == 1 ]; then
    errecho "  One or more S3 transfers failed."
  elif [ "$err_code" == 2 ]; then
    errecho "  Command line failed to parse."
  elif [ "$err_code" == 130 ]; then
    errecho "  Process received SIGINT."
  elif [ "$err_code" == 252 ]; then
    errecho "  Command syntax invalid."
  elif [ "$err_code" == 253 ]; then
    errecho "  The system environment or configuration was invalid."
  elif [ "$err_code" == 254 ]; then
    errecho "  The service returned an error."
  elif [ "$err_code" == 255 ]; then
    errecho "  255 is a catch-all error."
  fi

  return 0
}
```

- Per i dettagli sull'API, consulta DeleteTable AWS CLICommand Reference.

`DescribeTable`

Il seguente esempio di codice mostra come utilizzare`DescribeTable`.

AWS CLI con lo script Bash

> **Note**
>
> C'è altro da fare. GitHub Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codice AWS.

```
###############################################################################
# function dynamodb_describe_table
#
# This function returns the status of a DynamoDB table.
#
# Parameters:
#       -n table_name  -- The name of the table.
#
#  Response:
#       - TableStatus:
#     And:
#       0 - Table is active.
#       1 - If it fails.
###############################################################################
function dynamodb_describe_table {
  local table_name
  local option OPTARG # Required to use getopts command in a function.

  ######################################
  # Function usage explanation
  ######################################
  function usage() {
    echo "function dynamodb_describe_table"
    echo "Describe the status of a DynamoDB table."
    echo "  -n table_name  -- The name of the table."
    echo ""
  }

  # Retrieve the calling parameters.
  while getopts "n:h" option; do
    case "${option}" in
      n) table_name="${OPTARG}" ;;
      h)
        usage
```

```
        return 0
        ;;
      \?)
        echo "Invalid parameter"
        usage
        return 1
        ;;
    esac
  done
  export OPTIND=1

  if [[ -z "$table_name" ]]; then
    errecho "ERROR: You must provide a table name with the -n parameter."
    usage
    return 1
  fi

  local table_status
    table_status=$(
      aws dynamodb describe-table \
        --table-name "$table_name" \
        --output text \
        --query 'Table.TableStatus'
    )

   local error_code=${?}

    if [[ $error_code -ne 0 ]]; then
      aws_cli_error_log "$error_code"
      errecho "ERROR: AWS reports describe-table operation failed.$table_status"
      return 1
    fi

  echo "$table_status"

  return 0
}
```

Le funzioni di utilità utilizzate in questo esempio.

```
###############################################################################
# function errecho
```

```
#
# This function outputs everything sent to it to STDERR (standard error output).
#############################################################################
function errecho() {
  printf "%s\n" "$*" 1>&2
}

#############################################################################
# function aws_cli_error_log()
#
# This function is used to log the error messages from the AWS CLI.
#
# See https://docs.aws.amazon.com/cli/latest/topic/return-codes.html#cli-aws-help-
return-codes.
#
# The function expects the following argument:
#         $1 - The error code returned by the AWS CLI.
#
#  Returns:
#          0: - Success.
#
#############################################################################
function aws_cli_error_log() {
  local err_code=$1
  errecho "Error code : $err_code"
  if [ "$err_code" == 1 ]; then
    errecho "  One or more S3 transfers failed."
  elif [ "$err_code" == 2 ]; then
    errecho "  Command line failed to parse."
  elif [ "$err_code" == 130 ]; then
    errecho "  Process received SIGINT."
  elif [ "$err_code" == 252 ]; then
    errecho "  Command syntax invalid."
  elif [ "$err_code" == 253 ]; then
    errecho "  The system environment or configuration was invalid."
  elif [ "$err_code" == 254 ]; then
    errecho "  The service returned an error."
  elif [ "$err_code" == 255 ]; then
    errecho "  255 is a catch-all error."
  fi

  return 0
}
```

- Per i dettagli sull'API, consulta DescribeTable AWS CLICommand Reference.

`GetItem`

Il seguente esempio di codice mostra come utilizzare`GetItem`.

AWS CLI con lo script Bash

> **Note**
>
> C'è altro da fare. GitHub Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codice AWS.

```
###############################################################################
# function dynamodb_get_item
#
# This function gets an item from a DynamoDB table.
#
# Parameters:
#       -n table_name  -- The name of the table.
#       -k keys  -- Path to json file containing the keys that identify the item to
 get.
#       [-q query]  -- Optional JMESPath query expression.
#
#  Returns:
#       The item as text output.
#  And:
#       0 - If successful.
#       1 - If it fails.
###############################################################################
function dynamodb_get_item() {
  local table_name keys query response
  local option OPTARG # Required to use getopts command in a function.

  # ######################################
  # Function usage explanation
  ######################################
  function usage() {
    echo "function dynamodb_get_item"
```

```
    echo "Get an item from a DynamoDB table."
    echo " -n table_name  -- The name of the table."
    echo " -k keys  -- Path to json file containing the keys that identify the item
to get."
    echo " [-q query]  -- Optional JMESPath query expression."
    echo ""
  }
  query=""
  while getopts "n:k:q:h" option; do
    case "${option}" in
      n) table_name="${OPTARG}" ;;
      k) keys="${OPTARG}" ;;
      q) query="${OPTARG}" ;;
      h)
        usage
        return 0
        ;;
      \?)
        echo "Invalid parameter"
        usage
        return 1
        ;;
    esac
  done
  export OPTIND=1

  if [[ -z "$table_name" ]]; then
    errecho "ERROR: You must provide a table name with the -n parameter."
    usage
    return 1
  fi

  if [[ -z "$keys" ]]; then
    errecho "ERROR: You must provide a keys json file path the -k parameter."
    usage
    return 1
  fi

  if [[ -n "$query" ]]; then
    response=$(aws dynamodb get-item \
      --table-name "$table_name" \
      --key file://"$keys" \
      --output text \
      --query "$query")
```

```
  else
    response=$(
      aws dynamodb get-item \
        --table-name "$table_name" \
        --key file://"$keys" \
        --output text
    )
  fi

  local error_code=${?}

  if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports get-item operation failed.$response"
    return 1
  fi

  if [[ -n "$query" ]]; then
    echo "$response" | sed "/^\t/s/\t//1" # Remove initial tab that the JMSEPath
 query inserts on some strings.
  else
    echo "$response"
  fi

  return 0
}
```

Le funzioni di utilità utilizzate in questo esempio.

```
###############################################################################
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
###############################################################################
function errecho() {
  printf "%s\n" "$*" 1>&2
}

###############################################################################
# function aws_cli_error_log()
#
# This function is used to log the error messages from the AWS CLI.
```

```
#
# See https://docs.aws.amazon.com/cli/latest/topic/return-codes.html#cli-aws-help-
return-codes.
#
# The function expects the following argument:
#         $1 - The error code returned by the AWS CLI.
#
#  Returns:
#          0: - Success.
#
########################################################################
function aws_cli_error_log() {
  local err_code=$1
  errecho "Error code : $err_code"
  if [ "$err_code" == 1 ]; then
    errecho "  One or more S3 transfers failed."
  elif [ "$err_code" == 2 ]; then
    errecho "  Command line failed to parse."
  elif [ "$err_code" == 130 ]; then
    errecho "  Process received SIGINT."
  elif [ "$err_code" == 252 ]; then
    errecho "  Command syntax invalid."
  elif [ "$err_code" == 253 ]; then
    errecho "  The system environment or configuration was invalid."
  elif [ "$err_code" == 254 ]; then
    errecho "  The service returned an error."
  elif [ "$err_code" == 255 ]; then
    errecho "  255 is a catch-all error."
  fi

  return 0
}
```

- Per i dettagli sull'API, consulta GetItem AWS CLICommand Reference.

`ListTables`

Il seguente esempio di codice mostra come utilizzare`ListTables`.

AWS CLI con lo script Bash

> **Note**
>
> C'è altro da fare. GitHub Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codice AWS.

```
##############################################################################
# function dynamodb_list_tables
#
# This function lists all the tables in a DynamoDB.
#
# Returns:
#       0 - If successful.
#       1 - If it fails.
##############################################################################
function dynamodb_list_tables() {
  response=$(aws dynamodb list-tables \
    --output text \
    --query "TableNames")

  local error_code=${?}

  if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports batch-write-item operation failed.$response"
    return 1
  fi

  echo "$response" | tr -s "[:space:]" "\n"

  return 0
}
```

Le funzioni di utilità utilizzate in questo esempio.

```
##############################################################################
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
```

```
###############################################################################
function errecho() {
  printf "%s\n" "$*" 1>&2
}

###############################################################################
# function aws_cli_error_log()
#
# This function is used to log the error messages from the AWS CLI.
#
# See https://docs.aws.amazon.com/cli/latest/topic/return-codes.html#cli-aws-help-
return-codes.
#
# The function expects the following argument:
#         $1 - The error code returned by the AWS CLI.
#
#  Returns:
#          0: - Success.
#
###############################################################################
function aws_cli_error_log() {
  local err_code=$1
  errecho "Error code : $err_code"
  if [ "$err_code" == 1 ]; then
    errecho "  One or more S3 transfers failed."
  elif [ "$err_code" == 2 ]; then
    errecho "  Command line failed to parse."
  elif [ "$err_code" == 130 ]; then
    errecho "  Process received SIGINT."
  elif [ "$err_code" == 252 ]; then
    errecho "  Command syntax invalid."
  elif [ "$err_code" == 253 ]; then
    errecho "  The system environment or configuration was invalid."
  elif [ "$err_code" == 254 ]; then
    errecho "  The service returned an error."
  elif [ "$err_code" == 255 ]; then
    errecho "  255 is a catch-all error."
  fi

  return 0
}
```

- Per i dettagli sull'API, consulta ListTables AWS CLICommand Reference.

`PutItem`

Il seguente esempio di codice mostra come utilizzare`PutItem`.

AWS CLI con lo script Bash

> **Note**
>
> C'è altro da fare. GitHub Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codice AWS.

```
###############################################################################
# function dynamodb_put_item
#
# This function puts an item into a DynamoDB table.
#
# Parameters:
#       -n table_name  -- The name of the table.
#       -i item  -- Path to json file containing the item values.
#
#  Returns:
#       0 - If successful.
#       1 - If it fails.
###############################################################################
function dynamodb_put_item() {
  local table_name item response
  local option OPTARG # Required to use getopts command in a function.

  ######################################
  # Function usage explanation
  ######################################
  function usage() {
    echo "function dynamodb_put_item"
    echo "Put an item into a DynamoDB table."
    echo " -n table_name  -- The name of the table."
    echo " -i item  -- Path to json file containing the item values."
    echo ""
  }

  while getopts "n:i:h" option; do
    case "${option}" in
      n) table_name="${OPTARG}" ;;
```

```
    i) item="${OPTARG}" ;;
    h)
      usage
      return 0
      ;;
    \?)
      echo "Invalid parameter"
      usage
      return 1
      ;;
  esac
done
export OPTIND=1

if [[ -z "$table_name" ]]; then
  errecho "ERROR: You must provide a table name with the -n parameter."
  usage
  return 1
fi

if [[ -z "$item" ]]; then
  errecho "ERROR: You must provide an item with the -i parameter."
  usage
  return 1
fi

iecho "Parameters:\n"
iecho "    table_name:   $table_name"
iecho "    item:   $item"
iecho ""
iecho ""

response=$(aws dynamodb put-item \
  --table-name "$table_name" \
  --item file://"$item")

local error_code=${?}

if [[ $error_code -ne 0 ]]; then
  aws_cli_error_log $error_code
  errecho "ERROR: AWS reports put-item operation failed.$response"
  return 1
fi
```

```
  return 0

}
```

Le funzioni di utilità utilizzate in questo esempio.

```
################################################################################
# function iecho
#
# This function enables the script to display the specified text only if
# the global variable $VERBOSE is set to true.
################################################################################
function iecho() {
  if [[ $VERBOSE == true ]]; then
    echo "$@"
  fi
}

################################################################################
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
################################################################################
function errecho() {
  printf "%s\n" "$*" 1>&2
}

################################################################################
# function aws_cli_error_log()
#
# This function is used to log the error messages from the AWS CLI.
#
# See https://docs.aws.amazon.com/cli/latest/topic/return-codes.html#cli-aws-help-
return-codes.
#
# The function expects the following argument:
#         $1 - The error code returned by the AWS CLI.
#
#  Returns:
#          0: - Success.
#
################################################################################
```

```
function aws_cli_error_log() {
  local err_code=$1
  errecho "Error code : $err_code"
  if [ "$err_code" == 1 ]; then
    errecho "  One or more S3 transfers failed."
  elif [ "$err_code" == 2 ]; then
    errecho "  Command line failed to parse."
  elif [ "$err_code" == 130 ]; then
    errecho "  Process received SIGINT."
  elif [ "$err_code" == 252 ]; then
    errecho "  Command syntax invalid."
  elif [ "$err_code" == 253 ]; then
    errecho "  The system environment or configuration was invalid."
  elif [ "$err_code" == 254 ]; then
    errecho "  The service returned an error."
  elif [ "$err_code" == 255 ]; then
    errecho "  255 is a catch-all error."
  fi

  return 0
}
```

- Per i dettagli sull'API, consulta PutItem AWS CLICommand Reference.

`Query`

Il seguente esempio di codice mostra come utilizzare`Query`.

AWS CLI con lo script Bash

> **Note**
>
> C'è altro da fare. GitHub Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codice AWS.

```
###############################################################################
# function dynamodb_query
#
# This function queries a DynamoDB table.
#
```

```
# Parameters:
#       -n table_name  -- The name of the table.
#       -k key_condition_expression -- The key condition expression.
#       -a attribute_names -- Path to JSON file containing the attribute names.
#       -v attribute_values -- Path to JSON file containing the attribute values.
#       [-p projection_expression]  -- Optional projection expression.
#
#  Returns:
#       The items as json output.
#  And:
#       0 - If successful.
#       1 - If it fails.
###############################################################################
function dynamodb_query() {
  local table_name key_condition_expression attribute_names attribute_values
 projection_expression response
  local option OPTARG # Required to use getopts command in a function.

  # ######################################
  # Function usage explanation
  ######################################
  function usage() {
    echo "function dynamodb_query"
    echo "Query a DynamoDB table."
    echo " -n table_name  -- The name of the table."
    echo " -k key_condition_expression -- The key condition expression."
    echo " -a attribute_names -- Path to JSON file containing the attribute names."
    echo " -v attribute_values -- Path to JSON file containing the attribute
 values."
    echo " [-p projection_expression]  -- Optional projection expression."
    echo ""
  }

  while getopts "n:k:a:v:p:h" option; do
    case "${option}" in
      n) table_name="${OPTARG}" ;;
      k) key_condition_expression="${OPTARG}" ;;
      a) attribute_names="${OPTARG}" ;;
      v) attribute_values="${OPTARG}" ;;
      p) projection_expression="${OPTARG}" ;;
      h)
        usage
        return 0
        ;;
```

```
      \?)
        echo "Invalid parameter"
        usage
        return 1
        ;;
    esac
  done
  export OPTIND=1

  if [[ -z "$table_name" ]]; then
    errecho "ERROR: You must provide a table name with the -n parameter."
    usage
    return 1
  fi

  if [[ -z "$key_condition_expression" ]]; then
    errecho "ERROR: You must provide a key condition expression with the -k
parameter."
    usage
    return 1
  fi

  if [[ -z "$attribute_names" ]]; then
    errecho "ERROR: You must provide a attribute names with the -a parameter."
    usage
    return 1
  fi

  if [[ -z "$attribute_values" ]]; then
    errecho "ERROR: You must provide a attribute values with the -v parameter."
    usage
    return 1
  fi

  if [[ -z "$projection_expression" ]]; then
    response=$(aws dynamodb query \
      --table-name "$table_name" \
      --key-condition-expression "$key_condition_expression" \
      --expression-attribute-names file://"$attribute_names" \
      --expression-attribute-values file://"$attribute_values")
  else
    response=$(aws dynamodb query \
      --table-name "$table_name" \
      --key-condition-expression "$key_condition_expression" \
```

```
      --expression-attribute-names file://"$attribute_names" \
      --expression-attribute-values file://"$attribute_values" \
      --projection-expression "$projection_expression")
  fi

  local error_code=${?}

  if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports query operation failed.$response"
    return 1
  fi

  echo "$response"

  return 0
}
```

Le funzioni di utilità utilizzate in questo esempio.

```
###############################################################################
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
###############################################################################
function errecho() {
  printf "%s\n" "$*" 1>&2
}

###############################################################################
# function aws_cli_error_log()
#
# This function is used to log the error messages from the AWS CLI.
#
# See https://docs.aws.amazon.com/cli/latest/topic/return-codes.html#cli-aws-help-
return-codes.
#
# The function expects the following argument:
#         $1 - The error code returned by the AWS CLI.
#
#  Returns:
#          0: - Success.
```

```
#
################################################################################
function aws_cli_error_log() {
  local err_code=$1
  errecho "Error code : $err_code"
  if [ "$err_code" == 1 ]; then
    errecho "  One or more S3 transfers failed."
  elif [ "$err_code" == 2 ]; then
    errecho "  Command line failed to parse."
  elif [ "$err_code" == 130 ]; then
    errecho "  Process received SIGINT."
  elif [ "$err_code" == 252 ]; then
    errecho "  Command syntax invalid."
  elif [ "$err_code" == 253 ]; then
    errecho "  The system environment or configuration was invalid."
  elif [ "$err_code" == 254 ]; then
    errecho "  The service returned an error."
  elif [ "$err_code" == 255 ]; then
    errecho "  255 is a catch-all error."
  fi

  return 0
}
```

- Per informazioni dettagliate sull'API, consulta Query nella documentazione di riferimento dei comandi della AWS CLI .

`Scan`

Il seguente esempio di codice mostra come usare`Scan`.

AWS CLI con lo script Bash

> **Note**
>
> C'è altro da fare. GitHub Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codice AWS.

```
################################################################################
```

```
# function dynamodb_scan
#
# This function scans a DynamoDB table.
#
# Parameters:
#       -n table_name  -- The name of the table.
#       -f filter_expression  -- The filter expression.
#       -a expression_attribute_names -- Path to JSON file containing the expression
 attribute names.
#       -v expression_attribute_values -- Path to JSON file containing the
 expression attribute values.
#       [-p projection_expression]  -- Optional projection expression.
#
#  Returns:
#       The items as json output.
#  And:
#       0 - If successful.
#       1 - If it fails.
###############################################################################
function dynamodb_scan() {
  local table_name filter_expression expression_attribute_names
 expression_attribute_values projection_expression response
  local option OPTARG # Required to use getopts command in a function.

  # ######################################
  # Function usage explanation
  ######################################
  function usage() {
    echo "function dynamodb_scan"
    echo "Scan a DynamoDB table."
    echo " -n table_name  -- The name of the table."
    echo " -f filter_expression  -- The filter expression."
    echo " -a expression_attribute_names -- Path to JSON file containing the
 expression attribute names."
    echo " -v expression_attribute_values -- Path to JSON file containing the
 expression attribute values."
    echo " [-p projection_expression]  -- Optional projection expression."
    echo ""
  }

  while getopts "n:f:a:v:p:h" option; do
    case "${option}" in
      n) table_name="${OPTARG}" ;;
      f) filter_expression="${OPTARG}" ;;
```

```
    a) expression_attribute_names="${OPTARG}" ;;
    v) expression_attribute_values="${OPTARG}" ;;
    p) projection_expression="${OPTARG}" ;;
    h)
      usage
      return 0
      ;;
    \?)
      echo "Invalid parameter"
      usage
      return 1
      ;;
  esac
done
export OPTIND=1

if [[ -z "$table_name" ]]; then
  errecho "ERROR: You must provide a table name with the -n parameter."
  usage
  return 1
fi

if [[ -z "$filter_expression" ]]; then
  errecho "ERROR: You must provide a filter expression with the -f parameter."
  usage
  return 1
fi

if [[ -z "$expression_attribute_names" ]]; then
  errecho "ERROR: You must provide expression attribute names with the -a
parameter."
  usage
  return 1
fi

if [[ -z "$expression_attribute_values" ]]; then
  errecho "ERROR: You must provide expression attribute values with the -v
parameter."
  usage
  return 1
fi

if [[ -z "$projection_expression" ]]; then
  response=$(aws dynamodb scan \
```

```
      --table-name "$table_name" \
      --filter-expression "$filter_expression" \
      --expression-attribute-names file://"$expression_attribute_names" \
      --expression-attribute-values file://"$expression_attribute_values")
  else
    response=$(aws dynamodb scan \
      --table-name "$table_name" \
      --filter-expression "$filter_expression" \
      --expression-attribute-names file://"$expression_attribute_names" \
      --expression-attribute-values file://"$expression_attribute_values" \
      --projection-expression "$projection_expression")
  fi

  local error_code=${?}

  if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports scan operation failed.$response"
    return 1
  fi

  echo "$response"

  return 0
}
```

Le funzioni di utilità utilizzate in questo esempio.

```
###############################################################################
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
###############################################################################
function errecho() {
  printf "%s\n" "$*" 1>&2
}

###############################################################################
# function aws_cli_error_log()
#
# This function is used to log the error messages from the AWS CLI.
#
```

```
# See https://docs.aws.amazon.com/cli/latest/topic/return-codes.html#cli-aws-help-
return-codes.
#
# The function expects the following argument:
#         $1 - The error code returned by the AWS CLI.
#
#  Returns:
#          0: - Success.
#
################################################################################
function aws_cli_error_log() {
  local err_code=$1
  errecho "Error code : $err_code"
  if [ "$err_code" == 1 ]; then
    errecho "  One or more S3 transfers failed."
  elif [ "$err_code" == 2 ]; then
    errecho "  Command line failed to parse."
  elif [ "$err_code" == 130 ]; then
    errecho "  Process received SIGINT."
  elif [ "$err_code" == 252 ]; then
    errecho "  Command syntax invalid."
  elif [ "$err_code" == 253 ]; then
    errecho "  The system environment or configuration was invalid."
  elif [ "$err_code" == 254 ]; then
    errecho "  The service returned an error."
  elif [ "$err_code" == 255 ]; then
    errecho "  255 is a catch-all error."
  fi

  return 0
}
```

- Per informazioni dettagliate sull'API, consulta Scansione nella documentazione di riferimento dei comandi della AWS CLI .

`UpdateItem`

Il seguente esempio di codice mostra come usare`UpdateItem`.

AWS CLI con lo script Bash

> **Note**
>
> C'è altro da fare. GitHub Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codice AWS.

```
###############################################################################
# function dynamodb_update_item
#
# This function updates an item in a DynamoDB table.
#
#
# Parameters:
#       -n table_name  -- The name of the table.
#       -k keys  -- Path to json file containing the keys that identify the item to
 update.
#       -e update expression  -- An expression that defines one or more attributes
 to be updated.
#       -v values  -- Path to json file containing the update values.
#
#  Returns:
#       0 - If successful.
#       1 - If it fails.
###############################################################################
function dynamodb_update_item() {
  local table_name keys update_expression values response
  local option OPTARG # Required to use getopts command in a function.

  ######################################
  # Function usage explanation
  ######################################
  function usage() {
    echo "function dynamodb_update_item"
    echo "Update an item in a DynamoDB table."
    echo " -n table_name  -- The name of the table."
    echo " -k keys  -- Path to json file containing the keys that identify the item
 to update."
    echo " -e update expression  -- An expression that defines one or more
 attributes to be updated."
    echo " -v values  -- Path to json file containing the update values."
```

```
  echo ""
}

while getopts "n:k:e:v:h" option; do
  case "${option}" in
    n) table_name="${OPTARG}" ;;
    k) keys="${OPTARG}" ;;
    e) update_expression="${OPTARG}" ;;
    v) values="${OPTARG}" ;;
    h)
      usage
      return 0
      ;;
    \?)
      echo "Invalid parameter"
      usage
      return 1
      ;;
  esac
done
export OPTIND=1

if [[ -z "$table_name" ]]; then
  errecho "ERROR: You must provide a table name with the -n parameter."
  usage
  return 1
fi

if [[ -z "$keys" ]]; then
  errecho "ERROR: You must provide a keys json file path the -k parameter."
  usage
  return 1
fi
if [[ -z "$update_expression" ]]; then
  errecho "ERROR: You must provide an update expression with the -e parameter."
  usage
  return 1
fi

if [[ -z "$values" ]]; then
  errecho "ERROR: You must provide a values json file path the -v parameter."
  usage
  return 1
fi
```

```
  iecho "Parameters:\n"
  iecho "    table_name:   $table_name"
  iecho "    keys:   $keys"
  iecho "    update_expression:   $update_expression"
  iecho "    values:   $values"

  response=$(aws dynamodb update-item \
    --table-name "$table_name" \
    --key file://"$keys" \
    --update-expression "$update_expression" \
    --expression-attribute-values file://"$values")

  local error_code=${?}

  if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports update-item operation failed.$response"
    return 1
  fi

  return 0

}
```

Le funzioni di utilità utilizzate in questo esempio.

```
###############################################################################
# function iecho
#
# This function enables the script to display the specified text only if
# the global variable $VERBOSE is set to true.
###############################################################################
function iecho() {
  if [[ $VERBOSE == true ]]; then
    echo "$@"
  fi
}

###############################################################################
# function errecho
#
```

```
# This function outputs everything sent to it to STDERR (standard error output).
#########################################################################################
function errecho() {
  printf "%s\n" "$*" 1>&2
}

#########################################################################################
# function aws_cli_error_log()
#
# This function is used to log the error messages from the AWS CLI.
#
# See https://docs.aws.amazon.com/cli/latest/topic/return-codes.html#cli-aws-help-
return-codes.
#
# The function expects the following argument:
#         $1 - The error code returned by the AWS CLI.
#
#  Returns:
#          0: - Success.
#
#########################################################################################
function aws_cli_error_log() {
  local err_code=$1
  errecho "Error code : $err_code"
  if [ "$err_code" == 1 ]; then
    errecho "  One or more S3 transfers failed."
  elif [ "$err_code" == 2 ]; then
    errecho "  Command line failed to parse."
  elif [ "$err_code" == 130 ]; then
    errecho "  Process received SIGINT."
  elif [ "$err_code" == 252 ]; then
    errecho "  Command syntax invalid."
  elif [ "$err_code" == 253 ]; then
    errecho "  The system environment or configuration was invalid."
  elif [ "$err_code" == 254 ]; then
    errecho "  The service returned an error."
  elif [ "$err_code" == 255 ]; then
    errecho "  255 is a catch-all error."
  fi

  return 0
}
```

- Per i dettagli sull'API, consulta UpdateItem AWS CLICommand Reference.

Scenari

Scenari avanzati basati sull'indice secondario globale

L'esempio di codice seguente mostra come utilizzare le configurazioni degli indici secondari globali.

- Crea una tabella con più elementi GSIs.
- Creare una tabella con capacità on demand e indici secondari globali.
- Metti gli elementi in una tabella con più elementi GSIs.
- Interrogazione GSIs multipla con condizioni diverse.

AWS CLI con script Bash

Crea una tabella con più GSIs di una tabella.

```
# Create a table with multiple GSIs
aws dynamodb create-table \
    --table-name MusicLibrary \
    --attribute-definitions \
        AttributeName=Artist,AttributeType=S \
        AttributeName=SongTitle,AttributeType=S \
        AttributeName=AlbumTitle,AttributeType=S \
        AttributeName=Genre,AttributeType=S \
        AttributeName=Year,AttributeType=N \
    --key-schema \
        AttributeName=Artist,KeyType=HASH \
        AttributeName=SongTitle,KeyType=RANGE \
    --billing-mode PAY_PER_REQUEST \
    --global-secondary-indexes \
        "[
            {
                \"IndexName\": \"AlbumIndex\",
                \"KeySchema\": [{\"AttributeName\":\"AlbumTitle\",\"KeyType\":\"HASH
\"}],
                \"Projection\": {\"ProjectionType\":\"ALL\"}
            },
            {
                \"IndexName\": \"GenreYearIndex\",
```

```
                \"KeySchema\": [
                    {\"AttributeName\":\"Genre\",\"KeyType\":\"HASH\"},
                    {\"AttributeName\":\"Year\",\"KeyType\":\"RANGE\"}
                ],
                \"Projection\": {\"ProjectionType\":\"INCLUDE\",\"NonKeyAttributes
\":[\"Artist\",\"SongTitle\"]}
            }
        ]"
```

Creare una tabella con capacità on demand e indici secondari globali.

```
# Create a table with on-demand capacity and GSI
aws dynamodb create-table \
    --table-name MusicOnDemand \
    --attribute-definitions \
        AttributeName=Artist,AttributeType=S \
        AttributeName=SongTitle,AttributeType=S \
        AttributeName=Genre,AttributeType=S \
    --key-schema \
        AttributeName=Artist,KeyType=HASH \
        AttributeName=SongTitle,KeyType=RANGE \
    --billing-mode PAY_PER_REQUEST \
    --global-secondary-indexes \
        "[
            {
                \"IndexName\": \"GenreIndex\",
                \"KeySchema\": [{\"AttributeName\":\"Genre\",\"KeyType\":\"HASH\"}],
                \"Projection\": {\"ProjectionType\":\"ALL\"}
            }
        ]"
```

Metti gli elementi in una tabella con più elementi GSIs.

```
# Add items to MusicLibrary table
aws dynamodb put-item \
    --table-name MusicLibrary \
    --item '{
        "Artist": {"S": "The Beatles"},
        "SongTitle": {"S": "Hey Jude"},
        "AlbumTitle": {"S": "Past Masters"},
        "Genre": {"S": "Rock"},
```

```
        "Year": {"N": "1968"}
    }'

aws dynamodb put-item \
    --table-name MusicLibrary \
    --item '{
        "Artist": {"S": "Miles Davis"},
        "SongTitle": {"S": "So What"},
        "AlbumTitle": {"S": "Kind of Blue"},
        "Genre": {"S": "Jazz"},
        "Year": {"N": "1959"}
    }'
```

Interroga gli elementi da una tabella con più elementi GSIs.

```
# Query the AlbumIndex GSI
echo "Querying AlbumIndex GSI:"
aws dynamodb query \
    --table-name MusicLibrary \
    --index-name AlbumIndex \
    --key-condition-expression "AlbumTitle = :album" \
    --expression-attribute-values '{":album":{"S":"Kind of Blue"}}'

# Query the GenreYearIndex GSI with a range condition
echo "Querying GenreYearIndex GSI with range condition:"
aws dynamodb query \
    --table-name MusicLibrary \
    --index-name GenreYearIndex \
    --key-condition-expression "Genre = :genre AND #yr > :year" \
    --expression-attribute-names '{"#yr": "Year"}' \
    --expression-attribute-values '{":genre":{"S":"Rock"},":year":{"N":"1965"}}'
```

- Per informazioni dettagliate sull'API, consulta i seguenti argomenti nella documentazione di riferimento dei comandi della AWS CLI .
  - CreateTable
  - PutItem
  - Query

Creare e gestire tabelle globali MRSC

L'esempio di codice seguente mostra come creare e gestire le tabelle globali DynamoDB con consistenza avanzata multi-Regione (MRSC).

- Creare una tabella con consistenza avanzata multi-Regione.
- Verificare la configurazione MRSC e lo stato della replica.
- Verificare l'elevata consistenza tra Regioni con letture immediate.
- Eseguire scritture condizionali con garanzie MRSC.
- Eseguire la pulizia delle risorse della tabella globale MRSC.

AWS CLI con lo script Bash

Creare una tabella con consistenza avanzata multi-Regione.

```
# Step 1: Create a new table in us-east-2 (primary region for MRSC)
# Note: Table must be empty when enabling MRSC
aws dynamodb create-table \
    --table-name MusicTable \
    --attribute-definitions \
        AttributeName=Artist,AttributeType=S \
        AttributeName=SongTitle,AttributeType=S \
    --key-schema \
        AttributeName=Artist,KeyType=HASH \
        AttributeName=SongTitle,KeyType=RANGE \
    --billing-mode PAY_PER_REQUEST \
    --region us-east-2

# Wait for table to become active
aws dynamodb wait table-exists --table-name MusicTable --region us-east-2

# Step 2: Add replica and witness with Multi-Region Strong Consistency
# MRSC requires exactly three replicas in supported regions
aws dynamodb update-table \
    --table-name MusicTable \
    --replica-updates '[{"Create": {"RegionName": "us-east-1"}}]' \
    --global-table-witness-updates '[{"Create": {"RegionName": "us-west-2"}}]' \
    --multi-region-consistency STRONG \
    --region us-east-2
```

Verificare la configurazione MRSC e lo stato della replica.

```
# Verify the global table configuration and MRSC setting
aws dynamodb describe-table \
    --table-name MusicTable \
    --region us-east-2 \
    --query 'Table.
{TableName:TableName,TableStatus:TableStatus,MultiRegionConsistency:MultiRegionConsistency,R
{Region:RegionName,Status:ReplicaStatus}}'
```

Verifica l'elevata consistenza con letture immediate tra Regioni.

```
# Write an item to the primary region
aws dynamodb put-item \
    --table-name MusicTable \
    --item '{"Artist": {"S":"The Beatles"},"SongTitle": {"S":"Hey Jude"},"Album":
 {"S":"The Beatles 1967-1970"},"Year": {"N":"1968"}}' \
    --region us-east-2

# Read the item from replica region to verify strong consistency (cannot read or
 write to witness)
# No wait time needed - MRSC provides immediate consistency
echo "Reading from us-east-1 (immediate consistency):"
aws dynamodb get-item \
    --table-name MusicTable \
    --key '{"Artist": {"S":"The Beatles"},"SongTitle": {"S":"Hey Jude"}}' \
    --consistent-read \
    --region us-east-1
```

Eseguire scritture condizionali con garanzie MRSC.

```
# Perform a conditional update from a different region
# This demonstrates that conditions work consistently across all regions
aws dynamodb update-item \
    --table-name MusicTable \
    --key '{"Artist": {"S":"The Beatles"},"SongTitle": {"S":"Hey Jude"}}' \
    --update-expression "SET #rating = :rating" \
    --condition-expression "attribute_exists(Artist)" \
    --expression-attribute-names '{"#rating": "Rating"}' \
    --expression-attribute-values '{":rating": {"N":"5"}}' \
```

```
    --region us-east-1
```

Eseguire la pulizia delle risorse della tabella globale MRSC.

```
# Remove replica tables (must be done before deleting the primary table)
aws dynamodb update-table \
    --table-name MusicTable \
    --replica-updates '[{"Delete": {"RegionName": "us-east-1"}}]' \
    --global-table-witness-updates '[{"Delete": {"RegionName": "us-west-2"}}]' \
    --region us-east-2

# Wait for replicas to be deleted
echo "Waiting for replicas to be deleted..."
sleep 30

# Delete the primary table
aws dynamodb delete-table \
    --table-name MusicTable \
    --region us-east-2
```

- Per informazioni dettagliate sull'API, consulta i seguenti argomenti nella documentazione di riferimento dei comandi della AWS CLI .
  - CreateTable
  - DeleteTable
  - DescribeTable
  - GetItem
  - PutItem
  - UpdateItem
  - UpdateTable

Gestire gli indici secondari globali

L'esempio di codice seguente mostra come gestire l'intero ciclo di vita degli indici secondari globali.

- Creare una tabella con un indice secondario globale.
- Aggiungere un nuovo indice secondario globale a una tabella esistente.

- Aggiornare (aumentare) il throughput a caldo dell'indice secondario globale (GSI).
- Interroga i dati usando GSIs.
- Eliminare un indice secondario globale.

AWS CLI con lo script Bash

Creare una tabella con un indice secondario globale.

```
# Create a table with a GSI
aws dynamodb create-table \
    --table-name MusicCollection \
    --attribute-definitions \
        AttributeName=Artist,AttributeType=S \
        AttributeName=SongTitle,AttributeType=S \
        AttributeName=AlbumTitle,AttributeType=S \
    --key-schema \
        AttributeName=Artist,KeyType=HASH \
        AttributeName=SongTitle,KeyType=RANGE \
    --billing-mode PAY_PER_REQUEST \
    --global-secondary-indexes \
        "IndexName=AlbumIndex,\
        KeySchema=[{AttributeName=AlbumTitle,KeyType=HASH}],\
        Projection={ProjectionType=ALL}"
```

Aggiunge un nuovo indice secondario globale (on demand) a una tabella esistente.

```
# Add a new GSI to an existing table
aws dynamodb update-table \
    --table-name MusicCollection \
    --attribute-definitions \
        AttributeName=Genre,AttributeType=S \
    --global-secondary-index-updates \
        "[{\"Create\":{\"IndexName\":\"GenreIndex\",\
        \"KeySchema\":[{\"AttributeName\":\"Genre\",\"KeyType\":\"HASH\"}],\
        \"Projection\":{\"ProjectionType\":\"ALL\"}}}]"
```

Aggiornare (aumentare) il throughput a caldo dell'indice secondario globale (GSI).

```
# Increase the warm throughput of a GSI (default values are 12k reads, 4k writes)
```

```
aws dynamodb update-table \
    --table-name MusicCollection \
    --global-secondary-index-updates \
        "[{\"Update\":{\"IndexName\":\"AlbumIndex\",\
        \"WarmThroughput\":{\"ReadUnitsPerSecond\":15000,\"WriteUnitsPerSecond
\":6000}}}]"
```

Interroga i dati usando GSIs.

```
# Query the AlbumIndex GSI
aws dynamodb query \
    --table-name MusicCollection \
    --index-name AlbumIndex \
    --key-condition-expression "AlbumTitle = :album" \
    --expression-attribute-values '{":album":{"S":"Let It Be"}}'

# Query the GenreIndex GSI
aws dynamodb query \
    --table-name MusicCollection \
    --index-name GenreIndex \
    --key-condition-expression "Genre = :genre" \
    --expression-attribute-values '{":genre":{"S":"Jazz"}}'
```

Eliminare un indice secondario globale.

```
# Delete a GSI from a table
aws dynamodb update-table \
    --table-name MusicCollection \
    --global-secondary-index-updates \
        "[{\"Delete\":{\"IndexName\":\"GenreIndex\"}}]"
```

- Per informazioni dettagliate sull'API, consulta i seguenti argomenti nella documentazione di riferimento dei comandi della AWS CLI .
  - CreateTable
  - DeleteTable
  - Query
  - UpdateTable

Gestire le policy basate sulle risorse

L'esempio di codice seguente mostra come gestire l'intero ciclo di vita delle policy basate sulle risorse per le tabelle DynamoDB.

- Creare una tabella con una policy di risorse.
- Ottenere una policy di risorse.
- Aggiornare una policy di risorse.
- Eliminare una policy di risorse.

AWS CLI con lo script Bash

Creare una tabella con una policy di risorse.

```
# Step 1: Create a DynamoDB table
aws dynamodb create-table \
    --table-name MusicCollection \
    --attribute-definitions \
        AttributeName=Artist,AttributeType=S \
        AttributeName=SongTitle,AttributeType=S \
    --key-schema \
        AttributeName=Artist,KeyType=HASH \
        AttributeName=SongTitle,KeyType=RANGE \
    --billing-mode PAY_PER_REQUEST

# Step 2: Create a resource-based policy document
cat > policy.json << 'EOF'
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::123456789012:role/DynamoDBReadOnly"
      },
      "Action": [
        "dynamodb:GetItem",
        "dynamodb:BatchGetItem",
        "dynamodb:Query",
        "dynamodb:Scan"
      ],
      "Resource": "arn:aws:dynamodb:us-west-2:123456789012:table/MusicCollection"
```

```
    }
  ]
}
EOF

# Step 3: Attach the resource-based policy to the table
aws dynamodb put-resource-policy \
    --resource-arn arn:aws:dynamodb:us-west-2:123456789012:table/MusicCollection \
    --policy file://policy.json
```

Ottenere una policy di risorse.

```
# Get the resource-based policy attached to a table
aws dynamodb get-resource-policy \
    --resource-arn arn:aws:dynamodb:us-west-2:123456789012:table/MusicCollection
```

Aggiornare una policy di risorse.

```
# Step 1: Create an updated policy document
cat > updated-policy.json << 'EOF'
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": [
          "arn:aws:iam::123456789012:role/DynamoDBReadOnly",
          "arn:aws:iam::123456789012:role/DynamoDBAnalytics"
        ]
      },
      "Action": [
        "dynamodb:GetItem",
        "dynamodb:BatchGetItem",
        "dynamodb:Query",
        "dynamodb:Scan"
      ],
      "Resource": "arn:aws:dynamodb:us-west-2:123456789012:table/MusicCollection"
    }
  ]
}
```

```
EOF

# Step 2: Update the resource-based policy on the table
aws dynamodb put-resource-policy \
    --resource-arn arn:aws:dynamodb:us-west-2:123456789012:table/MusicCollection \
    --policy file://updated-policy.json
```

Eliminare una policy di risorse.

```
# Delete the resource-based policy from a table
aws dynamodb delete-resource-policy \
    --resource-arn arn:aws:dynamodb:us-west-2:123456789012:table/MusicCollection
```

- Per informazioni dettagliate sull'API, consulta i seguenti argomenti nella documentazione di riferimento dei comandi della AWS CLI .
  - CreateTable
  - DeleteResourcePolicy
  - GetResourcePolicy
  - PutResourcePolicy

Configurare il controllo degli accessi basato su attributi

L'esempio di codice seguente mostra come implementare il controllo degli accessi basato su attributi (ABAC) per DynamoDB.

- Creare una policy IAM per il controllo degli accessi basato su attributi (ABAC).
- Creare tabelle con tag per reparti diversi.
- Elencare e filtrare le tabelle in base ai tag.

AWS CLI con lo script Bash

Creare una policy IAM per il controllo degli accessi basato su attributi (ABAC).

```
# Step 1: Create a policy document for ABAC
cat > abac-policy.json << 'EOF'
{
  "Version":"2012-10-17",
```

```
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "dynamodb:GetItem",
        "dynamodb:BatchGetItem",
        "dynamodb:Query",
        "dynamodb:Scan"
      ],
      "Resource": "arn:aws:dynamodb:*:*:table/*",
      "Condition": {
        "StringEquals": {
          "aws:ResourceTag/Department": "${aws:PrincipalTag/Department}"
        }
      }
    },
    {
      "Effect": "Allow",
      "Action": [
        "dynamodb:PutItem",
        "dynamodb:UpdateItem",
        "dynamodb:DeleteItem",
        "dynamodb:BatchWriteItem"
      ],
      "Resource": "arn:aws:dynamodb:*:*:table/*",
      "Condition": {
        "StringEquals": {
          "aws:ResourceTag/Department": "${aws:PrincipalTag/Department}",
          "aws:ResourceTag/Environment": "Development"
        }
      }
    }
  ]
}
EOF

# Step 2: Create the IAM policy
aws iam create-policy \
    --policy-name DynamoDBDepartmentBasedAccess \
    --policy-document file://abac-policy.json
```

Creare tabelle con tag per reparti diversi.

```
# Create a DynamoDB table with tags for ABAC
aws dynamodb create-table \
    --table-name FinanceData \
    --attribute-definitions \
        AttributeName=RecordID,AttributeType=S \
    --key-schema \
        AttributeName=RecordID,KeyType=HASH \
    --billing-mode PAY_PER_REQUEST \
    --tags \
        Key=Department,Value=Finance \
        Key=Environment,Value=Development

# Create another table with different tags
aws dynamodb create-table \
    --table-name MarketingData \
    --attribute-definitions \
        AttributeName=RecordID,AttributeType=S \
    --key-schema \
        AttributeName=RecordID,KeyType=HASH \
    --billing-mode PAY_PER_REQUEST \
    --tags \
        Key=Department,Value=Marketing \
        Key=Environment,Value=Production
```

Elencare e filtrare le tabelle in base ai tag.

```
# List all DynamoDB tables
echo "Listing all tables:"
aws dynamodb list-tables

# Get ARNs for all tables
echo -e "\nGetting ARNs for all tables:"
TABLE_ARNS=$(aws dynamodb list-tables --query "TableNames[*]" --output text | xargs
 -I {} aws dynamodb describe-table --table-name {} --query "Table.TableArn" --output
 text)

# For each table ARN, list its tags
echo -e "\nListing tags for each table:"
for ARN in $TABLE_ARNS; do
    TABLE_NAME=$(echo $ARN | awk -F/ '{print $2}')
    echo -e "\nTags for table: $TABLE_NAME"
    aws dynamodb list-tags-of-resource --resource-arn $ARN
```

```
done

# Example: Find tables with a specific tag
echo -e "\nFinding tables with Environment=Production tag:"
for ARN in $TABLE_ARNS; do
    TABLE_NAME=$(echo $ARN | awk -F/ '{print $2}')
    TAGS=$(aws dynamodb list-tags-of-resource --resource-arn $ARN --query "Tags[?
Key=='Environment' && Value=='Production']" --output text)
    if [ ! -z "$TAGS" ]; then
        echo "Table with Production tag: $TABLE_NAME"
    fi
done
```

- Per informazioni dettagliate sull'API, consulta i seguenti argomenti nella documentazione di riferimento dei comandi della AWS CLI .
  - CreatePolicy
  - CreateTable
  - ListTables

Utilizzare gli indici secondari locali

L'esempio di codice seguente mostra come creare ed eseguire query su tabelle con indici secondari locali.

- Creare una tabella con un indice secondario locale.
- Crea una tabella LSIs con più tipi di proiezione.
- Interroga i dati utilizzando LSIs.

AWS CLI con lo script Bash

Crea una tabella con un indice secondario locale.

```
# Create a table with a Local Secondary Index
aws dynamodb create-table \
    --table-name CustomerOrders \
    --attribute-definitions \
        AttributeName=CustomerID,AttributeType=S \
        AttributeName=OrderID,AttributeType=S \
```

```
        AttributeName=OrderDate,AttributeType=S \
    --key-schema \
        AttributeName=CustomerID,KeyType=HASH \
        AttributeName=OrderID,KeyType=RANGE \
    --local-secondary-indexes \
        "IndexName=OrderDateIndex,\
        KeySchema=[{AttributeName=CustomerID,KeyType=HASH},
{AttributeName=OrderDate,KeyType=RANGE}],\
        Projection={ProjectionType=ALL}" \
    --billing-mode PAY_PER_REQUEST
```

Crea una tabella con più LSIs di una tabella.

```
# Create a table with multiple Local Secondary Indexes
aws dynamodb create-table \
    --table-name CustomerDetails \
    --attribute-definitions \
        AttributeName=CustomerID,AttributeType=S \
        AttributeName=Name,AttributeType=S \
        AttributeName=Email,AttributeType=S \
        AttributeName=RegistrationDate,AttributeType=S \
    --key-schema \
        AttributeName=CustomerID,KeyType=HASH \
        AttributeName=Name,KeyType=RANGE \
    --local-secondary-indexes \
        "[
            {
                \"IndexName\": \"EmailIndex\",
                \"KeySchema\": [
                    {\"AttributeName\":\"CustomerID\",\"KeyType\":\"HASH\"},
                    {\"AttributeName\":\"Email\",\"KeyType\":\"RANGE\"}
                ],
                \"Projection\": {\"ProjectionType\":\"INCLUDE\",\"NonKeyAttributes
\":[\"Address\",\"Phone\"]}
            },
            {
                \"IndexName\": \"RegistrationIndex\",
                \"KeySchema\": [
                    {\"AttributeName\":\"CustomerID\",\"KeyType\":\"HASH\"},
                    {\"AttributeName\":\"RegistrationDate\",\"KeyType\":\"RANGE\"}
                ],
                \"Projection\": {\"ProjectionType\":\"KEYS_ONLY\"}
```

```
            }
        ]" \
    --billing-mode PAY_PER_REQUEST
```

Interroga i dati utilizzando LSIs.

```
# Query the OrderDateIndex LSI
aws dynamodb query \
    --table-name CustomerOrders \
    --index-name OrderDateIndex \
    --key-condition-expression "CustomerID = :custId AND OrderDate BETWEEN :date1
 AND :date2" \
    --expression-attribute-values '{
        ":custId": {"S": "C1"},
        ":date1": {"S": "2023-01-01"},
        ":date2": {"S": "2023-02-01"}
    }'

# Query with a filter expression
aws dynamodb query \
    --table-name CustomerOrders \
    --index-name OrderDateIndex \
    --key-condition-expression "CustomerID = :custId" \
    --filter-expression "Amount > :amount" \
    --expression-attribute-values '{
        ":custId": {"S": "C1"},
        ":amount": {"N": "150"}
    }'
```

- Per informazioni dettagliate sull'API, consulta i seguenti argomenti nella documentazione di riferimento dei comandi della AWS CLI .
  - CreateTable
  - Query

Lavora con Streams e Time-to-Live

Il seguente esempio di codice mostra come gestire i flussi e le funzionalità di DynamoDB. Time-to-Live

- Creare una tabella con i flussi abilitati.
- Descrivere i flussi.
- Creare una funzione Lambda per l'elaborazione dei flussi.
- Abilitare TTL in una tabella.
- Aggiungere elementi con attributi TTL.
- Descrivere le impostazioni TTL.

AWS CLI con lo script Bash

Creare una tabella con i flussi abilitati.

```
# Create a table with DynamoDB Streams enabled
aws dynamodb create-table \
    --table-name StreamsDemo \
    --attribute-definitions \
        AttributeName=ID,AttributeType=S \
    --key-schema \
        AttributeName=ID,KeyType=HASH \
    --billing-mode PAY_PER_REQUEST \
    --stream-specification StreamEnabled=true,StreamViewType=NEW_AND_OLD_IMAGES
```

Descrivere i flussi.

```
# Get information about the stream
aws dynamodb describe-table \
    --table-name StreamsDemo \
    --query "Table.StreamSpecification"

# Get the stream ARN
STREAM_ARN=$(aws dynamodb describe-table \
    --table-name StreamsDemo \
    --query "Table.LatestStreamArn" \
    --output text)

echo "Stream ARN: $STREAM_ARN"

# Describe the stream
aws dynamodbstreams describe-stream \
```

```
    --stream-arn $STREAM_ARN
```

Crea una funzione Lambda per i flussi.

```
# Step 1: Create an IAM role for the Lambda function
cat > trust-policy.json << 'EOF'
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "lambda.amazonaws.com"
      },
      "Action": "sts:AssumeRole"
    }
  ]
}
EOF

aws iam create-role \
    --role-name DynamoDBStreamsLambdaRole \
    --assume-role-policy-document file://trust-policy.json

# Step 2: Attach permissions to the role
aws iam attach-role-policy \
    --role-name DynamoDBStreamsLambdaRole \
    --policy-arn arn:aws:iam::aws:policy/service-role/AWSLambdaDynamoDBExecutionRole

# Step 3: Create a Lambda function (code would be in a separate file)
echo "Lambda function creation would be done separately with appropriate code"

# Step 4: Create an event source mapping
echo "Example command to create event source mapping:"
echo "aws lambda create-event-source-mapping \\"
echo "    --function-name ProcessDynamoDBRecords \\"
echo "    --event-source $STREAM_ARN \\"
echo "    --batch-size 100 \\"
echo "    --starting-position LATEST"
```

Abilitare TTL in una tabella.

```
# Create a table for TTL demonstration
aws dynamodb create-table \
    --table-name TTLDemo \
    --attribute-definitions \
        AttributeName=ID,AttributeType=S \
    --key-schema \
        AttributeName=ID,KeyType=HASH \
    --billing-mode PAY_PER_REQUEST

# Wait for table to become active
aws dynamodb wait table-exists --table-name TTLDemo

# Enable TTL on the table
aws dynamodb update-time-to-live \
    --table-name TTLDemo \
    --time-to-live-specification "Enabled=true, AttributeName=ExpirationTime"
```

Aggiungere elementi con attributi TTL.

```
# Calculate expiration time (current time + 1 day in seconds)
EXPIRATION_TIME=$(date -d "+1 day" +%s)

# Add an item with TTL attribute
aws dynamodb put-item \
    --table-name TTLDemo \
    --item '{
        "ID": {"S": "item1"},
        "Data": {"S": "This item will expire in 1 day"},
        "ExpirationTime": {"N": "'$EXPIRATION_TIME'"}
    }'

# Add an item that expires in 1 hour
EXPIRATION_TIME_HOUR=$(date -d "+1 hour" +%s)
aws dynamodb put-item \
    --table-name TTLDemo \
    --item '{
        "ID": {"S": "item2"},
        "Data": {"S": "This item will expire in 1 hour"},
        "ExpirationTime": {"N": "'$EXPIRATION_TIME_HOUR'"}
    }'
```

Descrivere le impostazioni TTL.

```
# Describe TTL settings for a table
aws dynamodb describe-time-to-live \
    --table-name TTLDemo
```

- Per informazioni dettagliate sull'API, consulta i seguenti argomenti nella documentazione di riferimento dei comandi della AWS CLI .
  - AttachRolePolicy
  - CreateRole
  - CreateTable
  - DescribeTable
  - DescribeTimeToLive
  - PutItem
  - UpdateTimeToLive

Utilizzare tabelle globali e la replica multi-Regione con consistenza effettiva (MREC)

Gli esempi di codice seguenti mostrano come creare e gestire le tabelle DynamoDB globali con consistenza effettiva multi-Regione (MRSC).

- Creare una tabella con replica multi-Regione (MREC).
- Inserire e ottenere elementi dalle tabelle di replica.
- Rimuovi le repliche one-by-one.
- Eseguire la pulizia mediante l'eliminazione delle tabelle.

AWS CLI con lo script Bash

Creare una tabella con replica multi-Regione.

```
# Step 1: Create a new table (MusicTable) in US East (Ohio), with DynamoDB Streams
 enabled (NEW_AND_OLD_IMAGES)
aws dynamodb create-table \
    --table-name MusicTable \
    --attribute-definitions \
        AttributeName=Artist,AttributeType=S \
```

```
        AttributeName=SongTitle,AttributeType=S \
    --key-schema \
        AttributeName=Artist,KeyType=HASH \
        AttributeName=SongTitle,KeyType=RANGE \
    --billing-mode PAY_PER_REQUEST \
    --stream-specification StreamEnabled=true,StreamViewType=NEW_AND_OLD_IMAGES \
    --region us-east-2

# Step 2: Create an identical MusicTable table in US East (N. Virginia)
aws dynamodb update-table --table-name MusicTable --cli-input-json \
'{
  "ReplicaUpdates":
  [
    {
      "Create": {
        "RegionName": "us-east-1"
      }
    }
  ]
}' \
--region us-east-2

# Step 3: Create a table in Europe (Ireland)
aws dynamodb update-table --table-name MusicTable --cli-input-json \
'{
  "ReplicaUpdates":
  [
    {
      "Create": {
        "RegionName": "eu-west-1"
      }
    }
  ]
}' \
--region us-east-2
```

Descrivere la tabella multi-Regione.

```
# Step 4: View the list of replicas created using describe-table
aws dynamodb describe-table \
    --table-name MusicTable \
    --region us-east-2 \
```

```
    --query 'Table.
{TableName:TableName,TableStatus:TableStatus,MultiRegionConsistency:MultiRegionConsistency,R
{Region:RegionName,Status:ReplicaStatus}}'
```

Inserisce elementi in una tabella di replica.

```
# Step 5: To verify that replication is working, add a new item to the Music table
 in US East (Ohio)
aws dynamodb put-item \
    --table-name MusicTable \
    --item '{"Artist": {"S":"item_1"},"SongTitle": {"S":"Song Value 1"}}' \
    --region us-east-2
```

Ottiene elementi dalle tabelle di replica.

```
# Step 6: Wait for a few seconds, and then check to see whether the item has been
# successfully replicated to US East (N. Virginia) and Europe (Ireland)
aws dynamodb get-item \
    --table-name MusicTable \
    --key '{"Artist": {"S":"item_1"},"SongTitle": {"S":"Song Value 1"}}' \
    --region us-east-1

aws dynamodb get-item \
    --table-name MusicTable \
    --key '{"Artist": {"S":"item_1"},"SongTitle": {"S":"Song Value 1"}}' \
    --region eu-west-1
```

Rimuove le repliche.

```
# Step 7: Delete the replica table in Europe (Ireland) Region
aws dynamodb update-table --table-name MusicTable --cli-input-json \
'{
  "ReplicaUpdates":
  [
    {
      "Delete": {
        "RegionName": "eu-west-1"
      }
    }
```

```
  ]
}' \
--region us-east-2

# Delete the replica table in US East (N. Virginia) Region
aws dynamodb update-table --table-name MusicTable --cli-input-json \
'{
  "ReplicaUpdates":
  [
    {
      "Delete": {
        "RegionName": "us-east-1"
      }
    }
  ]
}' \
--region us-east-2
```

Eseguire la pulizia mediante l'eliminazione delle tabelle.

```
# Clean up: Delete the primary table
aws dynamodb delete-table --table-name MusicTable --region us-east-2

echo "Global table demonstration complete."
```

- Per informazioni dettagliate sull'API, consulta i seguenti argomenti nella documentazione di riferimento dei comandi della AWS CLI .
  - CreateTable
  - DeleteTable
  - DescribeTable
  - GetItem
  - PutItem
  - UpdateTable

Utilizzare il tagging delle risorse

L'esempio di codice seguente mostra come gestire i tag per le risorse DynamoDB.

- Creare una tabella con i tag.
- Elenca i tag associati a una risorsa.
- Aggiungere tag a una risorsa.
- Rimozione di tag da una risorsa.
- Filtrare le tabelle in base ai tag.

AWS CLI con lo script Bash

Creare una tabella con i tag.

```
# Create a table with tags
aws dynamodb create-table \
    --table-name TaggedTable \
    --attribute-definitions \
        AttributeName=ID,AttributeType=S \
    --key-schema \
        AttributeName=ID,KeyType=HASH \
    --billing-mode PAY_PER_REQUEST \
    --tags \
        Key=Environment,Value=Production \
        Key=Project,Value=Analytics \
        Key=Owner,Value=DataTeam
```

Elenca i tag associati a una risorsa.

```
# Get the table ARN
TABLE_ARN=$(aws dynamodb describe-table \
    --table-name TaggedTable \
    --query "Table.TableArn" \
    --output text)

# List tags for the table
aws dynamodb list-tags-of-resource \
    --resource-arn $TABLE_ARN
```

Aggiungere tag a una risorsa.

```
# Add tags to an existing table
```

```
aws dynamodb tag-resource \
    --resource-arn $TABLE_ARN \
    --tags \
        Key=CostCenter,Value=12345 \
        Key=BackupSchedule,Value=Daily
```

Rimozione di tag da una risorsa.

```
# Remove tags from a table
aws dynamodb untag-resource \
    --resource-arn $TABLE_ARN \
    --tag-keys Owner BackupSchedule
```

Filtrare le tabelle in base ai tag.

```
# Create another table with different tags
aws dynamodb create-table \
    --table-name AnotherTaggedTable \
    --attribute-definitions \
        AttributeName=ID,AttributeType=S \
    --key-schema \
        AttributeName=ID,KeyType=HASH \
    --billing-mode PAY_PER_REQUEST \
    --tags \
        Key=Environment,Value=Development \
        Key=Project,Value=Testing

# Wait for table to become active
aws dynamodb wait table-exists --table-name AnotherTaggedTable

# List all tables
echo "All tables:"
aws dynamodb list-tables

# Get ARNs for all tables
echo -e "\nFiltering tables by Environment=Production tag:"
TABLE_ARNS=$(aws dynamodb list-tables --query "TableNames[*]" --output text | xargs
 -I {} aws dynamodb describe-table --table-name {} --query "Table.TableArn" --output
 text)

# Find tables with specific tag
```

```
for ARN in $TABLE_ARNS; do
    TABLE_NAME=$(echo $ARN | awk -F/ '{print $2}')
    TAGS=$(aws dynamodb list-tags-of-resource --resource-arn $ARN --query "Tags[?
Key=='Environment' && Value=='Production']" --output text)
    if [ ! -z "$TAGS" ]; then
        echo "Table with Production tag: $TABLE_NAME"
    fi
done
```

- Per informazioni dettagliate sull'API, consulta i seguenti argomenti nella documentazione di riferimento dei comandi della AWS CLI .
  - CreateTable
  - ListTagsOfResource
  - TagResource
  - UntagResource

Utilizzare la crittografia delle tabelle

L'esempio di codice seguente mostra come gestire le opzioni di crittografia per le tabelle DynamoDB.

- Creare un tabella con la crittografia predefinita.
- Creare una tabella con una chiave gestita dal cliente (CMK).
- Aggiornare le impostazioni di crittografia delle tabelle.
- Descrivere la crittografia delle tabelle.

AWS CLI con lo script Bash

Creare un tabella con la crittografia predefinita.

```
# Create a table with default encryption (AWS owned key)
aws dynamodb create-table \
    --table-name CustomerData \
    --attribute-definitions \
        AttributeName=CustomerID,AttributeType=S \
    --key-schema \
        AttributeName=CustomerID,KeyType=HASH \
    --billing-mode PAY_PER_REQUEST \
```

```
    --sse-specification Enabled=true,SSEType=KMS
```

Creare una tabella con una chiave gestita dal cliente (CMK).

```
# Step 1: Create a customer managed key in KMS
aws kms create-key \
    --description "Key for DynamoDB table encryption" \
    --key-usage ENCRYPT_DECRYPT \
    --customer-master-key-spec SYMMETRIC_DEFAULT

# Store the key ID for later use
KEY_ID=$(aws kms list-keys --query "Keys[?contains(KeyArn, 'Key for
 DynamoDB')].KeyId" --output text)

# Step 2: Create a table with the customer managed key
aws dynamodb create-table \
    --table-name SensitiveData \
    --attribute-definitions \
        AttributeName=RecordID,AttributeType=S \
    --key-schema \
        AttributeName=RecordID,KeyType=HASH \
    --billing-mode PAY_PER_REQUEST \
    --sse-specification Enabled=true,SSEType=KMS,KMSMasterKeyId=$KEY_ID
```

Aggiornare la crittografia delle tabelle.

```
# Update a table to use a different KMS key
aws dynamodb update-table \
    --table-name CustomerData \
    --sse-specification Enabled=true,SSEType=KMS,KMSMasterKeyId=$KEY_ID
```

Descrivere la crittografia delle tabelle.

```
# Describe the table to see encryption settings
aws dynamodb describe-table \
    --table-name CustomerData \
    --query "Table.SSEDescription"
```

- Per informazioni dettagliate sull'API, consulta i seguenti argomenti nella documentazione di riferimento dei comandi della AWS CLI .
  - CreateKey
  - CreateTable
  - DescribeTable
  - UpdateTable

EC2 Esempi di Amazon che utilizzano AWS CLI lo script Bash

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando lo script AWS Command Line Interface with Bash con Amazon EC2.

Nozioni di base: esempi di codice che mostrano come eseguire le operazioni essenziali all'interno di un servizio.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Scenari: esempi di codice che mostrano come eseguire un'attività specifica chiamando più funzioni all'interno dello stesso servizio o combinate con altri Servizi AWS.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Nozioni di base
- Azioni
- Scenari

Nozioni di base

Informazioni di base

L'esempio di codice seguente mostra come:

- Creare una coppia di chiavi e un gruppo di sicurezza.

- Selezionare un'Amazon Machine Image (AMI) e un tipo di istanza compatibile e quindi creare un'istanza.
- Arrestare e riavviare l'istanza.
- Associazione di un indirizzo IP elastico all'istanza.
- Connettiti alla tua istanza con SSH, quindi elimina le risorse.

AWS CLI con lo script Bash

> **Note**
>
> C'è altro da fare. GitHub Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codice AWS.

Esegui uno scenario interattivo al prompt dei comandi.

```
###############################################################################
# function get_started_with_ec2_instances
#
# Runs an interactive scenario that shows how to get started using EC2 instances.
#
#     "EC2 access" permissions are needed to run this code.
#
# Returns:
#       0 - If successful.
#       1 - If an error occurred.
###############################################################################
function get_started_with_ec2_instances() {
  # Requires version 4 for mapfile.
  local required_version=4.0

  # Get the current Bash version
  # Check if BASH_VERSION is set
  local current_version
  if [[ -n "$BASH_VERSION" ]]; then
    # Convert BASH_VERSION to a number for comparison
    current_version=$BASH_VERSION
  else
    # Get the current Bash version using the bash command
    current_version=$(bash --version | head -n 1 | awk '{ print $4 }')
```

```
 fi

 # Convert version strings to numbers for comparison
 local required_version_num current_version_num
 required_version_num=$(echo "$required_version" | awk -F. '{ print ($1 * 10000) +
($2 * 100) + $3 }')
 current_version_num=$(echo "$current_version" | awk -F. '{ print ($1 * 10000) +
($2 * 100) + $3 }')

 # Compare versions
 if ((current_version_num < required_version_num)); then
   echo "Error: This script requires Bash version $required_version or higher."
   echo "Your current Bash version is number is $current_version."
   exit 1
 fi

 {
   if [ "$EC2_OPERATIONS_SOURCED" != "True" ]; then

     source ./ec2_operations.sh
   fi
 }

 echo_repeat "*" 88
 echo "Welcome to the Amazon Elastic Compute Cloud (Amazon EC2) get started with
instances demo."
 echo_repeat "*" 88
 echo

 echo "Let's create an RSA key pair that you can be use to securely connect to "
 echo "your EC2 instance."

 echo -n "Enter a unique name for your key: "
 get_input
 local key_name
 key_name=$get_input_result

 local temp_dir
 temp_dir=$(mktemp -d)
 local key_file_name="$temp_dir/${key_name}.pem"

 if ec2_create_keypair -n "${key_name}" -f "${key_file_name}"; then
   echo "Created a key pair $key_name and saved the private key to $key_file_name"
   echo
```

```
else
  errecho "The key pair failed to create. This demo will exit."
  return 1
fi

chmod 400 "${key_file_name}"

if yes_no_input "Do you want to list some of your key pairs? (y/n) "; then
  local keys_and_fingerprints
  keys_and_fingerprints="$(ec2_describe_key_pairs)" && {
    local image_name_and_id
    while IFS=$'\n' read -r image_name_and_id; do
      local entries
      IFS=$'\t' read -ra entries <<<"$image_name_and_id"
      echo "Found rsa key ${entries[0]} with fingerprint:"
      echo "     ${entries[1]}"
    done <<<"$keys_and_fingerprints"

  }
fi

echo_repeat "*" 88
echo_repeat "*" 88

echo "Let's create a security group to manage access to your instance."
echo -n "Enter a unique name for your security group: "
get_input
local security_group_name
security_group_name=$get_input_result
local security_group_id
security_group_id=$(ec2_create_security_group -n "$security_group_name" \
  -d "Security group for EC2 instance") || {
  errecho "The security failed to create. This demo will exit."
  clean_up "$key_name" "$key_file_name"
  return 1
}

echo "Security group created with ID $security_group_id"
echo

local public_ip
public_ip=$(curl -s http://checkip.amazonaws.com)

echo "Let's add a rule to allow SSH only from your current IP address."
```

```
  echo "Your public IP address is $public_ip"
  echo -n "press return to add this rule to your security group."
  get_input

  if ! ec2_authorize_security_group_ingress -g "$security_group_id" -i "$public_ip"
 -p tcp -f 22 -t 22; then
    errecho "The security group rules failed to update. This demo will exit."
    clean_up "$key_name" "$key_file_name" "$security_group_id"
    return 1
  fi

  echo "Security group rules updated"

  local security_group_description
  security_group_description="$(ec2_describe_security_groups -g
 "${security_group_id}")" || {
    errecho "Failed to describe security groups. This demo will exit."
    clean_up "$key_name" "$key_file_name" "$security_group_id"
    return 1
  }

  mapfile -t parameters <<<"$security_group_description"
  IFS=$'\t' read -ra entries <<<"${parameters[0]}"
  echo "Security group: ${entries[0]}"
  echo "    ID: ${entries[1]}"
  echo "    VPC: ${entries[2]}"
  echo "Inbound permissions:"
  IFS=$'\t' read -ra entries <<<"${parameters[1]}"
  echo "    IpProtocol: ${entries[0]}"
  echo "    FromPort: ${entries[1]}"
  echo "    ToPort: ${entries[2]}"
  echo "    CidrIp: ${parameters[2]}"

  local parameters
  parameters="$(ssm_get_parameters_by_path -p "/aws/service/ami-amazon-linux-
latest")" || {
    errecho "Failed to get parameters. This demo will exit."
    clean_up "$key_name" "$key_file_name" "$security_group_id"
    return 1

  }

  local image_ids=""
  mapfile -t parameters <<<"$parameters"
```

```
  for image_name_and_id in "${parameters[@]}"; do
    IFS=$'\t' read -ra values <<<"$image_name_and_id"
    if [[ "${values[0]}" == *"amzn2"* ]]; then
      image_ids+="${values[1]} "
    fi
  done

  local images
  images="$(ec2_describe_images -i "$image_ids")" || {
    errecho "Failed to describe images. This demo will exit."
    clean_up "$key_name" "$key_file_name" "$security_group_id"
    return 1

  }

  new_line_and_tab_to_list "$images"
  local images=("${list_result[@]}")

  # Get the size of the array
  local images_count=${#images[@]}

  if ((images_count == 0)); then
    errecho "No images found. This demo will exit."
    clean_up "$key_name" "$key_file_name" "$security_group_id"
    return 1
  fi

  echo_repeat "*" 88
  echo_repeat "*" 88

  echo "Let's create an instance from an Amazon Linux 2 AMI. Here are some options:"
  for ((i = 0; i < images_count; i += 3)); do
    echo "$(((i / 3) + 1)) - ${images[$i]}"
  done

  integer_input "Please enter the number of the AMI you want to use: " 1
"$((images_count / 3))"
  local choice=$get_input_result
  choice=$(((choice - 1) * 3))

  echo "Great choice."
  echo

  local architecture=${images[$((choice + 1))]}
```

```
  local image_id=${images[$((choice + 2))]}
  echo "Here are some instance types that support the ${architecture} architecture
 of the image:"
  response="$(ec2_describe_instance_types -a "${architecture}" -t
"*.micro,*.small")" || {
    errecho "Failed to describe instance types. This demo will exit."
    clean_up "$key_name" "$key_file_name" "$security_group_id"
    return 1
  }

  local instance_types
  mapfile -t instance_types <<<"$response"

  # Get the size of the array
  local instance_types_count=${#instance_types[@]}

  echo "Here are some options:"
  for ((i = 0; i < instance_types_count; i++)); do
    echo "$((i + 1)) - ${instance_types[$i]}"
  done

  integer_input "Which one do you want to use? " 1 "${#instance_types[@]}
"
  choice=$get_input_result
  local instance_type=${instance_types[$((choice - 1))]}
  echo "Another great choice."
  echo

  echo "Creating your instance and waiting for it to start..."
  local instance_id
  instance_id=$(ec2_run_instances -i "$image_id" -t "$instance_type" -k "$key_name"
 -s "$security_group_id") || {
    errecho "Failed to run instance. This demo will exit."
    clean_up "$key_name" "$key_file_name" "$security_group_id"
    return 1
  }

  ec2_wait_for_instance_running -i "$instance_id"
  echo "Your instance is ready:"
  echo

  local instance_details
  instance_details="$(ec2_describe_instances -i "${instance_id}")"
```

```
  echo
  print_instance_details "${instance_details}"

  local public_ip
  public_ip=$(echo "${instance_details}" | awk '{print $6}')
  echo
  echo "You can use SSH to connect to your instance"
  echo "If the connection attempt times out, you might have to manually update the
SSH ingress rule"
  echo "for your IP address in the AWS Management Console."
  connect_to_instance "$key_file_name" "$public_ip"

  echo -n "Press Enter when you're ready to continue the demo: "
  get_input

  echo_repeat "*" 88
  echo_repeat "*" 88

  echo "Let's stop and start your instance to see what changes."
  echo "Stopping your instance and waiting until it's stopped..."
  ec2_stop_instances -i "$instance_id"
  ec2_wait_for_instance_stopped -i "$instance_id"

  echo "Your instance is stopped. Restarting..."

  ec2_start_instances -i "$instance_id"
  ec2_wait_for_instance_running -i "$instance_id"

  echo "Your instance is running again."
  local instance_details
  instance_details="$(ec2_describe_instances -i "${instance_id}")"

  print_instance_details "${instance_details}"

  public_ip=$(echo "${instance_details}" | awk '{print $6}')

  echo "Every time your instance is restarted, its public IP address changes"
  connect_to_instance "$key_file_name" "$public_ip"

  echo -n "Press Enter when you're ready to continue the demo: "
  get_input

  echo_repeat "*" 88
  echo_repeat "*" 88
```

```
  echo "You can allocate an Elastic IP address and associate it with your instance"
  echo "to keep a consistent IP address even when your instance restarts."

  local result
  result=$(ec2_allocate_address -d vpc) || {
    errecho "Failed to allocate an address. This demo will exit."
    clean_up "$key_name" "$key_file_name" "$security_group_id" "$instance_id"
    return 1
  }

  local elastic_ip allocation_id
  elastic_ip=$(echo "$result" | awk '{print $1}')
  allocation_id=$(echo "$result" | awk '{print $2}')

  echo "Allocated static Elastic IP address: $elastic_ip"

  local association_id
  association_id=$(ec2_associate_address -i "$instance_id" -a "$allocation_id") || {
    errecho "Failed to associate an address. This demo will exit."
    clean_up "$key_name" "$key_file_name" "$security_group_id" "$instance_id"
"$allocation_id"
    return 1
  }

  echo "Associated your Elastic IP with your instance."
  echo "You can now use SSH to connect to your instance by using the Elastic IP."
  connect_to_instance "$key_file_name" "$elastic_ip"

  echo -n "Press Enter when you're ready to continue the demo: "
  get_input

  echo_repeat "*" 88
  echo_repeat "*" 88

  echo "Let's stop and start your instance to see what changes."
  echo "Stopping your instance and waiting until it's stopped..."
  ec2_stop_instances -i "$instance_id"
  ec2_wait_for_instance_stopped -i "$instance_id"

  echo "Your instance is stopped. Restarting..."

  ec2_start_instances -i "$instance_id"
  ec2_wait_for_instance_running -i "$instance_id"
```

```
  echo "Your instance is running again."
  local instance_details
  instance_details="$(ec2_describe_instances -i "${instance_id}")"

  print_instance_details "${instance_details}"

  echo "Because you have associated an Elastic IP with your instance, you can"
  echo "connect by using a consistent IP address after the instance restarts."
  connect_to_instance "$key_file_name" "$elastic_ip"

  echo -n "Press Enter when you're ready to continue the demo: "
  get_input

  echo_repeat "*" 88
  echo_repeat "*" 88

  if yes_no_input "Do you want to delete the resources created in this demo: (y/n)
 "; then
    clean_up "$key_name" "$key_file_name" "$security_group_id" "$instance_id" \
      "$allocation_id" "$association_id"
  else
    echo "The following resources were not deleted."
    echo "Key pair: $key_name"
    echo "Key file: $key_file_name"
    echo "Security group: $security_group_id"
    echo "Instance: $instance_id"
    echo "Elastic IP address: $elastic_ip"
  fi
}

###############################################################################
# function clean_up
#
# This function cleans up the created resources.
#     $1 - The name of the ec2 key pair to delete.
#     $2 - The name of the key file to delete.
#     $3 - The ID of the security group to delete.
#     $4 - The ID of the instance to terminate.
#     $5 - The ID of the elastic IP address to release.
#     $6 - The ID of the elastic IP address to disassociate.
#
# Returns:
#       0 - If successful.
```

```
#       1 - If an error occurred.
###############################################################################
function clean_up() {
  local result=0
  local key_pair_name=$1
  local key_file_name=$2
  local security_group_id=$3
  local instance_id=$4
  local allocation_id=$5
  local association_id=$6

  if [ -n "$association_id" ]; then
    # bashsupport disable=BP2002
    if (ec2_disassociate_address -a "$association_id"); then
      echo "Disassociated elastic IP address with ID $association_id"
    else
      errecho "The elastic IP address disassociation failed."
      result=1
    fi
  fi

  if [ -n "$allocation_id" ]; then
    # bashsupport disable=BP2002
    if (ec2_release_address -a "$allocation_id"); then
      echo "Released elastic IP address with ID $allocation_id"
    else
      errecho "The elastic IP address release failed."
      result=1
    fi
  fi

  if [ -n "$instance_id" ]; then
    # bashsupport disable=BP2002
    if (ec2_terminate_instances -i "$instance_id"); then
      echo "Started terminating instance with ID $instance_id"

      ec2_wait_for_instance_terminated -i "$instance_id"
    else
      errecho "The instance terminate failed."
      result=1
    fi
  fi

  if [ -n "$security_group_id" ]; then
```

```
    # bashsupport disable=BP2002
    if (ec2_delete_security_group -i "$security_group_id"); then
      echo "Deleted security group with ID $security_group_id"
    else
      errecho "The security group delete failed."
      result=1
    fi
  fi

  if [ -n "$key_pair_name" ]; then
    # bashsupport disable=BP2002
    if (ec2_delete_keypair -n "$key_pair_name"); then
      echo "Deleted key pair named $key_pair_name"
    else
      errecho "The key pair delete failed."
      result=1
    fi
  fi

  if [ -n "$key_file_name" ]; then
    rm -f "$key_file_name"
  fi

  return $result
}

###############################################################################
# function ssm_get_parameters_by_path
#
# This function retrieves one or more parameters from the AWS Systems Manager
 Parameter Store
# by specifying a parameter path.
#
# Parameters:
#       -p parameter_path - The path of the parameter(s) to retrieve.
#
# And:
#       0 - If successful.
#       1 - If it fails.
###############################################################################
function ssm_get_parameters_by_path() {
  local parameter_path response
  local option OPTARG # Required to use getopts command in a function.
```

```
 # bashsupport disable=BP5008
 function usage() {
   echo "function ssm_get_parameters_by_path"
   echo "Retrieves one or more parameters from the AWS Systems Manager Parameter
Store by specifying a parameter path."
   echo "  -p parameter_path - The path of the parameter(s) to retrieve."
   echo ""
 }

 # Retrieve the calling parameters.
 while getopts "p:h" option; do
   case "${option}" in
     p) parameter_path="${OPTARG}" ;;
     h)
       usage
       return 0
       ;;
     \?)
       echo "Invalid parameter"
       usage
       return 1
       ;;
   esac
 done
 export OPTIND=1

 if [[ -z "$parameter_path" ]]; then
   errecho "ERROR: You must provide a parameter path with the -p parameter."
   usage
   return 1
 fi

 response=$(aws ssm get-parameters-by-path \
   --path "$parameter_path" \
   --query "Parameters[*].[Name, Value]" \
   --output text) || {
   aws_cli_error_log $?
   errecho "ERROR: AWS reports get-parameters-by-path operation failed.$response"
   return 1
 }

 echo "$response"

 return 0
```

```
}

#############################################################################
# function print_instance_details
#
# This function prints the details of an Amazon Elastic Compute Cloud (Amazon EC2)
 instance.
#
# Parameters:
#       instance_details - The instance details in the format "InstanceId ImageId
 InstanceType KeyName VpcId PublicIpAddress State.Name".
#
# Returns:
#       0 - If successful.
#       1 - If it fails.
#############################################################################
function print_instance_details() {
  local instance_details="$1"

  if [[ -z "${instance_details}" ]]; then
    echo "Error: Missing required instance details argument."
    return 1
  fi

  local instance_id image_id instance_type key_name vpc_id public_ip state
  instance_id=$(echo "${instance_details}" | awk '{print $1}')
  image_id=$(echo "${instance_details}" | awk '{print $2}')
  instance_type=$(echo "${instance_details}" | awk '{print $3}')
  key_name=$(echo "${instance_details}" | awk '{print $4}')
  vpc_id=$(echo "${instance_details}" | awk '{print $5}')
  public_ip=$(echo "${instance_details}" | awk '{print $6}')
  state=$(echo "${instance_details}" | awk '{print $7}')

  echo "    ID: ${instance_id}"
  echo "    Image ID: ${image_id}"
  echo "    Instance type: ${instance_type}"
  echo "    Key name: ${key_name}"
  echo "    VPC ID: ${vpc_id}"
  echo "    Public IP: ${public_ip}"
  echo "    State: ${state}"

  return 0
}
```

```
###############################################################################
# function connect_to_instance
#
# This function displays the public IP address of an Amazon Elastic Compute Cloud
 (Amazon EC2) instance and prompts the user to connect to the instance via SSH.
#
# Parameters:
#       $1 - The name of the key file used to connect to the instance.
#       $2 - The public IP address of the instance.
#
# Returns:
#       None
###############################################################################
function connect_to_instance() {
  local key_file_name="$1"
  local public_ip="$2"

  # Validate the input parameters
  if [[ -z "$key_file_name" ]]; then
    echo "ERROR: You must provide a key file name as the first argument." >&2
    return 1
  fi

  if [[ -z "$public_ip" ]]; then
    echo "ERROR: You must provide a public IP address as the second argument." >&2
    return 1
  fi

  # Display the public IP address and connection command
  echo "To connect, run the following command:"
  echo "    ssh -i ${key_file_name} ec2-user@${public_ip}"

  # Prompt the user to connect to the instance
  if yes_no_input "Do you want to connect now? (y/n) "; then
    echo "After you have connected, you can return to this example by typing 'exit'"
    ssh -i "${key_file_name}" ec2-user@"${public_ip}"
  fi
}

###############################################################################
# function get_input
#
# This function gets user input from the command line.
#
```

```
# Outputs:
#   User input to stdout.
#
# Returns:
#       0
###############################################################################
function get_input() {

  if [ -z "${mock_input+x}" ]; then
    read -r get_input_result
  else

    if [ "$mock_input_array_index" -lt ${#mock_input_array[@]} ]; then
      get_input_result="${mock_input_array[$mock_input_array_index]}"
      # bashsupport disable=BP2001
      # shellcheck disable=SC2206
      ((mock_input_array_index++))
      echo -n "$get_input_result"
    else
      echo "MOCK_INPUT_ARRAY has no more elements" 1>&2
      return 1
    fi
  fi

  return 0
}

###############################################################################
# function yes_no_input
#
# This function requests a yes/no answer from the user, following to a prompt.
#
# Parameters:
#       $1 - The prompt.
#
# Returns:
#       0 - If yes.
#       1 - If no.
###############################################################################
function yes_no_input() {
  if [ -z "$1" ]; then
    echo "Internal error yes_no_input"
    return 1
  fi
```

```
  local index=0
  local response="N"
  while [[ $index -lt 10 ]]; do
    index=$((index + 1))
    echo -n "$1"
    if ! get_input; then
      return 1
    fi
    response=$(echo "$get_input_result" | tr '[:upper:]' '[:lower:]')
    if [ "$response" = "y" ] || [ "$response" = "n" ]; then
      break
    else
      echo -e "\nPlease enter or 'y' or 'n'."
    fi
  done

  echo

  if [ "$response" = "y" ]; then
    return 0
  else
    return 1
  fi
}

###############################################################################
# function integer_input
#
# This function prompts the user to enter an integer within a specified range
# and validates the input.
#
# Parameters:
#       $1 - The prompt message to display to the user.
#       $2 - The minimum value of the accepted range.
#       $3 - The maximum value of the accepted range.
#
# Returns:
#       The valid integer input from the user.
#       If the input is invalid or out of range, the function will continue
#       prompting the user until a valid input is provided.
###############################################################################
function integer_input() {
  local prompt="$1"
```

```
  local min_value="$2"
  local max_value="$3"
  local input=""

  while true; do
    # Display the prompt message and wait for user input
    echo -n "$prompt"

    if ! get_input; then
      return 1
    fi

    input="$get_input_result"

    # Check if the input is a valid integer
    if [[ "$input" =~ ^-?[0-9]+$ ]]; then
      # Check if the input is within the specified range
      if ((input >= min_value && input <= max_value)); then
        return 0
      else
        echo "Error: Input, $input, must be between $min_value and $max_value."
      fi
    else
      echo "Error: Invalid input- $input. Please enter an integer."
    fi
  done
}
################################################################################
# function new_line_and_tab_to_list
#
# This function takes a string input containing newlines and tabs, and
# converts it into a list (array) of elements.
#
# Parameters:
#       $1 - The input string containing newlines and tabs.
#
# Returns:
#       The resulting list (array) is stored in the global variable
#       'list_result'.
################################################################################
function new_line_and_tab_to_list() {
  local input=$1
  export list_result
```

```
  list_result=()
  mapfile -t lines <<<"$input"
  local line
  for line in "${lines[@]}"; do
    IFS=$'\t' read -ra parameters <<<"$line"
    list_result+=("${parameters[@]}")
  done
}

###############################################################################
# function echo_repeat
#
# This function prints a string 'n' times to stdout.
#
# Parameters:
#       $1 - The string.
#       $2 - Number of times to print the string.
#
# Outputs:
#   String 'n' times to stdout.
#
# Returns:
#       0
###############################################################################
function echo_repeat() {
  local end=$2
  for ((i = 0; i < end; i++)); do
    echo -n "$1"
  done
  echo
}
```

Le funzioni DynamoDB utilizzate in questo scenario.

```
###############################################################################
# function ec2_create_keypair
#
# This function creates an Amazon Elastic Compute Cloud (Amazon EC2) ED25519 or
 2048-bit RSA key pair
# and writes it to a file.
#
# Parameters:
```

```
#       -n key_pair_name - A key pair name.
#       -f file_path - File to store the key pair.
#
# And:
#       0 - If successful.
#       1 - If it fails.
###############################################################################
function ec2_create_keypair() {
  local key_pair_name file_path response
  local option OPTARG # Required to use getopts command in a function.

  # bashsupport disable=BP5008
  function usage() {
    echo "function ec2_create_keypair"
    echo "Creates an Amazon Elastic Compute Cloud (Amazon EC2) ED25519 or 2048-bit
 RSA key pair"
    echo " and writes it to a file."
    echo "  -n key_pair_name - A key pair name."
    echo "  -f file_path - File to store the key pair."
    echo ""
  }

  # Retrieve the calling parameters.
  while getopts "n:f:h" option; do
    case "${option}" in
      n) key_pair_name="${OPTARG}" ;;
      f) file_path="${OPTARG}" ;;
      h)
        usage
        return 0
        ;;
      \?)
        echo "Invalid parameter"
        usage
        return 1
        ;;
    esac
  done
  export OPTIND=1

  if [[ -z "$key_pair_name" ]]; then
    errecho "ERROR: You must provide a key name with the -n parameter."
    usage
    return 1
```

```
  fi

  if [[ -z "$file_path" ]]; then
    errecho "ERROR: You must provide a file path with the -f parameter."
    usage
    return 1
  fi

  response=$(aws ec2 create-key-pair \
    --key-name "$key_pair_name" \
    --query 'KeyMaterial' \
    --output text) || {
    aws_cli_error_log ${?}
    errecho "ERROR: AWS reports create-access-key operation failed.$response"
    return 1
  }

  if [[ -n "$file_path" ]]; then
    echo "$response" >"$file_path"
  fi

  return 0
}

###############################################################################
# function ec2_describe_key_pairs
#
# This function describes one or more Amazon Elastic Compute Cloud (Amazon EC2) key
 pairs.
#
# Parameters:
#       -h - Display help.
#
# And:
#       0 - If successful.
#       1 - If it fails.
###############################################################################
function ec2_describe_key_pairs() {
  local option OPTARG # Required to use getopts command in a function.

  # bashsupport disable=BP5008
  function usage() {
    echo "function ec2_describe_key_pairs"
```

```
    echo "Describes one or more Amazon Elastic Compute Cloud (Amazon EC2) key
 pairs."
    echo "  -h - Display help."
    echo ""
  }

  # Retrieve the calling parameters.
  while getopts "h" option; do
    case "${option}" in
      h)
        usage
        return 0
        ;;
      \?)
        echo "Invalid parameter"
        usage
        return 1
        ;;
    esac
  done
  export OPTIND=1

  local response

  response=$(aws ec2 describe-key-pairs \
    --query 'KeyPairs[*].[KeyName, KeyFingerprint]' \
    --output text) || {
    aws_cli_error_log ${?}
    errecho "ERROR: AWS reports describe-key-pairs operation failed.$response"
    return 1
  }

  echo "$response"

  return 0
}

################################################################################
# function ec2_create_security_group
#
# This function creates an Amazon Elastic Compute Cloud (Amazon EC2) security group.
#
# Parameters:
#       -n security_group_name - The name of the security group.
```

```
#       -d security_group_description - The description of the security group.
#
# Returns:
#       The ID of the created security group, or an error message if the operation
 fails.
# And:
#       0 - If successful.
#       1 - If it fails.
#
###############################################################################
function ec2_create_security_group() {
  local security_group_name security_group_description response

  # Function to display usage information
  function usage() {
    echo "function ec2_create_security_group"
    echo "Creates an Amazon Elastic Compute Cloud (Amazon EC2) security group."
    echo "  -n security_group_name - The name of the security group."
    echo "  -d security_group_description - The description of the security group."
    echo ""
  }

  # Parse the command-line arguments
  while getopts "n:d:h" option; do
    case "${option}" in
      n) security_group_name="${OPTARG}" ;;
      d) security_group_description="${OPTARG}" ;;
      h)
        usage
        return 0
        ;;
      \?)
        echo "Invalid parameter"
        usage
        return 1
        ;;
    esac
  done
  export OPTIND=1

  # Validate the input parameters
  if [[ -z "$security_group_name" ]]; then
    errecho "ERROR: You must provide a security group name with the -n parameter."
    return 1
```

```
  fi

  if [[ -z "$security_group_description" ]]; then
    errecho "ERROR: You must provide a security group description with the -d
 parameter."
    return 1
  fi

  # Create the security group
  response=$(aws ec2 create-security-group \
    --group-name "$security_group_name" \
    --description "$security_group_description" \
    --query "GroupId" \
    --output text) || {
    aws_cli_error_log ${?}
    errecho "ERROR: AWS reports create-security-group operation failed."
    errecho "$response"
    return 1
  }

  echo "$response"
  return 0
}

################################################################################
# function ec2_describe_security_groups
#
# This function describes one or more Amazon Elastic Compute Cloud (Amazon EC2)
 security groups.
#
# Parameters:
#       -g security_group_id - The ID of the security group to describe (optional).
#
# And:
#       0 - If successful.
#       1 - If it fails.
################################################################################
function ec2_describe_security_groups() {
  local security_group_id response
  local option OPTARG # Required to use getopts command in a function.

  # bashsupport disable=BP5008
  function usage() {
    echo "function ec2_describe_security_groups"
```

```
    echo "Describes one or more Amazon Elastic Compute Cloud (Amazon EC2) security
 groups."
    echo "  -g security_group_id - The ID of the security group to describe
 (optional)."
    echo ""
  }

  # Retrieve the calling parameters.
  while getopts "g:h" option; do
    case "${option}" in
      g) security_group_id="${OPTARG}" ;;
      h)
        usage
        return 0
        ;;
      \?)
        echo "Invalid parameter"
        usage
        return 1
        ;;
    esac
  done
  export OPTIND=1

  local query="SecurityGroups[*].[GroupName, GroupId, VpcId, IpPermissions[*].
[IpProtocol, FromPort, ToPort, IpRanges[*].CidrIp]]"

  if [[ -n "$security_group_id" ]]; then
    response=$(aws ec2 describe-security-groups --group-ids "$security_group_id" --
query "${query}" --output text)
  else
    response=$(aws ec2 describe-security-groups --query "${query}" --output text)
  fi

  local error_code=${?}

  if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports describe-security-groups operation failed.$response"
    return 1
  fi

  echo "$response"
```

```
  return 0
}

###############################################################################
# function ec2_authorize_security_group_ingress
#
# This function authorizes an ingress rule for an Amazon Elastic Compute Cloud
 (Amazon EC2) security group.
#
# Parameters:
#       -g security_group_id - The ID of the security group.
#       -i ip_address - The IP address or CIDR block to authorize.
#       -p protocol - The protocol to authorize (e.g., tcp, udp, icmp).
#       -f from_port - The start of the port range to authorize.
#       -t to_port - The end of the port range to authorize.
#
# And:
#       0 - If successful.
#       1 - If it fails.
###############################################################################
function ec2_authorize_security_group_ingress() {
  local security_group_id ip_address protocol from_port to_port response
  local option OPTARG # Required to use getopts command in a function.

  # bashsupport disable=BP5008
  function usage() {
    echo "function ec2_authorize_security_group_ingress"
    echo "Authorizes an ingress rule for an Amazon Elastic Compute Cloud (Amazon
 EC2) security group."
    echo "  -g security_group_id - The ID of the security group."
    echo "  -i ip_address - The IP address or CIDR block to authorize."
    echo "  -p protocol - The protocol to authorize (e.g., tcp, udp, icmp)."
    echo "  -f from_port - The start of the port range to authorize."
    echo "  -t to_port - The end of the port range to authorize."
    echo ""
  }

  # Retrieve the calling parameters.
  while getopts "g:i:p:f:t:h" option; do
    case "${option}" in
      g) security_group_id="${OPTARG}" ;;
      i) ip_address="${OPTARG}" ;;
      p) protocol="${OPTARG}" ;;
      f) from_port="${OPTARG}" ;;
```

```
      t) to_port="${OPTARG}" ;;
      h)
        usage
        return 0
        ;;
      \?)
        echo "Invalid parameter"
        usage
        return 1
        ;;
    esac
  done
  export OPTIND=1

  if [[ -z "$security_group_id" ]]; then
    errecho "ERROR: You must provide a security group ID with the -g parameter."
    usage
    return 1
  fi

  if [[ -z "$ip_address" ]]; then
    errecho "ERROR: You must provide an IP address or CIDR block with the -i
parameter."
    usage
    return 1
  fi

  if [[ -z "$protocol" ]]; then
    errecho "ERROR: You must provide a protocol with the -p parameter."
    usage
    return 1
  fi

  if [[ -z "$from_port" ]]; then
    errecho "ERROR: You must provide a start port with the -f parameter."
    usage
    return 1
  fi

  if [[ -z "$to_port" ]]; then
    errecho "ERROR: You must provide an end port with the -t parameter."
    usage
    return 1
  fi
```

```
  response=$(aws ec2 authorize-security-group-ingress \
    --group-id "$security_group_id" \
    --cidr "${ip_address}/32" \
    --protocol "$protocol" \
    --port "$from_port-$to_port" \
    --output text) || {
    aws_cli_error_log ${?}
    errecho "ERROR: AWS reports authorize-security-group-ingress operation failed.
$response"
    return 1
  }

  return 0
}

################################################################################
# function ec2_describe_images
#
# This function describes one or more Amazon Elastic Compute Cloud (Amazon EC2)
 images.
#
# Parameters:
#       -i image_ids - A space-separated  list of image IDs (optional).
#       -h - Display help.
#
# And:
#       0 - If successful.
#       1 - If it fails.
################################################################################
function ec2_describe_images() {
  local image_ids response
  local option OPTARG # Required to use getopts command in a function.

  # bashsupport disable=BP5008
  function usage() {
    echo "function ec2_describe_images"
    echo "Describes one or more Amazon Elastic Compute Cloud (Amazon EC2) images."
    echo "  -i image_ids - A space-separated list of image IDs (optional)."
    echo "  -h - Display help."
    echo ""
  }

  # Retrieve the calling parameters.
```

```
  while getopts "i:h" option; do
    case "${option}" in
      i) image_ids="${OPTARG}" ;;
      h)
        usage
        return 0
        ;;
      \?)
        echo "Invalid parameter"
        usage
        return 1
        ;;
    esac
  done
  export OPTIND=1

  local aws_cli_args=()

  if [[ -n "$image_ids" ]]; then
    # shellcheck disable=SC2206
    aws_cli_args+=("--image-ids" $image_ids)
  fi

  response=$(aws ec2 describe-images \
    "${aws_cli_args[@]}" \
    --query 'Images[*].[Description,Architecture,ImageId]' \
    --output text) || {
    aws_cli_error_log ${?}
    errecho "ERROR: AWS reports describe-images operation failed.$response"
    return 1
  }

  echo "$response"

  return 0
}

###############################################################################
# ec2_describe_instance_types
#
# This function describes EC2 instance types filtered by processor architecture
# and optionally by instance type. It takes the following arguments:
#
# -a, --architecture ARCHITECTURE  Specify the processor architecture (e.g., x86_64)
```

```
# -t, --type INSTANCE_TYPE         Comma-separated list of instance types (e.g.,
 t2.micro)
# -h, --help                       Show the usage help
#
# The function prints the instance type and supported architecture for each
# matching instance type.
#############################################################################
function ec2_describe_instance_types() {
  local architecture=""
  local instance_types=""

  # bashsupport disable=BP5008
  function usage() {
    echo "Usage: ec2_describe_instance_types [-a|--architecture ARCHITECTURE] [-t|--
type INSTANCE_TYPE] [-h|--help]"
    echo "  -a, --architecture ARCHITECTURE  Specify the processor architecture
 (e.g., x86_64)"
    echo "  -t, --type INSTANCE_TYPE         Comma-separated list of instance types
 (e.g., t2.micro)"
    echo "  -h, --help                       Show this help message"
  }

  while [[ $# -gt 0 ]]; do
    case "$1" in
      -a | --architecture)
        architecture="$2"
        shift 2
        ;;
      -t | --type)
        instance_types="$2"
        shift 2
        ;;
      -h | --help)
        usage
        return 0
        ;;
      *)
        echo "Unknown argument: $1"
        return 1
        ;;
    esac
  done

  if [[ -z "$architecture" ]]; then
```

```
  errecho "Error: Architecture not specified."
  usage
  return 1
fi

if [[ -z "$instance_types" ]]; then
  errecho "Error: Instance type not specified."
  usage
  return 1
fi

local tmp_json_file="temp_ec2.json"
echo -n '[
  {
    "Name": "processor-info.supported-architecture",
    "Values": [' >"$tmp_json_file"

local items
IFS=',' read -ra items <<<"$architecture"
local array_size
array_size=${#items[@]}
for i in $(seq 0 $((array_size - 1))); do
  echo -n '"'"${items[$i]}"'"' >>"$tmp_json_file"
  if [[ $i -lt $((array_size - 1)) ]]; then
    echo -n ',' >>"$tmp_json_file"
  fi
done
echo -n ']},
  {
  "Name": "instance-type",
    "Values": [' >>"$tmp_json_file"
IFS=',' read -ra items <<<"$instance_types"
local array_size
array_size=${#items[@]}
for i in $(seq 0 $((array_size - 1))); do
  echo -n '"'"${items[$i]}"'"' >>"$tmp_json_file"
  if [[ $i -lt $((array_size - 1)) ]]; then
    echo -n ',' >>"$tmp_json_file"
  fi
done

echo -n ']}]' >>"$tmp_json_file"

local response
```

```
  response=$(aws ec2 describe-instance-types --filters file://"$tmp_json_file" \
    --query 'InstanceTypes[*].[InstanceType]' --output text)

  local error_code=$?

  rm "$tmp_json_file"

  if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    echo "ERROR: AWS reports describe-instance-types operation failed."
    return 1
  fi

  echo "$response"
  return 0
}

###############################################################################
# function ec2_run_instances
#
# This function launches one or more Amazon Elastic Compute Cloud (Amazon EC2)
 instances.
#
# Parameters:
#       -i image_id - The ID of the Amazon Machine Image (AMI) to use.
#       -t instance_type - The instance type to use (e.g., t2.micro).
#       -k key_pair_name - The name of the key pair to use.
#       -s security_group_id - The ID of the security group to use.
#       -c count - The number of instances to launch (default: 1).
#       -h - Display help.
#
# Returns:
#       0 - If successful.
#       1 - If it fails.
###############################################################################
function ec2_run_instances() {
  local image_id instance_type key_pair_name security_group_id count response
  local option OPTARG # Required to use getopts command in a function.

  # bashsupport disable=BP5008
  function usage() {
    echo "function ec2_run_instances"
    echo "Launches one or more Amazon Elastic Compute Cloud (Amazon EC2) instances."
    echo "  -i image_id - The ID of the Amazon Machine Image (AMI) to use."
```

```
  echo "  -t instance_type - The instance type to use (e.g., t2.micro)."
  echo "  -k key_pair_name - The name of the key pair to use."
  echo "  -s security_group_id - The ID of the security group to use."
  echo "  -c count - The number of instances to launch (default: 1)."
  echo "  -h - Display help."
  echo ""
}

# Retrieve the calling parameters.
while getopts "i:t:k:s:c:h" option; do
  case "${option}" in
    i) image_id="${OPTARG}" ;;
    t) instance_type="${OPTARG}" ;;
    k) key_pair_name="${OPTARG}" ;;
    s) security_group_id="${OPTARG}" ;;
    c) count="${OPTARG}" ;;
    h)
      usage
      return 0
      ;;
    \?)
      echo "Invalid parameter"
      usage
      return 1
      ;;
  esac
done
export OPTIND=1

if [[ -z "$image_id" ]]; then
  errecho "ERROR: You must provide an Amazon Machine Image (AMI) ID with the -i
parameter."
  usage
  return 1
fi

if [[ -z "$instance_type" ]]; then
  errecho "ERROR: You must provide an instance type with the -t parameter."
  usage
  return 1
fi

if [[ -z "$key_pair_name" ]]; then
  errecho "ERROR: You must provide a key pair name with the -k parameter."
```

```
    usage
    return 1
  fi

  if [[ -z "$security_group_id" ]]; then
    errecho "ERROR: You must provide a security group ID with the -s parameter."
    usage
    return 1
  fi

  if [[ -z "$count" ]]; then
    count=1
  fi

  response=$(aws ec2 run-instances \
    --image-id "$image_id" \
    --instance-type "$instance_type" \
    --key-name "$key_pair_name" \
    --security-group-ids "$security_group_id" \
    --count "$count" \
    --query 'Instances[*].[InstanceId]' \
    --output text) || {
    aws_cli_error_log ${?}
    errecho "ERROR: AWS reports run-instances operation failed.$response"
    return 1
  }

  echo "$response"

  return 0
}

###############################################################################
# function ec2_describe_instances
#
# This function describes one or more Amazon Elastic Compute Cloud (Amazon EC2)
 instances.
#
# Parameters:
#       -i instance_id - The ID of the instance to describe (optional).
#       -q query - The query to filter the response (optional).
#       -h - Display help.
#
# Returns:
```

```
#       0 - If successful.
#       1 - If it fails.
###############################################################################
function ec2_describe_instances() {
  local instance_id query response
  local option OPTARG # Required to use getopts command in a function.

  # bashsupport disable=BP5008
  function usage() {
    echo "function ec2_describe_instances"
    echo "Describes one or more Amazon Elastic Compute Cloud (Amazon EC2)
 instances."
    echo "  -i instance_id - The ID of the instance to describe (optional)."
    echo "  -q query - The query to filter the response (optional)."
    echo "  -h - Display help."
    echo ""
  }

  # Retrieve the calling parameters.
  while getopts "i:q:h" option; do
    case "${option}" in
      i) instance_id="${OPTARG}" ;;
      q) query="${OPTARG}" ;;
      h)
        usage
        return 0
        ;;
      \?)
        echo "Invalid parameter"
        usage
        return 1
        ;;
    esac
  done
  export OPTIND=1

  local aws_cli_args=()

  if [[ -n "$instance_id" ]]; then
    # shellcheck disable=SC2206
    aws_cli_args+=("--instance-ids" $instance_id)
  fi

  local query_arg=""
```

```
  if [[ -n "$query" ]]; then
    query_arg="--query '$query'"
  else
    query_arg="--query Reservations[*].Instances[*].
[InstanceId,ImageId,InstanceType,KeyName,VpcId,PublicIpAddress,State.Name]"
  fi

  # shellcheck disable=SC2086
  response=$(aws ec2 describe-instances \
    "${aws_cli_args[@]}" \
    $query_arg \
    --output text) || {
    aws_cli_error_log ${?}
    errecho "ERROR: AWS reports describe-instances operation failed.$response"
    return 1
  }

  echo "$response"

  return 0
}

###############################################################################
# function ec2_stop_instances
#
# This function stops one or more Amazon Elastic Compute Cloud (Amazon EC2)
 instances.
#
# Parameters:
#       -i instance_id - The ID(s) of the instance(s) to stop (comma-separated).
#       -h - Display help.
#
# Returns:
#       0 - If successful.
#       1 - If it fails.
###############################################################################
function ec2_stop_instances() {
  local instance_ids
  local option OPTARG # Required to use getopts command in a function.

  # bashsupport disable=BP5008
  function usage() {
    echo "function ec2_stop_instances"
    echo "Stops one or more Amazon Elastic Compute Cloud (Amazon EC2) instances."
```

```
    echo "  -i instance_id - The ID(s) of the instance(s) to stop (comma-
separated)."
    echo "  -h - Display help."
    echo ""
  }

  # Retrieve the calling parameters.
  while getopts "i:h" option; do
    case "${option}" in
      i) instance_ids="${OPTARG}" ;;
      h)
        usage
        return 0
        ;;
      \?)
        echo "Invalid parameter"
        usage
        return 1
        ;;
    esac
  done
  export OPTIND=1

  if [[ -z "$instance_ids" ]]; then
    errecho "ERROR: You must provide one or more instance IDs with the -i
 parameter."
    usage
    return 1
  fi

  response=$(aws ec2 stop-instances \
    --instance-ids "${instance_ids}") || {
    aws_cli_error_log ${?}
    errecho "ERROR: AWS reports stop-instances operation failed with $response."
    return 1
  }

  return 0
}

###############################################################################
# function ec2_start_instances
#
```

```
# This function starts one or more Amazon Elastic Compute Cloud (Amazon EC2)
 instances.
#
# Parameters:
#       -i instance_id - The ID(s) of the instance(s) to start (comma-separated).
#       -h - Display help.
#
# Returns:
#       0 - If successful.
#       1 - If it fails.
###############################################################################
function ec2_start_instances() {
  local instance_ids
  local option OPTARG # Required to use getopts command in a function.

  # bashsupport disable=BP5008
  function usage() {
    echo "function ec2_start_instances"
    echo "Starts one or more Amazon Elastic Compute Cloud (Amazon EC2) instances."
    echo "  -i instance_id - The ID(s) of the instance(s) to start (comma-
separated)."
    echo "  -h - Display help."
    echo ""
  }

  # Retrieve the calling parameters.
  while getopts "i:h" option; do
    case "${option}" in
      i) instance_ids="${OPTARG}" ;;
      h)
        usage
        return 0
        ;;
      \?)
        echo "Invalid parameter"
        usage
        return 1
        ;;
    esac
  done
  export OPTIND=1

  if [[ -z "$instance_ids" ]]; then
```

```
    errecho "ERROR: You must provide one or more instance IDs with the -i
 parameter."
    usage
    return 1
  fi

  response=$(aws ec2 start-instances \
    --instance-ids "${instance_ids}") || {
    aws_cli_error_log ${?}
    errecho "ERROR: AWS reports start-instances operation failed with $response."
    return 1
  }

  return 0
}

###############################################################################
# function ec2_allocate_address
#
# This function allocates an Elastic IP address for use with Amazon Elastic Compute
 Cloud (Amazon EC2) instances in a specific AWS Region.
#
# Parameters:
#       -d domain - The domain for the Elastic IP address (either 'vpc' or
 'standard').
#
# Returns:
#       The allocated Elastic IP address, or an error message if the operation
 fails.
# And:
#       0 - If successful.
#       1 - If it fails.
#
###############################################################################
function ec2_allocate_address() {
  local domain response

  # Function to display usage information
  function usage() {
    echo "function ec2_allocate_address"
    echo "Allocates an Elastic IP address for use with Amazon Elastic Compute Cloud
 (Amazon EC2) instances in a specific AWS Region."
    echo "  -d domain - The domain for the Elastic IP address (either 'vpc' or
 'standard')."
```

```
  echo ""
}

# Parse the command-line arguments
while getopts "d:h" option; do
  case "${option}" in
    d) domain="${OPTARG}" ;;
    h)
      usage
      return 0
      ;;
    \?)
      echo "Invalid parameter"
      usage
      return 1
      ;;
  esac
done
export OPTIND=1

# Validate the input parameters
if [[ -z "$domain" ]]; then
  errecho "ERROR: You must provide a domain with the -d parameter (either 'vpc' or
'standard')."
  return 1
fi

if [[ "$domain" != "vpc" && "$domain" != "standard" ]]; then
  errecho "ERROR: Invalid domain value. Must be either 'vpc' or 'standard'."
  return 1
fi

# Allocate the Elastic IP address
response=$(aws ec2 allocate-address \
  --domain "$domain" \
  --query "[PublicIp,AllocationId]" \
  --output text) || {
  aws_cli_error_log ${?}
  errecho "ERROR: AWS reports allocate-address operation failed."
  errecho "$response"
  return 1
}

echo "$response"
```

```
  return 0
}

################################################################################
# function ec2_associate_address
#
# This function associates an Elastic IP address with an Amazon Elastic Compute
 Cloud (Amazon EC2) instance.
#
# Parameters:
#       -a allocation_id - The allocation ID of the Elastic IP address to associate.
#       -i instance_id - The ID of the EC2 instance to associate the Elastic IP
 address with.
#
# Returns:
#       0 - If successful.
#       1 - If it fails.
#
################################################################################
function ec2_associate_address() {
  local allocation_id instance_id response

  # Function to display usage information
  function usage() {
    echo "function ec2_associate_address"
    echo "Associates an Elastic IP address with an Amazon Elastic Compute Cloud
 (Amazon EC2) instance."
    echo "  -a allocation_id - The allocation ID of the Elastic IP address to
 associate."
    echo "  -i instance_id - The ID of the EC2 instance to associate the Elastic IP
 address with."
    echo ""
  }

  # Parse the command-line arguments
  while getopts "a:i:h" option; do
    case "${option}" in
      a) allocation_id="${OPTARG}" ;;
      i) instance_id="${OPTARG}" ;;
      h)
        usage
        return 0
        ;;
      \?)
```

```
        echo "Invalid parameter"
        usage
        return 1
        ;;
    esac
  done
  export OPTIND=1

  # Validate the input parameters
  if [[ -z "$allocation_id" ]]; then
    errecho "ERROR: You must provide an allocation ID with the -a parameter."
    return 1
  fi

  if [[ -z "$instance_id" ]]; then
    errecho "ERROR: You must provide an instance ID with the -i parameter."
    return 1
  fi

  # Associate the Elastic IP address
  response=$(aws ec2 associate-address \
    --allocation-id "$allocation_id" \
    --instance-id "$instance_id" \
    --query "AssociationId" \
    --output text) || {
    aws_cli_error_log ${?}
    errecho "ERROR: AWS reports associate-address operation failed."
    errecho "$response"
    return 1
  }

  echo "$response"
  return 0
}

###############################################################################
# function ec2_disassociate_address
#
# This function disassociates an Elastic IP address from an Amazon Elastic Compute
 Cloud (Amazon EC2) instance.
#
# Parameters:
#       -a association_id - The association ID that represents the association of
 the Elastic IP address with an instance.
```

```
#
# And:
#       0 - If successful.
#       1 - If it fails.
#
###############################################################################
function ec2_disassociate_address() {
  local association_id response

  # Function to display usage information
  function usage() {
    echo "function ec2_disassociate_address"
    echo "Disassociates an Elastic IP address from an Amazon Elastic Compute Cloud
 (Amazon EC2) instance."
    echo "  -a association_id - The association ID that represents the association
 of the Elastic IP address with an instance."
    echo ""
  }

  # Parse the command-line arguments
  while getopts "a:h" option; do
    case "${option}" in
      a) association_id="${OPTARG}" ;;
      h)
        usage
        return 0
        ;;
      \?)
        echo "Invalid parameter"
        usage
        return 1
        ;;
    esac
  done
  export OPTIND=1

  # Validate the input parameters
  if [[ -z "$association_id" ]]; then
    errecho "ERROR: You must provide an association ID with the -a parameter."
    return 1
  fi

  response=$(aws ec2 disassociate-address \
    --association-id "$association_id") || {
```

```
    aws_cli_error_log ${?}
    errecho "ERROR: AWS reports disassociate-address operation failed."
    errecho "$response"
    return 1
  }

  return 0
}

###############################################################################
# function ec2_release_address
#
# This function releases an Elastic IP address from an Amazon Elastic Compute Cloud
 (Amazon EC2) instance.
#
# Parameters:
#       -a allocation_id - The allocation ID of the Elastic IP address to release.
#
# Returns:
#       0 - If successful.
#       1 - If it fails.
#
###############################################################################
function ec2_release_address() {
  local allocation_id response

  # Function to display usage information
  function usage() {
    echo "function ec2_release_address"
    echo "Releases an Elastic IP address from an Amazon Elastic Compute Cloud
 (Amazon EC2) instance."
    echo "  -a allocation_id - The allocation ID of the Elastic IP address to
 release."
    echo ""
  }

  # Parse the command-line arguments
  while getopts "a:h" option; do
    case "${option}" in
      a) allocation_id="${OPTARG}" ;;
      h)
        usage
        return 0
        ;;
```

```
      \?)
        echo "Invalid parameter"
        usage
        return 1
        ;;
    esac
  done
  export OPTIND=1

  # Validate the input parameters
  if [[ -z "$allocation_id" ]]; then
    errecho "ERROR: You must provide an allocation ID with the -a parameter."
    return 1
  fi

  response=$(aws ec2 release-address \
    --allocation-id "$allocation_id") || {
    aws_cli_error_log ${?}
    errecho "ERROR: AWS reports release-address operation failed."
    errecho "$response"
    return 1
  }

  return 0
}

###############################################################################
# function ec2_terminate_instances
#
# This function terminates one or more Amazon Elastic Compute Cloud (Amazon EC2)
# instances using the AWS CLI.
#
# Parameters:
#       -i instance_ids - A space-separated list of instance IDs.
#       -h - Display help.
#
# Returns:
#       0 - If successful.
#       1 - If it fails.
###############################################################################
function ec2_terminate_instances() {
  local instance_ids response
  local option OPTARG # Required to use getopts command in a function.
```

```
 # bashsupport disable=BP5008
 function usage() {
   echo "function ec2_terminate_instances"
   echo "Terminates one or more Amazon Elastic Compute Cloud (Amazon EC2)
instances."
   echo "  -i instance_ids - A space-separated list of instance IDs."
   echo "  -h - Display help."
   echo ""
 }

 # Retrieve the calling parameters.
 while getopts "i:h" option; do
   case "${option}" in
     i) instance_ids="${OPTARG}" ;;
     h)
       usage
       return 0
       ;;
     \?)
       echo "Invalid parameter"
       usage
       return 1
       ;;
   esac
 done
 export OPTIND=1

 # Check if instance ID is provided
 if [[ -z "${instance_ids}" ]]; then
   echo "Error: Missing required instance IDs parameter."
   usage
   return 1
 fi

 # shellcheck disable=SC2086
 response=$(aws ec2 terminate-instances \
   "--instance-ids" $instance_ids \
   --query 'TerminatingInstances[*].[InstanceId,CurrentState.Name]' \
   --output text) || {
   aws_cli_error_log ${?}
   errecho "ERROR: AWS reports terminate-instances operation failed.$response"
   return 1
 }
```

```
  return 0
}

###############################################################################
# function ec2_delete_security_group
#
# This function deletes an Amazon Elastic Compute Cloud (Amazon EC2) security group.
#
# Parameters:
#       -i security_group_id - The ID of the security group to delete.
#
# And:
#       0 - If successful.
#       1 - If it fails.
###############################################################################
function ec2_delete_security_group() {
  local security_group_id response
  local option OPTARG # Required to use getopts command in a function.

  # bashsupport disable=BP5008
  function usage() {
    echo "function ec2_delete_security_group"
    echo "Deletes an Amazon Elastic Compute Cloud (Amazon EC2) security group."
    echo "  -i security_group_id - The ID of the security group to delete."
    echo ""
  }

  # Retrieve the calling parameters.
  while getopts "i:h" option; do
    case "${option}" in
      i) security_group_id="${OPTARG}" ;;
      h)
        usage
        return 0
        ;;
      \?)
        echo "Invalid parameter"
        usage
        return 1
        ;;
    esac
  done
  export OPTIND=1
```

```
  if [[ -z "$security_group_id" ]]; then
    errecho "ERROR: You must provide a security group ID with the -i parameter."
    usage
    return 1
  fi

  response=$(aws ec2 delete-security-group --group-id "$security_group_id" --output
 text) || {
    aws_cli_error_log ${?}
    errecho "ERROR: AWS reports delete-security-group operation failed.$response"
    return 1
  }

  return 0
}

###############################################################################
# function ec2_delete_keypair
#
# This function deletes an Amazon EC2 ED25519 or 2048-bit RSA key pair.
#
# Parameters:
#       -n key_pair_name - A key pair name.
#
# And:
#       0 - If successful.
#       1 - If it fails.
###############################################################################
function ec2_delete_keypair() {
  local key_pair_name response

  local option OPTARG # Required to use getopts command in a function.
  # bashsupport disable=BP5008
  function usage() {
    echo "function ec2_delete_keypair"
    echo "Deletes an Amazon EC2 ED25519 or 2048-bit RSA key pair."
    echo "  -n key_pair_name - A key pair name."
    echo ""
  }

  # Retrieve the calling parameters.
  while getopts "n:h" option; do
    case "${option}" in
      n) key_pair_name="${OPTARG}" ;;
```

```
      h)
        usage
        return 0
        ;;
      \?)
        echo "Invalid parameter"
        usage
        return 1
        ;;
    esac
  done
  export OPTIND=1

  if [[ -z "$key_pair_name" ]]; then
    errecho "ERROR: You must provide a key pair name with the -n parameter."
    usage
    return 1
  fi

  response=$(aws ec2 delete-key-pair \
    --key-name "$key_pair_name") || {
    aws_cli_error_log ${?}
    errecho "ERROR: AWS reports delete-key-pair operation failed.$response"
    return 1
  }

  return 0
}
```

Le funzioni di utility utilizzate in questo scenario.

```
###############################################################################
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
###############################################################################
function errecho() {
  printf "%s\n" "$*" 1>&2
}

###############################################################################
# function aws_cli_error_log()
```

```
#
# This function is used to log the error messages from the AWS CLI.
#
# The function expects the following argument:
#         $1 - The error code returned by the AWS CLI.
#
#  Returns:
#          0: - Success.
#
################################################################################
function aws_cli_error_log() {
  local err_code=$1
  errecho "Error code : $err_code"
  if [ "$err_code" == 1 ]; then
    errecho "  One or more S3 transfers failed."
  elif [ "$err_code" == 2 ]; then
    errecho "  Command line failed to parse."
  elif [ "$err_code" == 130 ]; then
    errecho "  Process received SIGINT."
  elif [ "$err_code" == 252 ]; then
    errecho "  Command syntax invalid."
  elif [ "$err_code" == 253 ]; then
    errecho "  The system environment or configuration was invalid."
  elif [ "$err_code" == 254 ]; then
    errecho "  The service returned an error."
  elif [ "$err_code" == 255 ]; then
    errecho "  255 is a catch-all error."
  fi

  return 0
}
```

- Per informazioni dettagliate sull'API, consulta i seguenti argomenti nella documentazione di riferimento dei comandi della AWS CLI .
  - AllocateAddress
  - AssociateAddress
  - AuthorizeSecurityGroupIngress
  - CreateKeyPair
  - CreateSecurityGroup
  - DeleteKeyPair

- DeleteSecurityGroup
- DescribeImages
- DescribeInstanceTypes
- DescribeInstances
- DescribeKeyPairs
- DescribeSecurityGroups
- DisassociateAddress
- ReleaseAddress
- RunInstances
- StartInstances
- StopInstances
- TerminateInstances
- UnmonitorInstances

Azioni

`AllocateAddress`

Il seguente esempio di codice mostra come usare`AllocateAddress`.

AWS CLI con lo script Bash

> **Note**
>
> C'è altro da fare. GitHub Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codice AWS.

```
###############################################################################
# function ec2_allocate_address
#
# This function allocates an Elastic IP address for use with Amazon Elastic Compute
 Cloud (Amazon EC2) instances in a specific AWS Region.
#
# Parameters:
```

```
#       -d domain - The domain for the Elastic IP address (either 'vpc' or
 'standard').
#
# Returns:
#       The allocated Elastic IP address, or an error message if the operation
 fails.
# And:
#       0 - If successful.
#       1 - If it fails.
#
###############################################################################
function ec2_allocate_address() {
  local domain response

  # Function to display usage information
  function usage() {
    echo "function ec2_allocate_address"
    echo "Allocates an Elastic IP address for use with Amazon Elastic Compute Cloud
 (Amazon EC2) instances in a specific AWS Region."
    echo "  -d domain - The domain for the Elastic IP address (either 'vpc' or
 'standard')."
    echo ""
  }

  # Parse the command-line arguments
  while getopts "d:h" option; do
    case "${option}" in
      d) domain="${OPTARG}" ;;
      h)
        usage
        return 0
        ;;
      \?)
        echo "Invalid parameter"
        usage
        return 1
        ;;
    esac
  done
  export OPTIND=1

  # Validate the input parameters
  if [[ -z "$domain" ]]; then
```

```
    errecho "ERROR: You must provide a domain with the -d parameter (either 'vpc' or
 'standard')."
    return 1
  fi

  if [[ "$domain" != "vpc" && "$domain" != "standard" ]]; then
    errecho "ERROR: Invalid domain value. Must be either 'vpc' or 'standard'."
    return 1
  fi

  # Allocate the Elastic IP address
  response=$(aws ec2 allocate-address \
    --domain "$domain" \
    --query "[PublicIp,AllocationId]" \
    --output text) || {
    aws_cli_error_log ${?}
    errecho "ERROR: AWS reports allocate-address operation failed."
    errecho "$response"
    return 1
  }

  echo "$response"
  return 0
}
```

Le funzioni di utilità utilizzate in questo esempio.

```
########################################################################################
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
########################################################################################
function errecho() {
  printf "%s\n" "$*" 1>&2
}

########################################################################################
# function aws_cli_error_log()
#
# This function is used to log the error messages from the AWS CLI.
#
# The function expects the following argument:
```

```
#         $1 - The error code returned by the AWS CLI.
#
#  Returns:
#          0: - Success.
#
################################################################################
function aws_cli_error_log() {
  local err_code=$1
  errecho "Error code : $err_code"
  if [ "$err_code" == 1 ]; then
    errecho "  One or more S3 transfers failed."
  elif [ "$err_code" == 2 ]; then
    errecho "  Command line failed to parse."
  elif [ "$err_code" == 130 ]; then
    errecho "  Process received SIGINT."
  elif [ "$err_code" == 252 ]; then
    errecho "  Command syntax invalid."
  elif [ "$err_code" == 253 ]; then
    errecho "  The system environment or configuration was invalid."
  elif [ "$err_code" == 254 ]; then
    errecho "  The service returned an error."
  elif [ "$err_code" == 255 ]; then
    errecho "  255 is a catch-all error."
  fi

  return 0
}
```

- Per i dettagli sull'API, consulta AllocateAddress AWS CLICommand Reference.

`AssociateAddress`

Il seguente esempio di codice mostra come utilizzare`AssociateAddress`.

AWS CLI con lo script Bash

> **Note**
>
> C'è altro da fare. GitHub Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codice AWS.

```
###############################################################################
# function ec2_associate_address
#
# This function associates an Elastic IP address with an Amazon Elastic Compute
 Cloud (Amazon EC2) instance.
#
# Parameters:
#       -a allocation_id - The allocation ID of the Elastic IP address to associate.
#       -i instance_id - The ID of the EC2 instance to associate the Elastic IP
 address with.
#
# Returns:
#       0 - If successful.
#       1 - If it fails.
#
###############################################################################
function ec2_associate_address() {
  local allocation_id instance_id response

  # Function to display usage information
  function usage() {
    echo "function ec2_associate_address"
    echo "Associates an Elastic IP address with an Amazon Elastic Compute Cloud
 (Amazon EC2) instance."
    echo "  -a allocation_id - The allocation ID of the Elastic IP address to
 associate."
    echo "  -i instance_id - The ID of the EC2 instance to associate the Elastic IP
 address with."
    echo ""
  }

  # Parse the command-line arguments
  while getopts "a:i:h" option; do
    case "${option}" in
      a) allocation_id="${OPTARG}" ;;
      i) instance_id="${OPTARG}" ;;
      h)
        usage
        return 0
        ;;
      \?)
        echo "Invalid parameter"
        usage
```

```
        return 1
        ;;
    esac
  done
  export OPTIND=1

  # Validate the input parameters
  if [[ -z "$allocation_id" ]]; then
    errecho "ERROR: You must provide an allocation ID with the -a parameter."
    return 1
  fi

  if [[ -z "$instance_id" ]]; then
    errecho "ERROR: You must provide an instance ID with the -i parameter."
    return 1
  fi

  # Associate the Elastic IP address
  response=$(aws ec2 associate-address \
    --allocation-id "$allocation_id" \
    --instance-id "$instance_id" \
    --query "AssociationId" \
    --output text) || {
    aws_cli_error_log ${?}
    errecho "ERROR: AWS reports associate-address operation failed."
    errecho "$response"
    return 1
  }

  echo "$response"
  return 0
}
```

Le funzioni di utilità utilizzate in questo esempio.

```
###############################################################################
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
###############################################################################
function errecho() {
  printf "%s\n" "$*" 1>&2
```

```
}

#############################################################################
# function aws_cli_error_log()
#
# This function is used to log the error messages from the AWS CLI.
#
# The function expects the following argument:
#         $1 - The error code returned by the AWS CLI.
#
#  Returns:
#          0: - Success.
#
#############################################################################
function aws_cli_error_log() {
  local err_code=$1
  errecho "Error code : $err_code"
  if [ "$err_code" == 1 ]; then
    errecho "  One or more S3 transfers failed."
  elif [ "$err_code" == 2 ]; then
    errecho "  Command line failed to parse."
  elif [ "$err_code" == 130 ]; then
    errecho "  Process received SIGINT."
  elif [ "$err_code" == 252 ]; then
    errecho "  Command syntax invalid."
  elif [ "$err_code" == 253 ]; then
    errecho "  The system environment or configuration was invalid."
  elif [ "$err_code" == 254 ]; then
    errecho "  The service returned an error."
  elif [ "$err_code" == 255 ]; then
    errecho "  255 is a catch-all error."
  fi

  return 0
}
```

- Per i dettagli sull'API, consulta AssociateAddress AWS CLICommand Reference.

`AuthorizeSecurityGroupIngress`

Il seguente esempio di codice mostra come utilizzare`AuthorizeSecurityGroupIngress`.

AWS CLI con lo script Bash

> **Note**
>
> C'è altro da fare. GitHub Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codice AWS.

```
#############################################################################
# function ec2_authorize_security_group_ingress
#
# This function authorizes an ingress rule for an Amazon Elastic Compute Cloud
 (Amazon EC2) security group.
#
# Parameters:
#       -g security_group_id - The ID of the security group.
#       -i ip_address - The IP address or CIDR block to authorize.
#       -p protocol - The protocol to authorize (e.g., tcp, udp, icmp).
#       -f from_port - The start of the port range to authorize.
#       -t to_port - The end of the port range to authorize.
#
# And:
#       0 - If successful.
#       1 - If it fails.
#############################################################################
function ec2_authorize_security_group_ingress() {
  local security_group_id ip_address protocol from_port to_port response
  local option OPTARG # Required to use getopts command in a function.

  # bashsupport disable=BP5008
  function usage() {
    echo "function ec2_authorize_security_group_ingress"
    echo "Authorizes an ingress rule for an Amazon Elastic Compute Cloud (Amazon
 EC2) security group."
    echo "  -g security_group_id - The ID of the security group."
    echo "  -i ip_address - The IP address or CIDR block to authorize."
    echo "  -p protocol - The protocol to authorize (e.g., tcp, udp, icmp)."
    echo "  -f from_port - The start of the port range to authorize."
    echo "  -t to_port - The end of the port range to authorize."
    echo ""
  }
```

```
  # Retrieve the calling parameters.
  while getopts "g:i:p:f:t:h" option; do
    case "${option}" in
      g) security_group_id="${OPTARG}" ;;
      i) ip_address="${OPTARG}" ;;
      p) protocol="${OPTARG}" ;;
      f) from_port="${OPTARG}" ;;
      t) to_port="${OPTARG}" ;;
      h)
        usage
        return 0
        ;;
      \?)
        echo "Invalid parameter"
        usage
        return 1
        ;;
    esac
  done
  export OPTIND=1

  if [[ -z "$security_group_id" ]]; then
    errecho "ERROR: You must provide a security group ID with the -g parameter."
    usage
    return 1
  fi

  if [[ -z "$ip_address" ]]; then
    errecho "ERROR: You must provide an IP address or CIDR block with the -i
parameter."
    usage
    return 1
  fi

  if [[ -z "$protocol" ]]; then
    errecho "ERROR: You must provide a protocol with the -p parameter."
    usage
    return 1
  fi

  if [[ -z "$from_port" ]]; then
    errecho "ERROR: You must provide a start port with the -f parameter."
    usage
    return 1
```

```
  fi

  if [[ -z "$to_port" ]]; then
    errecho "ERROR: You must provide an end port with the -t parameter."
    usage
    return 1
  fi

  response=$(aws ec2 authorize-security-group-ingress \
    --group-id "$security_group_id" \
    --cidr "${ip_address}/32" \
    --protocol "$protocol" \
    --port "$from_port-$to_port" \
    --output text) || {
    aws_cli_error_log ${?}
    errecho "ERROR: AWS reports authorize-security-group-ingress operation failed.
$response"
    return 1
  }

  return 0
}
```

Le funzioni di utilità utilizzate in questo esempio.

```
###############################################################################
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
###############################################################################
function errecho() {
  printf "%s\n" "$*" 1>&2
}

###############################################################################
# function aws_cli_error_log()
#
# This function is used to log the error messages from the AWS CLI.
#
# The function expects the following argument:
#         $1 - The error code returned by the AWS CLI.
#
```

```
#  Returns:
#          0: - Success.
#
###############################################################################
function aws_cli_error_log() {
  local err_code=$1
  errecho "Error code : $err_code"
  if [ "$err_code" == 1 ]; then
    errecho "  One or more S3 transfers failed."
  elif [ "$err_code" == 2 ]; then
    errecho "  Command line failed to parse."
  elif [ "$err_code" == 130 ]; then
    errecho "  Process received SIGINT."
  elif [ "$err_code" == 252 ]; then
    errecho "  Command syntax invalid."
  elif [ "$err_code" == 253 ]; then
    errecho "  The system environment or configuration was invalid."
  elif [ "$err_code" == 254 ]; then
    errecho "  The service returned an error."
  elif [ "$err_code" == 255 ]; then
    errecho "  255 is a catch-all error."
  fi

  return 0
}
```

- Per i dettagli sull'API, consulta AuthorizeSecurityGroupIngress AWS CLICommand Reference.

`CreateKeyPair`

Il seguente esempio di codice mostra come utilizzare`CreateKeyPair`.

AWS CLI con lo script Bash

> **Note**
>
> C'è altro da fare. GitHub Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codice AWS.

```
###############################################################################
```

```
# function ec2_create_keypair
#
# This function creates an Amazon Elastic Compute Cloud (Amazon EC2) ED25519 or
 2048-bit RSA key pair
# and writes it to a file.
#
# Parameters:
#       -n key_pair_name - A key pair name.
#       -f file_path - File to store the key pair.
#
# And:
#       0 - If successful.
#       1 - If it fails.
###############################################################################
function ec2_create_keypair() {
  local key_pair_name file_path response
  local option OPTARG # Required to use getopts command in a function.

  # bashsupport disable=BP5008
  function usage() {
    echo "function ec2_create_keypair"
    echo "Creates an Amazon Elastic Compute Cloud (Amazon EC2) ED25519 or 2048-bit
 RSA key pair"
    echo " and writes it to a file."
    echo "  -n key_pair_name - A key pair name."
    echo "  -f file_path - File to store the key pair."
    echo ""
  }

  # Retrieve the calling parameters.
  while getopts "n:f:h" option; do
    case "${option}" in
      n) key_pair_name="${OPTARG}" ;;
      f) file_path="${OPTARG}" ;;
      h)
        usage
        return 0
        ;;
      \?)
        echo "Invalid parameter"
        usage
        return 1
        ;;
    esac
```

```
  done
  export OPTIND=1

  if [[ -z "$key_pair_name" ]]; then
    errecho "ERROR: You must provide a key name with the -n parameter."
    usage
    return 1
  fi

  if [[ -z "$file_path" ]]; then
    errecho "ERROR: You must provide a file path with the -f parameter."
    usage
    return 1
  fi

  response=$(aws ec2 create-key-pair \
    --key-name "$key_pair_name" \
    --query 'KeyMaterial' \
    --output text) || {
    aws_cli_error_log ${?}
    errecho "ERROR: AWS reports create-access-key operation failed.$response"
    return 1
  }

  if [[ -n "$file_path" ]]; then
    echo "$response" >"$file_path"
  fi

  return 0
}
```

Le funzioni di utilità utilizzate in questo esempio.

```
###############################################################################
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
###############################################################################
function errecho() {
  printf "%s\n" "$*" 1>&2
}
```

```
###############################################################################
# function aws_cli_error_log()
#
# This function is used to log the error messages from the AWS CLI.
#
# The function expects the following argument:
#         $1 - The error code returned by the AWS CLI.
#
#  Returns:
#          0: - Success.
#
###############################################################################
function aws_cli_error_log() {
  local err_code=$1
  errecho "Error code : $err_code"
  if [ "$err_code" == 1 ]; then
    errecho "  One or more S3 transfers failed."
  elif [ "$err_code" == 2 ]; then
    errecho "  Command line failed to parse."
  elif [ "$err_code" == 130 ]; then
    errecho "  Process received SIGINT."
  elif [ "$err_code" == 252 ]; then
    errecho "  Command syntax invalid."
  elif [ "$err_code" == 253 ]; then
    errecho "  The system environment or configuration was invalid."
  elif [ "$err_code" == 254 ]; then
    errecho "  The service returned an error."
  elif [ "$err_code" == 255 ]; then
    errecho "  255 is a catch-all error."
  fi

  return 0
}
```

- Per i dettagli sull'API, consulta CreateKeyPair AWS CLICommand Reference.

`CreateSecurityGroup`

Il seguente esempio di codice mostra come utilizzare`CreateSecurityGroup`.

AWS CLI con lo script Bash

> **Note**
>
> C'è altro da fare. GitHub Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codice AWS.

```
#############################################################################
# function ec2_create_security_group
#
# This function creates an Amazon Elastic Compute Cloud (Amazon EC2) security group.
#
# Parameters:
#       -n security_group_name - The name of the security group.
#       -d security_group_description - The description of the security group.
#
# Returns:
#       The ID of the created security group, or an error message if the operation
 fails.
# And:
#       0 - If successful.
#       1 - If it fails.
#
#############################################################################
function ec2_create_security_group() {
  local security_group_name security_group_description response

  # Function to display usage information
  function usage() {
    echo "function ec2_create_security_group"
    echo "Creates an Amazon Elastic Compute Cloud (Amazon EC2) security group."
    echo "  -n security_group_name - The name of the security group."
    echo "  -d security_group_description - The description of the security group."
    echo ""
  }

  # Parse the command-line arguments
  while getopts "n:d:h" option; do
    case "${option}" in
      n) security_group_name="${OPTARG}" ;;
      d) security_group_description="${OPTARG}" ;;
```

```
      h)
        usage
        return 0
        ;;
      \?)
        echo "Invalid parameter"
        usage
        return 1
        ;;
    esac
  done
  export OPTIND=1

  # Validate the input parameters
  if [[ -z "$security_group_name" ]]; then
    errecho "ERROR: You must provide a security group name with the -n parameter."
    return 1
  fi

  if [[ -z "$security_group_description" ]]; then
    errecho "ERROR: You must provide a security group description with the -d
 parameter."
    return 1
  fi

  # Create the security group
  response=$(aws ec2 create-security-group \
    --group-name "$security_group_name" \
    --description "$security_group_description" \
    --query "GroupId" \
    --output text) || {
    aws_cli_error_log ${?}
    errecho "ERROR: AWS reports create-security-group operation failed."
    errecho "$response"
    return 1
  }

  echo "$response"
  return 0
}
```

Le funzioni di utilità utilizzate in questo esempio.

```
#############################################################################
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
#############################################################################
function errecho() {
  printf "%s\n" "$*" 1>&2
}

#############################################################################
# function aws_cli_error_log()
#
# This function is used to log the error messages from the AWS CLI.
#
# The function expects the following argument:
#         $1 - The error code returned by the AWS CLI.
#
#  Returns:
#          0: - Success.
#
#############################################################################
function aws_cli_error_log() {
  local err_code=$1
  errecho "Error code : $err_code"
  if [ "$err_code" == 1 ]; then
    errecho "  One or more S3 transfers failed."
  elif [ "$err_code" == 2 ]; then
    errecho "  Command line failed to parse."
  elif [ "$err_code" == 130 ]; then
    errecho "  Process received SIGINT."
  elif [ "$err_code" == 252 ]; then
    errecho "  Command syntax invalid."
  elif [ "$err_code" == 253 ]; then
    errecho "  The system environment or configuration was invalid."
  elif [ "$err_code" == 254 ]; then
    errecho "  The service returned an error."
  elif [ "$err_code" == 255 ]; then
    errecho "  255 is a catch-all error."
  fi

  return 0
}
```

- Per i dettagli sull'API, consulta CreateSecurityGroup AWS CLICommand Reference.

`DeleteKeyPair`

Il seguente esempio di codice mostra come utilizzare`DeleteKeyPair`.

AWS CLI con lo script Bash

> **Note**
>
> C'è altro da fare. GitHub Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codice AWS.

```
###############################################################################
# function ec2_delete_keypair
#
# This function deletes an Amazon EC2 ED25519 or 2048-bit RSA key pair.
#
# Parameters:
#       -n key_pair_name - A key pair name.
#
# And:
#       0 - If successful.
#       1 - If it fails.
###############################################################################
function ec2_delete_keypair() {
  local key_pair_name response

  local option OPTARG # Required to use getopts command in a function.
  # bashsupport disable=BP5008
  function usage() {
    echo "function ec2_delete_keypair"
    echo "Deletes an Amazon EC2 ED25519 or 2048-bit RSA key pair."
    echo "  -n key_pair_name - A key pair name."
    echo ""
  }

  # Retrieve the calling parameters.
  while getopts "n:h" option; do
    case "${option}" in
      n) key_pair_name="${OPTARG}" ;;
```

```
      h)
        usage
        return 0
        ;;
      \?)
        echo "Invalid parameter"
        usage
        return 1
        ;;
    esac
  done
  export OPTIND=1

  if [[ -z "$key_pair_name" ]]; then
    errecho "ERROR: You must provide a key pair name with the -n parameter."
    usage
    return 1
  fi

  response=$(aws ec2 delete-key-pair \
    --key-name "$key_pair_name") || {
    aws_cli_error_log ${?}
    errecho "ERROR: AWS reports delete-key-pair operation failed.$response"
    return 1
  }

  return 0
}
```

Le funzioni di utilità utilizzate in questo esempio.

```
###############################################################################
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
###############################################################################
function errecho() {
  printf "%s\n" "$*" 1>&2
}

###############################################################################
# function aws_cli_error_log()
```

```
#
# This function is used to log the error messages from the AWS CLI.
#
# The function expects the following argument:
#         $1 - The error code returned by the AWS CLI.
#
#  Returns:
#          0: - Success.
#
################################################################################
function aws_cli_error_log() {
  local err_code=$1
  errecho "Error code : $err_code"
  if [ "$err_code" == 1 ]; then
    errecho "  One or more S3 transfers failed."
  elif [ "$err_code" == 2 ]; then
    errecho "  Command line failed to parse."
  elif [ "$err_code" == 130 ]; then
    errecho "  Process received SIGINT."
  elif [ "$err_code" == 252 ]; then
    errecho "  Command syntax invalid."
  elif [ "$err_code" == 253 ]; then
    errecho "  The system environment or configuration was invalid."
  elif [ "$err_code" == 254 ]; then
    errecho "  The service returned an error."
  elif [ "$err_code" == 255 ]; then
    errecho "  255 is a catch-all error."
  fi

  return 0
}
```

- Per i dettagli sull'API, consulta DeleteKeyPair AWS CLICommand Reference.

`DeleteSecurityGroup`

Il seguente esempio di codice mostra come utilizzare`DeleteSecurityGroup`.

AWS CLI con lo script Bash

> **Note**
>
> C'è altro da fare. GitHub Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codice AWS.

```
###############################################################################
# function ec2_delete_security_group
#
# This function deletes an Amazon Elastic Compute Cloud (Amazon EC2) security group.
#
# Parameters:
#       -i security_group_id - The ID of the security group to delete.
#
# And:
#       0 - If successful.
#       1 - If it fails.
###############################################################################
function ec2_delete_security_group() {
  local security_group_id response
  local option OPTARG # Required to use getopts command in a function.

  # bashsupport disable=BP5008
  function usage() {
    echo "function ec2_delete_security_group"
    echo "Deletes an Amazon Elastic Compute Cloud (Amazon EC2) security group."
    echo "  -i security_group_id - The ID of the security group to delete."
    echo ""
  }

  # Retrieve the calling parameters.
  while getopts "i:h" option; do
    case "${option}" in
      i) security_group_id="${OPTARG}" ;;
      h)
        usage
        return 0
        ;;
      \?)
        echo "Invalid parameter"
```

```
        usage
        return 1
        ;;
    esac
  done
  export OPTIND=1

  if [[ -z "$security_group_id" ]]; then
    errecho "ERROR: You must provide a security group ID with the -i parameter."
    usage
    return 1
  fi

  response=$(aws ec2 delete-security-group --group-id "$security_group_id" --output
 text) || {
    aws_cli_error_log ${?}
    errecho "ERROR: AWS reports delete-security-group operation failed.$response"
    return 1
  }

  return 0
}
```

Le funzioni di utilità utilizzate in questo esempio.

```
###############################################################################
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
###############################################################################
function errecho() {
  printf "%s\n" "$*" 1>&2
}

###############################################################################
# function aws_cli_error_log()
#
# This function is used to log the error messages from the AWS CLI.
#
# The function expects the following argument:
#         $1 - The error code returned by the AWS CLI.
#
```

```
#  Returns:
#          0: - Success.
#
###############################################################################
function aws_cli_error_log() {
  local err_code=$1
  errecho "Error code : $err_code"
  if [ "$err_code" == 1 ]; then
    errecho "  One or more S3 transfers failed."
  elif [ "$err_code" == 2 ]; then
    errecho "  Command line failed to parse."
  elif [ "$err_code" == 130 ]; then
    errecho "  Process received SIGINT."
  elif [ "$err_code" == 252 ]; then
    errecho "  Command syntax invalid."
  elif [ "$err_code" == 253 ]; then
    errecho "  The system environment or configuration was invalid."
  elif [ "$err_code" == 254 ]; then
    errecho "  The service returned an error."
  elif [ "$err_code" == 255 ]; then
    errecho "  255 is a catch-all error."
  fi

  return 0
}
```

- Per i dettagli sull'API, consulta DeleteSecurityGroup AWS CLICommand Reference.

`DescribeImages`

Il seguente esempio di codice mostra come utilizzare`DescribeImages`.

AWS CLI con lo script Bash

> **Note**
>
> C'è altro da fare. GitHub Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codice AWS.

```
###############################################################################
```

```
# function ec2_describe_images
#
# This function describes one or more Amazon Elastic Compute Cloud (Amazon EC2)
 images.
#
# Parameters:
#       -i image_ids - A space-separated  list of image IDs (optional).
#       -h - Display help.
#
# And:
#       0 - If successful.
#       1 - If it fails.
###############################################################################
function ec2_describe_images() {
  local image_ids response
  local option OPTARG # Required to use getopts command in a function.

  # bashsupport disable=BP5008
  function usage() {
    echo "function ec2_describe_images"
    echo "Describes one or more Amazon Elastic Compute Cloud (Amazon EC2) images."
    echo "  -i image_ids - A space-separated list of image IDs (optional)."
    echo "  -h - Display help."
    echo ""
  }

  # Retrieve the calling parameters.
  while getopts "i:h" option; do
    case "${option}" in
      i) image_ids="${OPTARG}" ;;
      h)
        usage
        return 0
        ;;
      \?)
        echo "Invalid parameter"
        usage
        return 1
        ;;
    esac
  done
  export OPTIND=1

  local aws_cli_args=()
```

```
  if [[ -n "$image_ids" ]]; then
    # shellcheck disable=SC2206
    aws_cli_args+=("--image-ids" $image_ids)
  fi

  response=$(aws ec2 describe-images \
    "${aws_cli_args[@]}" \
    --query 'Images[*].[Description,Architecture,ImageId]' \
    --output text) || {
    aws_cli_error_log ${?}
    errecho "ERROR: AWS reports describe-images operation failed.$response"
    return 1
  }

  echo "$response"

  return 0
}
```

Le funzioni di utilità utilizzate in questo esempio.

```
################################################################################
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
################################################################################
function errecho() {
  printf "%s\n" "$*" 1>&2
}

################################################################################
# function aws_cli_error_log()
#
# This function is used to log the error messages from the AWS CLI.
#
# The function expects the following argument:
#         $1 - The error code returned by the AWS CLI.
#
#  Returns:
#          0: - Success.
#
```

```
###############################################################################
function aws_cli_error_log() {
  local err_code=$1
  errecho "Error code : $err_code"
  if [ "$err_code" == 1 ]; then
    errecho "  One or more S3 transfers failed."
  elif [ "$err_code" == 2 ]; then
    errecho "  Command line failed to parse."
  elif [ "$err_code" == 130 ]; then
    errecho "  Process received SIGINT."
  elif [ "$err_code" == 252 ]; then
    errecho "  Command syntax invalid."
  elif [ "$err_code" == 253 ]; then
    errecho "  The system environment or configuration was invalid."
  elif [ "$err_code" == 254 ]; then
    errecho "  The service returned an error."
  elif [ "$err_code" == 255 ]; then
    errecho "  255 is a catch-all error."
  fi

  return 0
}
```

- Per i dettagli sull'API, consulta DescribeImages AWS CLICommand Reference.

`DescribeInstanceTypes`

Il seguente esempio di codice mostra come utilizzare`DescribeInstanceTypes`.

AWS CLI con lo script Bash

> **Note**
>
> C'è altro da fare. GitHub Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codice AWS.

```
###############################################################################
# ec2_describe_instance_types
#
# This function describes EC2 instance types filtered by processor architecture
```

```
# and optionally by instance type. It takes the following arguments:
#
# -a, --architecture ARCHITECTURE  Specify the processor architecture (e.g., x86_64)
# -t, --type INSTANCE_TYPE         Comma-separated list of instance types (e.g.,
 t2.micro)
# -h, --help                       Show the usage help
#
# The function prints the instance type and supported architecture for each
# matching instance type.
##############################################################################
function ec2_describe_instance_types() {
  local architecture=""
  local instance_types=""

  # bashsupport disable=BP5008
  function usage() {
    echo "Usage: ec2_describe_instance_types [-a|--architecture ARCHITECTURE] [-t|--
type INSTANCE_TYPE] [-h|--help]"
    echo "  -a, --architecture ARCHITECTURE  Specify the processor architecture
 (e.g., x86_64)"
    echo "  -t, --type INSTANCE_TYPE         Comma-separated list of instance types
 (e.g., t2.micro)"
    echo "  -h, --help                       Show this help message"
  }

  while [[ $# -gt 0 ]]; do
    case "$1" in
      -a | --architecture)
        architecture="$2"
        shift 2
        ;;
      -t | --type)
        instance_types="$2"
        shift 2
        ;;
      -h | --help)
        usage
        return 0
        ;;
      *)
        echo "Unknown argument: $1"
        return 1
        ;;
    esac
```

```
done

if [[ -z "$architecture" ]]; then
  errecho "Error: Architecture not specified."
  usage
  return 1
fi

if [[ -z "$instance_types" ]]; then
  errecho "Error: Instance type not specified."
  usage
  return 1
fi

local tmp_json_file="temp_ec2.json"
echo -n '[
  {
    "Name": "processor-info.supported-architecture",
    "Values": [' >"$tmp_json_file"

local items
IFS=',' read -ra items <<<"$architecture"
local array_size
array_size=${#items[@]}
for i in $(seq 0 $((array_size - 1))); do
  echo -n '"'"${items[$i]}"'"' >>"$tmp_json_file"
  if [[ $i -lt $((array_size - 1)) ]]; then
    echo -n ',' >>"$tmp_json_file"
  fi
done
echo -n ']},
  {
  "Name": "instance-type",
    "Values": [' >>"$tmp_json_file"
IFS=',' read -ra items <<<"$instance_types"
local array_size
array_size=${#items[@]}
for i in $(seq 0 $((array_size - 1))); do
  echo -n '"'"${items[$i]}"'"' >>"$tmp_json_file"
  if [[ $i -lt $((array_size - 1)) ]]; then
    echo -n ',' >>"$tmp_json_file"
  fi
done
```

```
  echo -n ']}]' >>"$tmp_json_file"

  local response
  response=$(aws ec2 describe-instance-types --filters file://"$tmp_json_file" \
    --query 'InstanceTypes[*].[InstanceType]' --output text)

  local error_code=$?

  rm "$tmp_json_file"

  if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    echo "ERROR: AWS reports describe-instance-types operation failed."
    return 1
  fi

  echo "$response"
  return 0
}
```

Le funzioni di utilità utilizzate in questo esempio.

```
###############################################################################
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
###############################################################################
function errecho() {
  printf "%s\n" "$*" 1>&2
}

###############################################################################
# function aws_cli_error_log()
#
# This function is used to log the error messages from the AWS CLI.
#
# The function expects the following argument:
#         $1 - The error code returned by the AWS CLI.
#
#  Returns:
#          0: - Success.
#
```

```
###############################################################################
function aws_cli_error_log() {
  local err_code=$1
  errecho "Error code : $err_code"
  if [ "$err_code" == 1 ]; then
    errecho "  One or more S3 transfers failed."
  elif [ "$err_code" == 2 ]; then
    errecho "  Command line failed to parse."
  elif [ "$err_code" == 130 ]; then
    errecho "  Process received SIGINT."
  elif [ "$err_code" == 252 ]; then
    errecho "  Command syntax invalid."
  elif [ "$err_code" == 253 ]; then
    errecho "  The system environment or configuration was invalid."
  elif [ "$err_code" == 254 ]; then
    errecho "  The service returned an error."
  elif [ "$err_code" == 255 ]; then
    errecho "  255 is a catch-all error."
  fi

  return 0
}
```

- Per i dettagli sull'API, consulta DescribeInstanceTypes AWS CLICommand Reference.

`DescribeInstances`

Il seguente esempio di codice mostra come utilizzare`DescribeInstances`.

AWS CLI con lo script Bash

> **Note**
>
> C'è altro da fare. GitHub Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codice AWS.

```
###############################################################################
# function ec2_describe_instances
#
```

```
# This function describes one or more Amazon Elastic Compute Cloud (Amazon EC2)
 instances.
#
# Parameters:
#       -i instance_id - The ID of the instance to describe (optional).
#       -q query - The query to filter the response (optional).
#       -h - Display help.
#
# Returns:
#       0 - If successful.
#       1 - If it fails.
###############################################################################
function ec2_describe_instances() {
  local instance_id query response
  local option OPTARG # Required to use getopts command in a function.

  # bashsupport disable=BP5008
  function usage() {
    echo "function ec2_describe_instances"
    echo "Describes one or more Amazon Elastic Compute Cloud (Amazon EC2)
 instances."
    echo "  -i instance_id - The ID of the instance to describe (optional)."
    echo "  -q query - The query to filter the response (optional)."
    echo "  -h - Display help."
    echo ""
  }

  # Retrieve the calling parameters.
  while getopts "i:q:h" option; do
    case "${option}" in
      i) instance_id="${OPTARG}" ;;
      q) query="${OPTARG}" ;;
      h)
        usage
        return 0
        ;;
      \?)
        echo "Invalid parameter"
        usage
        return 1
        ;;
    esac
  done
  export OPTIND=1
```

```
  local aws_cli_args=()

  if [[ -n "$instance_id" ]]; then
    # shellcheck disable=SC2206
    aws_cli_args+=("--instance-ids" $instance_id)
  fi

  local query_arg=""
  if [[ -n "$query" ]]; then
    query_arg="--query '$query'"
  else
    query_arg="--query Reservations[*].Instances[*].
[InstanceId,ImageId,InstanceType,KeyName,VpcId,PublicIpAddress,State.Name]"
  fi

  # shellcheck disable=SC2086
  response=$(aws ec2 describe-instances \
    "${aws_cli_args[@]}" \
    $query_arg \
    --output text) || {
    aws_cli_error_log ${?}
    errecho "ERROR: AWS reports describe-instances operation failed.$response"
    return 1
  }

  echo "$response"

  return 0
}
```

Le funzioni di utilità utilizzate in questo esempio.

```
###############################################################################
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
###############################################################################
function errecho() {
  printf "%s\n" "$*" 1>&2
}
```

```
##############################################################################
# function aws_cli_error_log()
#
# This function is used to log the error messages from the AWS CLI.
#
# The function expects the following argument:
#         $1 - The error code returned by the AWS CLI.
#
#  Returns:
#          0: - Success.
#
##############################################################################
function aws_cli_error_log() {
  local err_code=$1
  errecho "Error code : $err_code"
  if [ "$err_code" == 1 ]; then
    errecho "  One or more S3 transfers failed."
  elif [ "$err_code" == 2 ]; then
    errecho "  Command line failed to parse."
  elif [ "$err_code" == 130 ]; then
    errecho "  Process received SIGINT."
  elif [ "$err_code" == 252 ]; then
    errecho "  Command syntax invalid."
  elif [ "$err_code" == 253 ]; then
    errecho "  The system environment or configuration was invalid."
  elif [ "$err_code" == 254 ]; then
    errecho "  The service returned an error."
  elif [ "$err_code" == 255 ]; then
    errecho "  255 is a catch-all error."
  fi

  return 0
}
```

- Per i dettagli sull'API, consulta DescribeInstances AWS CLICommand Reference.

`DescribeKeyPairs`

Il seguente esempio di codice mostra come utilizzare`DescribeKeyPairs`.

AWS CLI con lo script Bash

> **Note**
>
> C'è altro da fare. GitHub Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codice AWS.

```
###############################################################################
# function ec2_describe_key_pairs
#
# This function describes one or more Amazon Elastic Compute Cloud (Amazon EC2) key
 pairs.
#
# Parameters:
#       -h - Display help.
#
# And:
#       0 - If successful.
#       1 - If it fails.
###############################################################################
function ec2_describe_key_pairs() {
  local option OPTARG # Required to use getopts command in a function.

  # bashsupport disable=BP5008
  function usage() {
    echo "function ec2_describe_key_pairs"
    echo "Describes one or more Amazon Elastic Compute Cloud (Amazon EC2) key
 pairs."
    echo "  -h - Display help."
    echo ""
  }

  # Retrieve the calling parameters.
  while getopts "h" option; do
    case "${option}" in
      h)
        usage
        return 0
        ;;
      \?)
        echo "Invalid parameter"
```

```
        usage
        return 1
        ;;
    esac
  done
  export OPTIND=1

  local response

  response=$(aws ec2 describe-key-pairs \
    --query 'KeyPairs[*].[KeyName, KeyFingerprint]' \
    --output text) || {
    aws_cli_error_log ${?}
    errecho "ERROR: AWS reports describe-key-pairs operation failed.$response"
    return 1
  }

  echo "$response"

  return 0
}
```

Le funzioni di utilità utilizzate in questo esempio.

```
###############################################################################
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
###############################################################################
function errecho() {
  printf "%s\n" "$*" 1>&2
}

###############################################################################
# function aws_cli_error_log()
#
# This function is used to log the error messages from the AWS CLI.
#
# The function expects the following argument:
#         $1 - The error code returned by the AWS CLI.
#
#  Returns:
```

```
#          0: - Success.
#
###############################################################################
function aws_cli_error_log() {
  local err_code=$1
  errecho "Error code : $err_code"
  if [ "$err_code" == 1 ]; then
    errecho "  One or more S3 transfers failed."
  elif [ "$err_code" == 2 ]; then
    errecho "  Command line failed to parse."
  elif [ "$err_code" == 130 ]; then
    errecho "  Process received SIGINT."
  elif [ "$err_code" == 252 ]; then
    errecho "  Command syntax invalid."
  elif [ "$err_code" == 253 ]; then
    errecho "  The system environment or configuration was invalid."
  elif [ "$err_code" == 254 ]; then
    errecho "  The service returned an error."
  elif [ "$err_code" == 255 ]; then
    errecho "  255 is a catch-all error."
  fi

  return 0
}
```

- Per i dettagli sull'API, consulta DescribeKeyPairs AWS CLICommand Reference.

`DescribeSecurityGroups`

Il seguente esempio di codice mostra come utilizzare`DescribeSecurityGroups`.

AWS CLI con lo script Bash

> **Note**
>
> C'è altro da fare. GitHub Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codice AWS.

```
###############################################################################
# function ec2_describe_security_groups
```

```
#
# This function describes one or more Amazon Elastic Compute Cloud (Amazon EC2)
 security groups.
#
# Parameters:
#       -g security_group_id - The ID of the security group to describe (optional).
#
# And:
#       0 - If successful.
#       1 - If it fails.
###############################################################################
function ec2_describe_security_groups() {
  local security_group_id response
  local option OPTARG # Required to use getopts command in a function.

  # bashsupport disable=BP5008
  function usage() {
    echo "function ec2_describe_security_groups"
    echo "Describes one or more Amazon Elastic Compute Cloud (Amazon EC2) security
 groups."
    echo "  -g security_group_id - The ID of the security group to describe
 (optional)."
    echo ""
  }

  # Retrieve the calling parameters.
  while getopts "g:h" option; do
    case "${option}" in
      g) security_group_id="${OPTARG}" ;;
      h)
        usage
        return 0
        ;;
      \?)
        echo "Invalid parameter"
        usage
        return 1
        ;;
    esac
  done
  export OPTIND=1

  local query="SecurityGroups[*].[GroupName, GroupId, VpcId, IpPermissions[*].
[IpProtocol, FromPort, ToPort, IpRanges[*].CidrIp]]"
```

```
  if [[ -n "$security_group_id" ]]; then
    response=$(aws ec2 describe-security-groups --group-ids "$security_group_id" --
query "${query}" --output text)
  else
    response=$(aws ec2 describe-security-groups --query "${query}" --output text)
  fi

  local error_code=${?}

  if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports describe-security-groups operation failed.$response"
    return 1
  fi

  echo "$response"

  return 0
}
```

Le funzioni di utilità utilizzate in questo esempio.

```
###############################################################################
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
###############################################################################
function errecho() {
  printf "%s\n" "$*" 1>&2
}

###############################################################################
# function aws_cli_error_log()
#
# This function is used to log the error messages from the AWS CLI.
#
# The function expects the following argument:
#         $1 - The error code returned by the AWS CLI.
#
#  Returns:
#          0: - Success.
```

```
#
################################################################################
function aws_cli_error_log() {
  local err_code=$1
  errecho "Error code : $err_code"
  if [ "$err_code" == 1 ]; then
    errecho "  One or more S3 transfers failed."
  elif [ "$err_code" == 2 ]; then
    errecho "  Command line failed to parse."
  elif [ "$err_code" == 130 ]; then
    errecho "  Process received SIGINT."
  elif [ "$err_code" == 252 ]; then
    errecho "  Command syntax invalid."
  elif [ "$err_code" == 253 ]; then
    errecho "  The system environment or configuration was invalid."
  elif [ "$err_code" == 254 ]; then
    errecho "  The service returned an error."
  elif [ "$err_code" == 255 ]; then
    errecho "  255 is a catch-all error."
  fi

  return 0
}
```

- Per i dettagli sull'API, consulta DescribeSecurityGroups AWS CLICommand Reference.

`DisassociateAddress`

Il seguente esempio di codice mostra come utilizzare`DisassociateAddress`.

AWS CLI con lo script Bash

> **Note**
>
> C'è altro da fare. GitHub Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codice AWS.

```
################################################################################
# function ec2_disassociate_address
#
```

```
# This function disassociates an Elastic IP address from an Amazon Elastic Compute
 Cloud (Amazon EC2) instance.
#
# Parameters:
#       -a association_id - The association ID that represents the association of
 the Elastic IP address with an instance.
#
# And:
#       0 - If successful.
#       1 - If it fails.
#
###############################################################################
function ec2_disassociate_address() {
  local association_id response

  # Function to display usage information
  function usage() {
    echo "function ec2_disassociate_address"
    echo "Disassociates an Elastic IP address from an Amazon Elastic Compute Cloud
 (Amazon EC2) instance."
    echo "  -a association_id - The association ID that represents the association
 of the Elastic IP address with an instance."
    echo ""
  }

  # Parse the command-line arguments
  while getopts "a:h" option; do
    case "${option}" in
      a) association_id="${OPTARG}" ;;
      h)
        usage
        return 0
        ;;
      \?)
        echo "Invalid parameter"
        usage
        return 1
        ;;
    esac
  done
  export OPTIND=1

  # Validate the input parameters
  if [[ -z "$association_id" ]]; then
```

```
    errecho "ERROR: You must provide an association ID with the -a parameter."
    return 1
  fi

  response=$(aws ec2 disassociate-address \
    --association-id "$association_id") || {
    aws_cli_error_log ${?}
    errecho "ERROR: AWS reports disassociate-address operation failed."
    errecho "$response"
    return 1
  }

  return 0
}
```

Le funzioni di utilità utilizzate in questo esempio.

```
###############################################################################
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
###############################################################################
function errecho() {
  printf "%s\n" "$*" 1>&2
}

###############################################################################
# function aws_cli_error_log()
#
# This function is used to log the error messages from the AWS CLI.
#
# The function expects the following argument:
#         $1 - The error code returned by the AWS CLI.
#
#  Returns:
#          0: - Success.
#
###############################################################################
function aws_cli_error_log() {
  local err_code=$1
  errecho "Error code : $err_code"
  if [ "$err_code" == 1 ]; then
```

```
    errecho "  One or more S3 transfers failed."
  elif [ "$err_code" == 2 ]; then
    errecho "  Command line failed to parse."
  elif [ "$err_code" == 130 ]; then
    errecho "  Process received SIGINT."
  elif [ "$err_code" == 252 ]; then
    errecho "  Command syntax invalid."
  elif [ "$err_code" == 253 ]; then
    errecho "  The system environment or configuration was invalid."
  elif [ "$err_code" == 254 ]; then
    errecho "  The service returned an error."
  elif [ "$err_code" == 255 ]; then
    errecho "  255 is a catch-all error."
  fi

  return 0
}
```

- Per i dettagli sull'API, consulta DisassociateAddress AWS CLICommand Reference.

`ReleaseAddress`

Il seguente esempio di codice mostra come utilizzare`ReleaseAddress`.

AWS CLI con lo script Bash

> **Note**
>
> C'è altro da fare. GitHub Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codice AWS.

```
################################################################################
# function ec2_release_address
#
# This function releases an Elastic IP address from an Amazon Elastic Compute Cloud
 (Amazon EC2) instance.
#
# Parameters:
#       -a allocation_id - The allocation ID of the Elastic IP address to release.
#
```

```
# Returns:
#       0 - If successful.
#       1 - If it fails.
#
###############################################################################
function ec2_release_address() {
  local allocation_id response

  # Function to display usage information
  function usage() {
    echo "function ec2_release_address"
    echo "Releases an Elastic IP address from an Amazon Elastic Compute Cloud
 (Amazon EC2) instance."
    echo "  -a allocation_id - The allocation ID of the Elastic IP address to
 release."
    echo ""
  }

  # Parse the command-line arguments
  while getopts "a:h" option; do
    case "${option}" in
      a) allocation_id="${OPTARG}" ;;
      h)
        usage
        return 0
        ;;
      \?)
        echo "Invalid parameter"
        usage
        return 1
        ;;
    esac
  done
  export OPTIND=1

  # Validate the input parameters
  if [[ -z "$allocation_id" ]]; then
    errecho "ERROR: You must provide an allocation ID with the -a parameter."
    return 1
  fi

  response=$(aws ec2 release-address \
    --allocation-id "$allocation_id") || {
    aws_cli_error_log ${?}
```

```
    errecho "ERROR: AWS reports release-address operation failed."
    errecho "$response"
    return 1
  }

  return 0
}
```

Le funzioni di utilità utilizzate in questo esempio.

```
################################################################################
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
################################################################################
function errecho() {
  printf "%s\n" "$*" 1>&2
}

################################################################################
# function aws_cli_error_log()
#
# This function is used to log the error messages from the AWS CLI.
#
# The function expects the following argument:
#         $1 - The error code returned by the AWS CLI.
#
#  Returns:
#          0: - Success.
#
################################################################################
function aws_cli_error_log() {
  local err_code=$1
  errecho "Error code : $err_code"
  if [ "$err_code" == 1 ]; then
    errecho "  One or more S3 transfers failed."
  elif [ "$err_code" == 2 ]; then
    errecho "  Command line failed to parse."
  elif [ "$err_code" == 130 ]; then
    errecho "  Process received SIGINT."
  elif [ "$err_code" == 252 ]; then
    errecho "  Command syntax invalid."
```

```
  elif [ "$err_code" == 253 ]; then
    errecho "  The system environment or configuration was invalid."
  elif [ "$err_code" == 254 ]; then
    errecho "  The service returned an error."
  elif [ "$err_code" == 255 ]; then
    errecho "  255 is a catch-all error."
  fi

  return 0
}
```

- Per i dettagli sull'API, consulta ReleaseAddress AWS CLICommand Reference.

`RunInstances`

Il seguente esempio di codice mostra come utilizzare`RunInstances`.

AWS CLI con lo script Bash

> **Note**
>
> C'è altro da fare. GitHub Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codice AWS.

```
################################################################################
# function ec2_run_instances
#
# This function launches one or more Amazon Elastic Compute Cloud (Amazon EC2)
 instances.
#
# Parameters:
#       -i image_id - The ID of the Amazon Machine Image (AMI) to use.
#       -t instance_type - The instance type to use (e.g., t2.micro).
#       -k key_pair_name - The name of the key pair to use.
#       -s security_group_id - The ID of the security group to use.
#       -c count - The number of instances to launch (default: 1).
#       -h - Display help.
#
# Returns:
#       0 - If successful.
```

```
#       1 - If it fails.
###############################################################################
function ec2_run_instances() {
  local image_id instance_type key_pair_name security_group_id count response
  local option OPTARG # Required to use getopts command in a function.

  # bashsupport disable=BP5008
  function usage() {
    echo "function ec2_run_instances"
    echo "Launches one or more Amazon Elastic Compute Cloud (Amazon EC2) instances."
    echo "  -i image_id - The ID of the Amazon Machine Image (AMI) to use."
    echo "  -t instance_type - The instance type to use (e.g., t2.micro)."
    echo "  -k key_pair_name - The name of the key pair to use."
    echo "  -s security_group_id - The ID of the security group to use."
    echo "  -c count - The number of instances to launch (default: 1)."
    echo "  -h - Display help."
    echo ""
  }

  # Retrieve the calling parameters.
  while getopts "i:t:k:s:c:h" option; do
    case "${option}" in
      i) image_id="${OPTARG}" ;;
      t) instance_type="${OPTARG}" ;;
      k) key_pair_name="${OPTARG}" ;;
      s) security_group_id="${OPTARG}" ;;
      c) count="${OPTARG}" ;;
      h)
        usage
        return 0
        ;;
      \?)
        echo "Invalid parameter"
        usage
        return 1
        ;;
    esac
  done
  export OPTIND=1

  if [[ -z "$image_id" ]]; then
    errecho "ERROR: You must provide an Amazon Machine Image (AMI) ID with the -i
 parameter."
    usage
```

```
    return 1
  fi

  if [[ -z "$instance_type" ]]; then
    errecho "ERROR: You must provide an instance type with the -t parameter."
    usage
    return 1
  fi

  if [[ -z "$key_pair_name" ]]; then
    errecho "ERROR: You must provide a key pair name with the -k parameter."
    usage
    return 1
  fi

  if [[ -z "$security_group_id" ]]; then
    errecho "ERROR: You must provide a security group ID with the -s parameter."
    usage
    return 1
  fi

  if [[ -z "$count" ]]; then
    count=1
  fi

  response=$(aws ec2 run-instances \
    --image-id "$image_id" \
    --instance-type "$instance_type" \
    --key-name "$key_pair_name" \
    --security-group-ids "$security_group_id" \
    --count "$count" \
    --query 'Instances[*].[InstanceId]' \
    --output text) || {
    aws_cli_error_log ${?}
    errecho "ERROR: AWS reports run-instances operation failed.$response"
    return 1
  }

  echo "$response"

  return 0
}
```

Le funzioni di utilità utilizzate in questo esempio.

```
#############################################################################
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
#############################################################################
function errecho() {
  printf "%s\n" "$*" 1>&2
}

#############################################################################
# function aws_cli_error_log()
#
# This function is used to log the error messages from the AWS CLI.
#
# The function expects the following argument:
#         $1 - The error code returned by the AWS CLI.
#
#  Returns:
#          0: - Success.
#
#############################################################################
function aws_cli_error_log() {
  local err_code=$1
  errecho "Error code : $err_code"
  if [ "$err_code" == 1 ]; then
    errecho "  One or more S3 transfers failed."
  elif [ "$err_code" == 2 ]; then
    errecho "  Command line failed to parse."
  elif [ "$err_code" == 130 ]; then
    errecho "  Process received SIGINT."
  elif [ "$err_code" == 252 ]; then
    errecho "  Command syntax invalid."
  elif [ "$err_code" == 253 ]; then
    errecho "  The system environment or configuration was invalid."
  elif [ "$err_code" == 254 ]; then
    errecho "  The service returned an error."
  elif [ "$err_code" == 255 ]; then
    errecho "  255 is a catch-all error."
  fi

  return 0
}
```

- Per i dettagli sull'API, consulta RunInstances AWS CLICommand Reference.

`StartInstances`

Il seguente esempio di codice mostra come utilizzare`StartInstances`.

AWS CLI con lo script Bash

> **Note**
>
> C'è altro da fare. GitHub Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codice AWS.

```
###############################################################################
# function ec2_start_instances
#
# This function starts one or more Amazon Elastic Compute Cloud (Amazon EC2)
 instances.
#
# Parameters:
#       -i instance_id - The ID(s) of the instance(s) to start (comma-separated).
#       -h - Display help.
#
# Returns:
#       0 - If successful.
#       1 - If it fails.
###############################################################################
function ec2_start_instances() {
  local instance_ids
  local option OPTARG # Required to use getopts command in a function.

  # bashsupport disable=BP5008
  function usage() {
    echo "function ec2_start_instances"
    echo "Starts one or more Amazon Elastic Compute Cloud (Amazon EC2) instances."
    echo "  -i instance_id - The ID(s) of the instance(s) to start (comma-
separated)."
    echo "  -h - Display help."
    echo ""
```

```
  }

  # Retrieve the calling parameters.
  while getopts "i:h" option; do
    case "${option}" in
      i) instance_ids="${OPTARG}" ;;
      h)
        usage
        return 0
        ;;
      \?)
        echo "Invalid parameter"
        usage
        return 1
        ;;
    esac
  done
  export OPTIND=1

  if [[ -z "$instance_ids" ]]; then
    errecho "ERROR: You must provide one or more instance IDs with the -i
 parameter."
    usage
    return 1
  fi

  response=$(aws ec2 start-instances \
    --instance-ids "${instance_ids}") || {
    aws_cli_error_log ${?}
    errecho "ERROR: AWS reports start-instances operation failed with $response."
    return 1
  }

  return 0
}
```

Le funzioni di utilità utilizzate in questo esempio.

```
###############################################################################
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
```

```
###############################################################################
function errecho() {
  printf "%s\n" "$*" 1>&2
}

###############################################################################
# function aws_cli_error_log()
#
# This function is used to log the error messages from the AWS CLI.
#
# The function expects the following argument:
#         $1 - The error code returned by the AWS CLI.
#
#  Returns:
#          0: - Success.
#
###############################################################################
function aws_cli_error_log() {
  local err_code=$1
  errecho "Error code : $err_code"
  if [ "$err_code" == 1 ]; then
    errecho "  One or more S3 transfers failed."
  elif [ "$err_code" == 2 ]; then
    errecho "  Command line failed to parse."
  elif [ "$err_code" == 130 ]; then
    errecho "  Process received SIGINT."
  elif [ "$err_code" == 252 ]; then
    errecho "  Command syntax invalid."
  elif [ "$err_code" == 253 ]; then
    errecho "  The system environment or configuration was invalid."
  elif [ "$err_code" == 254 ]; then
    errecho "  The service returned an error."
  elif [ "$err_code" == 255 ]; then
    errecho "  255 is a catch-all error."
  fi

  return 0
}
```

- Per i dettagli sull'API, consulta StartInstances AWS CLICommand Reference.

`StopInstances`

Il seguente esempio di codice mostra come utilizzare`StopInstances`.

AWS CLI con lo script Bash

> **Note**
>
> C'è altro da fare. GitHub Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codice AWS.

```
###############################################################################
# function ec2_stop_instances
#
# This function stops one or more Amazon Elastic Compute Cloud (Amazon EC2)
 instances.
#
# Parameters:
#       -i instance_id - The ID(s) of the instance(s) to stop (comma-separated).
#       -h - Display help.
#
# Returns:
#       0 - If successful.
#       1 - If it fails.
###############################################################################
function ec2_stop_instances() {
  local instance_ids
  local option OPTARG # Required to use getopts command in a function.

  # bashsupport disable=BP5008
  function usage() {
    echo "function ec2_stop_instances"
    echo "Stops one or more Amazon Elastic Compute Cloud (Amazon EC2) instances."
    echo "  -i instance_id - The ID(s) of the instance(s) to stop (comma-
separated)."
    echo "  -h - Display help."
    echo ""
  }

  # Retrieve the calling parameters.
  while getopts "i:h" option; do
    case "${option}" in
```

```
      i) instance_ids="${OPTARG}" ;;
      h)
        usage
        return 0
        ;;
      \?)
        echo "Invalid parameter"
        usage
        return 1
        ;;
    esac
  done
  export OPTIND=1

  if [[ -z "$instance_ids" ]]; then
    errecho "ERROR: You must provide one or more instance IDs with the -i
 parameter."
    usage
    return 1
  fi

  response=$(aws ec2 stop-instances \
    --instance-ids "${instance_ids}") || {
    aws_cli_error_log ${?}
    errecho "ERROR: AWS reports stop-instances operation failed with $response."
    return 1
  }

  return 0
}
```

Le funzioni di utilità utilizzate in questo esempio.

```
###############################################################################
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
###############################################################################
function errecho() {
  printf "%s\n" "$*" 1>&2
}
```

```
##############################################################################
# function aws_cli_error_log()
#
# This function is used to log the error messages from the AWS CLI.
#
# The function expects the following argument:
#         $1 - The error code returned by the AWS CLI.
#
#  Returns:
#          0: - Success.
#
##############################################################################
function aws_cli_error_log() {
  local err_code=$1
  errecho "Error code : $err_code"
  if [ "$err_code" == 1 ]; then
    errecho "  One or more S3 transfers failed."
  elif [ "$err_code" == 2 ]; then
    errecho "  Command line failed to parse."
  elif [ "$err_code" == 130 ]; then
    errecho "  Process received SIGINT."
  elif [ "$err_code" == 252 ]; then
    errecho "  Command syntax invalid."
  elif [ "$err_code" == 253 ]; then
    errecho "  The system environment or configuration was invalid."
  elif [ "$err_code" == 254 ]; then
    errecho "  The service returned an error."
  elif [ "$err_code" == 255 ]; then
    errecho "  255 is a catch-all error."
  fi

  return 0
}
```

- Per i dettagli sull'API, consulta StopInstances AWS CLICommand Reference.

`TerminateInstances`

Il seguente esempio di codice mostra come utilizzare`TerminateInstances`.

AWS CLI con lo script Bash

> **Note**
>
> C'è altro da fare. GitHub Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codice AWS.

```
###############################################################################
# function ec2_terminate_instances
#
# This function terminates one or more Amazon Elastic Compute Cloud (Amazon EC2)
# instances using the AWS CLI.
#
# Parameters:
#       -i instance_ids - A space-separated list of instance IDs.
#       -h - Display help.
#
# Returns:
#       0 - If successful.
#       1 - If it fails.
###############################################################################
function ec2_terminate_instances() {
  local instance_ids response
  local option OPTARG # Required to use getopts command in a function.

  # bashsupport disable=BP5008
  function usage() {
    echo "function ec2_terminate_instances"
    echo "Terminates one or more Amazon Elastic Compute Cloud (Amazon EC2)
 instances."
    echo "  -i instance_ids - A space-separated list of instance IDs."
    echo "  -h - Display help."
    echo ""
  }

  # Retrieve the calling parameters.
  while getopts "i:h" option; do
    case "${option}" in
      i) instance_ids="${OPTARG}" ;;
      h)
        usage
```

```
        return 0
        ;;
      \?)
        echo "Invalid parameter"
        usage
        return 1
        ;;
    esac
  done
  export OPTIND=1

  # Check if instance ID is provided
  if [[ -z "${instance_ids}" ]]; then
    echo "Error: Missing required instance IDs parameter."
    usage
    return 1
  fi

  # shellcheck disable=SC2086
  response=$(aws ec2 terminate-instances \
    "--instance-ids" $instance_ids \
    --query 'TerminatingInstances[*].[InstanceId,CurrentState.Name]' \
    --output text) || {
    aws_cli_error_log ${?}
    errecho "ERROR: AWS reports terminate-instances operation failed.$response"
    return 1
  }

  return 0
}
```

Le funzioni di utilità utilizzate in questo esempio.

```
###############################################################################
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
###############################################################################
function errecho() {
  printf "%s\n" "$*" 1>&2
}
```

```
###############################################################################
# function aws_cli_error_log()
#
# This function is used to log the error messages from the AWS CLI.
#
# The function expects the following argument:
#         $1 - The error code returned by the AWS CLI.
#
#  Returns:
#          0: - Success.
#
###############################################################################
function aws_cli_error_log() {
  local err_code=$1
  errecho "Error code : $err_code"
  if [ "$err_code" == 1 ]; then
    errecho "  One or more S3 transfers failed."
  elif [ "$err_code" == 2 ]; then
    errecho "  Command line failed to parse."
  elif [ "$err_code" == 130 ]; then
    errecho "  Process received SIGINT."
  elif [ "$err_code" == 252 ]; then
    errecho "  Command syntax invalid."
  elif [ "$err_code" == 253 ]; then
    errecho "  The system environment or configuration was invalid."
  elif [ "$err_code" == 254 ]; then
    errecho "  The service returned an error."
  elif [ "$err_code" == 255 ]; then
    errecho "  255 is a catch-all error."
  fi

  return 0
}
```

- Per i dettagli sull'API, consulta TerminateInstances AWS CLICommand Reference.

Scenari

Creare un VPC con sottoreti private e gateway NAT

L'esempio di codice seguente mostra come:

- Creare un VPC con sottoreti private e gateway NAT con la CLI.
- Configurare i componenti necessari tra cui VPC, sottoreti, tabelle di routing e gateway NAT.
- Configurare i gruppi di sicurezza e i ruoli IAM per un accesso e una sicurezza adeguati.
- Utilizzare i comandi della CLI per automatizzare la creazione e la configurazione di queste risorse.

AWS CLI con lo script Bash

> **Note**
>
> C'è altro da fare. GitHub Trova l'esempio completo e scopri come eseguire la configurazione e l'esecuzione nel repository dei tutorial sugli esempi di codice per gli sviluppatori.

```
#!/bin/bash

# VPC with Private Subnets and NAT Gateways (IMDSv2 Compliant Version)
# This script creates a VPC with public and private subnets in two Availability
 Zones,
# NAT gateways, an internet gateway, route tables, a VPC endpoint for S3,
# security groups, a launch template, an Auto Scaling group, and an Application Load
 Balancer.

# Set up logging
LOG_FILE="vpc-private-subnets-nat.log"
exec > >(tee -a "$LOG_FILE") 2>&1

# Cleanup function to delete all created resources
cleanup_resources() {
  echo "Cleaning up resources..."

  # Delete Auto Scaling group if it exists
  if [ -n "${ASG_NAME:-}" ]; then
    echo "Deleting Auto Scaling group: $ASG_NAME"
    aws autoscaling delete-auto-scaling-group --auto-scaling-group-name "$ASG_NAME"
 --force-delete
    echo "Waiting for Auto Scaling group to be deleted..."
    aws autoscaling wait auto-scaling-groups-deleted --auto-scaling-group-names
 "$ASG_NAME"
```

```
fi

# Delete load balancer if it exists
if [ -n "${LB_ARN:-}" ]; then
  echo "Deleting load balancer: $LB_ARN"
  aws elbv2 delete-load-balancer --load-balancer-arn "$LB_ARN"
  # Wait for load balancer to be deleted
  sleep 30
fi

# Delete target group if it exists
if [ -n "${TARGET_GROUP_ARN:-}" ]; then
  echo "Deleting target group: $TARGET_GROUP_ARN"
  aws elbv2 delete-target-group --target-group-arn "$TARGET_GROUP_ARN"
fi

# Delete launch template if it exists
if [ -n "${LAUNCH_TEMPLATE_NAME:-}" ]; then
  echo "Deleting launch template: $LAUNCH_TEMPLATE_NAME"
  aws ec2 delete-launch-template --launch-template-name "$LAUNCH_TEMPLATE_NAME"
fi

# Delete NAT Gateways if they exist
if [ -n "${NAT_GW1_ID:-}" ]; then
  echo "Deleting NAT Gateway 1: $NAT_GW1_ID"
  aws ec2 delete-nat-gateway --nat-gateway-id "$NAT_GW1_ID"
fi

if [ -n "${NAT_GW2_ID:-}" ]; then
  echo "Deleting NAT Gateway 2: $NAT_GW2_ID"
  aws ec2 delete-nat-gateway --nat-gateway-id "$NAT_GW2_ID"
fi

# Wait for NAT Gateways to be deleted
if [ -n "${NAT_GW1_ID:-}" ] || [ -n "${NAT_GW2_ID:-}" ]; then
  echo "Waiting for NAT Gateways to be deleted..."
  sleep 60
fi

# Release Elastic IPs if they exist
if [ -n "${EIP1_ALLOC_ID:-}" ]; then
  echo "Releasing Elastic IP 1: $EIP1_ALLOC_ID"
  aws ec2 release-address --allocation-id "$EIP1_ALLOC_ID"
fi
```

```
  if [ -n "${EIP2_ALLOC_ID:-}" ]; then
    echo "Releasing Elastic IP 2: $EIP2_ALLOC_ID"
    aws ec2 release-address --allocation-id "$EIP2_ALLOC_ID"
  fi

  # Delete VPC endpoint if it exists
  if [ -n "${VPC_ENDPOINT_ID:-}" ]; then
    echo "Deleting VPC endpoint: $VPC_ENDPOINT_ID"
    aws ec2 delete-vpc-endpoints --vpc-endpoint-ids "$VPC_ENDPOINT_ID"
  fi

  # Delete security groups if they exist
  if [ -n "${APP_SG_ID:-}" ]; then
    echo "Deleting application security group: $APP_SG_ID"
    aws ec2 delete-security-group --group-id "$APP_SG_ID"
  fi

  if [ -n "${LB_SG_ID:-}" ]; then
    echo "Deleting load balancer security group: $LB_SG_ID"
    aws ec2 delete-security-group --group-id "$LB_SG_ID"
  fi

  # Detach and delete Internet Gateway if it exists
  if [ -n "${IGW_ID:-}" ] && [ -n "${VPC_ID:-}" ]; then
    echo "Detaching Internet Gateway: $IGW_ID from VPC: $VPC_ID"
    aws ec2 detach-internet-gateway --internet-gateway-id "$IGW_ID" --vpc-id
"$VPC_ID"
    echo "Deleting Internet Gateway: $IGW_ID"
    aws ec2 delete-internet-gateway --internet-gateway-id "$IGW_ID"
  fi

  # Delete route table associations and route tables if they exist
  if [ -n "${PUBLIC_RT_ASSOC1_ID:-}" ]; then
    echo "Disassociating public route table from subnet 1: $PUBLIC_RT_ASSOC1_ID"
    aws ec2 disassociate-route-table --association-id "$PUBLIC_RT_ASSOC1_ID"
  fi

  if [ -n "${PUBLIC_RT_ASSOC2_ID:-}" ]; then
    echo "Disassociating public route table from subnet 2: $PUBLIC_RT_ASSOC2_ID"
    aws ec2 disassociate-route-table --association-id "$PUBLIC_RT_ASSOC2_ID"
  fi

  if [ -n "${PRIVATE_RT1_ASSOC_ID:-}" ]; then
```

```
  echo "Disassociating private route table 1: $PRIVATE_RT1_ASSOC_ID"
  aws ec2 disassociate-route-table --association-id "$PRIVATE_RT1_ASSOC_ID"
fi

if [ -n "${PRIVATE_RT2_ASSOC_ID:-}" ]; then
  echo "Disassociating private route table 2: $PRIVATE_RT2_ASSOC_ID"
  aws ec2 disassociate-route-table --association-id "$PRIVATE_RT2_ASSOC_ID"
fi

if [ -n "${PUBLIC_RT_ID:-}" ]; then
  echo "Deleting public route table: $PUBLIC_RT_ID"
  aws ec2 delete-route-table --route-table-id "$PUBLIC_RT_ID"
fi

if [ -n "${PRIVATE_RT1_ID:-}" ]; then
  echo "Deleting private route table 1: $PRIVATE_RT1_ID"
  aws ec2 delete-route-table --route-table-id "$PRIVATE_RT1_ID"
fi

if [ -n "${PRIVATE_RT2_ID:-}" ]; then
  echo "Deleting private route table 2: $PRIVATE_RT2_ID"
  aws ec2 delete-route-table --route-table-id "$PRIVATE_RT2_ID"
fi

# Delete subnets if they exist
if [ -n "${PUBLIC_SUBNET1_ID:-}" ]; then
  echo "Deleting public subnet 1: $PUBLIC_SUBNET1_ID"
  aws ec2 delete-subnet --subnet-id "$PUBLIC_SUBNET1_ID"
fi

if [ -n "${PUBLIC_SUBNET2_ID:-}" ]; then
  echo "Deleting public subnet 2: $PUBLIC_SUBNET2_ID"
  aws ec2 delete-subnet --subnet-id "$PUBLIC_SUBNET2_ID"
fi

if [ -n "${PRIVATE_SUBNET1_ID:-}" ]; then
  echo "Deleting private subnet 1: $PRIVATE_SUBNET1_ID"
  aws ec2 delete-subnet --subnet-id "$PRIVATE_SUBNET1_ID"
fi

if [ -n "${PRIVATE_SUBNET2_ID:-}" ]; then
  echo "Deleting private subnet 2: $PRIVATE_SUBNET2_ID"
  aws ec2 delete-subnet --subnet-id "$PRIVATE_SUBNET2_ID"
fi
```

```
  # Delete VPC if it exists
  if [ -n "${VPC_ID:-}" ]; then
    echo "Deleting VPC: $VPC_ID"
    aws ec2 delete-vpc --vpc-id "$VPC_ID"
  fi

  echo "Cleanup completed."
}

# Error handling function
handle_error() {
  echo "ERROR: $1"
  echo "Attempting to clean up resources..."
  cleanup_resources
  exit 1
}

# Function to check command success
check_command() {
  if [ $? -ne 0 ]; then
    handle_error "$1"
  fi
}

# Generate a random identifier for resource names
RANDOM_ID=$(openssl rand -hex 4)
echo "Using random identifier: $RANDOM_ID"

# Create VPC
echo "Creating VPC..."
VPC_RESULT=$(aws ec2 create-vpc --cidr-block 10.0.0.0/16 --tag-specifications
 "ResourceType=vpc,Tags=[{Key=Name,Value=ProductionVPC-$RANDOM_ID}]")
check_command "Failed to create VPC"

VPC_ID=$(echo "$VPC_RESULT" | jq -r '.Vpc.VpcId')
echo "VPC created with ID: $VPC_ID"

# Get Availability Zones
echo "Getting Availability Zones..."
AZ_RESULT=$(aws ec2 describe-availability-zones --query
 'AvailabilityZones[0:2].ZoneName' --output text)
check_command "Failed to get Availability Zones"
```

```
# Convert space-separated output to array
read -r -a AZS <<< "$AZ_RESULT"
AZ1=${AZS[0]}
AZ2=${AZS[1]}
echo "Using Availability Zones: $AZ1 and $AZ2"

# Create subnets
echo "Creating subnets..."
PUBLIC_SUBNET1_RESULT=$(aws ec2 create-subnet --vpc-id "$VPC_ID" --
cidr-block 10.0.0.0/24 --availability-zone "$AZ1" --tag-specifications
 "ResourceType=subnet,Tags=[{Key=Name,Value=PublicSubnet1-$RANDOM_ID}]")
check_command "Failed to create public subnet 1"
PUBLIC_SUBNET1_ID=$(echo "$PUBLIC_SUBNET1_RESULT" | jq -r '.Subnet.SubnetId')

PRIVATE_SUBNET1_RESULT=$(aws ec2 create-subnet --vpc-id "$VPC_ID" --
cidr-block 10.0.1.0/24 --availability-zone "$AZ1" --tag-specifications
 "ResourceType=subnet,Tags=[{Key=Name,Value=PrivateSubnet1-$RANDOM_ID}]")
check_command "Failed to create private subnet 1"
PRIVATE_SUBNET1_ID=$(echo "$PRIVATE_SUBNET1_RESULT" | jq -r '.Subnet.SubnetId')

PUBLIC_SUBNET2_RESULT=$(aws ec2 create-subnet --vpc-id "$VPC_ID" --
cidr-block 10.0.2.0/24 --availability-zone "$AZ2" --tag-specifications
 "ResourceType=subnet,Tags=[{Key=Name,Value=PublicSubnet2-$RANDOM_ID}]")
check_command "Failed to create public subnet 2"
PUBLIC_SUBNET2_ID=$(echo "$PUBLIC_SUBNET2_RESULT" | jq -r '.Subnet.SubnetId')

PRIVATE_SUBNET2_RESULT=$(aws ec2 create-subnet --vpc-id "$VPC_ID" --
cidr-block 10.0.3.0/24 --availability-zone "$AZ2" --tag-specifications
 "ResourceType=subnet,Tags=[{Key=Name,Value=PrivateSubnet2-$RANDOM_ID}]")
check_command "Failed to create private subnet 2"
PRIVATE_SUBNET2_ID=$(echo "$PRIVATE_SUBNET2_RESULT" | jq -r '.Subnet.SubnetId')

echo "Subnets created with IDs:"
echo "Public Subnet 1: $PUBLIC_SUBNET1_ID"
echo "Private Subnet 1: $PRIVATE_SUBNET1_ID"
echo "Public Subnet 2: $PUBLIC_SUBNET2_ID"
echo "Private Subnet 2: $PRIVATE_SUBNET2_ID"

# Create Internet Gateway
echo "Creating Internet Gateway..."
IGW_RESULT=$(aws ec2 create-internet-gateway --tag-specifications
 "ResourceType=internet-gateway,Tags=[{Key=Name,Value=ProductionIGW-$RANDOM_ID}]")
check_command "Failed to create Internet Gateway"
IGW_ID=$(echo "$IGW_RESULT" | jq -r '.InternetGateway.InternetGatewayId')
```

```
echo "Internet Gateway created with ID: $IGW_ID"

# Attach Internet Gateway to VPC
echo "Attaching Internet Gateway to VPC..."
aws ec2 attach-internet-gateway --internet-gateway-id "$IGW_ID" --vpc-id "$VPC_ID"
check_command "Failed to attach Internet Gateway to VPC"

# Create route tables
echo "Creating route tables..."
PUBLIC_RT_RESULT=$(aws ec2 create-route-table --vpc-id "$VPC_ID" --tag-
specifications "ResourceType=route-table,Tags=[{Key=Name,Value=PublicRouteTable-
$RANDOM_ID}]")
check_command "Failed to create public route table"
PUBLIC_RT_ID=$(echo "$PUBLIC_RT_RESULT" | jq -r '.RouteTable.RouteTableId')

PRIVATE_RT1_RESULT=$(aws ec2 create-route-table --vpc-id "$VPC_ID" --tag-
specifications "ResourceType=route-table,Tags=[{Key=Name,Value=PrivateRouteTable1-
$RANDOM_ID}]")
check_command "Failed to create private route table 1"
PRIVATE_RT1_ID=$(echo "$PRIVATE_RT1_RESULT" | jq -r '.RouteTable.RouteTableId')

PRIVATE_RT2_RESULT=$(aws ec2 create-route-table --vpc-id "$VPC_ID" --tag-
specifications "ResourceType=route-table,Tags=[{Key=Name,Value=PrivateRouteTable2-
$RANDOM_ID}]")
check_command "Failed to create private route table 2"
PRIVATE_RT2_ID=$(echo "$PRIVATE_RT2_RESULT" | jq -r '.RouteTable.RouteTableId')

echo "Route tables created with IDs:"
echo "Public Route Table: $PUBLIC_RT_ID"
echo "Private Route Table 1: $PRIVATE_RT1_ID"
echo "Private Route Table 2: $PRIVATE_RT2_ID"

# Add route to Internet Gateway in public route table
echo "Adding route to Internet Gateway in public route table..."
aws ec2 create-route --route-table-id "$PUBLIC_RT_ID" --destination-cidr-block
 0.0.0.0/0 --gateway-id "$IGW_ID"
check_command "Failed to add route to Internet Gateway"

# Associate subnets with route tables
echo "Associating subnets with route tables..."
PUBLIC_RT_ASSOC1_RESULT=$(aws ec2 associate-route-table --route-table-id
 "$PUBLIC_RT_ID" --subnet-id "$PUBLIC_SUBNET1_ID")
check_command "Failed to associate public subnet 1 with route table"
PUBLIC_RT_ASSOC1_ID=$(echo "$PUBLIC_RT_ASSOC1_RESULT" | jq -r '.AssociationId')
```

```
PUBLIC_RT_ASSOC2_RESULT=$(aws ec2 associate-route-table --route-table-id
 "$PUBLIC_RT_ID" --subnet-id "$PUBLIC_SUBNET2_ID")
check_command "Failed to associate public subnet 2 with route table"
PUBLIC_RT_ASSOC2_ID=$(echo "$PUBLIC_RT_ASSOC2_RESULT" | jq -r '.AssociationId')

PRIVATE_RT1_ASSOC_RESULT=$(aws ec2 associate-route-table --route-table-id
 "$PRIVATE_RT1_ID" --subnet-id "$PRIVATE_SUBNET1_ID")
check_command "Failed to associate private subnet 1 with route table"
PRIVATE_RT1_ASSOC_ID=$(echo "$PRIVATE_RT1_ASSOC_RESULT" | jq -r '.AssociationId')

PRIVATE_RT2_ASSOC_RESULT=$(aws ec2 associate-route-table --route-table-id
 "$PRIVATE_RT2_ID" --subnet-id "$PRIVATE_SUBNET2_ID")
check_command "Failed to associate private subnet 2 with route table"
PRIVATE_RT2_ASSOC_ID=$(echo "$PRIVATE_RT2_ASSOC_RESULT" | jq -r '.AssociationId')

echo "Route table associations created with IDs:"
echo "Public Subnet 1 Association: $PUBLIC_RT_ASSOC1_ID"
echo "Public Subnet 2 Association: $PUBLIC_RT_ASSOC2_ID"
echo "Private Subnet 1 Association: $PRIVATE_RT1_ASSOC_ID"
echo "Private Subnet 2 Association: $PRIVATE_RT2_ASSOC_ID"

# Create NAT Gateways
echo "Creating NAT Gateways..."

# Allocate Elastic IPs for NAT Gateways
echo "Allocating Elastic IPs for NAT Gateways..."
EIP1_RESULT=$(aws ec2 allocate-address --domain vpc --tag-specifications
 "ResourceType=elastic-ip,Tags=[{Key=Name,Value=NAT1-EIP-$RANDOM_ID}]")
check_command "Failed to allocate Elastic IP 1"
EIP1_ALLOC_ID=$(echo "$EIP1_RESULT" | jq -r '.AllocationId')

EIP2_RESULT=$(aws ec2 allocate-address --domain vpc --tag-specifications
 "ResourceType=elastic-ip,Tags=[{Key=Name,Value=NAT2-EIP-$RANDOM_ID}]")
check_command "Failed to allocate Elastic IP 2"
EIP2_ALLOC_ID=$(echo "$EIP2_RESULT" | jq -r '.AllocationId')

echo "Elastic IPs allocated with IDs:"
echo "EIP 1 Allocation ID: $EIP1_ALLOC_ID"
echo "EIP 2 Allocation ID: $EIP2_ALLOC_ID"

# Create NAT Gateways
echo "Creating NAT Gateway in public subnet 1..."
```

```
NAT_GW1_RESULT=$(aws ec2 create-nat-gateway --subnet-id
 "$PUBLIC_SUBNET1_ID" --allocation-id "$EIP1_ALLOC_ID" --tag-specifications
 "ResourceType=natgateway,Tags=[{Key=Name,Value=NAT-Gateway1-$RANDOM_ID}]")
check_command "Failed to create NAT Gateway 1"
NAT_GW1_ID=$(echo "$NAT_GW1_RESULT" | jq -r '.NatGateway.NatGatewayId')

echo "Creating NAT Gateway in public subnet 2..."
NAT_GW2_RESULT=$(aws ec2 create-nat-gateway --subnet-id
 "$PUBLIC_SUBNET2_ID" --allocation-id "$EIP2_ALLOC_ID" --tag-specifications
 "ResourceType=natgateway,Tags=[{Key=Name,Value=NAT-Gateway2-$RANDOM_ID}]")
check_command "Failed to create NAT Gateway 2"
NAT_GW2_ID=$(echo "$NAT_GW2_RESULT" | jq -r '.NatGateway.NatGatewayId')

echo "NAT Gateways created with IDs:"
echo "NAT Gateway 1: $NAT_GW1_ID"
echo "NAT Gateway 2: $NAT_GW2_ID"

# Wait for NAT Gateways to be available
echo "Waiting for NAT Gateways to be available..."
aws ec2 wait nat-gateway-available --nat-gateway-ids "$NAT_GW1_ID"
check_command "NAT Gateway 1 did not become available"
aws ec2 wait nat-gateway-available --nat-gateway-ids "$NAT_GW2_ID"
check_command "NAT Gateway 2 did not become available"
echo "NAT Gateways are now available"

# Add routes to NAT Gateways in private route tables
echo "Adding routes to NAT Gateways in private route tables..."
aws ec2 create-route --route-table-id "$PRIVATE_RT1_ID" --destination-cidr-block
 0.0.0.0/0 --nat-gateway-id "$NAT_GW1_ID"
check_command "Failed to add route to NAT Gateway 1"

aws ec2 create-route --route-table-id "$PRIVATE_RT2_ID" --destination-cidr-block
 0.0.0.0/0 --nat-gateway-id "$NAT_GW2_ID"
check_command "Failed to add route to NAT Gateway 2"

# Create VPC Endpoint for S3
echo "Creating VPC Endpoint for S3..."
S3_PREFIX_LIST_ID=$(aws ec2 describe-prefix-lists --filters "Name=prefix-
list-name,Values=com.amazonaws.$(aws configure get region).s3" --query
 'PrefixLists[0].PrefixListId' --output text)
check_command "Failed to get S3 prefix list ID"

VPC_ENDPOINT_RESULT=$(aws ec2 create-vpc-endpoint --vpc-id "$VPC_ID" --
service-name "com.amazonaws.$(aws configure get region).s3" --route-table-ids
```

```
 "$PRIVATE_RT1_ID" "$PRIVATE_RT2_ID" --tag-specifications "ResourceType=vpc-
endpoint,Tags=[{Key=Name,Value=S3-Endpoint-$RANDOM_ID}]")
check_command "Failed to create VPC endpoint for S3"
VPC_ENDPOINT_ID=$(echo "$VPC_ENDPOINT_RESULT" | jq -r '.VpcEndpoint.VpcEndpointId')
echo "VPC Endpoint created with ID: $VPC_ENDPOINT_ID"

# Create security groups
echo "Creating security groups..."
LB_SG_RESULT=$(aws ec2 create-security-group --group-name "LoadBalancerSG-
$RANDOM_ID" --description "Security group for the load balancer"
 --vpc-id "$VPC_ID" --tag-specifications "ResourceType=security-
group,Tags=[{Key=Name,Value=LoadBalancerSG-$RANDOM_ID}]")
check_command "Failed to create load balancer security group"
LB_SG_ID=$(echo "$LB_SG_RESULT" | jq -r '.GroupId')

# Allow inbound HTTP traffic from anywhere to the load balancer
aws ec2 authorize-security-group-ingress --group-id "$LB_SG_ID" --protocol tcp --
port 80 --cidr 0.0.0.0/0
check_command "Failed to authorize ingress to load balancer security group"

APP_SG_RESULT=$(aws ec2 create-security-group --group-name "AppServerSG-$RANDOM_ID"
 --description "Security group for the application servers" --vpc-id "$VPC_ID" --
tag-specifications "ResourceType=security-group,Tags=[{Key=Name,Value=AppServerSG-
$RANDOM_ID}]")
check_command "Failed to create application server security group"
APP_SG_ID=$(echo "$APP_SG_RESULT" | jq -r '.GroupId')

# Allow inbound HTTP traffic from the load balancer security group to the
 application servers
aws ec2 authorize-security-group-ingress --group-id "$APP_SG_ID" --protocol tcp --
port 80 --source-group "$LB_SG_ID"
check_command "Failed to authorize ingress to application server security group"

echo "Security groups created with IDs:"
echo "Load Balancer Security Group: $LB_SG_ID"
echo "Application Server Security Group: $APP_SG_ID"

# Create a launch template
echo "Creating launch template..."

# Create user data script with IMDSv2 support
cat > user-data.sh << 'EOF'
#!/bin/bash
yum update -y
```

```
yum install -y httpd
systemctl start httpd
systemctl enable httpd

# Use IMDSv2 with session token
TOKEN=$(curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-ec2-
metadata-token-ttl-seconds: 21600")
AZ=$(curl -H "X-aws-ec2-metadata-token: $TOKEN" -s http://169.254.169.254/latest/
meta-data/placement/availability-zone)
HOSTNAME=$(hostname -f)

echo "<h1>Hello from $HOSTNAME in $AZ</h1>" > /var/www/html/index.html
EOF

# Encode user data
USER_DATA=$(base64 -w 0 user-data.sh)

# Get latest Amazon Linux 2 AMI
echo "Getting latest Amazon Linux 2 AMI..."
AMI_ID=$(aws ec2 describe-images --owners amazon --filters "Name=name,Values=amzn2-
ami-hvm-*-x86_64-gp2" "Name=state,Values=available" --query 'sort_by(Images,
 &CreationDate)[-1].ImageId' --output text)
check_command "Failed to get latest Amazon Linux 2 AMI"
echo "Using AMI: $AMI_ID"

# Create launch template with IMDSv2 required
LAUNCH_TEMPLATE_NAME="AppServerTemplate-$RANDOM_ID"
echo "Creating launch template: $LAUNCH_TEMPLATE_NAME"

aws ec2 create-launch-template \
  --launch-template-name "$LAUNCH_TEMPLATE_NAME" \
  --version-description "Initial version" \
  --tag-specifications "ResourceType=launch-template,Tags=[{Key=Name,Value=
$LAUNCH_TEMPLATE_NAME}]" \
  --launch-template-data "{
    \"NetworkInterfaces\": [{
      \"DeviceIndex\": 0,
      \"Groups\": [\"$APP_SG_ID\"],
      \"DeleteOnTermination\": true
    }],
    \"ImageId\": \"$AMI_ID\",
    \"InstanceType\": \"t3.micro\",
    \"UserData\": \"$USER_DATA\",
    \"MetadataOptions\": {
```

```
      \"HttpTokens\": \"required\",
      \"HttpEndpoint\": \"enabled\"
    },
    \"TagSpecifications\": [{
      \"ResourceType\": \"instance\",
      \"Tags\": [{
        \"Key\": \"Name\",
        \"Value\": \"AppServer-$RANDOM_ID\"
      }]
    }]
  }"
check_command "Failed to create launch template"

# Create target group
echo "Creating target group..."
TARGET_GROUP_NAME="AppTargetGroup-$RANDOM_ID"
TARGET_GROUP_RESULT=$(aws elbv2 create-target-group \
  --name "$TARGET_GROUP_NAME" \
  --protocol HTTP \
  --port 80 \
  --vpc-id "$VPC_ID" \
  --target-type instance \
  --health-check-protocol HTTP \
  --health-check-path "/" \
  --health-check-port traffic-port)
check_command "Failed to create target group"
TARGET_GROUP_ARN=$(echo "$TARGET_GROUP_RESULT" | jq -r
 '.TargetGroups[0].TargetGroupArn')
echo "Target group created with ARN: $TARGET_GROUP_ARN"

# Create load balancer
echo "Creating load balancer..."
LB_NAME="AppLoadBalancer-$RANDOM_ID"
LB_RESULT=$(aws elbv2 create-load-balancer \
  --name "$LB_NAME" \
  --subnets "$PUBLIC_SUBNET1_ID" "$PUBLIC_SUBNET2_ID" \
  --security-groups "$LB_SG_ID" \
  --tags "Key=Name,Value=$LB_NAME")
check_command "Failed to create load balancer"
LB_ARN=$(echo "$LB_RESULT" | jq -r '.LoadBalancers[0].LoadBalancerArn')
echo "Load balancer created with ARN: $LB_ARN"

# Wait for load balancer to be active
echo "Waiting for load balancer to be active..."
```

```
aws elbv2 wait load-balancer-available --load-balancer-arns "$LB_ARN"
check_command "Load balancer did not become available"

# Create listener
echo "Creating listener..."
LISTENER_RESULT=$(aws elbv2 create-listener \
  --load-balancer-arn "$LB_ARN" \
  --protocol HTTP \
  --port 80 \
  --default-actions "Type=forward,TargetGroupArn=$TARGET_GROUP_ARN")
check_command "Failed to create listener"
LISTENER_ARN=$(echo "$LISTENER_RESULT" | jq -r '.Listeners[0].ListenerArn')
echo "Listener created with ARN: $LISTENER_ARN"

# Create Auto Scaling group
echo "Creating Auto Scaling group..."
ASG_NAME="AppAutoScalingGroup-$RANDOM_ID"
aws autoscaling create-auto-scaling-group \
  --auto-scaling-group-name "$ASG_NAME" \
  --launch-template "LaunchTemplateName=$LAUNCH_TEMPLATE_NAME,Version=\$Latest" \
  --min-size 2 \
  --max-size 4 \
  --desired-capacity 2 \
  --vpc-zone-identifier "$PRIVATE_SUBNET1_ID,$PRIVATE_SUBNET2_ID" \
  --target-group-arns "$TARGET_GROUP_ARN" \
  --health-check-type ELB \
  --health-check-grace-period 300 \
  --tags "Key=Name,Value=AppServer-$RANDOM_ID,PropagateAtLaunch=true"
check_command "Failed to create Auto Scaling group"
echo "Auto Scaling group created with name: $ASG_NAME"

# Get load balancer DNS name
LB_DNS_NAME=$(aws elbv2 describe-load-balancers --load-balancer-arns "$LB_ARN" --
query 'LoadBalancers[0].DNSName' --output text)
check_command "Failed to get load balancer DNS name"

echo ""
echo "========================================="
echo "DEPLOYMENT COMPLETE"
echo "========================================="
echo "VPC ID: $VPC_ID"
echo "Public Subnet 1: $PUBLIC_SUBNET1_ID (AZ: $AZ1)"
echo "Private Subnet 1: $PRIVATE_SUBNET1_ID (AZ: $AZ1)"
echo "Public Subnet 2: $PUBLIC_SUBNET2_ID (AZ: $AZ2)"
```

```
echo "Private Subnet 2: $PRIVATE_SUBNET2_ID (AZ: $AZ2)"
echo "NAT Gateway 1: $NAT_GW1_ID"
echo "NAT Gateway 2: $NAT_GW2_ID"
echo "Load Balancer: $LB_NAME"
echo "Auto Scaling Group: $ASG_NAME"
echo ""
echo "Your application will be available at: http://$LB_DNS_NAME"
echo "It may take a few minutes for the instances to launch and pass health checks."
echo ""

# Add health check monitoring
echo "=========================================="
echo "MONITORING INSTANCE HEALTH AND LOAD BALANCER"
echo "=========================================="
echo "Waiting for instances to launch and pass health checks..."
echo "This may take 3-5 minutes. Checking every 30 seconds..."

# Monitor instance health and load balancer accessibility
MAX_ATTEMPTS=10
ATTEMPT=1
HEALTHY_INSTANCES=0

while [ $ATTEMPT -le $MAX_ATTEMPTS ] && [ $HEALTHY_INSTANCES -lt 2 ]; do
  echo "Check attempt $ATTEMPT of $MAX_ATTEMPTS..."

  # Check Auto Scaling group instances
  echo "Checking Auto Scaling group instances..."
  ASG_INSTANCES=$(aws autoscaling describe-auto-scaling-groups --auto-
scaling-group-names "$ASG_NAME" --query 'AutoScalingGroups[0].Instances[*].
[InstanceId,HealthStatus]' --output json)
  echo "ASG Instances status:"
  echo "$ASG_INSTANCES" | jq -r '.[] | "Instance: \(.[0]), Health: \(.[1])"'

  # Check target group health
  echo "Checking target group health..."
  TARGET_HEALTH=$(aws elbv2 describe-target-health --target-group-arn
 "$TARGET_GROUP_ARN" --output json)
  echo "Target health status:"
  echo "$TARGET_HEALTH" | jq -r '.TargetHealthDescriptions[] | "Instance:
 \(.Target.Id), State: \(.TargetHealth.State), Reason: \(.TargetHealth.Reason // "N/
A"), Description: \(.TargetHealth.Description // "N/A")"'

  # Count healthy instances
```

```
 HEALTHY_INSTANCES=$(echo "$TARGET_HEALTH" | jq -r '[.TargetHealthDescriptions[] |
select(.TargetHealth.State=="healthy")] | length')
 echo "Number of healthy instances: $HEALTHY_INSTANCES of 2 expected"

 # Check if we have healthy instances
 if [ $HEALTHY_INSTANCES -ge 2 ]; then
   echo "All instances are healthy!"

   # Test load balancer accessibility
   echo "Testing load balancer accessibility..."
   HTTP_STATUS=$(curl -s -o /dev/null -w "%{http_code}" "http://$LB_DNS_NAME")

   if [ "$HTTP_STATUS" = "200" ]; then
     echo "Load balancer is accessible! HTTP Status: $HTTP_STATUS"
     echo "You can access your application at: http://$LB_DNS_NAME"

     # Try to get the content to verify IMDSv2 is working
     echo "Fetching content to verify IMDSv2 functionality..."
     CONTENT=$(curl -s "http://$LB_DNS_NAME")
     echo "Response from server:"
     echo "$CONTENT"

     # Check if the content contains the expected pattern
     if [[ "$CONTENT" == *"Hello from"* && "$CONTENT" == *"in"* ]]; then
       echo "IMDSv2 is working correctly! The instance was able to access metadata
using the token-based approach."
     else
       echo "Warning: Content doesn't match expected pattern. IMDSv2 functionality
could not be verified."
     fi

     break
   else
     echo "Load balancer returned HTTP status: $HTTP_STATUS"
     echo "Will try again in 30 seconds..."
   fi
 else
   echo "Waiting for instances to become healthy..."
   echo "Will check again in 30 seconds..."
 fi

 ATTEMPT=$((ATTEMPT+1))

 if [ $ATTEMPT -le $MAX_ATTEMPTS ]; then
```

```
    sleep 30
  fi
done

if [ $HEALTHY_INSTANCES -lt 2 ]; then
  echo "Warning: Not all instances are healthy after maximum attempts."
  echo "You may need to wait longer or check for configuration issues."
fi

echo "To test your application, run:"
echo "curl http://$LB_DNS_NAME"
echo ""
echo "========================================="
echo "CLEANUP CONFIRMATION"
echo "========================================="
echo "Do you want to clean up all created resources? (y/n): "
read -r CLEANUP_CHOICE

if [[ "$CLEANUP_CHOICE" =~ ^[Yy]$ ]]; then
  cleanup_resources
  echo "All resources have been deleted."
else
  echo "Resources will not be deleted. You can manually delete them later."
  echo "To delete resources, run this script again and choose to clean up."
fi
```

- Per informazioni dettagliate sull'API, consulta i seguenti argomenti nella documentazione di riferimento dei comandi della AWS CLI .
  - AllocateAddress
  - AssociateRouteTable
  - AttachInternetGateway
  - AuthorizeSecurityGroupIngress
  - CreateInternetGateway
  - CreateLaunchTemplate
  - CreateNatGateway
  - CreateRoute
  - CreateRouteTable
  - CreateSecurityGroup

- CreateSubnet
- CreateVpc
- CreateVpcEndpoint
- DeleteAutoScalingGroup
- DeleteInternetGateway
- DeleteLaunchTemplate
- DeleteLoadBalancer
- DeleteNatGateway
- DeleteRouteTable
- DeleteSecurityGroup
- DeleteSubnet
- DeleteTargetGroup
- DeleteVpc
- DeleteVpcEndpoints
- DescribeAvailabilityZones
- DescribeImages
- DescribePrefixLists
- DetachInternetGateway
- ReleaseAddress

Nozioni di base su Amazon VPC

L'esempio di codice seguente mostra come:

- Configurare l'account
- Creare e configurare un VPC
- Configurare la rete
- Configurare la sicurezza
- Distribuire le risorse
- Testare e verificare
- Eseguire la pulizia delle risorse
- Valutare le implicazioni sulla produzione

- Valutare le implicazioni sulla produzione

AWS CLI con lo script Bash

> **Note**
>
> C'è altro da fare. GitHub Trova l'esempio completo e scopri come eseguire la configurazione e l'esecuzione nel repository dei tutorial sugli esempi di codice per gli sviluppatori.

```
#!/bin/bash

# VPC Creation Script
# This script creates a VPC with public and private subnets, internet gateway, NAT
 gateway, and security groups

# Set up logging
LOG_FILE="vpc_creation.log"
exec > >(tee -a "$LOG_FILE") 2>&1

# Function to handle errors
handle_error() {
  echo "ERROR: $1"
  echo "Resources created before error:"
  for resource in "${CREATED_RESOURCES[@]}"
  do
    echo "- $resource"
  done

  echo "Attempting to clean up resources..."
  cleanup_resources
  exit 1
}

# Function to clean up resources
cleanup_resources() {
  echo "Cleaning up resources in reverse order..."

  # Reverse the array to delete in reverse order of creation
```

```
 for ((i=${#CREATED_RESOURCES[@]}-1; i>=0; i--))
 do
   resource="${CREATED_RESOURCES[$i]}"
   resource_type=$(echo "$resource" | cut -d':' -f1)
   resource_id=$(echo "$resource" | cut -d':' -f2)

   case "$resource_type" in
     "INSTANCE")
       echo "Terminating EC2 instance: $resource_id"
       aws ec2 terminate-instances --instance-ids "$resource_id" || echo "Failed to
terminate instance: $resource_id"
       # Wait for instance to terminate
       echo "Waiting for instance to terminate..."
       aws ec2 wait instance-terminated --instance-ids "$resource_id" || echo
"Failed to wait for instance termination: $resource_id"
       ;;
     "KEY_PAIR")
       echo "Deleting key pair: $resource_id"
       aws ec2 delete-key-pair --key-name "$resource_id" || echo "Failed to delete
key pair: $resource_id"
       # Remove the .pem file if it exists
       if [ -f "${resource_id}.pem" ]; then
         rm -f "${resource_id}.pem"
       fi
       ;;
     "NAT_GATEWAY")
       echo "Deleting NAT Gateway: $resource_id"
       aws ec2 delete-nat-gateway --nat-gateway-id "$resource_id" || echo "Failed
to delete NAT Gateway: $resource_id"
       # NAT Gateway deletion takes time, wait for it to complete
       echo "Waiting for NAT Gateway to be deleted..."
       aws ec2 wait nat-gateway-deleted --nat-gateway-ids "$resource_id" || echo
"Failed to wait for NAT Gateway deletion: $resource_id"
       ;;
     "EIP")
       echo "Releasing Elastic IP: $resource_id"
       aws ec2 release-address --allocation-id "$resource_id" || echo "Failed to
release Elastic IP: $resource_id"
       ;;
     "ROUTE_TABLE_ASSOCIATION")
       echo "Disassociating Route Table: $resource_id"
       aws ec2 disassociate-route-table --association-id "$resource_id" || echo
"Failed to disassociate Route Table: $resource_id"
       ;;
```

```
      "ROUTE_TABLE")
        echo "Deleting Route Table: $resource_id"
        aws ec2 delete-route-table --route-table-id "$resource_id" || echo "Failed
 to delete Route Table: $resource_id"
        ;;
      "INTERNET_GATEWAY")
        echo "Detaching Internet Gateway: $resource_id from VPC: $VPC_ID"
        aws ec2 detach-internet-gateway --internet-gateway-id "$resource_id" --vpc-
id "$VPC_ID" || echo "Failed to detach Internet Gateway: $resource_id"
        echo "Deleting Internet Gateway: $resource_id"
        aws ec2 delete-internet-gateway --internet-gateway-id "$resource_id" || echo
 "Failed to delete Internet Gateway: $resource_id"
        ;;
      "SECURITY_GROUP")
        echo "Deleting Security Group: $resource_id"
        aws ec2 delete-security-group --group-id "$resource_id" || echo "Failed to
 delete Security Group: $resource_id"
        ;;
      "SUBNET")
        echo "Deleting Subnet: $resource_id"
        aws ec2 delete-subnet --subnet-id "$resource_id" || echo "Failed to delete
 Subnet: $resource_id"
        ;;
      "VPC")
        echo "Deleting VPC: $resource_id"
        aws ec2 delete-vpc --vpc-id "$resource_id" || echo "Failed to delete VPC:
 $resource_id"
        ;;
    esac
  done
}

# Initialize array to track created resources
CREATED_RESOURCES=()

echo "Starting VPC creation script at $(date)"

# Verify AWS CLI configuration
echo "Verifying AWS CLI configuration..."
aws configure list || handle_error "AWS CLI is not properly configured"

# Verify identity and permissions
echo "Verifying identity and permissions..."
if ! aws sts get-caller-identity; then
```

```
  echo "ERROR: Unable to verify AWS identity. This could be due to:"
  echo "  - Expired credentials"
  echo "  - Missing or invalid AWS credentials"
  echo "  - Insufficient permissions"
  echo ""
  echo "Please run 'aws configure' to update your credentials or check your IAM
 permissions."
  exit 1
fi

# Create VPC
echo "Creating VPC with CIDR block 10.0.0.0/16..."
VPC_ID=$(aws ec2 create-vpc --cidr-block 10.0.0.0/16 --tag-specifications
 'ResourceType=vpc,Tags=[{Key=Name,Value=MyVPC}]' --query 'Vpc.VpcId' --output text)

if [ -z "$VPC_ID" ]; then
  handle_error "Failed to create VPC"
fi

CREATED_RESOURCES+=("VPC:$VPC_ID")
echo "VPC created with ID: $VPC_ID"

# Enable DNS support and hostnames
echo "Enabling DNS support and hostnames for VPC..."
aws ec2 modify-vpc-attribute --vpc-id "$VPC_ID" --enable-dns-support || handle_error
 "Failed to enable DNS support"
aws ec2 modify-vpc-attribute --vpc-id "$VPC_ID" --enable-dns-hostnames ||
 handle_error "Failed to enable DNS hostnames"

# Get available Availability Zones
echo "Getting available Availability Zones..."
AZ1=$(aws ec2 describe-availability-zones --query 'AvailabilityZones[0].ZoneName' --
output text)
AZ2=$(aws ec2 describe-availability-zones --query 'AvailabilityZones[1].ZoneName' --
output text)

if [ -z "$AZ1" ] || [ -z "$AZ2" ]; then
  handle_error "Failed to get Availability Zones"
fi

echo "Using Availability Zones: $AZ1 and $AZ2"

# Create public subnets
echo "Creating public subnet in $AZ1..."
```

```
PUBLIC_SUBNET_AZ1=$(aws ec2 create-subnet \
  --vpc-id "$VPC_ID" \
  --cidr-block 10.0.0.0/24 \
  --availability-zone "$AZ1" \
  --tag-specifications 'ResourceType=subnet,Tags=[{Key=Name,Value=Public-Subnet-
AZ1}]' \
  --query 'Subnet.SubnetId' \
  --output text)

if [ -z "$PUBLIC_SUBNET_AZ1" ]; then
  handle_error "Failed to create public subnet in AZ1"
fi

CREATED_RESOURCES+=("SUBNET:$PUBLIC_SUBNET_AZ1")
echo "Public subnet created in $AZ1 with ID: $PUBLIC_SUBNET_AZ1"

echo "Creating public subnet in $AZ2..."
PUBLIC_SUBNET_AZ2=$(aws ec2 create-subnet \
  --vpc-id "$VPC_ID" \
  --cidr-block 10.0.1.0/24 \
  --availability-zone "$AZ2" \
  --tag-specifications 'ResourceType=subnet,Tags=[{Key=Name,Value=Public-Subnet-
AZ2}]' \
  --query 'Subnet.SubnetId' \
  --output text)

if [ -z "$PUBLIC_SUBNET_AZ2" ]; then
  handle_error "Failed to create public subnet in AZ2"
fi

CREATED_RESOURCES+=("SUBNET:$PUBLIC_SUBNET_AZ2")
echo "Public subnet created in $AZ2 with ID: $PUBLIC_SUBNET_AZ2"

# Create private subnets
echo "Creating private subnet in $AZ1..."
PRIVATE_SUBNET_AZ1=$(aws ec2 create-subnet \
  --vpc-id "$VPC_ID" \
  --cidr-block 10.0.2.0/24 \
  --availability-zone "$AZ1" \
  --tag-specifications 'ResourceType=subnet,Tags=[{Key=Name,Value=Private-Subnet-
AZ1}]' \
  --query 'Subnet.SubnetId' \
  --output text)
```

```
if [ -z "$PRIVATE_SUBNET_AZ1" ]; then
  handle_error "Failed to create private subnet in AZ1"
fi

CREATED_RESOURCES+=("SUBNET:$PRIVATE_SUBNET_AZ1")
echo "Private subnet created in $AZ1 with ID: $PRIVATE_SUBNET_AZ1"

echo "Creating private subnet in $AZ2..."
PRIVATE_SUBNET_AZ2=$(aws ec2 create-subnet \
  --vpc-id "$VPC_ID" \
  --cidr-block 10.0.3.0/24 \
  --availability-zone "$AZ2" \
  --tag-specifications 'ResourceType=subnet,Tags=[{Key=Name,Value=Private-Subnet-
AZ2}]' \
  --query 'Subnet.SubnetId' \
  --output text)

if [ -z "$PRIVATE_SUBNET_AZ2" ]; then
  handle_error "Failed to create private subnet in AZ2"
fi

CREATED_RESOURCES+=("SUBNET:$PRIVATE_SUBNET_AZ2")
echo "Private subnet created in $AZ2 with ID: $PRIVATE_SUBNET_AZ2"

# Create Internet Gateway
echo "Creating Internet Gateway..."
IGW_ID=$(aws ec2 create-internet-gateway \
  --tag-specifications 'ResourceType=internet-gateway,Tags=[{Key=Name,Value=MyIGW}]'
 \
  --query 'InternetGateway.InternetGatewayId' \
  --output text)

if [ -z "$IGW_ID" ]; then
  handle_error "Failed to create Internet Gateway"
fi

CREATED_RESOURCES+=("INTERNET_GATEWAY:$IGW_ID")
echo "Internet Gateway created with ID: $IGW_ID"

# Attach Internet Gateway to VPC
echo "Attaching Internet Gateway to VPC..."
aws ec2 attach-internet-gateway --internet-gateway-id "$IGW_ID" --vpc-id "$VPC_ID"
 || handle_error "Failed to attach Internet Gateway to VPC"
```

```
# Create public route table
echo "Creating public route table..."
PUBLIC_RT=$(aws ec2 create-route-table \
  --vpc-id "$VPC_ID" \
  --tag-specifications 'ResourceType=route-table,Tags=[{Key=Name,Value=Public-RT}]'
 \
  --query 'RouteTable.RouteTableId' \
  --output text)

if [ -z "$PUBLIC_RT" ]; then
  handle_error "Failed to create public route table"
fi

CREATED_RESOURCES+=("ROUTE_TABLE:$PUBLIC_RT")
echo "Public route table created with ID: $PUBLIC_RT"

# Add route to Internet Gateway
echo "Adding route to Internet Gateway in public route table..."
aws ec2 create-route --route-table-id "$PUBLIC_RT" --destination-cidr-block
 0.0.0.0/0 --gateway-id "$IGW_ID" || handle_error "Failed to add route to Internet
 Gateway"

# Associate public subnets with public route table
echo "Associating public subnet in $AZ1 with public route table..."
PUBLIC_RT_ASSOC_1=$(aws ec2 associate-route-table --route-table-id "$PUBLIC_RT" --
subnet-id "$PUBLIC_SUBNET_AZ1" --query 'AssociationId' --output text)

if [ -z "$PUBLIC_RT_ASSOC_1" ]; then
  handle_error "Failed to associate public subnet in AZ1 with public route table"
fi

CREATED_RESOURCES+=("ROUTE_TABLE_ASSOCIATION:$PUBLIC_RT_ASSOC_1")

echo "Associating public subnet in $AZ2 with public route table..."
PUBLIC_RT_ASSOC_2=$(aws ec2 associate-route-table --route-table-id "$PUBLIC_RT" --
subnet-id "$PUBLIC_SUBNET_AZ2" --query 'AssociationId' --output text)

if [ -z "$PUBLIC_RT_ASSOC_2" ]; then
  handle_error "Failed to associate public subnet in AZ2 with public route table"
fi

CREATED_RESOURCES+=("ROUTE_TABLE_ASSOCIATION:$PUBLIC_RT_ASSOC_2")

# Create private route table
```

```
echo "Creating private route table..."
PRIVATE_RT=$(aws ec2 create-route-table \
  --vpc-id "$VPC_ID" \
  --tag-specifications 'ResourceType=route-table,Tags=[{Key=Name,Value=Private-RT}]'
 \
  --query 'RouteTable.RouteTableId' \
  --output text)

if [ -z "$PRIVATE_RT" ]; then
  handle_error "Failed to create private route table"
fi

CREATED_RESOURCES+=("ROUTE_TABLE:$PRIVATE_RT")
echo "Private route table created with ID: $PRIVATE_RT"

# Associate private subnets with private route table
echo "Associating private subnet in $AZ1 with private route table..."
PRIVATE_RT_ASSOC_1=$(aws ec2 associate-route-table --route-table-id "$PRIVATE_RT" --
subnet-id "$PRIVATE_SUBNET_AZ1" --query 'AssociationId' --output text)

if [ -z "$PRIVATE_RT_ASSOC_1" ]; then
  handle_error "Failed to associate private subnet in AZ1 with private route table"
fi

CREATED_RESOURCES+=("ROUTE_TABLE_ASSOCIATION:$PRIVATE_RT_ASSOC_1")

echo "Associating private subnet in $AZ2 with private route table..."
PRIVATE_RT_ASSOC_2=$(aws ec2 associate-route-table --route-table-id "$PRIVATE_RT" --
subnet-id "$PRIVATE_SUBNET_AZ2" --query 'AssociationId' --output text)

if [ -z "$PRIVATE_RT_ASSOC_2" ]; then
  handle_error "Failed to associate private subnet in AZ2 with private route table"
fi

CREATED_RESOURCES+=("ROUTE_TABLE_ASSOCIATION:$PRIVATE_RT_ASSOC_2")

# Allocate Elastic IP for NAT Gateway
echo "Allocating Elastic IP for NAT Gateway..."
EIP_ALLOC=$(aws ec2 allocate-address --domain vpc --query 'AllocationId' --output
 text)

if [ -z "$EIP_ALLOC" ]; then
  handle_error "Failed to allocate Elastic IP"
fi
```

```
CREATED_RESOURCES+=("EIP:$EIP_ALLOC")
echo "Elastic IP allocated with ID: $EIP_ALLOC"

# Create NAT Gateway
echo "Creating NAT Gateway in public subnet in $AZ1..."
NAT_GW=$(aws ec2 create-nat-gateway \
  --subnet-id "$PUBLIC_SUBNET_AZ1" \
  --allocation-id "$EIP_ALLOC" \
  --tag-specifications
 'ResourceType=natgateway,Tags=[{Key=Name,Value=MyNATGateway}]' \
  --query 'NatGateway.NatGatewayId' \
  --output text)

if [ -z "$NAT_GW" ]; then
  handle_error "Failed to create NAT Gateway"
fi

CREATED_RESOURCES+=("NAT_GATEWAY:$NAT_GW")
echo "NAT Gateway created with ID: $NAT_GW"

# Wait for NAT Gateway to be available
echo "Waiting for NAT Gateway to be available..."
aws ec2 wait nat-gateway-available --nat-gateway-ids "$NAT_GW" || handle_error "NAT
 Gateway did not become available"

# Add route to NAT Gateway in private route table
echo "Adding route to NAT Gateway in private route table..."
aws ec2 create-route --route-table-id "$PRIVATE_RT" --destination-cidr-block
 0.0.0.0/0 --nat-gateway-id "$NAT_GW" || handle_error "Failed to add route to NAT
 Gateway"

# Enable auto-assign public IP for instances in public subnets
echo "Enabling auto-assign public IP for instances in public subnet in $AZ1..."
aws ec2 modify-subnet-attribute --subnet-id "$PUBLIC_SUBNET_AZ1" --map-public-ip-on-
launch || handle_error "Failed to enable auto-assign public IP for public subnet in
 AZ1"

echo "Enabling auto-assign public IP for instances in public subnet in $AZ2..."
aws ec2 modify-subnet-attribute --subnet-id "$PUBLIC_SUBNET_AZ2" --map-public-ip-on-
launch || handle_error "Failed to enable auto-assign public IP for public subnet in
 AZ2"

# Create security group for web servers
```

```
echo "Creating security group for web servers..."
WEB_SG=$(aws ec2 create-security-group \
  --group-name "WebServerSG-$(date +%s)" \
  --description "Security group for web servers" \
  --vpc-id "$VPC_ID" \
  --query 'GroupId' \
  --output text)

if [ -z "$WEB_SG" ]; then
  handle_error "Failed to create security group for web servers"
fi

CREATED_RESOURCES+=("SECURITY_GROUP:$WEB_SG")
echo "Security group for web servers created with ID: $WEB_SG"

# Allow HTTP and HTTPS traffic
echo "Allowing HTTP traffic to web servers security group..."
aws ec2 authorize-security-group-ingress --group-id "$WEB_SG" --protocol tcp --port
 80 --cidr 0.0.0.0/0 || handle_error "Failed to allow HTTP traffic"

echo "Allowing HTTPS traffic to web servers security group..."
aws ec2 authorize-security-group-ingress --group-id "$WEB_SG" --protocol tcp --port
 443 --cidr 0.0.0.0/0 || handle_error "Failed to allow HTTPS traffic"

# Note: In a production environment, you should restrict the source IP ranges for
 security
echo "NOTE: In a production environment, you should restrict the source IP ranges
 for HTTP and HTTPS traffic"

# Create security group for database servers
echo "Creating security group for database servers..."
DB_SG=$(aws ec2 create-security-group \
  --group-name "DBServerSG-$(date +%s)" \
  --description "Security group for database servers" \
  --vpc-id "$VPC_ID" \
  --query 'GroupId' \
  --output text)

if [ -z "$DB_SG" ]; then
  handle_error "Failed to create security group for database servers"
fi

CREATED_RESOURCES+=("SECURITY_GROUP:$DB_SG")
echo "Security group for database servers created with ID: $DB_SG"
```

```
# Allow MySQL/Aurora traffic from web servers only
echo "Allowing MySQL/Aurora traffic from web servers to database servers..."
aws ec2 authorize-security-group-ingress --group-id "$DB_SG" --protocol tcp --
port 3306 --source-group "$WEB_SG" || handle_error "Failed to allow MySQL/Aurora
 traffic"

# Verify VPC configuration
echo "Verifying VPC configuration..."
echo "VPC:"
aws ec2 describe-vpcs --vpc-id "$VPC_ID" || handle_error "Failed to describe VPC"

echo "Subnets:"
aws ec2 describe-subnets --filters "Name=vpc-id,Values=$VPC_ID" || handle_error
 "Failed to describe subnets"

echo "Route tables:"
aws ec2 describe-route-tables --filters "Name=vpc-id,Values=$VPC_ID" || handle_error
 "Failed to describe route tables"

echo "Internet gateway:"
aws ec2 describe-internet-gateways --filters "Name=attachment.vpc-id,Values=$VPC_ID"
 || handle_error "Failed to describe Internet Gateway"

echo "NAT gateway:"
aws ec2 describe-nat-gateways --filter "Name=vpc-id,Values=$VPC_ID" || handle_error
 "Failed to describe NAT Gateway"

echo "Security groups:"
aws ec2 describe-security-groups --filters "Name=vpc-id,Values=$VPC_ID" ||
 handle_error "Failed to describe security groups"

echo ""
# Summary of created resources
echo "VPC creation completed successfully!"
echo "Summary of created resources:"
echo "- VPC: $VPC_ID"
echo "- Public Subnet in $AZ1: $PUBLIC_SUBNET_AZ1"
echo "- Public Subnet in $AZ2: $PUBLIC_SUBNET_AZ2"
echo "- Private Subnet in $AZ1: $PRIVATE_SUBNET_AZ1"
echo "- Private Subnet in $AZ2: $PRIVATE_SUBNET_AZ2"
echo "- Internet Gateway: $IGW_ID"
echo "- Public Route Table: $PUBLIC_RT"
echo "- Private Route Table: $PRIVATE_RT"
```

```
echo "- Elastic IP: $EIP_ALLOC"
echo "- NAT Gateway: $NAT_GW"
echo "- Web Servers Security Group: $WEB_SG"
echo "- Database Servers Security Group: $DB_SG"

# Deploy EC2 instances
echo ""
echo "Deploying EC2 instances..."

# Create key pair for SSH access
KEY_NAME="vpc-tutorial-key-$(date +%s)"
echo "Creating key pair $KEY_NAME..."
aws ec2 create-key-pair --key-name "$KEY_NAME" --query 'KeyMaterial' --output text >
 "${KEY_NAME}.pem" || handle_error "Failed to create key pair"
chmod 400 "${KEY_NAME}.pem"
echo "Key pair saved to ${KEY_NAME}.pem"
CREATED_RESOURCES+=("KEY_PAIR:$KEY_NAME")

# Get latest Amazon Linux 2 AMI
echo "Getting latest Amazon Linux 2 AMI..."
AMI_ID=$(aws ec2 describe-images --owners amazon \
  --filters "Name=name,Values=amzn2-ami-hvm-*-x86_64-gp2"
 "Name=state,Values=available" \
  --query "sort_by(Images, &CreationDate)[-1].ImageId" --output text) ||
 handle_error "Failed to get AMI"
echo "Using AMI: $AMI_ID"

# Launch web server in public subnet
echo "Launching web server in public subnet..."
WEB_INSTANCE=$(aws ec2 run-instances \
  --image-id "$AMI_ID" \
  --count 1 \
  --instance-type t2.micro \
  --key-name "$KEY_NAME" \
  --security-group-ids "$WEB_SG" \
  --subnet-id "$PUBLIC_SUBNET_AZ1" \
  --associate-public-ip-address \
  --user-data '#!/bin/bash
    yum update -y
    yum install -y httpd
    systemctl start httpd
    systemctl enable httpd
    echo "<h1>Hello from $(hostname -f) in the public subnet</h1>" > /var/www/html/
index.html' \
```

```
  --tag-specifications 'ResourceType=instance,Tags=[{Key=Name,Value=WebServer}]' \
  --query 'Instances[0].InstanceId' \
  --output text) || handle_error "Failed to launch web server"
echo "Web server instance created with ID: $WEB_INSTANCE"
CREATED_RESOURCES+=("INSTANCE:$WEB_INSTANCE")

# Wait for web server to be running
echo "Waiting for web server to be running..."
aws ec2 wait instance-running --instance-ids "$WEB_INSTANCE"

# Get web server public IP
WEB_PUBLIC_IP=$(aws ec2 describe-instances --instance-ids "$WEB_INSTANCE" \
  --query 'Reservations[0].Instances[0].PublicIpAddress' --output text)
echo "Web server public IP: $WEB_PUBLIC_IP"
echo "You can access the web server at: http://$WEB_PUBLIC_IP"

# Launch database server in private subnet
echo "Launching database server in private subnet..."
DB_INSTANCE=$(aws ec2 run-instances \
  --image-id "$AMI_ID" \
  --count 1 \
  --instance-type t2.micro \
  --key-name "$KEY_NAME" \
  --security-group-ids "$DB_SG" \
  --subnet-id "$PRIVATE_SUBNET_AZ1" \
  --user-data '#!/bin/bash
    yum update -y
    yum install -y mariadb-server
    systemctl start mariadb
    systemctl enable mariadb' \
  --tag-specifications 'ResourceType=instance,Tags=[{Key=Name,Value=DBServer}]' \
  --query 'Instances[0].InstanceId' \
  --output text) || handle_error "Failed to launch database server"
echo "Database server instance created with ID: $DB_INSTANCE"
CREATED_RESOURCES+=("INSTANCE:$DB_INSTANCE")

# Wait for database server to be running
echo "Waiting for database server to be running..."
aws ec2 wait instance-running --instance-ids "$DB_INSTANCE"

# Get database server private IP
DB_PRIVATE_IP=$(aws ec2 describe-instances --instance-ids "$DB_INSTANCE" \
  --query 'Reservations[0].Instances[0].PrivateIpAddress' --output text)
echo "Database server private IP: $DB_PRIVATE_IP"
```

```
echo "EC2 instances deployed successfully!"
echo "- Web Server (Public): $WEB_INSTANCE ($WEB_PUBLIC_IP)"
echo "- Database Server (Private): $DB_INSTANCE ($DB_PRIVATE_IP)"
echo ""
echo "Note: To connect to the web server: ssh -i ${KEY_NAME}.pem ec2-user@
$WEB_PUBLIC_IP"
echo "To connect to the database server, you must first connect to the web server,
 then use it as a bastion host."
echo "=========================================="
echo "CLEANUP CONFIRMATION"
echo "=========================================="
echo "Do you want to clean up all created resources? (y/n): "
read -r CLEANUP_CHOICE
if [[ "$CLEANUP_CHOICE" =~ ^[Yy]$ ]]; then
  echo "Cleaning up resources..."
  cleanup_resources
  echo "All resources have been cleaned up."
else
  echo "Resources will not be cleaned up. You can manually clean them up later."
fi

echo "Script completed at $(date)"
```

- Per informazioni dettagliate sull'API, consulta i seguenti argomenti nella documentazione di riferimento dei comandi della AWS CLI .
  - AllocateAddress
  - AssociateRouteTable
  - AttachInternetGateway
  - AuthorizeSecurityGroupIngress
  - CreateInternetGateway
  - CreateKeyPair
  - CreateNatGateway
  - CreateRoute
  - CreateRouteTable
  - CreateSecurityGroup
  - CreateSubnet

- CreateVpc
- DeleteInternetGateway
- DeleteKeyPair
- DeleteNatGateway
- DeleteRouteTable
- DeleteSecurityGroup
- DeleteSubnet
- DeleteVpc
- DescribeAvailabilityZones
- DescribeImages
- DescribeInstances
- DescribeInternetGateways
- DescribeNatGateways
- DescribeRouteTables
- DescribeSecurityGroups
- DescribeSubnets
- DescribeVpcs
- DetachInternetGateway
- DisassociateRouteTable
- ModifySubnetAttribute
- ModifyVpcAttribute
- ReleaseAddress
- RunInstances
- TerminateInstances

Nozioni di base sull'utilizzo del gateway di transito

L'esempio di codice seguente mostra come:

- Creare un gateway di transito con supporto DNS e impostazioni predefinite della tabella di routing
- Attendere che il gateway di transito risulti disponibile
- Collegane due VPCs al gateway di transito utilizzando sottoreti

- Attendere che i collegamenti VPC risultino disponibili
- Aggiungi percorsi VPCs attraverso il gateway di transito
- Verificare la connettività tra le risorse VPC
- Eseguire la pulizia delle risorse, inclusi route, collegamenti e gateway di transito

AWS CLI con lo script Bash

> **Note**
>
> C'è altro da fare. GitHub Trova l'esempio completo e scopri come eseguire la configurazione e l'esecuzione nel repository dei tutorial sugli esempi di codice per gli sviluppatori.

```
#!/bin/bash

# Amazon VPC Transit Gateway CLI Script
# This script demonstrates how to create a transit gateway and connect two VPCs
# Modified to work with older AWS CLI versions that don't support transit gateway
 wait commands

# Error handling
set -e
LOG_FILE="transit-gateway-tutorial.log"
exec > >(tee -a "$LOG_FILE") 2>&1

# Function to wait for transit gateway to be available
wait_for_tgw() {
  local tgw_id=$1
  echo "Waiting for Transit Gateway $tgw_id to become available..."

  while true; do
    status=$(aws ec2 describe-transit-gateways --transit-gateway-ids "$tgw_id" --
query "TransitGateways[0].State" --output text)
    echo "Current status: $status"

    if [ "$status" = "available" ]; then
      echo "Transit Gateway is now available"
      break
    fi
```

```
    echo "Waiting for transit gateway to become available. Current state: $status"
    sleep 10
  done
}

# Function to wait for transit gateway attachment to be available
wait_for_tgw_attachment() {
  local attachment_id=$1
  echo "Waiting for Transit Gateway Attachment $attachment_id to become
 available..."

  while true; do
    status=$(aws ec2 describe-transit-gateway-vpc-attachments --transit-gateway-
attachment-ids "$attachment_id" --query "TransitGatewayVpcAttachments[0].State" --
output text)
    echo "Current status: $status"

    if [ "$status" = "available" ]; then
      echo "Transit Gateway Attachment is now available"
      break
    fi

    echo "Waiting for transit gateway attachment to become available. Current state:
 $status"
    sleep 10
  done
}

# Function to wait for transit gateway attachment to be deleted
wait_for_tgw_attachment_deleted() {
  local attachment_id=$1
  echo "Waiting for Transit Gateway Attachment $attachment_id to be deleted..."

  while true; do
    # Check if the attachment still exists
    count=$(aws ec2 describe-transit-gateway-vpc-attachments --filters
 "Name=transit-gateway-attachment-id,Values=$attachment_id" --query
 "length(TransitGatewayVpcAttachments)" --output text)

    if [ "$count" = "0" ]; then
      echo "Transit Gateway Attachment has been deleted"
      break
    fi
```

```
    status=$(aws ec2 describe-transit-gateway-vpc-attachments --transit-gateway-
attachment-ids "$attachment_id" --query "TransitGatewayVpcAttachments[0].State" --
output text 2>/dev/null || echo "deleted")

    if [ "$status" = "deleted" ]; then
      echo "Transit Gateway Attachment has been deleted"
      break
    fi

    echo "Waiting for transit gateway attachment to be deleted. Current state:
 $status"
    sleep 10
  done
}

# Function to clean up resources
cleanup() {
  echo "Error occurred. Cleaning up resources..."

  # Delete resources in reverse order
  if [ ! -z "$TGW_ATTACHMENT_1_ID" ]; then
    echo "Deleting Transit Gateway VPC Attachment 1: $TGW_ATTACHMENT_1_ID"
    aws ec2 delete-transit-gateway-vpc-attachment --transit-gateway-attachment-id
 "$TGW_ATTACHMENT_1_ID" || true
    wait_for_tgw_attachment_deleted "$TGW_ATTACHMENT_1_ID" || true
  fi

  if [ ! -z "$TGW_ATTACHMENT_2_ID" ]; then
    echo "Deleting Transit Gateway VPC Attachment 2: $TGW_ATTACHMENT_2_ID"
    aws ec2 delete-transit-gateway-vpc-attachment --transit-gateway-attachment-id
 "$TGW_ATTACHMENT_2_ID" || true
    wait_for_tgw_attachment_deleted "$TGW_ATTACHMENT_2_ID" || true
  fi

  if [ ! -z "$TGW_ID" ]; then
    echo "Deleting Transit Gateway: $TGW_ID"
    aws ec2 delete-transit-gateway --transit-gateway-id "$TGW_ID" || true
  fi

  exit 1
}

# Set up trap for error handling
```

```
trap cleanup ERR

echo "=== Amazon VPC Transit Gateway Tutorial ==="
echo "This script will create a transit gateway and connect two VPCs"
echo ""

# Get a valid availability zone dynamically
echo "Getting available AZ in current region..."
AZ=$(aws ec2 describe-availability-zones --query "AvailabilityZones[0].ZoneName" --
output text)
echo "Using availability zone: $AZ"

# Check if VPCs exist
echo "Checking for existing VPCs..."
VPC1_ID=$(aws ec2 describe-vpcs --filters "Name=tag:Name,Values=VPC1" --query
 "Vpcs[0].VpcId" --output text)
VPC2_ID=$(aws ec2 describe-vpcs --filters "Name=tag:Name,Values=VPC2" --query
 "Vpcs[0].VpcId" --output text)

if [ "$VPC1_ID" == "None" ] || [ -z "$VPC1_ID" ]; then
  echo "Creating VPC1..."
  VPC1_ID=$(aws ec2 create-vpc --cidr-block 10.1.0.0/16 --tag-specifications
 'ResourceType=vpc,Tags=[{Key=Name,Value=VPC1}]' --query Vpc.VpcId --output text)
  echo "Created VPC1: $VPC1_ID"

  # Create a subnet in VPC1
  echo "Creating subnet in VPC1..."
  SUBNET1_ID=$(aws ec2 create-subnet --vpc-id "$VPC1_ID" --cidr-
block 10.1.0.0/24 --availability-zone "$AZ" --tag-specifications
 'ResourceType=subnet,Tags=[{Key=Name,Value=VPC1-Subnet}]' --query Subnet.SubnetId
 --output text)
  echo "Created subnet in VPC1: $SUBNET1_ID"
else
  echo "Using existing VPC1: $VPC1_ID"
  SUBNET1_ID=$(aws ec2 describe-subnets --filters "Name=vpc-id,Values=$VPC1_ID" --
query "Subnets[0].SubnetId" --output text)
  if [ "$SUBNET1_ID" == "None" ] || [ -z "$SUBNET1_ID" ]; then
    echo "Creating subnet in VPC1..."
    SUBNET1_ID=$(aws ec2 create-subnet --vpc-id "$VPC1_ID" --cidr-
block 10.1.0.0/24 --availability-zone "$AZ" --tag-specifications
 'ResourceType=subnet,Tags=[{Key=Name,Value=VPC1-Subnet}]' --query Subnet.SubnetId
 --output text)
    echo "Created subnet in VPC1: $SUBNET1_ID"
  else
```

```
    echo "Using existing subnet in VPC1: $SUBNET1_ID"
  fi
fi

if [ "$VPC2_ID" == "None" ] || [ -z "$VPC2_ID" ]; then
  echo "Creating VPC2..."
  VPC2_ID=$(aws ec2 create-vpc --cidr-block 10.2.0.0/16 --tag-specifications
 'ResourceType=vpc,Tags=[{Key=Name,Value=VPC2}]' --query Vpc.VpcId --output text)
  echo "Created VPC2: $VPC2_ID"

  # Create a subnet in VPC2
  echo "Creating subnet in VPC2..."
  SUBNET2_ID=$(aws ec2 create-subnet --vpc-id "$VPC2_ID" --cidr-
block 10.2.0.0/24 --availability-zone "$AZ" --tag-specifications
 'ResourceType=subnet,Tags=[{Key=Name,Value=VPC2-Subnet}]' --query Subnet.SubnetId
 --output text)
  echo "Created subnet in VPC2: $SUBNET2_ID"
else
  echo "Using existing VPC2: $VPC2_ID"
  SUBNET2_ID=$(aws ec2 describe-subnets --filters "Name=vpc-id,Values=$VPC2_ID" --
query "Subnets[0].SubnetId" --output text)
  if [ "$SUBNET2_ID" == "None" ] || [ -z "$SUBNET2_ID" ]; then
    echo "Creating subnet in VPC2..."
    SUBNET2_ID=$(aws ec2 create-subnet --vpc-id "$VPC2_ID" --cidr-
block 10.2.0.0/24 --availability-zone "$AZ" --tag-specifications
 'ResourceType=subnet,Tags=[{Key=Name,Value=VPC2-Subnet}]' --query Subnet.SubnetId
 --output text)
    echo "Created subnet in VPC2: $SUBNET2_ID"
  else
    echo "Using existing subnet in VPC2: $SUBNET2_ID"
  fi
fi

# Get route tables for each VPC
RTB1_ID=$(aws ec2 describe-route-tables --filters "Name=vpc-id,Values=$VPC1_ID" --
query "RouteTables[0].RouteTableId" --output text)
RTB2_ID=$(aws ec2 describe-route-tables --filters "Name=vpc-id,Values=$VPC2_ID" --
query "RouteTables[0].RouteTableId" --output text)

echo "Route table for VPC1: $RTB1_ID"
echo "Route table for VPC2: $RTB2_ID"

# Step 1: Create the transit gateway
echo "Creating Transit Gateway..."
```

```
TGW_ID=$(aws ec2 create-transit-gateway \
  --description "My Transit Gateway" \
  --options
 AmazonSideAsn=64512,AutoAcceptSharedAttachments=disable,DefaultRouteTableAssociation=enable
 \
  --tag-specifications 'ResourceType=transit-
gateway,Tags=[{Key=Name,Value=MyTransitGateway}]' \
  --query TransitGateway.TransitGatewayId \
  --output text)

echo "Created Transit Gateway: $TGW_ID"

# Wait for the transit gateway to become available
wait_for_tgw "$TGW_ID"

# Step 2: Attach VPCs to the transit gateway
echo "Attaching VPC1 to Transit Gateway..."
TGW_ATTACHMENT_1_ID=$(aws ec2 create-transit-gateway-vpc-attachment \
  --transit-gateway-id "$TGW_ID" \
  --vpc-id "$VPC1_ID" \
  --subnet-ids "$SUBNET1_ID" \
  --tag-specifications 'ResourceType=transit-gateway-
attachment,Tags=[{Key=Name,Value=VPC1-Attachment}]' \
  --query TransitGatewayVpcAttachment.TransitGatewayAttachmentId \
  --output text)

echo "Created Transit Gateway VPC Attachment for VPC1: $TGW_ATTACHMENT_1_ID"

echo "Attaching VPC2 to Transit Gateway..."
TGW_ATTACHMENT_2_ID=$(aws ec2 create-transit-gateway-vpc-attachment \
  --transit-gateway-id "$TGW_ID" \
  --vpc-id "$VPC2_ID" \
  --subnet-ids "$SUBNET2_ID" \
  --tag-specifications 'ResourceType=transit-gateway-
attachment,Tags=[{Key=Name,Value=VPC2-Attachment}]' \
  --query TransitGatewayVpcAttachment.TransitGatewayAttachmentId \
  --output text)

echo "Created Transit Gateway VPC Attachment for VPC2: $TGW_ATTACHMENT_2_ID"

# Wait for the attachments to become available
wait_for_tgw_attachment "$TGW_ATTACHMENT_1_ID"
wait_for_tgw_attachment "$TGW_ATTACHMENT_2_ID"
```

```
# Step 3: Add routes between the transit gateway and VPCs
echo "Adding route from VPC1 to VPC2 via Transit Gateway..."
aws ec2 create-route \
  --route-table-id "$RTB1_ID" \
  --destination-cidr-block 10.2.0.0/16 \
  --transit-gateway-id "$TGW_ID"

echo "Adding route from VPC2 to VPC1 via Transit Gateway..."
aws ec2 create-route \
  --route-table-id "$RTB2_ID" \
  --destination-cidr-block 10.1.0.0/16 \
  --transit-gateway-id "$TGW_ID"

echo "Routes added successfully"

# Step 4: Display information for testing
echo ""
echo "=== Transit Gateway Setup Complete ==="
echo "Transit Gateway ID: $TGW_ID"
echo "VPC1 ID: $VPC1_ID"
echo "VPC2 ID: $VPC2_ID"
echo ""
echo "To test connectivity:"
echo "1. Launch an EC2 instance in each VPC"
echo "2. Configure security groups to allow ICMP traffic"
echo "3. Connect to one instance and ping the other instance's private IP"
echo ""

# Prompt user before cleanup
read -p "Press Enter to view created resources, or Ctrl+C to exit without
 cleanup..."

echo ""
echo "=== Resources Created ==="
echo "Transit Gateway: $TGW_ID"
echo "VPC1: $VPC1_ID"
echo "VPC2: $VPC2_ID"
echo "Subnet in VPC1: $SUBNET1_ID"
echo "Subnet in VPC2: $SUBNET2_ID"
echo "Transit Gateway Attachment for VPC1: $TGW_ATTACHMENT_1_ID"
echo "Transit Gateway Attachment for VPC2: $TGW_ATTACHMENT_2_ID"
echo ""

read -p "Do you want to clean up these resources? (y/n): " CLEANUP_CONFIRM
```

```
if [[ $CLEANUP_CONFIRM == "y" || $CLEANUP_CONFIRM == "Y" ]]; then
  echo "Starting cleanup..."

  # Delete routes
  echo "Deleting routes..."
  aws ec2 delete-route --route-table-id "$RTB1_ID" --destination-cidr-block
 10.2.0.0/16
  aws ec2 delete-route --route-table-id "$RTB2_ID" --destination-cidr-block
 10.1.0.0/16

  # Delete transit gateway attachments
  echo "Deleting Transit Gateway VPC Attachment for VPC1: $TGW_ATTACHMENT_1_ID"
  aws ec2 delete-transit-gateway-vpc-attachment --transit-gateway-attachment-id
 "$TGW_ATTACHMENT_1_ID"

  echo "Deleting Transit Gateway VPC Attachment for VPC2: $TGW_ATTACHMENT_2_ID"
  aws ec2 delete-transit-gateway-vpc-attachment --transit-gateway-attachment-id
 "$TGW_ATTACHMENT_2_ID"

  # Wait for attachments to be deleted
  wait_for_tgw_attachment_deleted "$TGW_ATTACHMENT_1_ID"
  wait_for_tgw_attachment_deleted "$TGW_ATTACHMENT_2_ID"

  # Delete transit gateway
  echo "Deleting Transit Gateway: $TGW_ID"
  aws ec2 delete-transit-gateway --transit-gateway-id "$TGW_ID"

  echo "Cleanup completed successfully"
else
  echo "Skipping cleanup. Resources will continue to incur charges until manually
 deleted."
fi

echo "Tutorial completed. See $LOG_FILE for detailed logs."
```

- Per informazioni dettagliate sull'API, consulta i seguenti argomenti nella documentazione di riferimento dei comandi della AWS CLI .
  - CreateRoute
  - CreateSubnet
  - CreateTransitGateway
  - CreateTransitGatewayVpcAttachment

- CreateVpc
- DeleteRoute
- DeleteTransitGateway
- DeleteTransitGatewayVpcAttachment
- DescribeAvailabilityZones
- DescribeRouteTables
- DescribeSubnets
- DescribeTransitGatewayAttachments
- DescribeTransitGatewayVpcAttachments
- DescribeTransitGateways
- DescribeVpcs

Nozioni di base su Gestione indirizzi IP (IPAM) di Amazon VPC

L'esempio di codice seguente mostra come:

- Impostare e configurare Gestione indirizzi IP (IPAM) di Amazon VPC con la CLI.
- Creare una risorsa IPAM con Regioni operative, ad esempio us-east-1, us-west-2.
- Recuperare l'ID dell'ambito privato per la risorsa IPAM.
- Crea una struttura gerarchica di IPv4 pool (pool di primo livello, regionali e di sviluppo).
- Eseguire il provisioning degli intervalli CIDR a ciascun pool, ad esempio 10.0.0.0/8, 10.0.0.0/16, 10.0.0.0/24.
- Creare un VPC con un CIDR allocato da un pool IPAM.
- Verificare le allocazioni dei pool IPAM e la creazione di VPC.
- Risolvere i problemi comuni come errori di autorizzazione, errori di allocazione CIDR e violazioni delle dipendenze.
- Pulisci le risorse IPAM (VPC CIDRs, pool e IPAM) per evitare addebiti inutili.
- Esplorare le fasi successive delle funzionalità IPAM avanzate.

AWS CLI con lo script Bash

> **Note**
>
> C'è altro da fare. GitHub Trova l'esempio completo e scopri come eseguire la configurazione e l'esecuzione nel repository dei tutorial sugli esempi di codice per gli sviluppatori.

```
#!/bin/bash

# IPAM Getting Started CLI Script - Version 7
# This script creates an IPAM, creates a hierarchy of IP address pools, and
 allocates a CIDR to a VPC
# Fixed to correctly identify the private scope ID, wait for resources to be
 available, add locale to development pool,
# use the correct parameter names for VPC creation, and wait for CIDR provisioning
 to complete

# Set up logging
LOG_FILE="ipam_script.log"
exec > >(tee -a "$LOG_FILE") 2>&1

echo "Starting IPAM setup script at $(date)"
echo "All commands and outputs will be logged to $LOG_FILE"

# Function to handle errors
handle_error() {
    echo "ERROR: $1"
    echo "Attempting to clean up resources..."
    cleanup_resources
    exit 1
}

# Function to clean up resources
cleanup_resources() {
    echo ""
    echo "========================================="
    echo "RESOURCES CREATED:"
    echo "========================================="
```

```
    if [ -n "$VPC_ID" ]; then
        echo "VPC: $VPC_ID"
    fi

    if [ -n "$DEV_POOL_ID" ]; then
        echo "Development Pool: $DEV_POOL_ID"
    fi

    if [ -n "$REGIONAL_POOL_ID" ]; then
        echo "Regional Pool: $REGIONAL_POOL_ID"
    fi

    if [ -n "$TOP_POOL_ID" ]; then
        echo "Top-level Pool: $TOP_POOL_ID"
    fi

    if [ -n "$IPAM_ID" ]; then
        echo "IPAM: $IPAM_ID"
    fi

    echo ""
    echo "========================================="
    echo "CLEANUP CONFIRMATION"
    echo "========================================="
    echo "Do you want to clean up all created resources? (y/n): "
    read -r CLEANUP_CHOICE

    if [[ "$CLEANUP_CHOICE" =~ ^[Yy]$ ]]; then
        echo "Starting cleanup..."

        # Delete resources in reverse order of creation to handle dependencies

        if [ -n "$VPC_ID" ]; then
            echo "Deleting VPC: $VPC_ID"
            aws ec2 delete-vpc --vpc-id "$VPC_ID" || echo "Failed to delete VPC"
            echo "Waiting for VPC to be deleted..."
            sleep 10
        fi

        if [ -n "$DEV_POOL_ID" ]; then
            echo "Deleting Development Pool: $DEV_POOL_ID"
            # First deprovision any CIDRs from the pool
            CIDRS=$(aws ec2 get-ipam-pool-cidrs --ipam-pool-id "$DEV_POOL_ID" --
query 'IpamPoolCidrs[].Cidr' --output text)
```

```
            for CIDR in $CIDRS; do
                echo "Deprovisioning CIDR $CIDR from Development Pool"
                aws ec2 deprovision-ipam-pool-cidr --ipam-pool-id "$DEV_POOL_ID" --
cidr "$CIDR" || echo "Failed to deprovision CIDR $CIDR"
                sleep 5
            done
            aws ec2 delete-ipam-pool --ipam-pool-id "$DEV_POOL_ID" || echo "Failed
 to delete Development Pool"
            echo "Waiting for Development Pool to be deleted..."
            sleep 10
        fi

        if [ -n "$REGIONAL_POOL_ID" ]; then
            echo "Deleting Regional Pool: $REGIONAL_POOL_ID"
            # First deprovision any CIDRs from the pool
            CIDRS=$(aws ec2 get-ipam-pool-cidrs --ipam-pool-id "$REGIONAL_POOL_ID"
 --query 'IpamPoolCidrs[].Cidr' --output text)
            for CIDR in $CIDRS; do
                echo "Deprovisioning CIDR $CIDR from Regional Pool"
                aws ec2 deprovision-ipam-pool-cidr --ipam-pool-id
 "$REGIONAL_POOL_ID" --cidr "$CIDR" || echo "Failed to deprovision CIDR $CIDR"
                sleep 5
            done
            aws ec2 delete-ipam-pool --ipam-pool-id "$REGIONAL_POOL_ID" || echo
 "Failed to delete Regional Pool"
            echo "Waiting for Regional Pool to be deleted..."
            sleep 10
        fi

        if [ -n "$TOP_POOL_ID" ]; then
            echo "Deleting Top-level Pool: $TOP_POOL_ID"
            # First deprovision any CIDRs from the pool
            CIDRS=$(aws ec2 get-ipam-pool-cidrs --ipam-pool-id "$TOP_POOL_ID" --
query 'IpamPoolCidrs[].Cidr' --output text)
            for CIDR in $CIDRS; do
                echo "Deprovisioning CIDR $CIDR from Top-level Pool"
                aws ec2 deprovision-ipam-pool-cidr --ipam-pool-id "$TOP_POOL_ID" --
cidr "$CIDR" || echo "Failed to deprovision CIDR $CIDR"
                sleep 5
            done
            aws ec2 delete-ipam-pool --ipam-pool-id "$TOP_POOL_ID" || echo "Failed
 to delete Top-level Pool"
            echo "Waiting for Top-level Pool to be deleted..."
            sleep 10
```

```
        fi

        if [ -n "$IPAM_ID" ]; then
            echo "Deleting IPAM: $IPAM_ID"
            aws ec2 delete-ipam --ipam-id "$IPAM_ID" || echo "Failed to delete IPAM"
        fi

        echo "Cleanup completed."
    else
        echo "Cleanup skipped. Resources will remain in your account."
    fi
}

# Function to wait for a pool to be in the 'create-complete' state
wait_for_pool() {
    local pool_id=$1
    local max_attempts=30
    local attempt=1
    local state=""

    echo "Waiting for pool $pool_id to be available..."

    while [ $attempt -le $max_attempts ]; do
        state=$(aws ec2 describe-ipam-pools --ipam-pool-ids "$pool_id" --query
 'IpamPools[0].State' --output text)

        if [ "$state" = "create-complete" ]; then
            echo "Pool $pool_id is now available (state: $state)"
            return 0
        fi

        echo "Attempt $attempt/$max_attempts: Pool $pool_id is in state: $state.
 Waiting..."
        sleep 10
        ((attempt++))
    done

    echo "Timed out waiting for pool $pool_id to be available"
    return 1
}

# Function to wait for a CIDR to be fully provisioned
wait_for_cidr_provisioning() {
    local pool_id=$1
```

```
    local cidr=$2
    local max_attempts=30
    local attempt=1
    local state=""

    echo "Waiting for CIDR $cidr to be fully provisioned in pool $pool_id..."

    while [ $attempt -le $max_attempts ]; do
        state=$(aws ec2 get-ipam-pool-cidrs --ipam-pool-id "$pool_id" --query
 "IpamPoolCidrs[?Cidr=='$cidr'].State" --output text)

        if [ "$state" = "provisioned" ]; then
            echo "CIDR $cidr is now fully provisioned (state: $state)"
            return 0
        fi

        echo "Attempt $attempt/$max_attempts: CIDR $cidr is in state: $state.
 Waiting..."
        sleep 10
        ((attempt++))
    done

    echo "Timed out waiting for CIDR $cidr to be provisioned"
    return 1
}

# Step 1: Create an IPAM
echo "Creating IPAM..."
IPAM_RESULT=$(aws ec2 create-ipam \
    --description "My IPAM" \
    --operating-regions RegionName=us-east-1 RegionName=us-west-2)

if [ $? -ne 0 ]; then
    handle_error "Failed to create IPAM"
fi

IPAM_ID=$(echo "$IPAM_RESULT" | grep -o '"IpamId": "[^"]*' | cut -d'"' -f4)
echo "IPAM created with ID: $IPAM_ID"

# Wait for IPAM to be created and available
echo "Waiting for IPAM to be available..."
sleep 20

# Step 2: Get the IPAM Scope ID - FIXED to correctly identify the private scope
```

```
echo "Getting IPAM Scope ID..."
SCOPE_RESULT=$(aws ec2 describe-ipams --ipam-id "$IPAM_ID")

if [ $? -ne 0 ]; then
    handle_error "Failed to get IPAM details"
fi

# Extract the private scope ID directly from the IPAM details
PRIVATE_SCOPE_ID=$(echo "$SCOPE_RESULT" | grep -o '"PrivateDefaultScopeId": "[^"]*'
 | cut -d'"' -f4)
echo "Private Scope ID: $PRIVATE_SCOPE_ID"

if [ -z "$PRIVATE_SCOPE_ID" ]; then
    handle_error "Failed to get Private Scope ID"
fi

# Step 3: Create a Top-Level IPv4 Pool
echo "Creating Top-level IPv4 Pool..."
TOP_POOL_RESULT=$(aws ec2 create-ipam-pool \
    --ipam-scope-id "$PRIVATE_SCOPE_ID" \
    --address-family ipv4 \
    --description "Top-level pool")

if [ $? -ne 0 ]; then
    handle_error "Failed to create Top-level Pool"
fi

TOP_POOL_ID=$(echo "$TOP_POOL_RESULT" | grep -o '"IpamPoolId": "[^"]*' | cut -d'"' -
f4)
echo "Top-level Pool created with ID: $TOP_POOL_ID"

# Wait for the top-level pool to be available
if ! wait_for_pool "$TOP_POOL_ID"; then
    handle_error "Top-level Pool did not become available in time"
fi

# Provision CIDR to the top-level pool
echo "Provisioning CIDR to Top-level Pool..."
TOP_POOL_CIDR="10.0.0.0/8"
PROVISION_RESULT=$(aws ec2 provision-ipam-pool-cidr \
    --ipam-pool-id "$TOP_POOL_ID" \
    --cidr "$TOP_POOL_CIDR")

if [ $? -ne 0 ]; then
```

```
    handle_error "Failed to provision CIDR to Top-level Pool"
fi

echo "$PROVISION_RESULT"

# Wait for the CIDR to be fully provisioned
if ! wait_for_cidr_provisioning "$TOP_POOL_ID" "$TOP_POOL_CIDR"; then
    handle_error "CIDR provisioning to Top-level Pool did not complete in time"
fi

# Step 4: Create a Regional IPv4 Pool
echo "Creating Regional IPv4 Pool..."
REGIONAL_POOL_RESULT=$(aws ec2 create-ipam-pool \
    --ipam-scope-id "$PRIVATE_SCOPE_ID" \
    --source-ipam-pool-id "$TOP_POOL_ID" \
    --locale us-east-1 \
    --address-family ipv4 \
    --description "Regional pool in us-east-1")

if [ $? -ne 0 ]; then
    handle_error "Failed to create Regional Pool"
fi

REGIONAL_POOL_ID=$(echo "$REGIONAL_POOL_RESULT" | grep -o '"IpamPoolId": "[^"]*' |
 cut -d'"' -f4)
echo "Regional Pool created with ID: $REGIONAL_POOL_ID"

# Wait for the regional pool to be available
if ! wait_for_pool "$REGIONAL_POOL_ID"; then
    handle_error "Regional Pool did not become available in time"
fi

# Provision CIDR to the regional pool
echo "Provisioning CIDR to Regional Pool..."
REGIONAL_POOL_CIDR="10.0.0.0/16"
PROVISION_RESULT=$(aws ec2 provision-ipam-pool-cidr \
    --ipam-pool-id "$REGIONAL_POOL_ID" \
    --cidr "$REGIONAL_POOL_CIDR")

if [ $? -ne 0 ]; then
    handle_error "Failed to provision CIDR to Regional Pool"
fi

echo "$PROVISION_RESULT"
```

```
# Wait for the CIDR to be fully provisioned
if ! wait_for_cidr_provisioning "$REGIONAL_POOL_ID" "$REGIONAL_POOL_CIDR"; then
    handle_error "CIDR provisioning to Regional Pool did not complete in time"
fi

# Step 5: Create a Development IPv4 Pool - FIXED to include locale
echo "Creating Development IPv4 Pool..."
DEV_POOL_RESULT=$(aws ec2 create-ipam-pool \
    --ipam-scope-id "$PRIVATE_SCOPE_ID" \
    --source-ipam-pool-id "$REGIONAL_POOL_ID" \
    --locale us-east-1 \
    --address-family ipv4 \
    --description "Development pool")

if [ $? -ne 0 ]; then
    handle_error "Failed to create Development Pool"
fi

DEV_POOL_ID=$(echo "$DEV_POOL_RESULT" | grep -o '"IpamPoolId": "[^"]*' | cut -d'"' -
f4)
echo "Development Pool created with ID: $DEV_POOL_ID"

# Wait for the development pool to be available
if ! wait_for_pool "$DEV_POOL_ID"; then
    handle_error "Development Pool did not become available in time"
fi

# Provision CIDR to the development pool
echo "Provisioning CIDR to Development Pool..."
DEV_POOL_CIDR="10.0.0.0/24"
PROVISION_RESULT=$(aws ec2 provision-ipam-pool-cidr \
    --ipam-pool-id "$DEV_POOL_ID" \
    --cidr "$DEV_POOL_CIDR")

if [ $? -ne 0 ]; then
    handle_error "Failed to provision CIDR to Development Pool"
fi

echo "$PROVISION_RESULT"

# Wait for the CIDR to be fully provisioned
if ! wait_for_cidr_provisioning "$DEV_POOL_ID" "$DEV_POOL_CIDR"; then
    handle_error "CIDR provisioning to Development Pool did not complete in time"
```

```
fi

# Step 6: Create a VPC Using an IPAM Pool CIDR - FIXED to use the correct parameter
 names and a smaller netmask length
echo "Creating VPC using IPAM Pool CIDR..."
VPC_RESULT=$(aws ec2 create-vpc \
    --ipv4-ipam-pool-id "$DEV_POOL_ID" \
    --ipv4-netmask-length 26 \
    --tag-specifications 'ResourceType=vpc,Tags=[{Key=Name,Value=IPAM-VPC}]')

if [ $? -ne 0 ]; then
    handle_error "Failed to create VPC"
fi

VPC_ID=$(echo "$VPC_RESULT" | grep -o '"VpcId": "[^"]*' | cut -d'"' -f4)
echo "VPC created with ID: $VPC_ID"

# Step 7: Verify the IPAM Pool Allocation
echo "Verifying IPAM Pool Allocation..."
ALLOCATION_RESULT=$(aws ec2 get-ipam-pool-allocations \
    --ipam-pool-id "$DEV_POOL_ID")

if [ $? -ne 0 ]; then
    handle_error "Failed to verify IPAM Pool Allocation"
fi

echo "IPAM Pool Allocation verified:"
echo "$ALLOCATION_RESULT" | grep -A 5 "Allocations"

echo ""
echo "IPAM setup completed successfully!"
echo ""

# Prompt for cleanup
cleanup_resources

echo "Script completed at $(date)"
exit 0
```

- Per informazioni dettagliate sull'API, consulta i seguenti argomenti nella documentazione di riferimento dei comandi della AWS CLI .
  - CreateIpam

- CreateIpamPool
- CreateVpc
- DeleteIpam
- DeleteIpamPool
- DeleteVpc
- DeprovisionIpamPoolCidr
- DescribeIpamPools
- DescribeIpams
- DescribeVpcs
- GetIpamPoolAllocations
- GetIpamPoolCidrs
- ProvisionIpamPoolCidr

HealthImaging esempi di utilizzo AWS CLI con lo script Bash

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando lo script AWS Command Line Interface with Bash con. HealthImaging

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`CreateDatastore`

Il seguente esempio di codice mostra come usare`CreateDatastore`.

AWS CLI con lo script Bash

```
###############################################################################
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
###############################################################################
function errecho() {
  printf "%s\n" "$*" 1>&2
}

###############################################################################
# function imaging_create_datastore
#
# This function creates an AWS HealthImaging data store for importing DICOM P10
 files.
#
# Parameters:
#       -n data_store_name - The name of the data store.
#
# Returns:
#       The datastore ID.
#    And:
#       0 - If successful.
#       1 - If it fails.
###############################################################################
function imaging_create_datastore() {
  local datastore_name response
  local option OPTARG # Required to use getopts command in a function.

  # bashsupport disable=BP5008
  function usage() {
    echo "function imaging_create_datastore"
    echo "Creates an AWS HealthImaging data store for importing DICOM P10 files."
    echo "  -n data_store_name - The name of the data store."
    echo ""
  }

  # Retrieve the calling parameters.
  while getopts "n:h" option; do
    case "${option}" in
      n) datastore_name="${OPTARG}" ;;
      h)
        usage
```

```
        return 0
        ;;
      \?)
        echo "Invalid parameter"
        usage
        return 1
        ;;
    esac
  done
  export OPTIND=1

  if [[ -z "$datastore_name" ]]; then
    errecho "ERROR: You must provide a data store name with the -n parameter."
    usage
    return 1
  fi

  response=$(aws medical-imaging create-datastore \
    --datastore-name "$datastore_name" \
    --output text \
    --query 'datastoreId')

  local error_code=${?}

  if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports medical-imaging create-datastore operation failed.
$response"
    return 1
  fi

  echo "$response"

  return 0
}
```

- Per i dettagli sull'API, vedere CreateDatastorein AWS CLI Command Reference.

> **Note**
>
> C'è altro su GitHub. Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codice AWS.

`DeleteDatastore`

Il seguente esempio di codice mostra come usare`DeleteDatastore`.

AWS CLI con lo script Bash

```
###############################################################################
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
###############################################################################
function errecho() {
  printf "%s\n" "$*" 1>&2
}

###############################################################################
# function imaging_delete_datastore
#
# This function deletes an AWS HealthImaging data store.
#
# Parameters:
#       -i datastore_id - The ID of the data store.
#
# Returns:
#       0 - If successful.
#       1 - If it fails.
###############################################################################
function imaging_delete_datastore() {
  local datastore_id response
  local option OPTARG # Required to use getopts command in a function.

  # bashsupport disable=BP5008
  function usage() {
    echo "function imaging_delete_datastore"
    echo "Deletes an AWS HealthImaging data store."
    echo "  -i datastore_id - The ID of the data store."
    echo ""
```

```
  }

  # Retrieve the calling parameters.
  while getopts "i:h" option; do
    case "${option}" in
      i) datastore_id="${OPTARG}" ;;
      h)
        usage
        return 0
        ;;
      \?)
        echo "Invalid parameter"
        usage
        return 1
        ;;
    esac
  done
  export OPTIND=1

  if [[ -z "$datastore_id" ]]; then
    errecho "ERROR: You must provide a data store ID with the -i parameter."
    usage
    return 1
  fi

  response=$(aws medical-imaging delete-datastore \
    --datastore-id "$datastore_id")

  local error_code=${?}

  if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports medical-imaging delete-datastore operation failed.
$response"
    return 1
  fi

  return 0
}
```

- Per i dettagli sull'API, vedere DeleteDatastorein AWS CLI Command Reference.

> **Note**
>
> C'è altro su GitHub. Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codice AWS.

`GetDatastore`

Il seguente esempio di codice mostra come usare`GetDatastore`.

AWS CLI con lo script Bash

```
###############################################################################
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
###############################################################################
function errecho() {
  printf "%s\n" "$*" 1>&2
}

###############################################################################
# function imaging_get_datastore
#
# Get a data store's properties.
#
# Parameters:
#       -i data_store_id - The ID of the data store.
#
# Returns:
#       [datastore_name, datastore_id, datastore_status, datastore_arn,  created_at,
 updated_at]
#    And:
#       0 - If successful.
#       1 - If it fails.
###############################################################################
function imaging_get_datastore() {
  local datastore_id option OPTARG # Required to use getopts command in a function.
  local error_code
  # bashsupport disable=BP5008
  function usage() {
    echo "function imaging_get_datastore"
    echo "Gets a data store's properties."
```

```
    echo "  -i datastore_id - The ID of the data store."
    echo ""
  }

  # Retrieve the calling parameters.
  while getopts "i:h" option; do
    case "${option}" in
      i) datastore_id="${OPTARG}" ;;
      h)
        usage
        return 0
        ;;
      \?)
        echo "Invalid parameter"
        usage
        return 1
        ;;
    esac
  done
  export OPTIND=1

  if [[ -z "$datastore_id" ]]; then
    errecho "ERROR: You must provide a data store ID with the -i parameter."
    usage
    return 1
  fi

  local response

  response=$(
    aws medical-imaging get-datastore \
      --datastore-id "$datastore_id" \
      --output text \
      --query "[ datastoreProperties.datastoreName,
datastoreProperties.datastoreId, datastoreProperties.datastoreStatus,
datastoreProperties.datastoreArn,  datastoreProperties.createdAt,
datastoreProperties.updatedAt]"
  )
  error_code=${?}

  if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports list-datastores operation failed.$response"
    return 1
```

```
  fi

  echo "$response"

  return 0
}
```

- Per i dettagli sull'API, vedere GetDatastorein AWS CLI Command Reference.

> **Note**
>
> C'è altro su GitHub. Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codice AWS.

`ListDatastores`

Il seguente esempio di codice mostra come usare`ListDatastores`.

AWS CLI con lo script Bash

```
###############################################################################
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
###############################################################################
function errecho() {
  printf "%s\n" "$*" 1>&2
}

###############################################################################
# function imaging_list_datastores
#
# List the HealthImaging data stores in the account.
#
# Returns:
#       [[datastore_name, datastore_id, datastore_status]]
#    And:
#       0 - If successful.
#       1 - If it fails.
###############################################################################
function imaging_list_datastores() {
```

```
  local option OPTARG # Required to use getopts command in a function.
  local error_code
  # bashsupport disable=BP5008
  function usage() {
    echo "function imaging_list_datastores"
    echo "Lists the AWS HealthImaging data stores in the account."
    echo ""
  }

  # Retrieve the calling parameters.
  while getopts "h" option; do
    case "${option}" in
      h)
        usage
        return 0
        ;;
      \?)
        echo "Invalid parameter"
        usage
        return 1
        ;;
    esac
  done
  export OPTIND=1

  local response
  response=$(aws medical-imaging list-datastores \
    --output text \
    --query "datastoreSummaries[*][datastoreName, datastoreId, datastoreStatus]")
  error_code=${?}

  if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports list-datastores operation failed.$response"
    return 1
  fi

  echo "$response"

  return 0
}
```

- Per i dettagli sull'API, vedere ListDatastoresin AWS CLI Command Reference.

> **Note**
>
> C'è altro su GitHub. Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codice AWS.

Esempi IAM che utilizzano AWS CLI lo script Bash

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando lo script AWS Command Line Interface with Bash con IAM.

Nozioni di base: esempi di codice che mostrano come eseguire le operazioni essenziali all'interno di un servizio.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Scenari: esempi di codice che mostrano come eseguire un'attività specifica chiamando più funzioni all'interno dello stesso servizio o combinate con altri Servizi AWS.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Nozioni di base
- Azioni
- Scenari

Nozioni di base

Informazioni di base

L'esempio di codice seguente mostra come creare un utente e assumere un ruolo.

> **Warning**
>
> Per evitare rischi per la sicurezza, non utilizzare gli utenti IAM per l'autenticazione quando sviluppi software creato ad hoc o lavori con dati reali. Utilizza invece la federazione con un provider di identità come AWS IAM Identity Center.

- Crea un utente che non disponga di autorizzazioni.
- Crea un ruolo che conceda l'autorizzazione per elencare i bucket Amazon S3 per l'account.
- Aggiungi una policy per consentire all'utente di assumere il ruolo.
- Assumi il ruolo ed elenca i bucket S3 utilizzando le credenziali temporanee, quindi ripulisci le risorse.

AWS CLI con lo script Bash

> **Note**
>
> C'è altro da fare. GitHub Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codice AWS.

```
###############################################################################
# function iam_create_user_assume_role
#
# Scenario to create an IAM user, create an IAM role, and apply the role to the
 user.
#
#     "IAM access" permissions are needed to run this code.
#     "STS assume role" permissions are needed to run this code. (Note: It might be
 necessary to
#           create a custom policy).
#
# Returns:
#       0 - If successful.
#       1 - If an error occurred.
###############################################################################
function iam_create_user_assume_role() {
  {
    if [ "$IAM_OPERATIONS_SOURCED" != "True" ]; then
```

```
    source ./iam_operations.sh
  fi
}

echo_repeat "*" 88
echo "Welcome to the IAM create user and assume role demo."
echo
echo "This demo will create an IAM user, create an IAM role, and apply the role to
the user."
echo_repeat "*" 88
echo

echo -n "Enter a name for a new IAM user: "
get_input
user_name=$get_input_result

local user_arn
user_arn=$(iam_create_user -u "$user_name")

# shellcheck disable=SC2181
if [[ ${?} == 0 ]]; then
  echo "Created demo IAM user named $user_name"
else
  errecho "$user_arn"
  errecho "The user failed to create. This demo will exit."
  return 1
fi

local access_key_response
access_key_response=$(iam_create_user_access_key -u "$user_name")
# shellcheck disable=SC2181
if [[ ${?} != 0 ]]; then
  errecho "The access key failed to create. This demo will exit."
  clean_up "$user_name"
  return 1
fi

IFS=$'\t ' read -r -a access_key_values <<<"$access_key_response"
local key_name=${access_key_values[0]}
local key_secret=${access_key_values[1]}

echo "Created access key named $key_name"
```

```
  echo "Wait 10 seconds for the user to be ready."
  sleep 10
  echo_repeat "*" 88
  echo

  local iam_role_name
  iam_role_name=$(generate_random_name "test-role")
  echo "Creating a role named $iam_role_name with user $user_name as the principal."

  local assume_role_policy_document="{
    \"Version\": \"2012-10-17\",
    \"Statement\": [{
        \"Effect\": \"Allow\",
        \"Principal\": {\"AWS\": \"$user_arn\"},
        \"Action\": \"sts:AssumeRole\"
        }]
    }"

  local role_arn
  role_arn=$(iam_create_role -n "$iam_role_name" -p "$assume_role_policy_document")

  # shellcheck disable=SC2181
  if [ ${?} == 0 ]; then
    echo "Created IAM role named $iam_role_name"
  else
    errecho "The role failed to create. This demo will exit."
    clean_up "$user_name" "$key_name"
    return 1
  fi

  local policy_name
  policy_name=$(generate_random_name "test-policy")
  local policy_document="{
                \"Version\": \"2012-10-17\",
                \"Statement\": [{
                    \"Effect\": \"Allow\",
                    \"Action\": \"s3:ListAllMyBuckets\",
                    \"Resource\": \"arn:aws:s3:::*\"}]}"

  local policy_arn
  policy_arn=$(iam_create_policy -n "$policy_name" -p "$policy_document")
  # shellcheck disable=SC2181
  if [[ ${?} == 0 ]]; then
    echo "Created  IAM policy named $policy_name"
```

```
  else
    errecho "The policy failed to create."
    clean_up "$user_name" "$key_name" "$iam_role_name"
    return 1
  fi

  if (iam_attach_role_policy -n "$iam_role_name" -p "$policy_arn"); then
    echo "Attached policy $policy_arn to role $iam_role_name"
  else
    errecho "The policy failed to attach."
    clean_up "$user_name" "$key_name" "$iam_role_name" "$policy_arn"
    return 1
  fi

  local assume_role_policy_document="{
                \"Version\": \"2012-10-17\",
                \"Statement\": [{
                    \"Effect\": \"Allow\",
                    \"Action\": \"sts:AssumeRole\",
                    \"Resource\": \"$role_arn\"}]}"

  local assume_role_policy_name
  assume_role_policy_name=$(generate_random_name "test-assume-role-")

  # shellcheck disable=SC2181
  local assume_role_policy_arn
  assume_role_policy_arn=$(iam_create_policy -n "$assume_role_policy_name" -p
"$assume_role_policy_document")
  # shellcheck disable=SC2181
  if [ ${?} == 0 ]; then
    echo "Created  IAM policy named $assume_role_policy_name for sts assume role"
  else
    errecho "The policy failed to create."
    clean_up "$user_name" "$key_name" "$iam_role_name" "$policy_arn" "$policy_arn"
    return 1
  fi

  echo "Wait 10 seconds to give AWS time to propagate these new resources and
connections."
  sleep 10
  echo_repeat "*" 88
  echo

  echo "Try to list buckets without the new user assuming the role."
```

```
 echo_repeat "*" 88
 echo

 # Set the environment variables for the created user.
 # bashsupport disable=BP2001
 export AWS_ACCESS_KEY_ID=$key_name
 # bashsupport disable=BP2001
 export AWS_SECRET_ACCESS_KEY=$key_secret

 local buckets
 buckets=$(s3_list_buckets)

 # shellcheck disable=SC2181
 if [ ${?} == 0 ]; then
   local bucket_count
   bucket_count=$(echo "$buckets" | wc -w | xargs)
   echo "There are $bucket_count buckets in the account. This should not have
happened."
 else
   errecho "Because the role with permissions has not been assumed, listing buckets
failed."
 fi

 echo
 echo_repeat "*" 88
 echo "Now assume the role $iam_role_name and list the buckets."
 echo_repeat "*" 88
 echo

 local credentials

 credentials=$(sts_assume_role -r "$role_arn" -n "AssumeRoleDemoSession")
 # shellcheck disable=SC2181
 if [ ${?} == 0 ]; then
   echo "Assumed role $iam_role_name"
 else
   errecho "Failed to assume role."
   export AWS_ACCESS_KEY_ID=""
   export AWS_SECRET_ACCESS_KEY=""
   clean_up "$user_name" "$key_name" "$iam_role_name" "$policy_arn" "$policy_arn"
"$assume_role_policy_arn"
   return 1
 fi
```

```
  IFS=$'\t ' read -r -a credentials <<<"$credentials"

  export AWS_ACCESS_KEY_ID=${credentials[0]}
  export AWS_SECRET_ACCESS_KEY=${credentials[1]}
  # bashsupport disable=BP2001
  export AWS_SESSION_TOKEN=${credentials[2]}

  buckets=$(s3_list_buckets)

  # shellcheck disable=SC2181
  if [ ${?} == 0 ]; then
    local bucket_count
    bucket_count=$(echo "$buckets" | wc -w | xargs)
    echo "There are $bucket_count buckets in the account. Listing buckets succeeded
because of "
    echo "the assumed role."
  else
    errecho "Failed to list buckets. This should not happen."
    export AWS_ACCESS_KEY_ID=""
    export AWS_SECRET_ACCESS_KEY=""
    export AWS_SESSION_TOKEN=""
    clean_up "$user_name" "$key_name" "$iam_role_name" "$policy_arn" "$policy_arn"
"$assume_role_policy_arn"
    return 1
  fi

  local result=0
  export AWS_ACCESS_KEY_ID=""
  export AWS_SECRET_ACCESS_KEY=""

  echo
  echo_repeat "*" 88
  echo "The created resources will now be deleted."
  echo_repeat "*" 88
  echo

  clean_up "$user_name" "$key_name" "$iam_role_name" "$policy_arn" "$policy_arn"
"$assume_role_policy_arn"

  # shellcheck disable=SC2181
  if [[ ${?} -ne 0 ]]; then
    result=1
  fi
```

```
  return $result
}
```

Le funzioni IAM utilizzate in questo scenario.

```
###############################################################################
# function iam_user_exists
#
# This function checks to see if the specified AWS Identity and Access Management
 (IAM) user already exists.
#
# Parameters:
#       $1 - The name of the IAM user to check.
#
# Returns:
#       0 - If the user already exists.
#       1 - If the user doesn't exist.
###############################################################################
function iam_user_exists() {
  local user_name
  user_name=$1

  # Check whether the IAM user already exists.
  # We suppress all output - we're interested only in the return code.

  local errors
  errors=$(aws iam get-user \
    --user-name "$user_name" 2>&1 >/dev/null)

  local error_code=${?}

  if [[ $error_code -eq 0 ]]; then
    return 0 # 0 in Bash script means true.
  else
    if [[ $errors != *"error"*"(NoSuchEntity)"* ]]; then
      aws_cli_error_log $error_code
      errecho "Error calling iam get-user $errors"
    fi

    return 1 # 1 in Bash script means false.
  fi
}
```

```
###############################################################################
# function iam_create_user
#
# This function creates the specified IAM user, unless
# it already exists.
#
# Parameters:
#       -u user_name  -- The name of the user to create.
#
# Returns:
#       The ARN of the user.
#     And:
#       0 - If successful.
#       1 - If it fails.
###############################################################################
function iam_create_user() {
  local user_name response
  local option OPTARG # Required to use getopts command in a function.

  # bashsupport disable=BP5008
  function usage() {
    echo "function iam_create_user"
    echo "Creates an AWS Identity and Access Management (IAM) user. You must supply
 a username:"
    echo "  -u user_name    The name of the user. It must be unique within the
 account."
    echo ""
  }

  # Retrieve the calling parameters.
  while getopts "u:h" option; do
    case "${option}" in
      u) user_name="${OPTARG}" ;;
      h)
        usage
        return 0
        ;;
      \?)
        echo "Invalid parameter"
        usage
        return 1
        ;;
    esac
```

```
  done
  export OPTIND=1

  if [[ -z "$user_name" ]]; then
    errecho "ERROR: You must provide a username with the -u parameter."
    usage
    return 1
  fi

  iecho "Parameters:\n"
  iecho "    User name:   $user_name"
  iecho ""

  # If the user already exists, we don't want to try to create it.
  if (iam_user_exists "$user_name"); then
    errecho "ERROR: A user with that name already exists in the account."
    return 1
  fi

  response=$(aws iam create-user --user-name "$user_name" \
    --output text \
    --query 'User.Arn')

  local error_code=${?}

  if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports create-user operation failed.$response"
    return 1
  fi

  echo "$response"

  return 0
}

###############################################################################
# function iam_create_user_access_key
#
# This function creates an IAM access key for the specified user.
#
# Parameters:
#       -u user_name -- The name of the IAM user.
#       [-f file_name] -- The optional file name for the access key output.
```

```
#
# Returns:
#       [access_key_id access_key_secret]
#     And:
#       0 - If successful.
#       1 - If it fails.
###############################################################################
function iam_create_user_access_key() {
  local user_name file_name response
  local option OPTARG # Required to use getopts command in a function.

  # bashsupport disable=BP5008
  function usage() {
    echo "function iam_create_user_access_key"
    echo "Creates an AWS Identity and Access Management (IAM) key pair."
    echo "  -u user_name   The name of the IAM user."
    echo "  [-f file_name]   Optional file name for the access key output."
    echo ""
  }

  # Retrieve the calling parameters.
  while getopts "u:f:h" option; do
    case "${option}" in
      u) user_name="${OPTARG}" ;;
      f) file_name="${OPTARG}" ;;
      h)
        usage
        return 0
        ;;
      \?)
        echo "Invalid parameter"
        usage
        return 1
        ;;
    esac
  done
  export OPTIND=1

  if [[ -z "$user_name" ]]; then
    errecho "ERROR: You must provide a username with the -u parameter."
    usage
    return 1
  fi
```

```
  response=$(aws iam create-access-key \
    --user-name "$user_name" \
    --output text)

  local error_code=${?}

  if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports create-access-key operation failed.$response"
    return 1
  fi

  if [[ -n "$file_name" ]]; then
    echo "$response" >"$file_name"
  fi

  local key_id key_secret
  # shellcheck disable=SC2086
  key_id=$(echo $response | cut -f 2 -d ' ')
  # shellcheck disable=SC2086
  key_secret=$(echo $response | cut -f 4 -d ' ')

  echo "$key_id $key_secret"

  return 0
}

###############################################################################
# function iam_create_role
#
# This function creates an IAM role.
#
# Parameters:
#       -n role_name -- The name of the IAM role.
#       -p policy_json -- The assume role policy document.
#
# Returns:
#       The ARN of the role.
#     And:
#       0 - If successful.
#       1 - If it fails.
###############################################################################
function iam_create_role() {
  local role_name policy_document response
```

```
  local option OPTARG # Required to use getopts command in a function.

  # bashsupport disable=BP5008
  function usage() {
    echo "function iam_create_user_access_key"
    echo "Creates an AWS Identity and Access Management (IAM) role."
    echo "  -n role_name   The name of the IAM role."
    echo "  -p policy_json -- The assume role policy document."
    echo ""
  }

  # Retrieve the calling parameters.
  while getopts "n:p:h" option; do
    case "${option}" in
      n) role_name="${OPTARG}" ;;
      p) policy_document="${OPTARG}" ;;
      h)
        usage
        return 0
        ;;
      \?)
        echo "Invalid parameter"
        usage
        return 1
        ;;
    esac
  done
  export OPTIND=1

  if [[ -z "$role_name" ]]; then
    errecho "ERROR: You must provide a role name with the -n parameter."
    usage
    return 1
  fi

  if [[ -z "$policy_document" ]]; then
    errecho "ERROR: You must provide a policy document with the -p parameter."
    usage
    return 1
  fi

  response=$(aws iam create-role \
    --role-name "$role_name" \
    --assume-role-policy-document "$policy_document" \
```

```
    --output text \
    --query Role.Arn)

  local error_code=${?}

  if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports create-role operation failed.\n$response"
    return 1
  fi

  echo "$response"

  return 0
}

###############################################################################
# function iam_create_policy
#
# This function creates an IAM policy.
#
# Parameters:
#       -n policy_name -- The name of the IAM policy.
#       -p policy_json -- The policy document.
#
# Returns:
#       0 - If successful.
#       1 - If it fails.
###############################################################################
function iam_create_policy() {
  local policy_name policy_document response
  local option OPTARG # Required to use getopts command in a function.

  # bashsupport disable=BP5008
  function usage() {
    echo "function iam_create_policy"
    echo "Creates an AWS Identity and Access Management (IAM) policy."
    echo "  -n policy_name   The name of the IAM policy."
    echo "  -p policy_json -- The policy document."
    echo ""
  }

  # Retrieve the calling parameters.
  while getopts "n:p:h" option; do
```

```
    case "${option}" in
      n) policy_name="${OPTARG}" ;;
      p) policy_document="${OPTARG}" ;;
      h)
        usage
        return 0
        ;;
      \?)
        echo "Invalid parameter"
        usage
        return 1
        ;;
    esac
  done
  export OPTIND=1

  if [[ -z "$policy_name" ]]; then
    errecho "ERROR: You must provide a policy name with the -n parameter."
    usage
    return 1
  fi

  if [[ -z "$policy_document" ]]; then
    errecho "ERROR: You must provide a policy document with the -p parameter."
    usage
    return 1
  fi

  response=$(aws iam create-policy \
    --policy-name "$policy_name" \
    --policy-document "$policy_document" \
    --output text \
    --query Policy.Arn)

  local error_code=${?}

  if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports create-policy operation failed.\n$response"
    return 1
  fi

  echo "$response"
}
```

```
###############################################################################
# function iam_attach_role_policy
#
# This function attaches an IAM policy to a tole.
#
# Parameters:
#       -n role_name -- The name of the IAM role.
#       -p policy_ARN -- The IAM policy document ARN..
#
# Returns:
#       0 - If successful.
#       1 - If it fails.
###############################################################################
function iam_attach_role_policy() {
  local role_name policy_arn response
  local option OPTARG # Required to use getopts command in a function.

  # bashsupport disable=BP5008
  function usage() {
    echo "function iam_attach_role_policy"
    echo "Attaches an AWS Identity and Access Management (IAM) policy to an IAM
 role."
    echo "  -n role_name   The name of the IAM role."
    echo "  -p policy_ARN -- The IAM policy document ARN."
    echo ""
  }

  # Retrieve the calling parameters.
  while getopts "n:p:h" option; do
    case "${option}" in
      n) role_name="${OPTARG}" ;;
      p) policy_arn="${OPTARG}" ;;
      h)
        usage
        return 0
        ;;
      \?)
        echo "Invalid parameter"
        usage
        return 1
        ;;
    esac
  done
```

```
  export OPTIND=1

  if [[ -z "$role_name" ]]; then
    errecho "ERROR: You must provide a role name with the -n parameter."
    usage
    return 1
  fi

  if [[ -z "$policy_arn" ]]; then
    errecho "ERROR: You must provide a policy ARN with the -p parameter."
    usage
    return 1
  fi

  response=$(aws iam attach-role-policy \
    --role-name "$role_name" \
    --policy-arn "$policy_arn")

  local error_code=${?}

  if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports attach-role-policy operation failed.\n$response"
    return 1
  fi

  echo "$response"

  return 0
}

###############################################################################
# function iam_detach_role_policy
#
# This function detaches an IAM policy to a tole.
#
# Parameters:
#       -n role_name -- The name of the IAM role.
#       -p policy_ARN -- The IAM policy document ARN..
#
# Returns:
#       0 - If successful.
#       1 - If it fails.
###############################################################################
```

```
function iam_detach_role_policy() {
  local role_name policy_arn response
  local option OPTARG # Required to use getopts command in a function.

  # bashsupport disable=BP5008
  function usage() {
    echo "function iam_detach_role_policy"
    echo "Detaches an AWS Identity and Access Management (IAM) policy to an IAM
 role."
    echo "  -n role_name   The name of the IAM role."
    echo "  -p policy_ARN -- The IAM policy document ARN."
    echo ""
  }

  # Retrieve the calling parameters.
  while getopts "n:p:h" option; do
    case "${option}" in
      n) role_name="${OPTARG}" ;;
      p) policy_arn="${OPTARG}" ;;
      h)
        usage
        return 0
        ;;
      \?)
        echo "Invalid parameter"
        usage
        return 1
        ;;
    esac
  done
  export OPTIND=1

  if [[ -z "$role_name" ]]; then
    errecho "ERROR: You must provide a role name with the -n parameter."
    usage
    return 1
  fi

  if [[ -z "$policy_arn" ]]; then
    errecho "ERROR: You must provide a policy ARN with the -p parameter."
    usage
    return 1
  fi
```

```
  response=$(aws iam detach-role-policy \
    --role-name "$role_name" \
    --policy-arn "$policy_arn")

  local error_code=${?}

  if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports detach-role-policy operation failed.\n$response"
    return 1
  fi

  echo "$response"

  return 0
}

###############################################################################
# function iam_delete_policy
#
# This function deletes an IAM policy.
#
# Parameters:
#       -n policy_arn -- The name of the IAM policy arn.
#
# Returns:
#       0 - If successful.
#       1 - If it fails.
###############################################################################
function iam_delete_policy() {
  local policy_arn response
  local option OPTARG # Required to use getopts command in a function.

  # bashsupport disable=BP5008
  function usage() {
    echo "function iam_delete_policy"
    echo "Deletes an AWS Identity and Access Management (IAM) policy"
    echo "  -n policy_arn -- The name of the IAM policy arn."
    echo ""
  }

  # Retrieve the calling parameters.
  while getopts "n:h" option; do
    case "${option}" in
```

```
      n) policy_arn="${OPTARG}" ;;
      h)
        usage
        return 0
        ;;
      \?)
        echo "Invalid parameter"
        usage
        return 1
        ;;
    esac
  done
  export OPTIND=1

  if [[ -z "$policy_arn" ]]; then
    errecho "ERROR: You must provide a policy arn with the -n parameter."
    usage
    return 1
  fi

  iecho "Parameters:\n"
  iecho "    Policy arn:  $policy_arn"
  iecho ""

  response=$(aws iam delete-policy \
    --policy-arn "$policy_arn")

  local error_code=${?}

  if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports delete-policy operation failed.\n$response"
    return 1
  fi

  iecho "delete-policy response:$response"
  iecho

  return 0
}

###############################################################################
# function iam_delete_role
#
```

```
# This function deletes an IAM role.
#
# Parameters:
#       -n role_name -- The name of the IAM role.
#
# Returns:
#       0 - If successful.
#       1 - If it fails.
###############################################################################
function iam_delete_role() {
  local role_name response
  local option OPTARG # Required to use getopts command in a function.

  # bashsupport disable=BP5008
  function usage() {
    echo "function iam_delete_role"
    echo "Deletes an AWS Identity and Access Management (IAM) role"
    echo "  -n role_name -- The name of the IAM role."
    echo ""
  }

  # Retrieve the calling parameters.
  while getopts "n:h" option; do
    case "${option}" in
      n) role_name="${OPTARG}" ;;
      h)
        usage
        return 0
        ;;
      \?)
        echo "Invalid parameter"
        usage
        return 1
        ;;
    esac
  done
  export OPTIND=1

  echo "role_name:$role_name"
  if [[ -z "$role_name" ]]; then
    errecho "ERROR: You must provide a role name with the -n parameter."
    usage
    return 1
  fi
```

```
  iecho "Parameters:\n"
  iecho "    Role name:  $role_name"
  iecho ""

  response=$(aws iam delete-role \
    --role-name "$role_name")

  local error_code=${?}

  if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports delete-role operation failed.\n$response"
    return 1
  fi

  iecho "delete-role response:$response"
  iecho

  return 0
}

###############################################################################
# function iam_delete_access_key
#
# This function deletes an IAM access key for the specified IAM user.
#
# Parameters:
#       -u user_name  -- The name of the user.
#       -k access_key -- The access key to delete.
#
# Returns:
#       0 - If successful.
#       1 - If it fails.
###############################################################################
function iam_delete_access_key() {
  local user_name access_key response
  local option OPTARG # Required to use getopts command in a function.

  # bashsupport disable=BP5008
  function usage() {
    echo "function iam_delete_access_key"
    echo "Deletes an AWS Identity and Access Management (IAM) access key for the
 specified IAM user"
```

```
  echo "  -u user_name    The name of the user."
  echo "  -k access_key   The access key to delete."
  echo ""
}

# Retrieve the calling parameters.
while getopts "u:k:h" option; do
  case "${option}" in
    u) user_name="${OPTARG}" ;;
    k) access_key="${OPTARG}" ;;
    h)
      usage
      return 0
      ;;
    \?)
      echo "Invalid parameter"
      usage
      return 1
      ;;
  esac
done
export OPTIND=1

if [[ -z "$user_name" ]]; then
  errecho "ERROR: You must provide a username with the -u parameter."
  usage
  return 1
fi

if [[ -z "$access_key" ]]; then
  errecho "ERROR: You must provide an access key with the -k parameter."
  usage
  return 1
fi

iecho "Parameters:\n"
iecho "    Username:   $user_name"
iecho "    Access key:   $access_key"
iecho ""

response=$(aws iam delete-access-key \
  --user-name "$user_name" \
  --access-key-id "$access_key")
```

```
  local error_code=${?}

  if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports delete-access-key operation failed.\n$response"
    return 1
  fi

  iecho "delete-access-key response:$response"
  iecho

  return 0
}

################################################################################
# function iam_delete_user
#
# This function deletes the specified IAM user.
#
# Parameters:
#       -u user_name  -- The name of the user to create.
#
# Returns:
#       0 - If successful.
#       1 - If it fails.
################################################################################
function iam_delete_user() {
  local user_name response
  local option OPTARG # Required to use getopts command in a function.

  # bashsupport disable=BP5008
  function usage() {
    echo "function iam_delete_user"
    echo "Deletes an AWS Identity and Access Management (IAM) user. You must supply
 a username:"
    echo "  -u user_name    The name of the user."
    echo ""
  }

  # Retrieve the calling parameters.
  while getopts "u:h" option; do
    case "${option}" in
      u) user_name="${OPTARG}" ;;
      h)
```

```
        usage
        return 0
        ;;
      \?)
        echo "Invalid parameter"
        usage
        return 1
        ;;
    esac
  done
  export OPTIND=1

  if [[ -z "$user_name" ]]; then
    errecho "ERROR: You must provide a username with the -u parameter."
    usage
    return 1
  fi

  iecho "Parameters:\n"
  iecho "    User name:   $user_name"
  iecho ""

  # If the user does not exist, we don't want to try to delete it.
  if (! iam_user_exists "$user_name"); then
    errecho "ERROR: A user with that name does not exist in the account."
    return 1
  fi

  response=$(aws iam delete-user \
    --user-name "$user_name")

  local error_code=${?}

  if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports delete-user operation failed.$response"
    return 1
  fi

  iecho "delete-user response:$response"
  iecho

  return 0
}
```

- Per informazioni dettagliate sull'API, consulta i seguenti argomenti nella documentazione di riferimento dei comandi della AWS CLI .
  - AttachRolePolicy
  - CreateAccessKey
  - CreatePolicy
  - CreateRole
  - CreateUser
  - DeleteAccessKey
  - DeletePolicy
  - DeleteRole
  - DeleteUser
  - DeleteUserPolicy
  - DetachRolePolicy
  - PutUserPolicy

Azioni

`AttachRolePolicy`

Il seguente esempio di codice mostra come usare`AttachRolePolicy`.

AWS CLI con lo script Bash

> **Note**
>
> C'è altro da fare. GitHub Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codice AWS.

```
###############################################################################
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
###############################################################################
```

```
function errecho() {
  printf "%s\n" "$*" 1>&2
}

#############################################################################
# function iam_attach_role_policy
#
# This function attaches an IAM policy to a tole.
#
# Parameters:
#       -n role_name -- The name of the IAM role.
#       -p policy_ARN -- The IAM policy document ARN..
#
# Returns:
#       0 - If successful.
#       1 - If it fails.
#############################################################################
function iam_attach_role_policy() {
  local role_name policy_arn response
  local option OPTARG # Required to use getopts command in a function.

  # bashsupport disable=BP5008
  function usage() {
    echo "function iam_attach_role_policy"
    echo "Attaches an AWS Identity and Access Management (IAM) policy to an IAM
 role."
    echo "  -n role_name   The name of the IAM role."
    echo "  -p policy_ARN -- The IAM policy document ARN."
    echo ""
  }

  # Retrieve the calling parameters.
  while getopts "n:p:h" option; do
    case "${option}" in
      n) role_name="${OPTARG}" ;;
      p) policy_arn="${OPTARG}" ;;
      h)
        usage
        return 0
        ;;
      \?)
        echo "Invalid parameter"
        usage
        return 1
```

```
        ;;
    esac
  done
  export OPTIND=1

  if [[ -z "$role_name" ]]; then
    errecho "ERROR: You must provide a role name with the -n parameter."
    usage
    return 1
  fi

  if [[ -z "$policy_arn" ]]; then
    errecho "ERROR: You must provide a policy ARN with the -p parameter."
    usage
    return 1
  fi

  response=$(aws iam attach-role-policy \
    --role-name "$role_name" \
    --policy-arn "$policy_arn")

  local error_code=${?}

  if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports attach-role-policy operation failed.\n$response"
    return 1
  fi

  echo "$response"

  return 0
}
```

- Per i dettagli sull'API, consulta AttachRolePolicy AWS CLICommand Reference.

`CreateAccessKey`

Il seguente esempio di codice mostra come utilizzare`CreateAccessKey`.

AWS CLI con lo script Bash

> **Note**
>
> C'è altro da fare. GitHub Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codice AWS.

```
#############################################################################
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
#############################################################################
function errecho() {
  printf "%s\n" "$*" 1>&2
}

#############################################################################
# function iam_create_user_access_key
#
# This function creates an IAM access key for the specified user.
#
# Parameters:
#       -u user_name -- The name of the IAM user.
#       [-f file_name] -- The optional file name for the access key output.
#
# Returns:
#       [access_key_id access_key_secret]
#     And:
#       0 - If successful.
#       1 - If it fails.
#############################################################################
function iam_create_user_access_key() {
  local user_name file_name response
  local option OPTARG # Required to use getopts command in a function.

  # bashsupport disable=BP5008
  function usage() {
    echo "function iam_create_user_access_key"
    echo "Creates an AWS Identity and Access Management (IAM) key pair."
    echo "  -u user_name   The name of the IAM user."
    echo "  [-f file_name]   Optional file name for the access key output."
```

```
  echo ""
}

# Retrieve the calling parameters.
while getopts "u:f:h" option; do
  case "${option}" in
    u) user_name="${OPTARG}" ;;
    f) file_name="${OPTARG}" ;;
    h)
      usage
      return 0
      ;;
    \?)
      echo "Invalid parameter"
      usage
      return 1
      ;;
  esac
done
export OPTIND=1

if [[ -z "$user_name" ]]; then
  errecho "ERROR: You must provide a username with the -u parameter."
  usage
  return 1
fi

response=$(aws iam create-access-key \
  --user-name "$user_name" \
  --output text)

local error_code=${?}

if [[ $error_code -ne 0 ]]; then
  aws_cli_error_log $error_code
  errecho "ERROR: AWS reports create-access-key operation failed.$response"
  return 1
fi

if [[ -n "$file_name" ]]; then
  echo "$response" >"$file_name"
fi

local key_id key_secret
```

```
  # shellcheck disable=SC2086
  key_id=$(echo $response | cut -f 2 -d ' ')
  # shellcheck disable=SC2086
  key_secret=$(echo $response | cut -f 4 -d ' ')

  echo "$key_id $key_secret"

  return 0
}
```

- Per i dettagli sull'API, consulta CreateAccessKey AWS CLICommand Reference.

`CreatePolicy`

Il seguente esempio di codice mostra come utilizzare`CreatePolicy`.

AWS CLI con lo script Bash

> **Note**
>
> C'è altro da fare. GitHub Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codice AWS.

```
###############################################################################
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
###############################################################################
function errecho() {
  printf "%s\n" "$*" 1>&2
}

###############################################################################
# function iam_create_policy
#
# This function creates an IAM policy.
#
# Parameters:
#       -n policy_name -- The name of the IAM policy.
#       -p policy_json -- The policy document.
```

```
#
# Returns:
#       0 - If successful.
#       1 - If it fails.
###############################################################################
function iam_create_policy() {
  local policy_name policy_document response
  local option OPTARG # Required to use getopts command in a function.

  # bashsupport disable=BP5008
  function usage() {
    echo "function iam_create_policy"
    echo "Creates an AWS Identity and Access Management (IAM) policy."
    echo "  -n policy_name   The name of the IAM policy."
    echo "  -p policy_json -- The policy document."
    echo ""
  }

  # Retrieve the calling parameters.
  while getopts "n:p:h" option; do
    case "${option}" in
      n) policy_name="${OPTARG}" ;;
      p) policy_document="${OPTARG}" ;;
      h)
        usage
        return 0
        ;;
      \?)
        echo "Invalid parameter"
        usage
        return 1
        ;;
    esac
  done
  export OPTIND=1

  if [[ -z "$policy_name" ]]; then
    errecho "ERROR: You must provide a policy name with the -n parameter."
    usage
    return 1
  fi

  if [[ -z "$policy_document" ]]; then
    errecho "ERROR: You must provide a policy document with the -p parameter."
```

```
    usage
    return 1
  fi

  response=$(aws iam create-policy \
    --policy-name "$policy_name" \
    --policy-document "$policy_document" \
    --output text \
    --query Policy.Arn)

  local error_code=${?}

  if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports create-policy operation failed.\n$response"
    return 1
  fi

  echo "$response"
}
```

- Per i dettagli sull'API, consulta CreatePolicy AWS CLICommand Reference.

`CreateRole`

Il seguente esempio di codice mostra come utilizzare`CreateRole`.

AWS CLI con lo script Bash

> **Note**
>
> C'è altro da fare. GitHub Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codice AWS.

```
#####################################################################################
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
#####################################################################################
function errecho() {
```

```
  printf "%s\n" "$*" 1>&2
}

###############################################################################
# function iam_create_role
#
# This function creates an IAM role.
#
# Parameters:
#       -n role_name -- The name of the IAM role.
#       -p policy_json -- The assume role policy document.
#
# Returns:
#       The ARN of the role.
#     And:
#       0 - If successful.
#       1 - If it fails.
###############################################################################
function iam_create_role() {
  local role_name policy_document response
  local option OPTARG # Required to use getopts command in a function.

  # bashsupport disable=BP5008
  function usage() {
    echo "function iam_create_user_access_key"
    echo "Creates an AWS Identity and Access Management (IAM) role."
    echo "  -n role_name   The name of the IAM role."
    echo "  -p policy_json -- The assume role policy document."
    echo ""
  }

  # Retrieve the calling parameters.
  while getopts "n:p:h" option; do
    case "${option}" in
      n) role_name="${OPTARG}" ;;
      p) policy_document="${OPTARG}" ;;
      h)
        usage
        return 0
        ;;
      \?)
        echo "Invalid parameter"
        usage
        return 1
```

```
        ;;
    esac
  done
  export OPTIND=1

  if [[ -z "$role_name" ]]; then
    errecho "ERROR: You must provide a role name with the -n parameter."
    usage
    return 1
  fi

  if [[ -z "$policy_document" ]]; then
    errecho "ERROR: You must provide a policy document with the -p parameter."
    usage
    return 1
  fi

  response=$(aws iam create-role \
    --role-name "$role_name" \
    --assume-role-policy-document "$policy_document" \
    --output text \
    --query Role.Arn)

  local error_code=${?}

  if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports create-role operation failed.\n$response"
    return 1
  fi

  echo "$response"

  return 0
}
```

- Per i dettagli sull'API, consulta CreateRole AWS CLICommand Reference.

`CreateUser`

Il seguente esempio di codice mostra come utilizzare`CreateUser`.

AWS CLI con lo script Bash

> **Note**
>
> C'è altro da fare. GitHub Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codice AWS.

```
###############################################################################
# function iecho
#
# This function enables the script to display the specified text only if
# the global variable $VERBOSE is set to true.
###############################################################################
function iecho() {
  if [[ $VERBOSE == true ]]; then
    echo "$@"
  fi
}

###############################################################################
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
###############################################################################
function errecho() {
  printf "%s\n" "$*" 1>&2
}

###############################################################################
# function iam_create_user
#
# This function creates the specified IAM user, unless
# it already exists.
#
# Parameters:
#       -u user_name  -- The name of the user to create.
#
# Returns:
#       The ARN of the user.
#     And:
#       0 - If successful.
```

```
#       1 - If it fails.
###############################################################################
function iam_create_user() {
  local user_name response
  local option OPTARG # Required to use getopts command in a function.

  # bashsupport disable=BP5008
  function usage() {
    echo "function iam_create_user"
    echo "Creates an AWS Identity and Access Management (IAM) user. You must supply
 a username:"
    echo "  -u user_name    The name of the user. It must be unique within the
 account."
    echo ""
  }

  # Retrieve the calling parameters.
  while getopts "u:h" option; do
    case "${option}" in
      u) user_name="${OPTARG}" ;;
      h)
        usage
        return 0
        ;;
      \?)
        echo "Invalid parameter"
        usage
        return 1
        ;;
    esac
  done
  export OPTIND=1

  if [[ -z "$user_name" ]]; then
    errecho "ERROR: You must provide a username with the -u parameter."
    usage
    return 1
  fi

  iecho "Parameters:\n"
  iecho "    User name:   $user_name"
  iecho ""

  # If the user already exists, we don't want to try to create it.
```

```
  if (iam_user_exists "$user_name"); then
    errecho "ERROR: A user with that name already exists in the account."
    return 1
  fi

  response=$(aws iam create-user --user-name "$user_name" \
    --output text \
    --query 'User.Arn')

  local error_code=${?}

  if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports create-user operation failed.$response"
    return 1
  fi

  echo "$response"

  return 0
}
```

- Per i dettagli sull'API, consulta CreateUser AWS CLICommand Reference.

`DeleteAccessKey`

Il seguente esempio di codice mostra come utilizzare`DeleteAccessKey`.

AWS CLI con lo script Bash

> **Note**
>
> C'è altro da fare. GitHub Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codice AWS.

```
##############################################################################
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
##############################################################################
```

```
function errecho() {
  printf "%s\n" "$*" 1>&2
}

###############################################################################
# function iam_delete_access_key
#
# This function deletes an IAM access key for the specified IAM user.
#
# Parameters:
#       -u user_name  -- The name of the user.
#       -k access_key -- The access key to delete.
#
# Returns:
#       0 - If successful.
#       1 - If it fails.
###############################################################################
function iam_delete_access_key() {
  local user_name access_key response
  local option OPTARG # Required to use getopts command in a function.

  # bashsupport disable=BP5008
  function usage() {
    echo "function iam_delete_access_key"
    echo "Deletes an AWS Identity and Access Management (IAM) access key for the
 specified IAM user"
    echo "  -u user_name    The name of the user."
    echo "  -k access_key   The access key to delete."
    echo ""
  }

  # Retrieve the calling parameters.
  while getopts "u:k:h" option; do
    case "${option}" in
      u) user_name="${OPTARG}" ;;
      k) access_key="${OPTARG}" ;;
      h)
        usage
        return 0
        ;;
      \?)
        echo "Invalid parameter"
        usage
        return 1
```

```
        ;;
    esac
  done
  export OPTIND=1

  if [[ -z "$user_name" ]]; then
    errecho "ERROR: You must provide a username with the -u parameter."
    usage
    return 1
  fi

  if [[ -z "$access_key" ]]; then
    errecho "ERROR: You must provide an access key with the -k parameter."
    usage
    return 1
  fi

  iecho "Parameters:\n"
  iecho "    Username:   $user_name"
  iecho "    Access key:   $access_key"
  iecho ""

  response=$(aws iam delete-access-key \
    --user-name "$user_name" \
    --access-key-id "$access_key")

  local error_code=${?}

  if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports delete-access-key operation failed.\n$response"
    return 1
  fi

  iecho "delete-access-key response:$response"
  iecho

  return 0
}
```

- Per i dettagli sull'API, consulta DeleteAccessKey AWS CLICommand Reference.

`DeletePolicy`

Il seguente esempio di codice mostra come utilizzare`DeletePolicy`.

AWS CLI con lo script Bash

> **Note**
>
> C'è altro da fare. GitHub Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codice AWS.

```
##############################################################################
# function iecho
#
# This function enables the script to display the specified text only if
# the global variable $VERBOSE is set to true.
##############################################################################
function iecho() {
  if [[ $VERBOSE == true ]]; then
    echo "$@"
  fi
}

##############################################################################
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
##############################################################################
function errecho() {
  printf "%s\n" "$*" 1>&2
}

##############################################################################
# function iam_delete_policy
#
# This function deletes an IAM policy.
#
# Parameters:
#       -n policy_arn -- The name of the IAM policy arn.
#
# Returns:
#       0 - If successful.
```

```
#       1 - If it fails.
###############################################################################
function iam_delete_policy() {
  local policy_arn response
  local option OPTARG # Required to use getopts command in a function.

  # bashsupport disable=BP5008
  function usage() {
    echo "function iam_delete_policy"
    echo "Deletes an AWS Identity and Access Management (IAM) policy"
    echo "  -n policy_arn -- The name of the IAM policy arn."
    echo ""
  }

  # Retrieve the calling parameters.
  while getopts "n:h" option; do
    case "${option}" in
      n) policy_arn="${OPTARG}" ;;
      h)
        usage
        return 0
        ;;
      \?)
        echo "Invalid parameter"
        usage
        return 1
        ;;
    esac
  done
  export OPTIND=1

  if [[ -z "$policy_arn" ]]; then
    errecho "ERROR: You must provide a policy arn with the -n parameter."
    usage
    return 1
  fi

  iecho "Parameters:\n"
  iecho "    Policy arn:  $policy_arn"
  iecho ""

  response=$(aws iam delete-policy \
    --policy-arn "$policy_arn")
```

```
  local error_code=${?}

  if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports delete-policy operation failed.\n$response"
    return 1
  fi

  iecho "delete-policy response:$response"
  iecho

  return 0
}
```

- Per i dettagli sull'API, consulta DeletePolicy AWS CLICommand Reference.

`DeleteRole`

Il seguente esempio di codice mostra come utilizzare`DeleteRole`.

AWS CLI con lo script Bash

> **Note**
>
> C'è altro da fare. GitHub Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codice AWS.

```
##############################################################################
# function iecho
#
# This function enables the script to display the specified text only if
# the global variable $VERBOSE is set to true.
##############################################################################
function iecho() {
  if [[ $VERBOSE == true ]]; then
    echo "$@"
  fi
}

##############################################################################
```

```
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
###############################################################################
function errecho() {
  printf "%s\n" "$*" 1>&2
}

###############################################################################
# function iam_delete_role
#
# This function deletes an IAM role.
#
# Parameters:
#       -n role_name -- The name of the IAM role.
#
# Returns:
#       0 - If successful.
#       1 - If it fails.
###############################################################################
function iam_delete_role() {
  local role_name response
  local option OPTARG # Required to use getopts command in a function.

  # bashsupport disable=BP5008
  function usage() {
    echo "function iam_delete_role"
    echo "Deletes an AWS Identity and Access Management (IAM) role"
    echo "  -n role_name -- The name of the IAM role."
    echo ""
  }

  # Retrieve the calling parameters.
  while getopts "n:h" option; do
    case "${option}" in
      n) role_name="${OPTARG}" ;;
      h)
        usage
        return 0
        ;;
      \?)
        echo "Invalid parameter"
        usage
        return 1
```

```
        ;;
    esac
  done
  export OPTIND=1

  echo "role_name:$role_name"
  if [[ -z "$role_name" ]]; then
    errecho "ERROR: You must provide a role name with the -n parameter."
    usage
    return 1
  fi

  iecho "Parameters:\n"
  iecho "    Role name:  $role_name"
  iecho ""

  response=$(aws iam delete-role \
    --role-name "$role_name")

  local error_code=${?}

  if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports delete-role operation failed.\n$response"
    return 1
  fi

  iecho "delete-role response:$response"
  iecho

  return 0
}
```

- Per i dettagli sull'API, consulta DeleteRole AWS CLICommand Reference.

`DeleteUser`

Il seguente esempio di codice mostra come utilizzare`DeleteUser`.

AWS CLI con lo script Bash

> **Note**
>
> C'è altro da fare. GitHub Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codice AWS.

```
##############################################################################
# function iecho
#
# This function enables the script to display the specified text only if
# the global variable $VERBOSE is set to true.
##############################################################################
function iecho() {
  if [[ $VERBOSE == true ]]; then
    echo "$@"
  fi
}

##############################################################################
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
##############################################################################
function errecho() {
  printf "%s\n" "$*" 1>&2
}

##############################################################################
# function iam_delete_user
#
# This function deletes the specified IAM user.
#
# Parameters:
#       -u user_name  -- The name of the user to create.
#
# Returns:
#       0 - If successful.
#       1 - If it fails.
##############################################################################
function iam_delete_user() {
```

```
  local user_name response
  local option OPTARG # Required to use getopts command in a function.

  # bashsupport disable=BP5008
  function usage() {
    echo "function iam_delete_user"
    echo "Deletes an AWS Identity and Access Management (IAM) user. You must supply
a username:"
    echo "  -u user_name    The name of the user."
    echo ""
  }

  # Retrieve the calling parameters.
  while getopts "u:h" option; do
    case "${option}" in
      u) user_name="${OPTARG}" ;;
      h)
        usage
        return 0
        ;;
      \?)
        echo "Invalid parameter"
        usage
        return 1
        ;;
    esac
  done
  export OPTIND=1

  if [[ -z "$user_name" ]]; then
    errecho "ERROR: You must provide a username with the -u parameter."
    usage
    return 1
  fi

  iecho "Parameters:\n"
  iecho "    User name:   $user_name"
  iecho ""

  # If the user does not exist, we don't want to try to delete it.
  if (! iam_user_exists "$user_name"); then
    errecho "ERROR: A user with that name does not exist in the account."
    return 1
  fi
```

```
  response=$(aws iam delete-user \
    --user-name "$user_name")

  local error_code=${?}

  if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports delete-user operation failed.$response"
    return 1
  fi

  iecho "delete-user response:$response"
  iecho

  return 0
}
```

- Per i dettagli sull'API, consulta DeleteUser AWS CLICommand Reference.

`DetachRolePolicy`

Il seguente esempio di codice mostra come utilizzare`DetachRolePolicy`.

AWS CLI con lo script Bash

> **Note**
>
> C'è altro da fare. GitHub Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codice AWS.

```
##############################################################################
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
##############################################################################
function errecho() {
  printf "%s\n" "$*" 1>&2
}
```

```
###############################################################################
# function iam_detach_role_policy
#
# This function detaches an IAM policy to a tole.
#
# Parameters:
#       -n role_name -- The name of the IAM role.
#       -p policy_ARN -- The IAM policy document ARN..
#
# Returns:
#       0 - If successful.
#       1 - If it fails.
###############################################################################
function iam_detach_role_policy() {
  local role_name policy_arn response
  local option OPTARG # Required to use getopts command in a function.

  # bashsupport disable=BP5008
  function usage() {
    echo "function iam_detach_role_policy"
    echo "Detaches an AWS Identity and Access Management (IAM) policy to an IAM
 role."
    echo "  -n role_name   The name of the IAM role."
    echo "  -p policy_ARN -- The IAM policy document ARN."
    echo ""
  }

  # Retrieve the calling parameters.
  while getopts "n:p:h" option; do
    case "${option}" in
      n) role_name="${OPTARG}" ;;
      p) policy_arn="${OPTARG}" ;;
      h)
        usage
        return 0
        ;;
      \?)
        echo "Invalid parameter"
        usage
        return 1
        ;;
    esac
  done
  export OPTIND=1
```

```
  if [[ -z "$role_name" ]]; then
    errecho "ERROR: You must provide a role name with the -n parameter."
    usage
    return 1
  fi

  if [[ -z "$policy_arn" ]]; then
    errecho "ERROR: You must provide a policy ARN with the -p parameter."
    usage
    return 1
  fi

  response=$(aws iam detach-role-policy \
    --role-name "$role_name" \
    --policy-arn "$policy_arn")

  local error_code=${?}

  if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports detach-role-policy operation failed.\n$response"
    return 1
  fi

  echo "$response"

  return 0
}
```

- Per i dettagli sull'API, consulta DetachRolePolicy AWS CLICommand Reference.

`GetUser`

Il seguente esempio di codice mostra come utilizzare`GetUser`.

AWS CLI con lo script Bash

> **Note**
>
> C'è altro da fare. GitHub Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codice AWS.

```
#############################################################################
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
#############################################################################
function errecho() {
  printf "%s\n" "$*" 1>&2
}

#############################################################################
# function iam_user_exists
#
# This function checks to see if the specified AWS Identity and Access Management
 (IAM) user already exists.
#
# Parameters:
#       $1 - The name of the IAM user to check.
#
# Returns:
#       0 - If the user already exists.
#       1 - If the user doesn't exist.
#############################################################################
function iam_user_exists() {
  local user_name
  user_name=$1

  # Check whether the IAM user already exists.
  # We suppress all output - we're interested only in the return code.

  local errors
  errors=$(aws iam get-user \
    --user-name "$user_name" 2>&1 >/dev/null)

  local error_code=${?}
```

```
  if [[ $error_code -eq 0 ]]; then
    return 0 # 0 in Bash script means true.
  else
    if [[ $errors != *"error"*"(NoSuchEntity)"* ]]; then
      aws_cli_error_log $error_code
      errecho "Error calling iam get-user $errors"
    fi

    return 1 # 1 in Bash script means false.
  fi
}
```

- Per i dettagli sull'API, consulta GetUser AWS CLICommand Reference.

`ListAccessKeys`

Il seguente esempio di codice mostra come utilizzare`ListAccessKeys`.

AWS CLI con lo script Bash

> **Note**
>
> C'è altro da fare. GitHub Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codice AWS.

```
##############################################################################
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
##############################################################################
function errecho() {
  printf "%s\n" "$*" 1>&2
}

##############################################################################
# function iam_list_access_keys
#
# This function lists the access keys for the specified user.
#
```

```
# Parameters:
#       -u user_name -- The name of the IAM user.
#
# Returns:
#       access_key_ids
#     And:
#       0 - If successful.
#       1 - If it fails.
###############################################################################
function iam_list_access_keys() {

  # bashsupport disable=BP5008
  function usage() {
    echo "function iam_list_access_keys"
    echo "Lists the AWS Identity and Access Management (IAM) access key IDs for the
 specified user."
    echo "  -u user_name   The name of the IAM user."
    echo ""
  }

  local user_name response
  local option OPTARG # Required to use getopts command in a function.
  # Retrieve the calling parameters.
  while getopts "u:h" option; do
    case "${option}" in
      u) user_name="${OPTARG}" ;;
      h)
        usage
        return 0
        ;;
      \?)
        echo "Invalid parameter"
        usage
        return 1
        ;;
    esac
  done
  export OPTIND=1

  if [[ -z "$user_name" ]]; then
    errecho "ERROR: You must provide a username with the -u parameter."
    usage
    return 1
  fi
```

```
  response=$(aws iam list-access-keys \
    --user-name "$user_name" \
    --output text \
    --query 'AccessKeyMetadata[].AccessKeyId')

  local error_code=${?}

  if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports list-access-keys operation failed.$response"
    return 1
  fi

  echo "$response"

  return 0
}
```

- Per i dettagli sull'API, consulta ListAccessKeys AWS CLICommand Reference.

`ListUsers`

Il seguente esempio di codice mostra come utilizzare`ListUsers`.

AWS CLI con lo script Bash

> **Note**
>
> C'è altro da fare. GitHub Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codice AWS.

```
##############################################################################
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
##############################################################################
function errecho() {
  printf "%s\n" "$*" 1>&2
}
```

```
###############################################################################
# function iam_list_users
#
# List the IAM users in the account.
#
# Returns:
#       The list of users names
#    And:
#       0 - If the user already exists.
#       1 - If the user doesn't exist.
###############################################################################
function iam_list_users() {
  local option OPTARG # Required to use getopts command in a function.
  local error_code
  # bashsupport disable=BP5008
  function usage() {
    echo "function iam_list_users"
    echo "Lists the AWS Identity and Access Management (IAM) user in the account."
    echo ""
  }

  # Retrieve the calling parameters.
  while getopts "h" option; do
    case "${option}" in
      h)
        usage
        return 0
        ;;
      \?)
        echo "Invalid parameter"
        usage
        return 1
        ;;
    esac
  done
  export OPTIND=1

  local response

  response=$(aws iam list-users \
    --output text \
    --query "Users[].UserName")
  error_code=${?}
```

```
  if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports list-users operation failed.$response"
    return 1
  fi

  echo "$response"

  return 0
}
```

- Per i dettagli sull'API, consulta ListUsers AWS CLICommand Reference.

`UpdateAccessKey`

Il seguente esempio di codice mostra come utilizzare`UpdateAccessKey`.

AWS CLI con lo script Bash

> **Note**
>
> C'è altro da fare. GitHub Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codice AWS.

```
##############################################################################
# function iam_update_access_key
#
# This function can activate or deactivate an IAM access key for the specified IAM
 user.
#
# Parameters:
#       -u user_name  -- The name of the user.
#       -k access_key -- The access key to update.
#       -a            -- Activate the selected access key.
#       -d            -- Deactivate the selected access key.
#
# Example:
#       # To deactivate the selected access key for IAM user Bob
#       iam_update_access_key -u Bob -k AKIAIOSFODNN7EXAMPLE -d
```

```
#
# Returns:
#       0 - If successful.
#       1 - If it fails.
###############################################################################
function iam_update_access_key() {
  local user_name access_key status response
  local option OPTARG # Required to use getopts command in a function.
  local activate_flag=false deactivate_flag=false

  # bashsupport disable=BP5008
  function usage() {
    echo "function iam_update_access_key"
    echo "Updates the status of an AWS Identity and Access Management (IAM) access
 key for the specified IAM user"
    echo "  -u user_name    The name of the user."
    echo "  -k access_key   The access key to update."
    echo "  -a              Activate the access key."
    echo "  -d              Deactivate the access key."
    echo ""
  }

  # Retrieve the calling parameters.
    while getopts "u:k:adh" option; do
      case "${option}" in
        u) user_name="${OPTARG}" ;;
        k) access_key="${OPTARG}" ;;
        a) activate_flag=true ;;
        d) deactivate_flag=true ;;
        h)
          usage
          return 0
          ;;
        \?)
          echo "Invalid parameter"
          usage
          return 1
          ;;
      esac
    done
    export OPTIND=1

   # Validate input parameters
    if [[ -z "$user_name" ]]; then
```

```
    errecho "ERROR: You must provide a username with the -u parameter."
    usage
    return 1
  fi

  if [[ -z "$access_key" ]]; then
    errecho "ERROR: You must provide an access key with the -k parameter."
    usage
    return 1
  fi

  # Ensure that only -a or -d is specified
  if [[ "$activate_flag" == true && "$deactivate_flag" == true ]]; then
    errecho "ERROR: You cannot specify both -a (activate) and -d (deactivate) at
the same time."
    usage
    return 1
  fi

  # If neither -a nor -d is provided, return an error
  if [[ "$activate_flag" == false && "$deactivate_flag" == false ]]; then
    errecho "ERROR: You must specify either -a (activate) or -d (deactivate)."
    usage
    return 1
  fi

  # Determine the status based on the flag
  if [[ "$activate_flag" == true ]]; then
    status="Active"
  elif [[ "$deactivate_flag" == true ]]; then
    status="Inactive"
  fi

  iecho "Parameters:\n"
  iecho "    Username:   $user_name"
  iecho "    Access key: $access_key"
  iecho "    New status: $status"
  iecho ""

  # Update the access key status
  response=$(aws iam update-access-key \
    --user-name "$user_name" \
    --access-key-id "$access_key" \
    --status "$status" 2>&1)
```

```
    local error_code=${?}

    if [[ $error_code -ne 0 ]]; then
      aws_cli_error_log $error_code
      errecho "ERROR: AWS reports update-access-key operation failed.\n$response"
      return 1
    fi

    iecho "update-access-key response: $response"
    iecho

    return 0
}
```

- Per i dettagli sull'API, consulta UpdateAccessKey AWS CLICommand Reference.

Scenari

Configurare il controllo degli accessi basato su attributi

L'esempio di codice seguente mostra come implementare il controllo degli accessi basato su attributi (ABAC) per DynamoDB.

- Creare una policy IAM per il controllo degli accessi basato su attributi (ABAC).
- Creare tabelle con tag per reparti diversi.
- Elencare e filtrare le tabelle in base ai tag.

AWS CLI con lo script Bash

Creare una policy IAM per il controllo degli accessi basato su attributi (ABAC).

```
# Step 1: Create a policy document for ABAC
cat > abac-policy.json << 'EOF'
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
```

```
        "dynamodb:GetItem",
        "dynamodb:BatchGetItem",
        "dynamodb:Query",
        "dynamodb:Scan"
      ],
      "Resource": "arn:aws:dynamodb:*:*:table/*",
      "Condition": {
        "StringEquals": {
          "aws:ResourceTag/Department": "${aws:PrincipalTag/Department}"
        }
      }
    },
    {
      "Effect": "Allow",
      "Action": [
        "dynamodb:PutItem",
        "dynamodb:UpdateItem",
        "dynamodb:DeleteItem",
        "dynamodb:BatchWriteItem"
      ],
      "Resource": "arn:aws:dynamodb:*:*:table/*",
      "Condition": {
        "StringEquals": {
          "aws:ResourceTag/Department": "${aws:PrincipalTag/Department}",
          "aws:ResourceTag/Environment": "Development"
        }
      }
    }
  ]
}
EOF

# Step 2: Create the IAM policy
aws iam create-policy \
    --policy-name DynamoDBDepartmentBasedAccess \
    --policy-document file://abac-policy.json
```

Creare tabelle con tag per reparti diversi.

```
# Create a DynamoDB table with tags for ABAC
aws dynamodb create-table \
    --table-name FinanceData \
```

```
    --attribute-definitions \
        AttributeName=RecordID,AttributeType=S \
    --key-schema \
        AttributeName=RecordID,KeyType=HASH \
    --billing-mode PAY_PER_REQUEST \
    --tags \
        Key=Department,Value=Finance \
        Key=Environment,Value=Development

# Create another table with different tags
aws dynamodb create-table \
    --table-name MarketingData \
    --attribute-definitions \
        AttributeName=RecordID,AttributeType=S \
    --key-schema \
        AttributeName=RecordID,KeyType=HASH \
    --billing-mode PAY_PER_REQUEST \
    --tags \
        Key=Department,Value=Marketing \
        Key=Environment,Value=Production
```

Elencare e filtrare le tabelle in base ai tag.

```
# List all DynamoDB tables
echo "Listing all tables:"
aws dynamodb list-tables

# Get ARNs for all tables
echo -e "\nGetting ARNs for all tables:"
TABLE_ARNS=$(aws dynamodb list-tables --query "TableNames[*]" --output text | xargs
 -I {} aws dynamodb describe-table --table-name {} --query "Table.TableArn" --output
 text)

# For each table ARN, list its tags
echo -e "\nListing tags for each table:"
for ARN in $TABLE_ARNS; do
    TABLE_NAME=$(echo $ARN | awk -F/ '{print $2}')
    echo -e "\nTags for table: $TABLE_NAME"
    aws dynamodb list-tags-of-resource --resource-arn $ARN
done

# Example: Find tables with a specific tag
```

```
echo -e "\nFinding tables with Environment=Production tag:"
for ARN in $TABLE_ARNS; do
    TABLE_NAME=$(echo $ARN | awk -F/ '{print $2}')
    TAGS=$(aws dynamodb list-tags-of-resource --resource-arn $ARN --query "Tags[?
Key=='Environment' && Value=='Production']" --output text)
    if [ ! -z "$TAGS" ]; then
        echo "Table with Production tag: $TABLE_NAME"
    fi
done
```

- Per informazioni dettagliate sull'API, consulta i seguenti argomenti nella documentazione di riferimento dei comandi della AWS CLI .
  - CreatePolicy
  - CreateTable
  - ListTables

Lavora con Streams e Time-to-Live

Il seguente esempio di codice mostra come gestire i flussi e le funzionalità di DynamoDB. Time-to-Live

- Creare una tabella con i flussi abilitati.
- Descrivere i flussi.
- Creare una funzione Lambda per l'elaborazione dei flussi.
- Abilitare TTL in una tabella.
- Aggiungere elementi con attributi TTL.
- Descrivere le impostazioni TTL.

AWS CLI con lo script Bash

Creare una tabella con i flussi abilitati.

```
# Create a table with DynamoDB Streams enabled
aws dynamodb create-table \
    --table-name StreamsDemo \
    --attribute-definitions \
        AttributeName=ID,AttributeType=S \
```

```
    --key-schema \
        AttributeName=ID,KeyType=HASH \
    --billing-mode PAY_PER_REQUEST \
    --stream-specification StreamEnabled=true,StreamViewType=NEW_AND_OLD_IMAGES
```

Descrivere i flussi.

```
# Get information about the stream
aws dynamodb describe-table \
    --table-name StreamsDemo \
    --query "Table.StreamSpecification"

# Get the stream ARN
STREAM_ARN=$(aws dynamodb describe-table \
    --table-name StreamsDemo \
    --query "Table.LatestStreamArn" \
    --output text)

echo "Stream ARN: $STREAM_ARN"

# Describe the stream
aws dynamodbstreams describe-stream \
    --stream-arn $STREAM_ARN
```

Crea una funzione Lambda per i flussi.

```
# Step 1: Create an IAM role for the Lambda function
cat > trust-policy.json << 'EOF'
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "lambda.amazonaws.com"
      },
      "Action": "sts:AssumeRole"
    }
  ]
}
EOF
```

```
aws iam create-role \
    --role-name DynamoDBStreamsLambdaRole \
    --assume-role-policy-document file://trust-policy.json

# Step 2: Attach permissions to the role
aws iam attach-role-policy \
    --role-name DynamoDBStreamsLambdaRole \
    --policy-arn arn:aws:iam::aws:policy/service-role/AWSLambdaDynamoDBExecutionRole

# Step 3: Create a Lambda function (code would be in a separate file)
echo "Lambda function creation would be done separately with appropriate code"

# Step 4: Create an event source mapping
echo "Example command to create event source mapping:"
echo "aws lambda create-event-source-mapping \\"
echo "    --function-name ProcessDynamoDBRecords \\"
echo "    --event-source $STREAM_ARN \\"
echo "    --batch-size 100 \\"
echo "    --starting-position LATEST"
```

Abilitare TTL in una tabella.

```
# Create a table for TTL demonstration
aws dynamodb create-table \
    --table-name TTLDemo \
    --attribute-definitions \
        AttributeName=ID,AttributeType=S \
    --key-schema \
        AttributeName=ID,KeyType=HASH \
    --billing-mode PAY_PER_REQUEST

# Wait for table to become active
aws dynamodb wait table-exists --table-name TTLDemo

# Enable TTL on the table
aws dynamodb update-time-to-live \
    --table-name TTLDemo \
    --time-to-live-specification "Enabled=true, AttributeName=ExpirationTime"
```

Aggiungere elementi con attributi TTL.

```
# Calculate expiration time (current time + 1 day in seconds)
EXPIRATION_TIME=$(date -d "+1 day" +%s)

# Add an item with TTL attribute
aws dynamodb put-item \
    --table-name TTLDemo \
    --item '{
        "ID": {"S": "item1"},
        "Data": {"S": "This item will expire in 1 day"},
        "ExpirationTime": {"N": "'$EXPIRATION_TIME'"}
    }'

# Add an item that expires in 1 hour
EXPIRATION_TIME_HOUR=$(date -d "+1 hour" +%s)
aws dynamodb put-item \
    --table-name TTLDemo \
    --item '{
        "ID": {"S": "item2"},
        "Data": {"S": "This item will expire in 1 hour"},
        "ExpirationTime": {"N": "'$EXPIRATION_TIME_HOUR'"}
    }'
```

Descrivere le impostazioni TTL.

```
# Describe TTL settings for a table
aws dynamodb describe-time-to-live \
    --table-name TTLDemo
```

- Per informazioni dettagliate sull'API, consulta i seguenti argomenti nella documentazione di riferimento dei comandi della AWS CLI .
  - AttachRolePolicy
  - CreateRole
  - CreateTable
  - DescribeTable
  - DescribeTimeToLive
  - PutItem
  - UpdateTimeToLive

AWS KMS esempi di utilizzo AWS CLI con lo script Bash

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando lo script AWS Command Line Interface with Bash con. AWS KMS

Scenari: esempi di codice che mostrano come eseguire un'attività specifica chiamando più funzioni all'interno dello stesso servizio o combinate con altri Servizi AWS.

Ogni esempio include un link al codice sorgente completo, dove è possibile trovare le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Scenari

Scenari

Utilizzare la crittografia delle tabelle

L'esempio di codice seguente mostra come gestire le opzioni di crittografia per le tabelle DynamoDB.

- Creare un tabella con la crittografia predefinita.
- Creare una tabella con una chiave gestita dal cliente (CMK).
- Aggiornare le impostazioni di crittografia delle tabelle.
- Descrivere la crittografia delle tabelle.

AWS CLI con lo script Bash

Creare un tabella con la crittografia predefinita.

```
# Create a table with default encryption (AWS owned key)
aws dynamodb create-table \
    --table-name CustomerData \
    --attribute-definitions \
        AttributeName=CustomerID,AttributeType=S \
    --key-schema \
        AttributeName=CustomerID,KeyType=HASH \
    --billing-mode PAY_PER_REQUEST \
    --sse-specification Enabled=true,SSEType=KMS
```

Creare una tabella con una chiave gestita dal cliente (CMK).

```
# Step 1: Create a customer managed key in KMS
aws kms create-key \
    --description "Key for DynamoDB table encryption" \
    --key-usage ENCRYPT_DECRYPT \
    --customer-master-key-spec SYMMETRIC_DEFAULT

# Store the key ID for later use
KEY_ID=$(aws kms list-keys --query "Keys[?contains(KeyArn, 'Key for
 DynamoDB')].KeyId" --output text)

# Step 2: Create a table with the customer managed key
aws dynamodb create-table \
    --table-name SensitiveData \
    --attribute-definitions \
        AttributeName=RecordID,AttributeType=S \
    --key-schema \
        AttributeName=RecordID,KeyType=HASH \
    --billing-mode PAY_PER_REQUEST \
    --sse-specification Enabled=true,SSEType=KMS,KMSMasterKeyId=$KEY_ID
```

Aggiornare la crittografia delle tabelle.

```
# Update a table to use a different KMS key
aws dynamodb update-table \
    --table-name CustomerData \
    --sse-specification Enabled=true,SSEType=KMS,KMSMasterKeyId=$KEY_ID
```

Descrivere la crittografia delle tabelle.

```
# Describe the table to see encryption settings
aws dynamodb describe-table \
    --table-name CustomerData \
    --query "Table.SSEDescription"
```

- Per informazioni dettagliate sull'API, consulta i seguenti argomenti nella documentazione di riferimento dei comandi della AWS CLI .

- CreateKey
- CreateTable
- DescribeTable
- UpdateTable

Esempi AWS CLI di Lightsail con lo script Bash

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando lo script AWS Command Line Interface with Bash con Lightsail.

Scenari: esempi di codice che mostrano come eseguire un'attività specifica chiamando più funzioni all'interno dello stesso servizio o combinate con altri Servizi AWS.

Ogni esempio include un link al codice sorgente completo, dove è possibile trovare le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Scenari

Scenari

Inizia a usare Lightsail

L'esempio di codice seguente mostra come:

- Esplorare i blueprint e i bundle disponibili per la creazione delle istanze
- Crea un'istanza Lightsail con Amazon Linux 2023
- Monitorare lo stato dell'istanza fino all'esecuzione
- Scaricare la coppia di chiavi predefinita per l'accesso SSH
- Creare e collegare un disco di archiviazione a blocchi aggiuntivo
- Creare uno snapshot dell'istanza peri l backup
- Eseguire la pulizia delle risorse per evitare addebiti continui

AWS CLI con script Bash

> **Note**
>
> C'è altro da fare. GitHub Trova l'esempio completo e scopri come eseguire la configurazione e l'esecuzione nel repository dei tutorial sugli esempi di codice per gli sviluppatori.

```
#!/bin/bash

# Amazon Lightsail Getting Started CLI Script
# This script demonstrates how to create and manage Lightsail resources using the
 AWS CLI

# FIXES APPLIED:
# 1. Added polling mechanism to check disk state before attaching
# 2. Added polling mechanism to check snapshot state before proceeding with cleanup
# 3. Set AWS_REGION variable to us-west-2 for consistent region usage

# Set AWS region
export AWS_REGION="us-west-2"
echo "Using AWS region: $AWS_REGION"

# Set up logging
LOG_FILE="lightsail-script.log"
exec > >(tee -a "$LOG_FILE") 2>&1

echo "Starting Lightsail Getting Started script at $(date)"

# Error handling function
handle_error() {
  echo "ERROR: $1"
  echo "Attempting to clean up resources..."
  cleanup_resources
  exit 1
}

# Function to check if a command succeeded
check_status() {
  if [ $? -ne 0 ]; then
    handle_error "$1"
```

```
  fi
}

# Generate a random identifier for resource names
RANDOM_ID=$(openssl rand -hex 4)
INSTANCE_NAME="LightsailInstance-${RANDOM_ID}"
DISK_NAME="LightsailDisk-${RANDOM_ID}"
SNAPSHOT_NAME="LightsailSnapshot-${RANDOM_ID}"

# Array to track created resources
declare -a CREATED_RESOURCES

# Function to add a resource to the tracking array
track_resource() {
  CREATED_RESOURCES+=("$1:$2")
  echo "Created $1: $2"
}

# Function to clean up resources
cleanup_resources() {
  echo "Resources created by this script:"
  for resource in "${CREATED_RESOURCES[@]}"; do
    echo "  $resource"
  done

  # Reverse the array to delete resources in reverse order
  for (( idx=${#CREATED_RESOURCES[@]}-1 ; idx>=0 ; idx-- )); do
    IFS=':' read -r type name <<< "${CREATED_RESOURCES[idx]}"

    case "$type" in
      "instance_snapshot")
        echo "Deleting instance snapshot: $name"
        aws lightsail delete-instance-snapshot --instance-snapshot-name "$name" --
region $AWS_REGION
        ;;
      "disk_snapshot")
        echo "Deleting disk snapshot: $name"
        aws lightsail delete-disk-snapshot --disk-snapshot-name "$name" --region
 $AWS_REGION
        ;;
      "disk")
        echo "Detaching disk: $name"
        aws lightsail detach-disk --disk-name "$name" --region $AWS_REGION
        sleep 10 # Wait for detach to complete
```

```
        echo "Deleting disk: $name"
        aws lightsail delete-disk --disk-name "$name" --region $AWS_REGION
        ;;
      "instance")
        echo "Deleting instance: $name"
        # Check instance state before attempting to delete
        INSTANCE_STATE=$(aws lightsail get-instance-state --instance-name "$name" --
region $AWS_REGION --query 'state.name' --output text 2>/dev/null)
        if [ "$INSTANCE_STATE" == "pending" ]; then
          echo "Instance is in pending state. Waiting for it to be ready before
 deleting..."
          MAX_WAIT=30
          WAITED=0
          while [ "$INSTANCE_STATE" == "pending" ] && [ $WAITED -lt $MAX_WAIT ]; do
            sleep 10
            WAITED=$((WAITED+1))
            INSTANCE_STATE=$(aws lightsail get-instance-state --instance-name
 "$name" --region $AWS_REGION --query 'state.name' --output text 2>/dev/null)
            echo "Instance state: $INSTANCE_STATE"
          done
        fi
        aws lightsail delete-instance --instance-name "$name" --region $AWS_REGION
        ;;
    esac
  done

  echo "Cleanup completed"
}

# Step 1: Verify AWS CLI configuration
echo "Step 1: Verifying AWS CLI configuration"
aws configure list
check_status "Failed to verify AWS CLI configuration"

# Step 2: Get available blueprints and bundles
echo "Step 2: Getting available blueprints and bundles"
echo "Available blueprints (showing first 5):"
aws lightsail get-blueprints --region $AWS_REGION --query 'blueprints[0:5].
[blueprintId,name]' --output table
check_status "Failed to get blueprints"

echo "Available bundles (showing first 5):"
aws lightsail get-bundles --region $AWS_REGION --query 'bundles[0:5].
[bundleId,name,price]' --output table
```

```
check_status "Failed to get bundles"

# Get available regions and availability zones
echo "Getting available regions and availability zones"
# Use a specific availability zone in us-west-2 region
AVAILABILITY_ZONE="us-west-2a"
echo "Using availability zone: $AVAILABILITY_ZONE"

# Step 3: Create a Lightsail instance
echo "Step 3: Creating Lightsail instance: $INSTANCE_NAME"
aws lightsail create-instances \
  --instance-names "$INSTANCE_NAME" \
  --availability-zone "$AVAILABILITY_ZONE" \
  --blueprint-id amazon_linux_2023 \
  --bundle-id nano_3_0 \
  --region $AWS_REGION
check_status "Failed to create Lightsail instance"
track_resource "instance" "$INSTANCE_NAME"

# Wait for the instance to be in a running state
echo "Waiting for instance to be in running state..."
# Wait for the instance to be ready (polling approach)
MAX_ATTEMPTS=30
ATTEMPTS=0
while [ $ATTEMPTS -lt $MAX_ATTEMPTS ]; do
  STATUS=$(aws lightsail get-instance-state --instance-name "$INSTANCE_NAME" --
region $AWS_REGION --query 'state.name' --output text)
  if [ "$STATUS" == "running" ]; then
    echo "Instance is now running"
    break
  fi
  echo "Instance status: $STATUS. Waiting..."
  ATTEMPTS=$((ATTEMPTS+1))
  sleep 10
done

if [ $ATTEMPTS -eq $MAX_ATTEMPTS ]; then
  handle_error "Instance failed to reach running state after 5 minutes"
fi

# Get instance details
echo "Getting instance details"
INSTANCE_IP=$(aws lightsail get-instance --instance-name "$INSTANCE_NAME" --region
 $AWS_REGION --query 'instance.publicIpAddress' --output text)
```

```
check_status "Failed to get instance IP address"
echo "Instance IP address: $INSTANCE_IP"

# Step 4: Download the default key pair
echo "Step 4: Downloading default key pair"
KEY_FILE="lightsail_key_${RANDOM_ID}.pem"
aws lightsail download-default-key-pair --region $AWS_REGION --output text >
 "$KEY_FILE"
check_status "Failed to download key pair"
chmod 400 "$KEY_FILE"
check_status "Failed to set permissions on key pair"
echo "Key pair downloaded to $KEY_FILE"

echo "To connect to your instance, use:"
echo "ssh -i $KEY_FILE ec2-user@$INSTANCE_IP"

# Step 5: Create a block storage disk
echo "Step 5: Creating block storage disk: $DISK_NAME"
aws lightsail create-disk \
  --disk-name "$DISK_NAME" \
  --availability-zone "$AVAILABILITY_ZONE" \
  --size-in-gb 8 \
  --region $AWS_REGION
check_status "Failed to create disk"
track_resource "disk" "$DISK_NAME"

# FIX: Wait for the disk to be available using polling instead of fixed sleep
echo "Waiting for disk to be available..."
MAX_ATTEMPTS=30
ATTEMPTS=0
while [ $ATTEMPTS -lt $MAX_ATTEMPTS ]; do
  DISK_STATE=$(aws lightsail get-disk --disk-name "$DISK_NAME" --region $AWS_REGION
 --query 'disk.state' --output text 2>/dev/null)
  if [ "$DISK_STATE" == "available" ]; then
    echo "Disk is now available"
    break
  fi
  echo "Disk status: $DISK_STATE. Waiting..."
  ATTEMPTS=$((ATTEMPTS+1))
  sleep 10
done

if [ $ATTEMPTS -eq $MAX_ATTEMPTS ]; then
  handle_error "Disk failed to become available after 5 minutes"
```

```
fi

# Attach the disk to the instance
echo "Attaching disk to instance"
aws lightsail attach-disk \
  --disk-name "$DISK_NAME" \
  --instance-name "$INSTANCE_NAME" \
  --disk-path /dev/xvdf \
  --region $AWS_REGION
check_status "Failed to attach disk to instance"

echo "Disk attached. To format and mount the disk, connect to your instance and
 run:"
echo "sudo mkfs -t ext4 /dev/xvdf"
echo "sudo mkdir -p /mnt/my-data"
echo "sudo mount /dev/xvdf /mnt/my-data"
echo "sudo chown ec2-user:ec2-user /mnt/my-data"

# Step 6: Create a snapshot of the instance
echo "Step 6: Creating snapshot of the instance: $SNAPSHOT_NAME"
aws lightsail create-instance-snapshot \
  --instance-name "$INSTANCE_NAME" \
  --instance-snapshot-name "$SNAPSHOT_NAME" \
  --region $AWS_REGION
check_status "Failed to create instance snapshot"
track_resource "instance_snapshot" "$SNAPSHOT_NAME"

# FIX: Wait for the snapshot to complete using polling instead of fixed sleep
echo "Waiting for snapshot to complete... (this may take several minutes)"
MAX_ATTEMPTS=60  # Increased timeout for snapshot creation
ATTEMPTS=0
while [ $ATTEMPTS -lt $MAX_ATTEMPTS ]; do
  SNAPSHOT_STATE=$(aws lightsail get-instance-snapshot --instance-snapshot-name
 "$SNAPSHOT_NAME" --region $AWS_REGION --query 'instanceSnapshot.state' --output
 text 2>/dev/null)
  if [ "$SNAPSHOT_STATE" == "completed" ]; then
    echo "Snapshot creation completed"
    break
  fi
  echo "Snapshot status: $SNAPSHOT_STATE. Waiting... ($ATTEMPTS/$MAX_ATTEMPTS)"
  ATTEMPTS=$((ATTEMPTS+1))
  sleep 10
done
```

```
if [ $ATTEMPTS -eq $MAX_ATTEMPTS ]; then
  echo "Warning: Snapshot creation is taking longer than expected but will continue
 in the background."
  echo "You can check its status later with: aws lightsail get-instance-snapshot --
instance-snapshot-name $SNAPSHOT_NAME --region $AWS_REGION"
fi

# Step 7: Clean up resources
echo "Step 7: Clean up resources"
echo "The script has created the following resources:"
for resource in "${CREATED_RESOURCES[@]}"; do
  echo "  $resource"
done

read -p "Do you want to clean up these resources? (y/n): " CLEANUP_CONFIRM
if [[ "$CLEANUP_CONFIRM" == "y" || "$CLEANUP_CONFIRM" == "Y" ]]; then
  cleanup_resources
else
  echo "Resources will not be cleaned up. You can manually delete them later."
  echo "To clean up manually, use the following commands:"
  echo "aws lightsail delete-instance-snapshot --instance-snapshot-name
 $SNAPSHOT_NAME --region $AWS_REGION"
  echo "aws lightsail detach-disk --disk-name $DISK_NAME --region $AWS_REGION"
  echo "aws lightsail delete-disk --disk-name $DISK_NAME --region $AWS_REGION"
  echo "aws lightsail delete-instance --instance-name $INSTANCE_NAME --region
 $AWS_REGION"
fi

echo "Script completed at $(date)"
```

- Per informazioni dettagliate sull'API, consulta i seguenti argomenti nella documentazione di riferimento dei comandi della AWS CLI .
  - AttachDisk
  - CreateDisk
  - CreateInstanceSnapshot
  - CreateInstances
  - DeleteDisk
  - DeleteInstance
  - DeleteInstanceSnapshot

- DetachDisk
- DownloadDefaultKeyPair
- GetBlueprints
- GetBundles
- GetDisk
- GetInstance
- GetInstanceSnapshot
- GetInstanceState

Esempi di Amazon S3 che utilizzano lo script AWS CLI Bash

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando lo script AWS Command Line Interface with Bash con Amazon S3.

Nozioni di base: esempi di codice che mostrano come eseguire le operazioni essenziali all'interno di un servizio.

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Scenari: esempi di codice che mostrano come eseguire un'attività specifica chiamando più funzioni all'interno dello stesso servizio o combinate con altri Servizi AWS.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Nozioni di base
- Azioni
- Scenari

Nozioni di base

Informazioni di base

L'esempio di codice seguente mostra come:

- Crea un bucket e carica un file in tale bucket.
- Scaricare un oggetto da un bucket.
- Copiare un oggetto in una sottocartella in un bucket.
- Elencare gli oggetti in un bucket.
- Elimina il bucket e tutti gli oggetti in esso contenuti.

AWS CLI con lo script Bash

> **Note**
>
> C'è altro da fare. GitHub Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codice AWS.

```
###############################################################################
# function s3_getting_started
#
# This function creates, copies, and deletes S3 buckets and objects.
#
# Returns:
#       0 - If successful.
#       1 - If an error occurred.
###############################################################################
function s3_getting_started() {
  {
    if [ "$BUCKET_OPERATIONS_SOURCED" != "True" ]; then
      cd bucket-lifecycle-operations || exit

      source ./bucket_operations.sh
      cd ..
    fi
  }

  echo_repeat "*" 88
  echo "Welcome to the Amazon S3 getting started demo."
  echo_repeat "*" 88
    echo "A unique bucket will be created by appending a Universally Unique
 Identifier to a bucket name prefix."
    echo -n "Enter a prefix for the S3 bucket that will be used in this demo: "
    get_input
```

```
    bucket_name_prefix=$get_input_result
  local bucket_name
  bucket_name=$(generate_random_name "$bucket_name_prefix")

  local region_code
  region_code=$(aws configure get region)

  if create_bucket -b "$bucket_name" -r "$region_code"; then
    echo "Created demo bucket named $bucket_name"
  else
    errecho "The bucket failed to create. This demo will exit."
    return 1
  fi

  local file_name
  while [ -z "$file_name" ]; do
    echo -n "Enter a file you want to upload to your bucket: "
    get_input
    file_name=$get_input_result

    if [ ! -f "$file_name" ]; then
      echo "Could not find file $file_name. Are you sure it exists?"
      file_name=""
    fi
  done

  local key
  key="$(basename "$file_name")"

  local result=0
  if copy_file_to_bucket "$bucket_name" "$file_name" "$key"; then
    echo "Uploaded file $file_name into bucket $bucket_name with key $key."
  else
    result=1
  fi

  local destination_file
  destination_file="$file_name.download"
  if yes_no_input "Would you like to download $key to the file $destination_file?
(y/n) "; then
    if download_object_from_bucket "$bucket_name" "$destination_file" "$key"; then
      echo "Downloaded $key in the bucket $bucket_name to the file
$destination_file."
    else
```

```
      result=1
    fi
  fi

  if yes_no_input "Would you like to copy $key a new object key in your bucket? (y/
n) "; then
    local to_key
    to_key="demo/$key"
    if copy_item_in_bucket "$bucket_name" "$key" "$to_key"; then
      echo "Copied $key in the bucket $bucket_name to the  $to_key."
    else
      result=1
    fi
  fi

  local bucket_items
  bucket_items=$(list_items_in_bucket "$bucket_name")

  # shellcheck disable=SC2181
  if [[ $? -ne 0 ]]; then
    result=1
  fi

  echo "Your bucket contains the following items."
  echo -e "Name\t\tSize"
  echo "$bucket_items"

  if yes_no_input "Delete the bucket, $bucket_name, as well as the objects in it?
 (y/n) "; then
    bucket_items=$(echo "$bucket_items" | cut -f 1)

    if delete_items_in_bucket "$bucket_name" "$bucket_items"; then
      echo "The following items were deleted from the bucket $bucket_name"
      echo "$bucket_items"
    else
      result=1
    fi

    if delete_bucket "$bucket_name"; then
      echo "Deleted the bucket $bucket_name"
    else
      result=1
    fi
  fi
```

```
  return $result
}
```

Funzioni Amazon S3 utilizzate in questo scenario.

```
###############################################################################
# function create-bucket
#
# This function creates the specified bucket in the specified AWS Region, unless
# it already exists.
#
# Parameters:
#       -b bucket_name  -- The name of the bucket to create.
#       -r region_code  -- The code for an AWS Region in which to
#                          create the bucket.
#
# Returns:
#       The URL of the bucket that was created.
#     And:
#       0 - If successful.
#       1 - If it fails.
###############################################################################
function create_bucket() {
  local bucket_name region_code response
  local option OPTARG # Required to use getopts command in a function.

  # bashsupport disable=BP5008
  function usage() {
    echo "function create_bucket"
    echo "Creates an Amazon S3 bucket. You must supply a bucket name:"
    echo "  -b bucket_name    The name of the bucket. It must be globally unique."
    echo "  [-r region_code]    The code for an AWS Region in which the bucket is
 created."
    echo ""
  }

  # Retrieve the calling parameters.
  while getopts "b:r:h" option; do
    case "${option}" in
      b) bucket_name="${OPTARG}" ;;
      r) region_code="${OPTARG}" ;;
```

```
      h)
        usage
        return 0
        ;;
      \?)
        echo "Invalid parameter"
        usage
        return 1
        ;;
    esac
  done

  if [[ -z "$bucket_name" ]]; then
    errecho "ERROR: You must provide a bucket name with the -b parameter."
    usage
    return 1
  fi

  local bucket_config_arg
  # A location constraint for "us-east-1" returns an error.
  if [[ -n "$region_code" ]] && [[ "$region_code" != "us-east-1" ]]; then
    bucket_config_arg="--create-bucket-configuration LocationConstraint=
$region_code"
  fi

  iecho "Parameters:\n"
  iecho "    Bucket name:   $bucket_name"
  iecho "    Region code:   $region_code"
  iecho ""

  # If the bucket already exists, we don't want to try to create it.
  if (bucket_exists "$bucket_name"); then
    errecho "ERROR: A bucket with that name already exists. Try again."
    return 1
  fi

  # shellcheck disable=SC2086
  response=$(aws s3api create-bucket \
    --bucket "$bucket_name" \
    $bucket_config_arg)

  # shellcheck disable=SC2181
  if [[ ${?} -ne 0 ]]; then
    errecho "ERROR: AWS reports create-bucket operation failed.\n$response"
```

```
    return 1
  fi
}

################################################################################
# function copy_file_to_bucket
#
# This function creates a file in the specified bucket.
#
# Parameters:
#       $1 - The name of the bucket to copy the file to.
#       $2 - The path and file name of the local file to copy to the bucket.
#       $3 - The key (name) to call the copy of the file in the bucket.
#
# Returns:
#       0 - If successful.
#       1 - If it fails.
################################################################################
function copy_file_to_bucket() {
  local response bucket_name source_file destination_file_name
  bucket_name=$1
  source_file=$2
  destination_file_name=$3

  response=$(aws s3api put-object \
    --bucket "$bucket_name" \
    --body "$source_file" \
    --key "$destination_file_name")

  # shellcheck disable=SC2181
  if [[ ${?} -ne 0 ]]; then
    errecho "ERROR: AWS reports put-object operation failed.\n$response"
    return 1
  fi
}

################################################################################
# function download_object_from_bucket
#
# This function downloads an object in a bucket to a file.
#
# Parameters:
#       $1 - The name of the bucket to download the object from.
#       $2 - The path and file name to store the downloaded bucket.
```

```
#       $3 - The key (name) of the object in the bucket.
#
# Returns:
#       0 - If successful.
#       1 - If it fails.
###############################################################################
function download_object_from_bucket() {
  local bucket_name=$1
  local destination_file_name=$2
  local object_name=$3
  local response

  response=$(aws s3api get-object \
    --bucket "$bucket_name" \
    --key "$object_name" \
    "$destination_file_name")

  # shellcheck disable=SC2181
  if [[ ${?} -ne 0 ]]; then
    errecho "ERROR: AWS reports put-object operation failed.\n$response"
    return 1
  fi
}

###############################################################################
# function copy_item_in_bucket
#
# This function creates a copy of the specified file in the same bucket.
#
# Parameters:
#       $1 - The name of the bucket to copy the file from and to.
#       $2 - The key of the source file to copy.
#       $3 - The key of the destination file.
#
# Returns:
#       0 - If successful.
#       1 - If it fails.
###############################################################################
function copy_item_in_bucket() {
  local bucket_name=$1
  local source_key=$2
  local destination_key=$3
  local response
```

```
  response=$(aws s3api copy-object \
    --bucket "$bucket_name" \
    --copy-source "$bucket_name/$source_key" \
    --key "$destination_key")

  # shellcheck disable=SC2181
  if [[ $? -ne 0 ]]; then
    errecho "ERROR:  AWS reports s3api copy-object operation failed.\n$response"
    return 1
  fi
}

################################################################################
# function list_items_in_bucket
#
# This function displays a list of the files in the bucket with each file's
# size. The function uses the --query parameter to retrieve only the key and
# size fields from the Contents collection.
#
# Parameters:
#       $1 - The name of the bucket.
#
# Returns:
#       The list of files in text format.
#     And:
#       0 - If successful.
#       1 - If it fails.
################################################################################
function list_items_in_bucket() {
  local bucket_name=$1
  local response

  response=$(aws s3api list-objects \
    --bucket "$bucket_name" \
    --output text \
    --query 'Contents[].{Key: Key, Size: Size}')

  # shellcheck disable=SC2181
  if [[ ${?} -eq 0 ]]; then
    echo "$response"
  else
    errecho "ERROR: AWS reports s3api list-objects operation failed.\n$response"
    return 1
  fi
```

```
}

###############################################################################
# function delete_items_in_bucket
#
# This function deletes the specified list of keys from the specified bucket.
#
# Parameters:
#       $1 - The name of the bucket.
#       $2 - A list of keys in the bucket to delete.

# Returns:
#       0 - If successful.
#       1 - If it fails.
###############################################################################
function delete_items_in_bucket() {
  local bucket_name=$1
  local keys=$2
  local response

  # Create the JSON for the items to delete.
  local delete_items
  delete_items="{\"Objects\":["
  for key in $keys; do
    delete_items="$delete_items{\"Key\": \"$key\"},"
  done
  delete_items=${delete_items%?} # Remove the final comma.
  delete_items="$delete_items]}"

  response=$(aws s3api delete-objects \
    --bucket "$bucket_name" \
    --delete "$delete_items")

  # shellcheck disable=SC2181
  if [[ $? -ne 0 ]]; then
    errecho "ERROR:  AWS reports s3api delete-object operation failed.\n$response"
    return 1
  fi
}

###############################################################################
# function delete_bucket
#
# This function deletes the specified bucket.
```

```
#
# Parameters:
#       $1 - The name of the bucket.

# Returns:
#       0 - If successful.
#       1 - If it fails.
###############################################################################
function delete_bucket() {
  local bucket_name=$1
  local response

  response=$(aws s3api delete-bucket \
    --bucket "$bucket_name")

  # shellcheck disable=SC2181
  if [[ $? -ne 0 ]]; then
    errecho "ERROR: AWS reports s3api delete-bucket failed.\n$response"
    return 1
  fi
}
```

- Per informazioni dettagliate sull'API, consulta i seguenti argomenti nella documentazione di riferimento dei comandi della AWS CLI .
  - CopyObject
  - CreateBucket
  - DeleteBucket
  - DeleteObjects
  - GetObject
  - ListObjectsV2
  - PutObject

Azioni

`CopyObject`

Il seguente esempio di codice mostra come usare`CopyObject`.

AWS CLI con lo script Bash

> **Note**
>
> C'è altro da fare. GitHub Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codice AWS.

```
#############################################################################
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
#############################################################################
function errecho() {
  printf "%s\n" "$*" 1>&2
}

#############################################################################
# function copy_item_in_bucket
#
# This function creates a copy of the specified file in the same bucket.
#
# Parameters:
#       $1 - The name of the bucket to copy the file from and to.
#       $2 - The key of the source file to copy.
#       $3 - The key of the destination file.
#
# Returns:
#       0 - If successful.
#       1 - If it fails.
#############################################################################
function copy_item_in_bucket() {
  local bucket_name=$1
  local source_key=$2
  local destination_key=$3
  local response

  response=$(aws s3api copy-object \
    --bucket "$bucket_name" \
    --copy-source "$bucket_name/$source_key" \
    --key "$destination_key")
```

```
  # shellcheck disable=SC2181
  if [[ $? -ne 0 ]]; then
    errecho "ERROR:  AWS reports s3api copy-object operation failed.\n$response"
    return 1
  fi
}
```

- Per i dettagli sull'API, consulta CopyObject AWS CLICommand Reference.

`CreateBucket`

Il seguente esempio di codice mostra come utilizzare`CreateBucket`.

AWS CLI con lo script Bash

> **Note**
>
> C'è altro da fare. GitHub Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codice AWS.

```
##############################################################################
# function iecho
#
# This function enables the script to display the specified text only if
# the global variable $VERBOSE is set to true.
##############################################################################
function iecho() {
  if [[ $VERBOSE == true ]]; then
    echo "$@"
  fi
}

##############################################################################
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
##############################################################################
function errecho() {
  printf "%s\n" "$*" 1>&2
}
```

```
###############################################################################
# function create-bucket
#
# This function creates the specified bucket in the specified AWS Region, unless
# it already exists.
#
# Parameters:
#       -b bucket_name  -- The name of the bucket to create.
#       -r region_code  -- The code for an AWS Region in which to
#                          create the bucket.
#
# Returns:
#       The URL of the bucket that was created.
#     And:
#       0 - If successful.
#       1 - If it fails.
###############################################################################
function create_bucket() {
  local bucket_name region_code response
  local option OPTARG # Required to use getopts command in a function.

  # bashsupport disable=BP5008
  function usage() {
    echo "function create_bucket"
    echo "Creates an Amazon S3 bucket. You must supply a bucket name:"
    echo "  -b bucket_name    The name of the bucket. It must be globally unique."
    echo "  [-r region_code]    The code for an AWS Region in which the bucket is
 created."
    echo ""
  }

  # Retrieve the calling parameters.
  while getopts "b:r:h" option; do
    case "${option}" in
      b) bucket_name="${OPTARG}" ;;
      r) region_code="${OPTARG}" ;;
      h)
        usage
        return 0
        ;;
      \?)
        echo "Invalid parameter"
        usage
```

```
        return 1
        ;;
    esac
  done

  if [[ -z "$bucket_name" ]]; then
    errecho "ERROR: You must provide a bucket name with the -b parameter."
    usage
    return 1
  fi

  local bucket_config_arg
  # A location constraint for "us-east-1" returns an error.
  if [[ -n "$region_code" ]] && [[ "$region_code" != "us-east-1" ]]; then
    bucket_config_arg="--create-bucket-configuration LocationConstraint=
$region_code"
  fi

  iecho "Parameters:\n"
  iecho "    Bucket name:   $bucket_name"
  iecho "    Region code:   $region_code"
  iecho ""

  # If the bucket already exists, we don't want to try to create it.
  if (bucket_exists "$bucket_name"); then
    errecho "ERROR: A bucket with that name already exists. Try again."
    return 1
  fi

  # shellcheck disable=SC2086
  response=$(aws s3api create-bucket \
    --bucket "$bucket_name" \
    $bucket_config_arg)

  # shellcheck disable=SC2181
  if [[ ${?} -ne 0 ]]; then
    errecho "ERROR: AWS reports create-bucket operation failed.\n$response"
    return 1
  fi
}
```

- Per i dettagli sull'API, consulta CreateBucket AWS CLICommand Reference.

`DeleteBucket`

Il seguente esempio di codice mostra come utilizzare`DeleteBucket`.

AWS CLI con lo script Bash

> **Note**
>
> C'è altro da fare. GitHub Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codice AWS.

```
##############################################################################
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
##############################################################################
function errecho() {
  printf "%s\n" "$*" 1>&2
}

##############################################################################
# function delete_bucket
#
# This function deletes the specified bucket.
#
# Parameters:
#       $1 - The name of the bucket.

# Returns:
#       0 - If successful.
#       1 - If it fails.
##############################################################################
function delete_bucket() {
  local bucket_name=$1
  local response

  response=$(aws s3api delete-bucket \
    --bucket "$bucket_name")

  # shellcheck disable=SC2181
  if [[ $? -ne 0 ]]; then
    errecho "ERROR: AWS reports s3api delete-bucket failed.\n$response"
```

```
    return 1
  fi
}
```

- Per i dettagli sull'API, consulta DeleteBucket AWS CLICommand Reference.

`DeleteObject`

Il seguente esempio di codice mostra come utilizzare`DeleteObject`.

AWS CLI con lo script Bash

> **Note**
>
> C'è altro da fare. GitHub Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codice AWS.

```
##############################################################################
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
##############################################################################
function errecho() {
  printf "%s\n" "$*" 1>&2
}

##############################################################################
# function delete_item_in_bucket
#
# This function deletes the specified file from the specified bucket.
#
# Parameters:
#       $1 - The name of the bucket.
#       $2 - The key (file name) in the bucket to delete.

# Returns:
#       0 - If successful.
#       1 - If it fails.
##############################################################################
```

```
function delete_item_in_bucket() {
  local bucket_name=$1
  local key=$2
  local response

  response=$(aws s3api delete-object \
    --bucket "$bucket_name" \
    --key "$key")

  # shellcheck disable=SC2181
  if [[ $? -ne 0 ]]; then
    errecho "ERROR:  AWS reports s3api delete-object operation failed.\n$response"
    return 1
  fi
}
```

- Per i dettagli sull'API, consulta DeleteObject AWS CLICommand Reference.

`DeleteObjects`

Il seguente esempio di codice mostra come utilizzare`DeleteObjects`.

AWS CLI con lo script Bash

> **Note**
>
> C'è altro da fare. GitHub Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codice AWS.

```
##############################################################################
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
##############################################################################
function errecho() {
  printf "%s\n" "$*" 1>&2
}

##############################################################################
```

```
# function delete_items_in_bucket
#
# This function deletes the specified list of keys from the specified bucket.
#
# Parameters:
#       $1 - The name of the bucket.
#       $2 - A list of keys in the bucket to delete.

# Returns:
#       0 - If successful.
#       1 - If it fails.
###############################################################################
function delete_items_in_bucket() {
  local bucket_name=$1
  local keys=$2
  local response

  # Create the JSON for the items to delete.
  local delete_items
  delete_items="{\"Objects\":["
  for key in $keys; do
    delete_items="$delete_items{\"Key\": \"$key\"},"
  done
  delete_items=${delete_items%?} # Remove the final comma.
  delete_items="$delete_items]}"

  response=$(aws s3api delete-objects \
    --bucket "$bucket_name" \
    --delete "$delete_items")

  # shellcheck disable=SC2181
  if [[ $? -ne 0 ]]; then
    errecho "ERROR:  AWS reports s3api delete-object operation failed.\n$response"
    return 1
  fi
}
```

- Per i dettagli sull'API, consulta DeleteObjects AWS CLICommand Reference.

`GetObject`

Il seguente esempio di codice mostra come utilizzare`GetObject`.

AWS CLI con lo script Bash

> **Note**
>
> C'è altro da fare. GitHub Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codice AWS.

```
#############################################################################
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
#############################################################################
function errecho() {
  printf "%s\n" "$*" 1>&2
}

#############################################################################
# function download_object_from_bucket
#
# This function downloads an object in a bucket to a file.
#
# Parameters:
#       $1 - The name of the bucket to download the object from.
#       $2 - The path and file name to store the downloaded bucket.
#       $3 - The key (name) of the object in the bucket.
#
# Returns:
#       0 - If successful.
#       1 - If it fails.
#############################################################################
function download_object_from_bucket() {
  local bucket_name=$1
  local destination_file_name=$2
  local object_name=$3
  local response

  response=$(aws s3api get-object \
    --bucket "$bucket_name" \
    --key "$object_name" \
    "$destination_file_name")
```

```
  # shellcheck disable=SC2181
  if [[ ${?} -ne 0 ]]; then
    errecho "ERROR: AWS reports put-object operation failed.\n$response"
    return 1
  fi
}
```

- Per i dettagli sull'API, consulta GetObject AWS CLICommand Reference.

`HeadBucket`

Il seguente esempio di codice mostra come utilizzare`HeadBucket`.

AWS CLI con lo script Bash

> **Note**
>
> C'è altro da fare. GitHub Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codice AWS.

```
###############################################################################
# function bucket_exists
#
# This function checks to see if the specified bucket already exists.
#
# Parameters:
#       $1 - The name of the bucket to check.
#
# Returns:
#       0 - If the bucket already exists.
#       1 - If the bucket doesn't exist.
###############################################################################
function bucket_exists() {
  local bucket_name
  bucket_name=$1

  # Check whether the bucket already exists.
  # We suppress all output - we're interested only in the return code.
```

```
  if aws s3api head-bucket \
    --bucket "$bucket_name" \
    >/dev/null 2>&1; then
    return 0 # 0 in Bash script means true.
  else
    return 1 # 1 in Bash script means false.
  fi
}
```

- Per i dettagli sull'API, consulta HeadBucket AWS CLICommand Reference.

`ListObjectsV2`

Il seguente esempio di codice mostra come utilizzare`ListObjectsV2`.

AWS CLI con lo script Bash

> **Note**
>
> C'è altro da fare. GitHub Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codice AWS.

```
#############################################################################
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
#############################################################################
function errecho() {
  printf "%s\n" "$*" 1>&2
}

#############################################################################
# function list_items_in_bucket
#
# This function displays a list of the files in the bucket with each file's
# size. The function uses the --query parameter to retrieve only the key and
# size fields from the Contents collection.
#
# Parameters:
#       $1 - The name of the bucket.
```

```
#
# Returns:
#       The list of files in text format.
#     And:
#       0 - If successful.
#       1 - If it fails.
###############################################################################
function list_items_in_bucket() {
  local bucket_name=$1
  local response

  response=$(aws s3api list-objects \
    --bucket "$bucket_name" \
    --output text \
    --query 'Contents[].{Key: Key, Size: Size}')

  # shellcheck disable=SC2181
  if [[ ${?} -eq 0 ]]; then
    echo "$response"
  else
    errecho "ERROR: AWS reports s3api list-objects operation failed.\n$response"
    return 1
  fi
}
```

- Per i dettagli sull'API, consulta ListObjectsV2 in AWS CLI Command Reference.

`PutObject`

Il seguente esempio di codice mostra come utilizzare`PutObject`.

AWS CLI con lo script Bash

> **Note**
>
> C'è altro da fare. GitHub Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codice AWS.

```
###############################################################################
```

```
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
###############################################################################
function errecho() {
  printf "%s\n" "$*" 1>&2
}

###############################################################################
# function copy_file_to_bucket
#
# This function creates a file in the specified bucket.
#
# Parameters:
#       $1 - The name of the bucket to copy the file to.
#       $2 - The path and file name of the local file to copy to the bucket.
#       $3 - The key (name) to call the copy of the file in the bucket.
#
# Returns:
#       0 - If successful.
#       1 - If it fails.
###############################################################################
function copy_file_to_bucket() {
  local response bucket_name source_file destination_file_name
  bucket_name=$1
  source_file=$2
  destination_file_name=$3

  response=$(aws s3api put-object \
    --bucket "$bucket_name" \
    --body "$source_file" \
    --key "$destination_file_name")

  # shellcheck disable=SC2181
  if [[ ${?} -ne 0 ]]; then
    errecho "ERROR: AWS reports put-object operation failed.\n$response"
    return 1
  fi
}
```

- Per i dettagli sull'API, consulta PutObject AWS CLICommand Reference.

Scenari

Nozioni di base su S3

L'esempio di codice seguente mostra come:

- Creare un bucket S3 con configurazione di denominazione e Regione univoca
- Configurare le impostazioni di sicurezza del bucket, incluso il blocco dell'accesso pubblico
- Abilitare il controllo delle versioni e la crittografia predefinita per la protezione dei dati
- Caricare oggetti con e senza metadati personalizzati
- Scaricare oggetti dal bucket all'archiviazione locale
- Copiare gli oggetti all'interno del bucket per organizzare i dati in cartelle
- Elencare i contenuti e gli oggetti del bucket con prefissi specifici
- Aggiungere tag ai bucket per la gestione delle risorse
- Eseguire la pulizia di tutte le risorse, inclusi gli oggetti con controllo delle versioni

AWS CLI con lo script Bash

> **Note**
>
> C'è altro da fare. GitHub Trova l'esempio completo e scopri come eseguire la configurazione e l'esecuzione nel repository dei tutorial sugli esempi di codice per gli sviluppatori.

```
#!/bin/bash

# Amazon S3 Getting Started Tutorial Script
# This script demonstrates basic S3 operations including:
# - Creating a bucket
# - Configuring bucket settings
# - Uploading, downloading, and copying objects
# - Deleting objects and buckets

# Latest fixes:
# 1. Fixed folder creation using temporary file
# 2. Corrected versioned object deletion in cleanup
# 3. Improved error handling for cleanup operations
```

```
# Set up error handling
set -e
trap 'cleanup_handler $?' EXIT

# Log file setup
LOG_FILE="s3-tutorial-$(date +%Y%m%d-%H%M%S).log"
exec > >(tee -a "$LOG_FILE") 2>&1

# Function to log messages
log() {
    echo "[$(date +"%Y-%m-%d %H:%M:%S")] $1"
}

# Function to handle errors
handle_error() {
    log "ERROR: $1"
    exit 1
}

# Function to check if a bucket exists
bucket_exists() {
    if aws s3api head-bucket --bucket "$1" 2>/dev/null; then
        return 0
    else
        return 1
    fi
}

# Function to delete all versions of objects in a bucket
delete_all_versions() {
    local bucket=$1
    log "Deleting all object versions from bucket $bucket..."

    # Get and delete all versions
    versions=$(aws s3api list-object-versions --bucket "$bucket" --query
 'Versions[].{Key:Key,VersionId:VersionId}' --output json 2>/dev/null)
    if [ -n "$versions" ] && [ "$versions" != "null" ]; then
        echo "{\"Objects\": $versions}" | aws s3api delete-objects --bucket
 "$bucket" --delete file:///dev/stdin >/dev/null 2>&1 || log "Warning: Some versions
 could not be deleted"
    fi

    # Get and delete all delete markers
```

```
    markers=$(aws s3api list-object-versions --bucket "$bucket" --query
 'DeleteMarkers[].{Key:Key,VersionId:VersionId}' --output json 2>/dev/null)
    if [ -n "$markers" ] && [ "$markers" != "null" ]; then
        echo "{\"Objects\": $markers}" | aws s3api delete-objects --bucket "$bucket"
 --delete file:///dev/stdin >/dev/null 2>&1 || log "Warning: Some delete markers
 could not be deleted"
    fi
}

# Function to handle cleanup on exit
cleanup_handler() {
    local exit_code=$1

    # Only run cleanup if it hasn't been run already
    if [ -z "$CLEANUP_DONE" ]; then
        cleanup
    fi

    exit $exit_code
}

# Function to clean up resources
cleanup() {
    log "Starting cleanup process..."
    CLEANUP_DONE=1

    # List all resources created for confirmation
    log "Resources created:"
    if [ -n "$BUCKET_NAME" ]; then
        log "- S3 Bucket: $BUCKET_NAME"

        # Only try to list objects if the bucket exists
        if bucket_exists "$BUCKET_NAME"; then
            # Check if any objects were created
            OBJECTS=$(aws s3api list-objects-v2 --bucket "$BUCKET_NAME" --query
 'Contents[].Key' --output text 2>/dev/null || echo "")
            if [ -n "$OBJECTS" ]; then
                log "- Objects in bucket:"
                echo "$OBJECTS" | tr '\t' '\n' | while read -r obj; do
                    log "  - $obj"
                done
            fi

            # Ask for confirmation before cleanup
```

```
            read -p "Do you want to proceed with cleanup and delete all resources?
 (y/n): " confirm
            if [[ $confirm != [yY] && $confirm != [yY][eE][sS] ]]; then
                log "Cleanup aborted by user."
                return
            fi

            # Delete all versions of objects
            delete_all_versions "$BUCKET_NAME"

            # Delete the bucket
            log "Deleting bucket $BUCKET_NAME..."
            aws s3api delete-bucket --bucket "$BUCKET_NAME" || log "Warning: Failed
 to delete bucket"
        else
            log "Bucket $BUCKET_NAME does not exist, skipping cleanup"
        fi
    fi

    # Clean up local files
    log "Removing local files..."
    rm -f sample-file.txt sample-document.txt downloaded-sample-file.txt empty-
file.tmp

    log "Cleanup completed."
}

# Generate a random bucket name
generate_bucket_name() {
    local hex_id
    hex_id=$(openssl rand -hex 6)
    echo "demo-s3-bucket-$hex_id"
}

# Main script execution
main() {
    log "Starting Amazon S3 Getting Started Tutorial"

    # Generate a unique bucket name
    BUCKET_NAME=$(generate_bucket_name)
    log "Generated bucket name: $BUCKET_NAME"

    # Step 1: Create a bucket
    log "Step 1: Creating S3 bucket..."
```

```
   # Get the current region or default to us-east-1
   REGION=$(aws configure get region)
   REGION=${REGION:-us-east-1}
   log "Using region: $REGION"

   if [ "$REGION" = "us-east-1" ]; then
       aws s3api create-bucket --bucket "$BUCKET_NAME" || handle_error "Failed to
create bucket"
   else
       aws s3api create-bucket \
           --bucket "$BUCKET_NAME" \
           --region "$REGION" \
           --create-bucket-configuration LocationConstraint="$REGION" ||
handle_error "Failed to create bucket"
   fi
   log "Bucket created successfully"

   # Configure bucket settings
   log "Configuring bucket settings..."

   # Block public access (security best practice)
   log "Blocking public access..."
   aws s3api put-public-access-block \
       --bucket "$BUCKET_NAME" \
       --public-access-block-configuration
"BlockPublicAcls=true,IgnorePublicAcls=true,BlockPublicPolicy=true,RestrictPublicBuckets=tr
|| handle_error "Failed to configure public access block"

   # Enable versioning
   log "Enabling versioning..."
   aws s3api put-bucket-versioning \
       --bucket "$BUCKET_NAME" \
       --versioning-configuration Status=Enabled || handle_error "Failed to enable
versioning"

   # Set default encryption
   log "Setting default encryption..."
   aws s3api put-bucket-encryption \
       --bucket "$BUCKET_NAME" \
       --server-side-encryption-configuration '{"Rules":
[{"ApplyServerSideEncryptionByDefault": {"SSEAlgorithm": "AES256"}}]}' ||
handle_error "Failed to set encryption"
```

```
    # Step 2: Upload an object
    log "Step 2: Uploading objects to bucket..."

    # Create a sample file
    echo "This is a sample file for the S3 tutorial." > sample-file.txt

    # Upload the file
    aws s3api put-object \
        --bucket "$BUCKET_NAME" \
        --key "sample-file.txt" \
        --body "sample-file.txt" || handle_error "Failed to upload object"
    log "Object uploaded successfully"

    # Upload with metadata
    echo "This is a document with metadata." > sample-document.txt
    aws s3api put-object \
        --bucket "$BUCKET_NAME" \
        --key "documents/sample-document.txt" \
        --body "sample-document.txt" \
        --content-type "text/plain" \
        --metadata "author=AWSDocumentation,purpose=tutorial" || handle_error
"Failed to upload object with metadata"
    log "Object with metadata uploaded successfully"

    # Step 3: Download an object
    log "Step 3: Downloading object from bucket..."
    aws s3api get-object \
        --bucket "$BUCKET_NAME" \
        --key "sample-file.txt" \
        "downloaded-sample-file.txt" || handle_error "Failed to download object"
    log "Object downloaded successfully"

    # Check if an object exists
    log "Checking if object exists..."
    aws s3api head-object \
        --bucket "$BUCKET_NAME" \
        --key "sample-file.txt" || handle_error "Object does not exist"
    log "Object exists"

    # Step 4: Copy object to a folder
    log "Step 4: Copying object to a folder..."

    # Create a folder structure using a temporary empty file
    log "Creating folder structure..."
```

```
    touch empty-file.tmp
    aws s3api put-object \
        --bucket "$BUCKET_NAME" \
        --key "favorite-files/" \
        --body empty-file.tmp || handle_error "Failed to create folder"

    # Copy the object
    log "Copying object..."
    aws s3api copy-object \
        --bucket "$BUCKET_NAME" \
        --copy-source "$BUCKET_NAME/sample-file.txt" \
        --key "favorite-files/sample-file.txt" || handle_error "Failed to copy
 object"
    log "Object copied successfully"

    # List objects in the bucket
    log "Listing all objects in the bucket..."
    aws s3api list-objects-v2 \
        --bucket "$BUCKET_NAME" \
        --query 'Contents[].Key' \
        --output table || handle_error "Failed to list objects"

    # List objects with a specific prefix
    log "Listing objects in the favorite-files folder..."
    aws s3api list-objects-v2 \
        --bucket "$BUCKET_NAME" \
        --prefix "favorite-files/" \
        --query 'Contents[].Key' \
        --output table || handle_error "Failed to list objects with prefix"

    # Add tags to the bucket
    log "Adding tags to the bucket..."
    aws s3api put-bucket-tagging \
        --bucket "$BUCKET_NAME" \
        --tagging 'TagSet=[{Key=Project,Value=S3Tutorial},
{Key=Environment,Value=Demo}]' || handle_error "Failed to add tags"
    log "Tags added successfully"

    log "Tutorial completed successfully!"
}

# Execute the main function
main
```

- Per informazioni dettagliate sull'API, consulta i seguenti argomenti nella documentazione di riferimento dei comandi della AWS CLI .
  - CopyObject
  - CreateBucket
  - DeleteBucket
  - DeleteObjects
  - GetObject
  - HeadObject
  - ListObjectVersions
  - ListObjectsV2
  - PutBucketEncryption
  - PutBucketTagging
  - PutBucketVersioning
  - PutObject
  - PutPublicAccessBlock

AWS STS esempi di utilizzo AWS CLI con lo script Bash

I seguenti esempi di codice mostrano come eseguire azioni e implementare scenari comuni utilizzando lo script AWS Command Line Interface with Bash con. AWS STS

Le azioni sono estratti di codice da programmi più grandi e devono essere eseguite nel contesto. Sebbene le azioni mostrino come richiamare le singole funzioni del servizio, è possibile visualizzarle contestualizzate negli scenari correlati.

Ogni esempio include un link al codice sorgente completo, in cui vengono fornite le istruzioni su come configurare ed eseguire il codice nel contesto.

Argomenti

- Azioni

Azioni

`AssumeRole`

Il seguente esempio di codice mostra come usare`AssumeRole`.

AWS CLI con lo script Bash

> **Note**
>
> C'è altro da fare. GitHub Trova l'esempio completo e scopri di più sulla configurazione e l'esecuzione nel Repository di esempi di codice AWS.

```
##############################################################################
# function iecho
#
# This function enables the script to display the specified text only if
# the global variable $VERBOSE is set to true.
##############################################################################
function iecho() {
  if [[ $VERBOSE == true ]]; then
    echo "$@"
  fi
}

##############################################################################
# function errecho
#
# This function outputs everything sent to it to STDERR (standard error output).
##############################################################################
function errecho() {
  printf "%s\n" "$*" 1>&2
}

##############################################################################
# function sts_assume_role
#
# This function assumes a role in the AWS account and returns the temporary
#  credentials.
#
# Parameters:
#       -n role_session_name -- The name of the session.
```

```
#       -r role_arn -- The ARN of the role to assume.
#
# Returns:
#       [access_key_id, secret_access_key, session_token]
#     And:
#       0 - If successful.
#       1 - If an error occurred.
###############################################################################
function sts_assume_role() {
  local role_session_name role_arn response
  local option OPTARG # Required to use getopts command in a function.

  # bashsupport disable=BP5008
  function usage() {
    echo "function sts_assume_role"
    echo "Assumes a role in the AWS account and returns the temporary credentials:"
    echo "  -n role_session_name -- The name of the session."
    echo "  -r role_arn -- The ARN of the role to assume."
    echo ""
  }

  while getopts n:r:h option; do
    case "${option}" in
      n) role_session_name=${OPTARG} ;;
      r) role_arn=${OPTARG} ;;
      h)
        usage
        return 0
        ;;
      \?)
        echo "Invalid parameter"
        usage
        return 1
        ;;
    esac
  done

  response=$(aws sts assume-role \
    --role-session-name "$role_session_name" \
    --role-arn "$role_arn" \
    --output text \
    --query "Credentials.[AccessKeyId, SecretAccessKey, SessionToken]")

  local error_code=${?}
```

```
  if [[ $error_code -ne 0 ]]; then
    aws_cli_error_log $error_code
    errecho "ERROR: AWS reports create-role operation failed.\n$response"
    return 1
  fi

  echo "$response"

  return 0
}
```

- Per i dettagli sull'API, consulta AssumeRole AWS CLICommand Reference.

Sicurezza in AWS CLI

Per AWS, la sicurezza del cloud ha la massima priorità. In quanto cliente AWS, è possibile trarre vantaggio da un'architettura di data center e di rete progettata per soddisfare i requisiti delle organizzazioni più esigenti a livello di sicurezza.

La sicurezza è una responsabilità condivisa tra te e AWS. Il modello di responsabilità condivisa descrive questo approccio come sicurezza del cloud e sicurezza nel cloud:

- La sicurezza del cloud: AWSè responsabile della protezione dell'infrastruttura che esegue i servizi AWSnel cloud AWS. AWSfornisce inoltre servizi che puoi utilizzare in sicurezza. Revisori di terze parti testano regolarmente e verificano l'efficacia della nostra sicurezza nell'ambito dei Programmi di conformità AWS. Per informazioni sui programmi di conformità applicabili a AWS Command Line Interface, consulta Servizi AWScoperti dal programma di conformità.
- Sicurezza nel cloud: la tua responsabilità è determinata dal servizio AWSche utilizzi. Sei anche responsabile di altri fattori, tra cui la riservatezza dei dati, i requisiti della tua azienda e le leggi e normative vigenti.

Questa documentazione aiuta a comprendere come applicare il modello di responsabilità condivisa quando si usa AWS Command Line Interface (AWS CLI). I seguenti argomenti illustrano come configurare AWS CLI per soddisfare gli obiettivi di sicurezza e conformità. Viene illustrato anche come utilizzare l'AWS CLI per aiutare a monitorare e proteggere le risorse AWS.

Argomenti

- Protezione dei dati in AWS CLI
- Identity and Access Management
- Convalida della conformità per questo AWS prodotto o servizio
- Resilienza per questo AWS prodotto o servizio
- Sicurezza dell'infrastruttura per questo AWS prodotto o servizio
- Applicazione di una versione minima di TLS 1.2 per AWS CLI

Protezione dei dati in AWS CLI

Il modello di responsabilità condivisa di AWSsi applica alla protezione dei dati in AWS Command Line Interface. Come descritto in questo modello, AWSè responsabile della protezione dell'infrastruttura

globale che esegue tutto l'Cloud AWS. L'utente è responsabile del controllo dei contenuti ospitati su questa infrastruttura. L'utente è inoltre responsabile della configurazione della protezione e delle attività di gestione per i Servizi AWS utilizzati. Per ulteriori informazioni sulla privacy dei dati, vedi le Domande frequenti sulla privacy dei dati. Per informazioni sulla protezione dei dati in Europa, consulta il post del blog relativo al Modello di responsabilità condivisa AWSe GDPR nel Blog sulla sicurezza AWS.

Per garantire la protezione dei dati, ti suggeriamo di proteggere le credenziali Account AWSe di configurare i singoli utenti con AWS IAM Identity Centero AWS Identity and Access Management(IAM). In tal modo, a ogni utente verranno assegnate solo le autorizzazioni necessarie per svolgere i suoi compiti. Ti suggeriamo, inoltre, di proteggere i dati nei seguenti modi:

- Utilizza l'autenticazione a più fattori (MFA) con ogni account.
- Utilizza SSL/TLS per comunicare con le risorse AWS. È richiesto TLS 1.2 ed è consigliato TLS 1.3.
- Configura l'API e la registrazione delle attività degli utenti con AWS CloudTrail. Per informazioni sull'utilizzo dei percorsi CloudTrail per acquisire le attività AWS, consulta Utilizzo dei percorsi CloudTrail nella Guida per l'utente di AWS CloudTrail.
- Utilizza le soluzioni di crittografia AWS, insieme a tutti i controlli di sicurezza di default all'interno dei Servizi AWS.
- Utilizza i servizi di sicurezza gestiti avanzati, come Amazon Macie, che aiutano a individuare e proteggere i dati sensibili archiviati in Amazon S3.
- Se necessiti di moduli crittografici convalidati FIPS 140-3 quando accedi ad AWS attraverso un'interfaccia a riga di comando o un'API, utilizza un endpoint FIPS. Per ulteriori informazioni sugli endpoint FIPS disponibili, consulta il Federal Information Processing Standard (FIPS) 140-3.

Ti consigliamo di non inserire mai informazioni riservate o sensibili, ad esempio gli indirizzi e-mail dei clienti, nei tag o nei campi di testo in formato libero, ad esempio nel campo Nome. Questo vale quando si lavora con l'AWS CLI e altri Servizi AWS utilizzando la console, l'API, la AWS CLI o gli SDK di AWS. I dati inseriti nei tag o nei campi di testo in formato libero utilizzati per i nomi possono essere utilizzati per i la fatturazione o i log di diagnostica. Quando fornisci un URL a un server esterno, ti suggeriamo vivamente di non includere informazioni sulle credenziali nell'URL per convalidare la tua richiesta al server.

Crittografia dei dati

Una caratteristica fondamentale di qualsiasi servizio sicuro è che le informazioni vengano crittografate quando non sono utilizzate attivamente.

Crittografia a riposo

AWS CLI non memorizza dati del cliente diversi dalle credenziali di cui ha bisogno per interagire con i servizi AWS per conto dell'utente.

Se utilizzi AWS CLI per richiamare un servizio AWS che trasmette i dati dei clienti al computer locale per l'archiviazione, fai riferimento al capitolo "Sicurezza e conformità" della Guida per l'utente di tale servizio per informazioni su come tali dati vengono archiviati, protetti e crittografati.

Crittografia in transito

Per impostazione predefinita, tutti i dati trasmessi dal computer client che esegue gli endpoint di AWS CLI e del servizio AWS vengono crittografati inviando ogni elemento tramite una connessione HTTPS/TLS.

Non devi fare nulla per abilitare l'uso di HTTPS/TLS. È sempre abilitato a meno che non si disattiva esplicitamente per un singolo comando utilizzando l'opzione della riga di comando `--no-verify-ssl`.

Identity and Access Management

AWS Identity and Access Management (IAM) è un software Servizio AWS che aiuta un amministratore a controllare in modo sicuro l'accesso alle AWS risorse. Gli amministratori IAM controllano chi può essere autenticato (effettuato l'accesso) e autorizzato (disporre delle autorizzazioni) a utilizzare le risorse. AWS IAM è uno Servizio AWS strumento che puoi utilizzare senza costi aggiuntivi.

Argomenti

- Destinatari
- Autenticazione con identità
- Gestione dell'accesso tramite policy
- Come Servizi AWS lavorare con IAM
- Risoluzione dei problemi di AWS identità e accesso

Destinatari

Il modo in cui usi AWS Identity and Access Management (IAM) varia a seconda del lavoro che AWS svolgi.

Utente del servizio: se lo utilizzi Servizi AWS per svolgere il tuo lavoro, l'amministratore ti fornisce le credenziali e le autorizzazioni necessarie. Man mano che utilizzi più AWS funzionalità per svolgere il tuo lavoro, potresti aver bisogno di autorizzazioni aggiuntive. La comprensione della gestione dell'accesso consente di richiedere le autorizzazioni corrette all'amministratore. Se non riesci ad accedere a una funzionalità di AWS, consulta Risoluzione dei problemi di AWS identità e accesso o consulta la guida per l'utente della funzionalità Servizio AWS che stai utilizzando.

Amministratore del servizio: se sei responsabile delle AWS risorse della tua azienda, probabilmente hai pieno accesso a AWS. È tuo compito determinare a quali AWS funzionalità e risorse devono accedere gli utenti del servizio. Devi quindi inviare le richieste all'amministratore IAM per modificare le autorizzazioni degli utenti del servizio. Esamina le informazioni contenute in questa pagina per comprendere i concetti di base relativi a IAM. Per saperne di più su come la tua azienda può utilizzare IAM con AWS, consulta la guida per l'utente del Servizio AWS software che stai utilizzando.

Amministratore IAM: un amministratore IAM potrebbe essere interessato a ottenere dei dettagli su come scrivere policy per gestire l'accesso a AWS. Per visualizzare esempi di policy AWS basate sull'identità che puoi utilizzare in IAM, consulta la guida per l'utente di quella Servizio AWS che stai utilizzando.

Autenticazione con identità

L'autenticazione è il modo in cui accedi AWS utilizzando le tue credenziali di identità. Devi autenticarti come utente IAM o assumendo un ruolo IAM. Utente root dell'account AWS

Puoi accedere come identità federata utilizzando credenziali provenienti da una fonte di identità come AWS IAM Identity Center (IAM Identity Center), autenticazione Single Sign-On o credenziali. Google/ Facebook Per ulteriori informazioni sull'accesso, consulta Come accedere all' Account AWS nella Guida per l'utente di Accedi ad AWS .

Per l'accesso programmatico, AWS fornisce un SDK e una CLI per firmare crittograficamente le richieste. Per ulteriori informazioni, consulta AWS Signature Version 4 per le richieste API nella Guida per l'utente di IAM.

Account AWS utente root

Quando si crea un Account AWS, si inizia con un'identità di accesso denominata utente Account AWS root che ha accesso completo a tutte Servizi AWS le risorse. Consigliamo vivamente di non utilizzare l'utente root per le attività quotidiane. Per le attività che richiedono le credenziali dell'utente root, consulta Attività che richiedono le credenziali dell'utente root nella Guida per l'utente IAM.

Identità federata

Come procedura ottimale, richiedi agli utenti umani di utilizzare la federazione con un provider di identità per accedere Servizi AWS utilizzando credenziali temporanee.

Un'identità federata è un utente della directory aziendale, del provider di identità Web o Directory Service che accede Servizi AWS utilizzando le credenziali di una fonte di identità. Le identità federate assumono ruoli che forniscono credenziali temporanee.

Per la gestione centralizzata degli accessi, si consiglia di utilizzare AWS IAM Identity Center. Per ulteriori informazioni, consulta Che cos'è il Centro identità IAM? nella Guida per l'utente di AWS IAM Identity Center .

Utenti e gruppi IAM

Un utente IAM è una identità che dispone di autorizzazioni specifiche per una singola persona o applicazione. Ti consigliamo di utilizzare credenziali temporanee invece di utenti IAM con credenziali a lungo termine. Per ulteriori informazioni, consulta Richiedere agli utenti umani di utilizzare la federazione con un provider di identità per accedere AWS utilizzando credenziali temporanee nella Guida per l'utente IAM.

Un gruppo IAM specifica una raccolta di utenti IAM e semplifica la gestione delle autorizzazioni per gestire gruppi di utenti di grandi dimensioni. Per ulteriori informazioni, consulta Casi d'uso per utenti IAM nella Guida per l'utente di IAM.

Ruoli IAM

Un ruolo IAM è un'identità con autorizzazioni specifiche che fornisce credenziali temporanee. Puoi assumere un ruolo passando da un ruolo utente a un ruolo IAM (console) o chiamando un'operazione AWS CLI o AWS API. Per ulteriori informazioni, consulta Metodi per assumere un ruolo nella Guida per l'utente di IAM.

I ruoli IAM sono utili per l'accesso federato degli utenti, le autorizzazioni utente IAM temporanee, l'accesso tra account, l'accesso tra servizi e le applicazioni in esecuzione su Amazon. EC2 Per maggiori informazioni, consultare Accesso a risorse multi-account in IAM nella Guida per l'utente IAM.

Gestione dell'accesso tramite policy

Puoi controllare l'accesso AWS creando policy e collegandole a identità o risorse. AWS Una policy definisce le autorizzazioni quando è associata a un'identità o a una risorsa. AWS valuta queste

politiche quando un preside effettua una richiesta. La maggior parte delle politiche viene archiviata AWS come documenti JSON. Per maggiori informazioni sui documenti delle policy JSON, consulta Panoramica delle policy JSON nella Guida per l'utente IAM.

Utilizzando le policy, gli amministratori specificano chi ha accesso a cosa definendo quale principale può eseguire azioni su quali risorse e in quali condizioni.

Per impostazione predefinita, utenti e ruoli non dispongono di autorizzazioni. Un amministratore IAM crea le policy IAM e le aggiunge ai ruoli, che gli utenti possono quindi assumere. Le policy IAM definiscono le autorizzazioni indipendentemente dal metodo utilizzato per eseguirle.

Policy basate sull'identità

Le policy basate su identità sono documenti di policy di autorizzazione JSON che è possibile collegare a un'identità (utente, gruppo o ruolo). Tali policy controllano le operazioni autorizzate per l'identità, nonché le risorse e le condizioni in cui possono essere eseguite. Per informazioni su come creare una policy basata su identità, consultare Definizione di autorizzazioni personalizzate IAM con policy gestite dal cliente nella Guida per l'utente IAM.

Le policy basate su identità possono essere policy in linea (con embedding direttamente in una singola identità) o policy gestite (policy autonome collegate a più identità). Per informazioni su come scegliere tra una policy gestita o una policy inline, consulta Scegliere tra policy gestite e policy in linea nella Guida per l'utente di IAM.

Policy basate sulle risorse

Le policy basate su risorse sono documenti di policy JSON che è possibile collegare a una risorsa. Gli esempi includono le policy di trust dei ruoli IAM e le policy dei bucket di Amazon S3. Nei servizi che supportano policy basate sulle risorse, gli amministratori dei servizi possono utilizzarli per controllare l'accesso a una risorsa specifica. In una policy basata sulle risorse è obbligatorio specificare un'entità principale.

Le policy basate sulle risorse sono policy inline che si trovano in tale servizio. Non è possibile utilizzare le policy AWS gestite di IAM in una policy basata sulle risorse.

Liste di controllo degli accessi () ACLs

Le liste di controllo degli accessi (ACLs) controllano quali principali (membri dell'account, utenti o ruoli) dispongono delle autorizzazioni per accedere a una risorsa. ACLs sono simili alle politiche basate sulle risorse, sebbene non utilizzino il formato del documento di policy JSON.

Amazon S3 e Amazon VPC sono esempi di servizi che supportano. AWS WAF ACLs Per ulteriori informazioni ACLs, consulta la panoramica della lista di controllo degli accessi (ACL) nella Amazon Simple Storage Service Developer Guide.

Altri tipi di policy

AWS supporta tipi di policy aggiuntivi che possono impostare le autorizzazioni massime concesse dai tipi di policy più comuni:

- Limiti delle autorizzazioni: imposta il numero massimo di autorizzazioni che una policy basata su identità ha la possibilità di concedere a un'entità IAM. Per ulteriori informazioni, consulta Limiti delle autorizzazioni per le entità IAM nella Guida per l'utente di IAM.
- Politiche di controllo del servizio (SCPs): specificano le autorizzazioni massime per un'organizzazione o un'unità organizzativa in. AWS Organizations Per ulteriori informazioni, consultare Policy di controllo dei servizi nella Guida per l'utente di AWS Organizations .
- Politiche di controllo delle risorse (RCPs): imposta le autorizzazioni massime disponibili per le risorse nei tuoi account. Per ulteriori informazioni, consulta Politiche di controllo delle risorse (RCPs) nella Guida per l'AWS Organizations utente.
- Policy di sessione: policy avanzate passate come parametro quando si crea una sessione temporanea per un ruolo o un utente federato. Per maggiori informazioni, consultare Policy di sessione nella Guida per l'utente IAM.

Più tipi di policy

Quando a una richiesta si applicano più tipi di policy, le autorizzazioni risultanti sono più complicate da comprendere. Per scoprire come si AWS determina se consentire o meno una richiesta quando sono coinvolti più tipi di policy, consulta Logica di valutazione delle policy nella IAM User Guide.

Come Servizi AWS lavorare con IAM

Per avere una visione di alto livello di come Servizi AWS funziona la maggior parte delle funzionalità IAM, consulta AWS i servizi che funzionano con IAM nella IAM User Guide.

Per scoprire come utilizzare uno specifico Servizio AWS con IAM, consulta la sezione sulla sicurezza della Guida per l'utente del servizio pertinente.

Risoluzione dei problemi di AWS identità e accesso

Utilizza le seguenti informazioni per aiutarti a diagnosticare e risolvere i problemi più comuni che potresti riscontrare quando lavori con un AWS IAM.

Argomenti

- Non sono autorizzato a eseguire alcuna azione in AWS
- Non sono autorizzato a eseguire iam: PassRole
- Voglio consentire a persone esterne a me di accedere Account AWS alle mie AWS risorse

Non sono autorizzato a eseguire alcuna azione in AWS

Se ricevi un errore che indica che non sei autorizzato a eseguire un'operazione, le tue policy devono essere aggiornate per poter eseguire l'operazione.

L'errore di esempio seguente si verifica quando l'utente IAM `mateojackson` prova a utilizzare la console per visualizzare i dettagli relativi a una risorsa *`my-example-widget`* fittizia ma non dispone di autorizzazioni `awes:`*`GetWidget`* fittizie.

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform:
 awes:GetWidget on resource: my-example-widget
```

In questo caso, la policy per l'utente `mateojackson` deve essere aggiornata per consentire l'accesso alla risorsa *`my-example-widget`* utilizzando l'azione `awes:`*`GetWidget`*.

Se hai bisogno di aiuto, contatta il tuo AWS amministratore. L'amministratore è la persona che ti ha fornito le credenziali di accesso.

Non sono autorizzato a eseguire iam: PassRole

Se ricevi un errore che indica che non sei autorizzato a eseguire l'operazione `iam:PassRole`, le tue policy devono essere aggiornate per poter passare un ruolo a AWS.

Alcuni Servizi AWS consentono di passare un ruolo esistente a quel servizio invece di creare un nuovo ruolo di servizio o un ruolo collegato al servizio. Per eseguire questa operazione, è necessario disporre delle autorizzazioni per trasmettere il ruolo al servizio.

L'errore di esempio seguente si verifica quando un utente IAM denominato `marymajor` cerca di utilizzare la console per eseguire un'operazione in AWS. Tuttavia, l'operazione richiede che il servizio

disponga delle autorizzazioni concesse da un ruolo di servizio. Mary non dispone delle autorizzazioni per trasmettere il ruolo al servizio.

```
User: arn:aws:iam::123456789012:user/marymajor is not authorized to perform:
 iam:PassRole
```

In questo caso, le policy di Mary devono essere aggiornate per poter eseguire l'operazione `iam:PassRole`.

Se hai bisogno di aiuto, contatta il tuo AWS amministratore. L'amministratore è la persona che ti ha fornito le credenziali di accesso.

Voglio consentire a persone esterne a me di accedere Account AWS alle mie AWS risorse

È possibile creare un ruolo con il quale utenti in altri account o persone esterne all'organizzazione possono accedere alle tue risorse. È possibile specificare chi è attendibile per l'assunzione del ruolo. Per i servizi che supportano politiche basate sulle risorse o liste di controllo degli accessi (ACLs), puoi utilizzare tali politiche per concedere alle persone l'accesso alle tue risorse.

Per maggiori informazioni, consulta gli argomenti seguenti:

- Per sapere se AWS supporta queste funzionalità, consulta. Come Servizi AWS lavorare con IAM
- Per scoprire come fornire l'accesso alle tue risorse attraverso Account AWS le risorse di tua proprietà, consulta Fornire l'accesso a un utente IAM in un altro Account AWS di tua proprietà nella IAM User Guide.
- Per scoprire come fornire l'accesso alle tue risorse a terze parti Account AWS, consulta Fornire l'accesso a soggetti Account AWS di proprietà di terze parti nella Guida per l'utente IAM.
- Per informazioni su come fornire l'accesso tramite la federazione delle identità, consulta Fornire l'accesso a utenti autenticati esternamente (federazione delle identità) nella Guida per l'utente IAM.
- Per informazioni sulle differenze di utilizzo tra ruoli e policy basate su risorse per l'accesso multi-account, consulta Accesso a risorse multi-account in IAM nella Guida per l'utente IAM.

Convalida della conformità per questo AWS prodotto o servizio

Per sapere se un Servizio AWS programma rientra nell'ambito di specifici programmi di conformità, consulta Servizi AWS la sezione Scope by Compliance Program Servizi AWS e scegli il programma

di conformità che ti interessa. Per informazioni generali, consulta Programmi di AWS conformità Programmi di di .

È possibile scaricare report di audit di terze parti utilizzando AWS Artifact. Per ulteriori informazioni, consulta Scaricamento dei report in AWS Artifact .

La vostra responsabilità di conformità durante l'utilizzo Servizi AWS è determinata dalla sensibilità dei dati, dagli obiettivi di conformità dell'azienda e dalle leggi e dai regolamenti applicabili. Per ulteriori informazioni sulla responsabilità di conformità durante l'utilizzo Servizi AWS, consulta la Documentazione AWS sulla sicurezza.

Questo AWS prodotto o servizio segue il modello di responsabilità condivisa attraverso i servizi specifici di Amazon Web Services (AWS) che supporta. Per informazioni sulla sicurezza dei AWS servizi, consulta la pagina della documentazione sulla sicurezza del AWS servizio e AWS i servizi che rientrano nell'ambito delle iniziative di AWS conformità previste dal programma di conformità.

Resilienza per questo AWS prodotto o servizio

L'infrastruttura AWS globale è costruita attorno a zone Regioni AWS di disponibilità.

Regioni AWS forniscono più zone di disponibilità fisicamente separate e isolate, collegate con reti a bassa latenza, ad alto throughput e altamente ridondanti.

Con le zone di disponibilità è possibile progettare e gestire applicazioni e database che eseguono automaticamente il failover tra zone di disponibilità senza interruzioni. Le zone di disponibilità sono più disponibili, tolleranti ai guasti e scalabili rispetto alle infrastrutture a data center singolo o multiplo tradizionali.

Per ulteriori informazioni su AWS regioni e zone di disponibilità, vedere Global Infrastructure.AWS

Questo AWS prodotto o servizio segue il modello di responsabilità condivisa attraverso i servizi specifici di Amazon Web Services (AWS) che supporta. Per informazioni sulla sicurezza dei AWS servizi, consulta la pagina della documentazione sulla sicurezza del AWS servizio e AWS i servizi che rientrano nell'ambito delle iniziative di AWS conformità previste dal programma di conformità.

Sicurezza dell'infrastruttura per questo AWS prodotto o servizio

Questo AWS prodotto o servizio utilizza servizi gestiti ed è pertanto protetto dalla sicurezza di rete AWS globale. Per informazioni sui servizi AWS di sicurezza e su come AWS protegge l'infrastruttura, consulta AWS Cloud Security. Per progettare il tuo AWS ambiente utilizzando le migliori pratiche per

la sicurezza dell'infrastruttura, vedi Infrastructure Protection in Security Pillar AWS Well-Architected Framework.

Utilizzate chiamate API AWS pubblicate per accedere a questo AWS Prodotto o Servizio attraverso la rete. I client devono supportare quanto segue:

- Transport Layer Security (TLS). È richiesto TLS 1.2 ed è consigliato TLS 1.3.
- Suite di cifratura con Perfect Forward Secrecy (PFS), ad esempio Ephemeral Diffie-Hellman (DHE) o Elliptic Curve Ephemeral Diffie-Hellman (ECDHE). La maggior parte dei sistemi moderni, come Java 7 e versioni successive, supporta tali modalità.

Inoltre, le richieste devono essere firmate utilizzando un ID chiave di accesso e una chiave di accesso segreta associata a un principale IAM. In alternativa è possibile utilizzare AWS Security Token Service (AWS STS) per generare credenziali di sicurezza temporanee per sottoscrivere le richieste.

Questo AWS prodotto o servizio segue il modello di responsabilità condivisa attraverso gli specifici servizi Amazon Web Services (AWS) che supporta. Per informazioni sulla sicurezza dei AWS servizi, consulta la pagina della documentazione sulla sicurezza del AWS servizio e AWS i servizi che rientrano nell'ambito delle iniziative di AWS conformità previste dal programma di conformità.

Applicazione di una versione minima di TLS 1.2 per AWS CLI

Quando si utilizza AWS Command Line Interface (AWS CLI), il protocollo Transport Layer Security (TLS) svolge un ruolo cruciale nel proteggere la comunicazione tra AWS CLI e Servizi AWS. Per aumentare la sicurezza durante la comunicazione con i servizi AWS, è necessario configurare TLS 1.2 o versioni successive.

La AWS CLI versione 2 utilizza uno script Python interno compilato per utilizzare una versione minima di TLS 1.2 quando il servizio con cui sta comunicando la supporta. Finché si utilizza la versione 2 della AWS CLI, non sono necessarie ulteriori fasi per applicare questo minimo. Per assicurarti di ottenere una maggiore sicurezza, accertati di eseguire l'aggiornamento a una versione recente della AWS CLI.

La AWS CLI e un Servizio AWS possono scambiare dati in modo sicuro, con il protocollo TLS che fornisce crittografia, autenticazione e integrità dei dati. Sfruttando il protocollo TLS, la AWS CLI garantisce che le interazioni con i Servizi AWS siano protette da accessi non autorizzati e violazioni dei dati, migliorando la sicurezza generale del tuo ecosistema AWS.

Il modello di responsabilità condivisa di AWSsi applica alla protezione dei dati in AWS Command Line Interface. Come descritto in questo modello, AWSè responsabile della protezione dell'infrastruttura globale che esegue tutto l'Servizi AWS. L'utente è responsabile del controllo dei contenuti ospitati su questa infrastruttura. L'utente è inoltre responsabile della configurazione della protezione e delle attività di gestione per i Servizi AWS utilizzati. Per ulteriori informazioni sulla protezione dei dati, consulta the section called "Protezione dei dati".

Guida alla migrazione per la AWS CLI versione 2

Questa sezione contiene le istruzioni per aggiornare la AWS CLI versione 1 alla AWS CLI versione 2. La AWS CLI versione 2 si basa sulla AWS CLI versione 1 e include funzionalità e miglioramenti basati sul feedback della community. La AWS CLI versione 2 è la versione principale più recente di AWS CLI e supporta tutte le funzionalità più recenti. Alcune funzionalità introdotte nella versione 2 non sono compatibili con la versione 1 ed è necessario eseguire l'aggiornamento per accedere a tali funzionalità.

Per evitare problemi imprevisti, prima di migrare alla versione 2, scopri le differenze tra le versioni. La AWS CLI versione 2 include nuove funzionalità e modifiche che potrebbero richiedere l'aggiornamento degli script o dei comandi per garantire la compatibilità con le versioni precedenti.

AWS CLI le versioni 1 e 2 utilizzano lo stesso nome di `aws` comando. Se sono state installate entrambe le versioni, il computer utilizza la prima trovata nel percorso di ricerca. Ciò potrebbe far sì che il nome del `aws` comando richiami la vecchia AWS CLI versione, anche quando è installata quella nuova.

Per eseguire l'aggiornamento alla AWS CLI versione 2, segui una delle seguenti istruzioni:

- Se in precedenza hai installato AWS CLI la versione 1, segui le istruzioni riportate inthe section called "Istruzioni di migrazione".
- Se non avete precedentemente installato la AWS CLI versione 1, seguite le istruzioni riportate inNozioni di base.

Utilizzo degli strumenti di migrazione AWS CLI per mitigare le interruzioni

Esistono cambiamenti nel comportamento tra la versione AWS CLI 1 e la AWS CLI versione 2 che potrebbero richiedere l'aggiornamento degli script o dei comandi. L'aggiornamento alla versione AWS CLI 2 senza verificare che gli script esistenti siano migrati in modo sicuro rischia di riscontrare effetti imprevisti durante l'esecuzione degli script con AWS CLI versione 2, come apportare modifiche involontarie alle risorse degli account. AWS

Esistono due strumenti di proprietà che è possibile utilizzare per guidare la AWS migrazione. Indipendentemente dallo strumento utilizzato, si consiglia di fare riferimento Interruzione delle

modifiche tra AWS CLI la versione 1 e la AWS CLI versione 2 alla Nuove funzionalità e modifiche nella AWS CLI versione 2 guida.

Aggiorna la modalità di debug

La modalità di debug dell'aggiornamento è una funzionalità della AWS CLI versione 1 disponibile nelle `1.44.0` versioni e successive. Quando questa funzionalità è abilitata, ogni comando AWS CLI eseguito viene controllato per determinare se è probabile che abbia un comportamento diverso nella versione AWS CLI 2, supponendo che l'ambiente di esecuzione rimanga lo stesso. Se un comando eseguito ha un comportamento diverso nella versione AWS CLI 2, verrà visualizzato un avviso accanto all'output del comando con le azioni suggerite che possono essere intraprese per preservare il comportamento della versione 1 della AWS CLI nella versione AWS CLI 2.

Il motivo principale per cui si utilizza la modalità di debug di aggiornamento per guidare la migrazione è che offre un'elevata copertura per rilevare se i comandi subiranno un cambiamento di comportamento. 14 delle 15 modifiche principali descritte in Interruzione delle modifiche tra AWS CLI la versione 1 e la AWS CLI versione 2 possono essere rilevate con la modalità di debug di aggiornamento. Utilizzo della modalità di aggiornamento di debug per aggiornare la versione AWS CLI 1 alla versione AWS CLI 2 Per i limiti di questa funzionalità, consulta Limitazioni la sezione dedicata. D'altra parte, lo svantaggio principale dell'utilizzo della modalità di debug di aggiornamento è che il suo output è rilevante solo all'interno del suo ambiente di esecuzione; se le impostazioni di configurazione della AWS CLI cambiano nell'ambiente dopo la migrazione alla versione AWS CLI 2, potrebbero comunque verificarsi modifiche sostanziali e dovresti verificare nuovamente utilizzando la modalità di debug di aggiornamento nell'ambiente aggiornato.

AWS Strumento di migrazione CLI da v1 a v2

Lo strumento di migrazione AWS CLI v1-to-v2 è uno strumento autonomo che può essere utilizzato con Python 3.9 e versioni successive. Questo strumento è un linter statico che funziona indipendentemente dalla AWS CLI. Elenca gli script di bash contenenti comandi della versione 1 della AWS CLI che possono avere un comportamento diverso nella versione CLI 2. AWS Per la maggior parte dei comandi che rileva come suscettibili di avere un comportamento diverso nella versione AWS CLI 2, può aggiornare automaticamente il comando all'interno dello script per preservare il comportamento della AWS CLI versione 1 nella AWS versione CLI 2.

Il motivo principale per utilizzare lo strumento di migrazione da AWS CLI v1 a v2 per guidare la migrazione è che può aggiornare automaticamente gli script per preservare il comportamento della CLI versione 1 nella versione AWS CLI 2, risparmiando tempo e AWS fatica nell'effettuare gli aggiornamenti da soli. D'altra parte, lo svantaggio principale dell'utilizzo dello strumento di

migrazione è che offre una copertura inferiore per rilevare se i comandi subiranno un cambiamento di comportamento. Essendo uno strumento statico, non verifica le modifiche di comportamento determinate in fase di esecuzione.

Scelta tra la modalità di debug dell'aggiornamento e AWS lo strumento di migrazione CLI da v1 a v2

La tabella seguente evidenzia le principali differenze tra la modalità di debug dell'aggiornamento e lo strumento di migrazione AWS CLI da v1 a v2 e può essere utilizzata per aiutarti a scegliere lo strumento giusto per il tuo caso d'uso.

Aggiorna la modalità di debug	AWS Strumento di migrazione CLI da v1 a v2
Rileva 14 delle 15 modifiche sostanziali tra la versione AWS CLI 1 e la versione AWS CLI 2.	Rileva 7 delle 15 modifiche sostanziali tra la versione AWS CLI 1 e la versione AWS CLI 2.
Compatibile con qualsiasi programma terminale in grado di eseguire la AWS CLI.	Supporta solo script di linting bash.
Richiede la versione AWS CLI 1, versione `1.44.0` o superiore.	Funziona indipendentemente dalla AWS CLI; non richiede l'installazione della AWS CLI.
È necessario apportare correzioni manuali per preservare il comportamento della versione 1 della AWS CLI nella versione AWS CLI 2.	Corregge automaticamente la maggior parte dei comandi AWS CLI versione 1 che rileva come suscettibili di avere un comportamento diverso nella versione AWS CLI 2.
I rilevamenti e le correzioni suggerite sono rilevanti solo per l'ambiente di esecuzione (ad esempio la configurazione AWS CLI) in cui vengono eseguiti.	I rilevamenti e le correzioni suggerite sono indipendenti dall'ambiente.
Richiede l'esecuzione di ogni comando AWS CLI in fase di test per modificare il comportam ento.	Non richiede l'esecuzione di alcun AWS comando CLI. Deve essere eseguito solo una volta per script bash.
Ha esattamente 5 casi noti in cui produrrà rilevamenti di falsi positivi.	Presenta esattamente 3 casi noti in cui produrrà rilevamenti di falsi positivi.

La tabella seguente mostra quali Interruzione delle modifiche tra AWS CLI la versione 1 e la AWS CLI versione 2 sono supportati in ciascuno dei due strumenti di migrazione.

Cambiamento rivoluzionario	Rilevato dalla modalità di debug dell'aggiornamento	Rilevato dallo strumento di AWS migrazione CLI v1-to-v2
Aggiunta di una variabile di ambiente per impostare la codifica dei file di testo	Sì	No
Per impostazione predefini ta, i parametri binari vengono passati come stringhe con codifica Base64.	Sì	Sì
Gestione migliorata delle proprietà e dei tag dei file per copie in più parti in Amazon S3	Sì	Sì
Nessun recupero automatico di `http://` o per i parametri `https://` URLs	Sì	No
Pager utilizzato per impostazi one predefinita per tutto l'output	Sì	Sì
I valori di output del timestamp sono standardizzati sul formato ISO 8601	Sì	No
Migliore gestione delle CloudFormation distribuzioni che non comportano modifiche	Sì	Sì
Modifica del comportamento predefinito per l'endpoint	Sì	No

Cambiamento rivoluzionario	Rilevato dalla modalità di debug dell'aggiornamento	Rilevato dallo strumento di AWS migrazione CLI v1-to-v2
Amazon S3 regionale per la Regione `us-east-1`		
Comportamento predefinito modificato per gli endpoint regionali AWS STS	Sì	No
`ecr get-login` rimosso e sostituito con `ecr get-login-password`	Sì	Sì
AWS CLI il supporto della versione 2 per i plugin sta cambiando	Sì	No
Rimozione del supporto per gli alias nascosti	Sì	Sì
L'impostazione del file di configurazione `api_versions` non è supportata	Sì	No
AWS CLI la versione 2 utilizza solo Signature v4 per autenticare le richieste Amazon S3	Sì	No
AWS CLI la versione 2 è più coerente con i parametri di paging	Sì	Sì
AWS CLI la versione 2 fornisce codici di ritorno più coerenti per tutti i comandi	No	No

Argomenti aggiuntivi in questa sezione

- the section called "Nuove funzionalità e modifiche"
- the section called "Istruzioni di migrazione"
- the section called "Utilizzo della modalità di debug di aggiornamento"
- the section called "Utilizzo dello strumento di AWS migrazione CLI da v1 a v2"

Nuove funzionalità e modifiche nella AWS CLI versione 2

Questo argomento descrive le nuove funzionalità e i cambiamenti di comportamento tra la AWS CLI versione 1 e AWS CLI la versione 2. In seguito a queste modifiche, potrebbe essere necessario aggiornare gli script o i comandi per ottenere nella versione 2 un comportamento che era previsto nella versione 1.

Argomenti

- AWS CLI nuove funzionalità della versione 2
- Interruzione delle modifiche tra AWS CLI la versione 1 e la AWS CLI versione 2

AWS CLI nuove funzionalità della versione 2

La AWS CLI versione 2 è la versione principale più recente di AWS CLI e supporta tutte le funzionalità più recenti. Alcune funzionalità introdotte nella versione 2 non sono compatibili con la versione 1 ed è necessario eseguire l'aggiornamento per accedere a tali funzionalità. Le caratteristiche principali comprendono:

Non è necessario un interprete Python

> La AWS CLI versione 2 non richiede un'installazione separata di Python. Include una versione incorporata.

Procedure guidate

> È possibile utilizzare una procedura guidata con la AWS CLI versione 2. La procedura guidata illustra come costruire determinati comandi.

Autenticazione in Centro identità IAM

Se la tua organizzazione utilizza AWS IAM Identity Center (IAM Identity Center), i tuoi utenti possono accedere ad Active Directory, a una directory IAM Identity Center integrata o a un altro IdP connesso a IAM Identity Center. Quindi, vengono mappati su un ruolo AWS Identity and Access Management (IAM) che consente di eseguire AWS CLI comandi.

Prompt automatico

Se abilitata, la AWS CLI versione 2 può richiedere comandi, parametri e risorse quando si esegue un `aws` comando.

Esecuzione delle immagini ufficiali di Amazon ECR Public o Docker per la AWS CLI

L'immagine Docker ufficiale di AWS CLI fornisce isolamento, portabilità e sicurezza che supporta e mantiene AWS direttamente. In questo modo, puoi utilizzare la AWS CLI versione 2 in un ambiente basato su container senza dover gestire personalmente l'installazione.

Pager lato client

La AWS CLI versione 2 prevede l'uso di un programma pager lato client per l'output. Per impostazione predefinita, questa funzionalità è attivata e restituisce tutto l'output attraverso il programma pager predefinito del sistema operativo.

`aws configure import`

Importa le credenziali `.csv` generate dalla Console di gestione AWS. Viene importato un file `.csv` con il nome del profilo corrispondente al nome utente IAM.

`aws configure list-profiles`

Elenca i nomi di tutti i profili che hai configurato.

the section called "Formato di output dei flussi YAML"

Il formato `yaml` e `yaml-stream` sfrutta il formato YAML fornendo al contempo una visualizzazione più reattiva di set di dati di grandi dimensioni mediante lo streaming dei dati. È possibile iniziare a visualizzare e utilizzare i dati YAML prima del download dell'intera query.

Nuovi comandi **`ddb`** di alto livello per DynamoDB

La AWS CLI versione 2 include i comandi Amazon `ddb put` `ddb select`DynamoDB di alto livello e. Questi comandi forniscono un'interfaccia semplificata per inserire elementi nelle tabelle DynamoDB ed eseguire ricerche in una tabella o in un indice DynamoDB.

`aws logs tail`

La AWS CLI versione 2 include un `aws logs tail` comando personalizzato che traccia i log per un gruppo Amazon CloudWatch Logs. Per impostazione predefinita, il comando restituisce i log di tutti i flussi di CloudWatch Logs associati degli ultimi dieci minuti.

È stato aggiunto il supporto dei metadati per i comandi **`s3`** di alto livello

La AWS CLI versione 2 aggiunge il `--copy-props` parametro ai comandi di alto livello. `s3` Questo parametro consente di configurare metadati e tag aggiuntivi per Amazon Simple Storage Service (Amazon S3).

`AWS_REGION`

La AWS CLI versione 2 ha una variabile di ambiente AWS compatibile con SDK chiamata. `AWS_REGION` Questa variabile specifica a cui Regione AWS inviare le richieste. Sostituisce la variabile di ambiente `AWS_DEFAULT_REGION`, che è applicabile solo nella AWS CLI.

Interruzione delle modifiche tra AWS CLI la versione 1 e la AWS CLI versione 2

Questa sezione descrive tutte le modifiche di comportamento tra la AWS CLI versione 1 e la AWS CLI versione 2. In seguito a queste modifiche, potrebbe essere necessario aggiornare gli script o i comandi per ottenere nella versione 2 un comportamento che era previsto nella versione 1.

Argomenti

- Aggiunta di una variabile di ambiente per impostare la codifica dei file di testo
- Per impostazione predefinita, i parametri binari vengono passati come stringhe con codifica Base64.
- Gestione migliorata delle proprietà e dei tag dei file per copie in più parti in Amazon S3
- Nessun recupero automatico di http:// o per i parametri https:// URLs
- Pager utilizzato per impostazione predefinita per tutto l'output
- I valori di output del timestamp sono standardizzati sul formato ISO 8601
- Migliore gestione delle CloudFormation distribuzioni che non comportano modifiche
- Modifica del comportamento predefinito per l'endpoint Amazon S3 regionale per la Regione us-east-1
- Comportamento predefinito modificato per gli endpoint regionali AWS STS

- ecr get-login rimosso e sostituito con ecr get-login-password
- AWS CLI il supporto della versione 2 per i plugin sta cambiando
- Rimozione del supporto per gli alias nascosti
- L'impostazione del file di configurazione api_versions non è supportata
- AWS CLI la versione 2 utilizza solo Signature v4 per autenticare le richieste Amazon S3
- AWS CLI la versione 2 è più coerente con i parametri di paging
- AWS CLI la versione 2 fornisce codici di ritorno più coerenti per tutti i comandi

Aggiunta di una variabile di ambiente per impostare la codifica dei file di testo

Per impostazione predefinita, i file di testo per the section called "Blob" utilizzano la stessa codifica delle impostazioni locali installate. Poiché la AWS CLI versione 2 utilizza una versione incorporata di Python, le variabili di `PYTHONIOENCODING` ambiente `PYTHONUTF8` e non sono supportate. Per impostare la codifica per i file di testo in modo che sia diversa dalle impostazioni locali, utilizzare la variabile di ambiente `AWS_CLI_FILE_ENCODING`. L'esempio seguente imposta l'apertura AWS CLI di file di testo `UTF-8` in Windows.

```
AWS_CLI_FILE_ENCODING=UTF-8
```

Per ulteriori informazioni, consulta Configurazione delle variabili d'ambiente per AWS CLI.

Per impostazione predefinita, i parametri binari vengono passati come stringhe con codifica Base64.

In AWS CLI, alcuni comandi richiedevano stringhe di byte con codifica in base64, mentre altri richiedevano stringhe di byte con codifica UTF-8. Nella AWS CLI versione 1, il passaggio di dati tra due tipi di stringhe codificate richiedeva spesso un'elaborazione intermedia. La AWS CLI versione 2 rende la gestione dei parametri binari più coerente, il che aiuta a passare i valori da un comando all'altro in modo più affidabile.

Per impostazione predefinita, la AWS CLI versione 2 passa tutti i parametri binari di input e output binari come stringa `blobs` con codifica Base64 (binary large object). Per ulteriori informazioni, consulta the section called "Blob".

Per ripristinare il comportamento della AWS CLI versione 1, utilizzate la configurazione del file o il `cli_binary_format` parametro. `--cli-binary-format`

Gestione migliorata delle proprietà e dei tag dei file per copie in più parti in Amazon S3

Quando si utilizzano i comandi della AWS CLI versione 1 nello spazio dei `aws s3` nomi per copiare un file da una posizione del bucket S3 a un'altra e tale operazione utilizza la copia in più parti, nessuna proprietà del file dall'oggetto di origine viene copiata nell'oggetto di destinazione.

Per impostazione predefinita, i comandi corrispondenti nella AWS CLI versione 2 trasferiscono tutti i tag e alcune proprietà dalla copia di origine a quella di destinazione. Rispetto alla AWS CLI versione 1, ciò può comportare un maggior numero di chiamate AWS API all'endpoint Amazon S3. Per modificare il comportamento predefinito dei `s3` comandi nella AWS CLI versione 2, usa il `--copy-props` parametro.

Per ulteriori informazioni, consulta the section called "Proprietà e tag dei file nelle copie in più parti".

Nessun recupero automatico di **`http://`** o per i parametri **`https://`** URLs

La AWS CLI versione 2 non esegue un'`GET`operazione quando il valore di un parametro inizia con `http://` o `https://` e non utilizza il contenuto restituito come valore del parametro. Di conseguenza, l'opzione della riga di comando associata `cli_follow_urlparam` viene rimossa dalla AWS CLI versione 2.

Se è necessario recuperare un URL e passare il contenuto dell'URL come valore di un parametro, è consigliabile utilizzare `curl` o uno strumento simile per scaricare il contenuto dell'URL in un file locale. Quindi puoi utilizzare la sintassi `file://` per leggere il contenuto di quel file e utilizzarlo come valore del parametro.

Ad esempio, il comando seguente non tenta più di recuperare il contenuto della pagina trovata all'indirizzo `http://www.example.com` e passare tali contenuto come parametro. Invece, passa la stringa di testo letterale `https://example.com` come parametro.

```
$ aws ssm put-parameter \
    --value http://www.example.com \
    --name prod.microservice1.db.secret \
    --type String 2
```

Per recuperare e utilizzare il contenuto di un URL Web come parametro, puoi procedere nel modo seguente nella versione 2.

```
$ curl https://my.example.com/mypolicyfile.json -o mypolicyfile.json
```

```
$ aws iam put-role-policy \
    --policy-document file://./mypolicyfile.json \
    --role-name MyRole \
    --policy-name MyReadOnlyPolicy
```

Nell'esempio precedente, il parametro `-o` indica a `curl` di salvare il file nella cartella corrente con lo stesso nome del file di origine. Il secondo comando recupera il contenuto di tale file scaricato e lo passa come il valore di `--policy-document`.

Pager utilizzato per impostazione predefinita per tutto l'output

Per impostazione predefinita, la AWS CLI versione 2 restituisce tutto l'output tramite il programma pager predefinito del sistema operativo. Questo programma è `less` in Linux o macOS e `more` in Windows. Può essere utile per navigare in grandi quantità di output di un servizio visualizzando l'output una pagina alla volta.

È possibile configurare la AWS CLI versione 2 per utilizzare un programma di paging diverso o non utilizzarne affatto. Per ulteriori informazioni, consulta the section called "Pager lato client".

I valori di output del timestamp sono standardizzati sul formato ISO 8601

Per impostazione predefinita, la AWS CLI versione 2 restituisce tutti i valori di risposta del timestamp nel formato ISO 8601. Nella AWS CLI versione 1, i comandi restituivano i valori del timestamp nel formato restituito dalla risposta dell'API HTTP, che poteva variare da servizio a servizio.

Per visualizzare i timestamp nel formato restituito dalla risposta dell'API HTTP, aggiungi il valore `wire` nel file `config`. Per ulteriori informazioni, consulta `cli_timestamp_format`.

Migliore gestione delle CloudFormation distribuzioni che non comportano modifiche

Per impostazione predefinita, nella AWS CLI versione 1, se si distribuisce un CloudFormation modello che non comporta modifiche, AWS CLI restituisce un codice di errore non riuscito. Questo causa problemi se non lo consideri un errore e desideri che il tuo script continui. È possibile ovviare a questo problema nella AWS CLI versione 1 aggiungendo il flag`--no-fail-on-empty-changeset`, che restituisce`0`.

Poiché si tratta di un caso d'uso comune, la AWS CLI versione 2 restituisce per impostazione predefinita un codice di uscita corretto, vale a dire `0` quando non vi sono modifiche causate da una distribuzione e l'operazione restituisce un set di modifiche vuoto.

Per ripristinare il comportamento originale, aggiungi il flag `--fail-on-empty-changeset`.

Modifica del comportamento predefinito per l'endpoint Amazon S3 regionale per la Regione **`us-east-1`**

Quando si configura la AWS CLI versione 1 per utilizzare la `us-east-1` regione, AWS CLI utilizza l'`s3.amazonaws.com`endpoint globale ospitato fisicamente nella regione. `us-east-1` La AWS CLI versione 2 utilizza il vero endpoint regionale `s3.us-east-1.amazonaws.com` quando viene specificata tale regione. Per forzare la AWS CLI versione 2 a utilizzare l'endpoint globale, puoi impostare la Regione per un comando su. `aws-global`

Comportamento predefinito modificato per gli endpoint regionali AWS STS

Per impostazione predefinita, la AWS CLI versione 2 invia tutte le richieste API AWS Security Token Service (AWS STS) all'endpoint regionale per quanto attualmente configurato. Regione AWS

Per impostazione predefinita, qualsiasi versione precedente della `1.42.0` AWS CLI versione 1 invia AWS STS richieste all' AWS STS endpoint globale. È possibile controllare questo comportamento predefinito nella versione 1 utilizzando l'impostazione `sts_regional_endpoints`.

Tutte le versioni successive o successive `1.42.0` utilizzano l'endpoint regionale come impostazione predefinita. Se si esegue la migrazione alla AWS CLI versione 2 da queste versioni più recenti, questo comportamento rimane invariato.

`ecr get-login` rimosso e sostituito con **`ecr get-login-password`**

La AWS CLI versione 2 sostituisce il comando `aws ecr get-login` con il `aws ecr get-login-password` comando che migliora l'integrazione automatica con l'autenticazione dei contenitori.

Il comando `aws ecr get-login-password` riduce il rischio di esporre le credenziali nell'elenco dei processi, nella cronologia della shell o in altri file di log. Migliora anche la compatibilità con il comando `docker login` per migliorare l'automazione.

Il `aws ecr get-login-password` comando è disponibile nella AWS CLI versione 1.17.10 e successive e nella AWS CLI versione 2. Il `aws ecr get-login` comando precedente è ancora disponibile nella AWS CLI versione 1 per motivi di compatibilità con le versioni precedenti.

Con il comando `aws ecr get-login-password` è possibile sostituire il codice seguente che recupera una password.

```
$ (aws ecr get-login --no-include-email)
```

Per ridurre il rischio di esporre la password alla cronologia o ai log della shell, utilizza invece il seguente comando di esempio. In questo esempio, la password viene reindirizzata direttamente al comando `docker login`, dove viene assegnata al parametro password tramite l'opzione `--password-stdin`.

```
$ aws ecr get-login-password | docker login --username AWS --password-stdin MY-REGISTRY-URL
```

Per ulteriori informazioni, consulta `aws ecr get-login-password` nella Guida di riferimento alla versione 2 della AWS CLI .

AWS CLI il supporto della versione 2 per i plugin sta cambiando

Il supporto per i plugin nella AWS CLI versione 2 è completamente provvisorio e ha lo scopo di aiutare gli utenti a migrare dalla AWS CLI versione 1 fino al rilascio di un'interfaccia plugin stabile e aggiornata. Non ci sono garanzie che un particolare plug-in o l'interfaccia del AWS CLI plug-in siano supportati nelle future versioni della AWS CLI versione 2. Se ti affidi ai plugin, assicurati di collegarti a una versione particolare del plug-in AWS CLI e di testarne la funzionalità quando esegui l'aggiornamento.

Per abilitare il supporto plugin, crea una sezione `[plugins]` in `~/.aws/config`.

```
[plugins]
cli_legacy_plugin_path = <path-to-plugins>/python3.7/site-packages
<plugin-name> = <plugin-module>
```

Nella sezione `[plugins]`, definisci la variabile `cli_legacy_plugin_path` e imposta il suo valore sul percorso dei pacchetti del sito Python in cui si trova il modulo plugin. Puoi quindi configurare un plugin fornendo un nome per il plugin (`plugin-name`) e il nome file del modulo Python (`plugin-module`) che contiene il codice sorgente per il plugin. AWS CLI Carica ogni plugin importandone `plugin-module` e chiamando la sua funzione. `awscli_initialize`

Rimozione del supporto per gli alias nascosti

AWS CLI la versione 2 non supporta più i seguenti alias nascosti che erano supportati nella versione 1.

Nella tabella seguente, la prima colonna mostra il servizio, il comando e il parametro che funzionano in tutte le versioni, inclusa la AWS CLI versione 2. La seconda colonna mostra l'alias che non funziona più nella AWS CLI versione 2.

Servizio, comando e parametro che funzionano	Alias obsoleto
cognito-identity create-identity-pool open-id-connect-provider -arns	open-id-connect-provider-ar-ns
storagegateway describe-tapes tape-arns	tape-ar-ns
gateway di archiviazione. describe-tape-archives.tape-arns	tape-ar-ns
gateway di archiviazione. describe-vtl-devices. vtl-device-arns	vtl-device-ar-ns
gateway di archiviazione. describe-cached-iscsi-volumes.volume -arns	volume-ar-ns
gateway di archiviazione. describe-stored-iscsi-volumes.volume-arns	volume-ar-ns
route53domains.view-billing.start-time	rapida
dispiegare. create-deployment-group.ec2-tag-set	ec-2-tag-set
dispiegare. list-application-revisions.s3-bucket	s-3-bucket
dispiegare. list-application-revisions.prefisso a 3 chiavi	s-3-key-prefix
distribuire. update-deployment-group.ec2-tag-set	ec-2-tag-set
obiettivo. enable-mfa-device.codice di autenticazione1	authentication-code-1
io. enable-mfa-device.codice di autenticazione2	authentication-code-2
io. resync-mfa-device.codice di autenticazione1	authentication-code-1
io. resync-mfa-device.codice di autenticazione2	authentication-code-2
importare/esportare. get-shipping-label.via 1	street-1

Servizio, comando e parametro che funzionano	Alias obsoleto
importazione/esportazione. get-shipping-label.strada 2	street-2
importazione/esportazione. get-shipping-label.strada 3	street-3
lambda.publish-version.code-sha256	code-sha-256
vela leggera. import-key-pair. public-key-base64	public-key-base-64
opsworks.register-volume.ec2-volume-id	ec-2-volume-id

L'impostazione del file di configurazione **`api_versions`** non è supportata

La AWS CLI versione 2 non supporta la chiamata a versioni precedenti del AWS servizio APIs utilizzando l'impostazione del file `api_versions` di configurazione. Tutti AWS CLI i comandi ora richiamano la versione più recente del servizio APIs attualmente supportata dall'endpoint.

AWS CLI la versione 2 utilizza solo Signature v4 per autenticare le richieste Amazon S3

La AWS CLI versione 2 non supporta algoritmi di firma precedenti per autenticare crittograficamente le richieste di servizio inviate agli endpoint Amazon S3. Questo processo di firma avviene automaticamente a ogni richiesta Amazon S3 ed è supportato solo il processo di firma Signature Version 4. Non è possibile configurare la versione della firma. Tutti i bucket Amazon S3 predefiniti URLs ora utilizzano solo SigV4 e hanno una durata di scadenza massima di una settimana.

AWS CLI la versione 2 è più coerente con i parametri di paging

Nella AWS CLI versione 1, se si specificano i parametri di impaginazione sulla riga di comando, l'impaginazione automatica viene disattivata come previsto. Quando invece si specificano i parametri di paginazione utilizzando un file con il parametro `--cli-input-json`, la paginazione automatica non viene disattivata ed è possibile che venga generato un output imprevisto. La AWS CLI versione 2 disattiva l'impaginazione automatica indipendentemente dal modo in cui vengono forniti i parametri.

AWS CLI la versione 2 fornisce codici di ritorno più coerenti per tutti i comandi

La AWS CLI versione 2 è più coerente tra tutti i comandi e restituisce correttamente un codice di uscita appropriato rispetto alla AWS CLI versione 1. Sono anche stati aggiunti i codici di uscita 252, 253 e 254. Per ulteriori informazioni sui codici di uscita, consulta the section called "Codici restituiti".

Se dipendi dal modo in cui la AWS CLI versione 1 utilizza i valori del codice di ritorno, ti consigliamo di controllare i codici di uscita per assicurarti di ottenere i valori previsti.

Installazione della AWS CLI versione 2 dalla AWS CLI versione 1

Questo argomento fornisce istruzioni per la migrazione dalla AWS CLI versione 1 alla AWS CLI versione 2.

AWS CLI le versioni 1 e 2 utilizzano lo stesso nome di `aws` comando. Se sono state installate entrambe le versioni, il computer utilizza la prima trovata nel percorso di ricerca. Se in precedenza hai installato la AWS CLI versione 1, ti consigliamo di eseguire una delle seguenti operazioni per utilizzare la AWS CLI versione 2:

- Consigliata: disinstalla la AWS CLI versione 1 e utilizza solo AWS CLI la versione 2.
- Per avere entrambe le versioni installate, utilizzare la capacità del sistema operativo di creare un collegamento simbolico (symlink) o alias con un nome diverso per uno dei due comandi `aws`.

Per informazioni sulle modifiche sostanziali tra la versione 1 e la versione 2, consulta the section called "Nuove funzionalità e modifiche".

Sostituzione della versione 1 con la versione 2

Effettuare le seguenti operazioni per sostituire AWS CLI la versione 1 con AWS CLI la versione 2.

Per sostituire AWS CLI la versione 1 con AWS CLI la versione 2

1. Prepara tutti gli script esistenti per la migrazione confermando eventuali modifiche sostanziali tra la versione 1 e la versione 2 in the section called "Nuove funzionalità e modifiche". È possibile utilizzare la funzionalità della modalità di debug di aggiornamento per guidare la preparazione seguendo le istruzioni riportate inthe section called "Utilizzo della modalità di debug di aggiornamento".
2. Disinstallate la AWS CLI versione 1 seguendo le istruzioni di disinstallazione per il sistema operativo in Installazione, aggiornamento e disinstallazione della AWS CLI versione 1.

3. Verificate che AWS CLI sia stata completamente disinstallata utilizzando il seguente comando.

```
$ aws --version
```

   Completa una delle seguenti fasi in base all'output:

   - Nessuna versione restituita: la AWS CLI versione 1 è stata disinstallata correttamente e puoi procedere al passaggio successivo.
   - Viene restituita una versione: hai ancora un'installazione della AWS CLI versione 1. Per la risoluzione dei problemi, consulta the section called "Il comando `aws --version` restituisce una versione dopo la disinstallazione di AWS CLI". Esegui le procedure di risoluzione dei problemi fino a quando non viene ricevuto alcun output in uscita.

4. Installa la AWS CLI versione 2 seguendo le istruzioni di installazione appropriate per il tuo sistema operativo inInstallazione o aggiornamento dell'ultima versione di AWS CLI.

ide-by-sideInstallazione S

Per avere entrambe le versioni installate, utilizzare la capacità del sistema operativo di creare un collegamento simbolico (symlink) o alias con un nome diverso per uno dei due comandi `aws`.

1. Installa la AWS CLI versione 2 seguendo le istruzioni di installazione appropriate per il tuo sistema operativo inInstallazione o aggiornamento dell'ultima versione di AWS CLI.
2. Usa la capacità del tuo sistema operativo di creare un collegamento simbolico o un alias con un nome diverso per uno dei due `aws` comandi, ad esempio utilizzando *`aws2`* per la AWS CLI versione 2. Di seguito sono riportati alcuni esempi di collegamenti simbolici per la versione 2 AWS CLI . Sostituisci il file *`PATH`* con il tuo percorso di installazione.

   Linux and macOS

   Puoi utilizzare un link simbolico o un alias su Linux e macOS.

```
$ alias aws2='PATH'
```

   Windows command prompt

   `DOSKEY` in Windows.

```
C:\> doskey aws2=PATH
```

Utilizzo della modalità di aggiornamento di debug per aggiornare la versione AWS CLI 1 alla versione AWS CLI 2

Questo argomento descrive la modalità di debug dell'aggiornamento nella versione 1 della AWS CLI.

Consigliamo agli utenti della AWS CLI versione 1 di eseguire l'aggiornamento alla versione AWS CLI 2 per accedere a nuove funzionalità e prestazioni migliorate. Esistono cambiamenti nel comportamento tra la versione AWS CLI 1 e la versione AWS CLI 2 che potrebbero richiedere l'aggiornamento degli script o dei comandi per ottenere lo stesso comportamento. La modalità di debug di aggiornamento nella versione AWS CLI 1 emette avvisi quando si utilizzano funzionalità che hanno un comportamento diverso nella AWS versione CLI 2. Questa funzionalità migliora l'esperienza di aggiornamento rilevando automaticamente quali comandi della AWS CLI versione 1 devono essere modificati prima dell'aggiornamento alla versione CLI 2 AWS per evitare problemi imprevisti.

Per ulteriori dettagli, vedere in. Interruzione delle modifiche tra AWS CLI la versione 1 e la AWS CLI versione 2 Nuove funzionalità e modifiche nella AWS CLI versione 2

Prerequisiti

La funzionalità della modalità di debug di aggiornamento è stata introdotta nella versione AWS CLI. `1.44.0`

Utilizzando la versione AWS CLI 1`aws --version`, esegui e verifica che la versione AWS CLI sia o superiore. `1.44.0`

Se la versione è inferiore a`1.44.0`, consulta Installazione, aggiornamento e disinstallazione della AWS CLI.

Come funziona

Se abilitata, la modalità di debug dell'aggiornamento rileva l'utilizzo di funzionalità che sono state aggiornate con modifiche sostanziali nella versione 2 della AWS CLI. Se si utilizza un comando o una funzionalità elencata nel nostro Interruzione delle modifiche tra AWS CLI la versione 1 e la

AWS CLI versione 2 dopo l'aggiornamento alla versione AWS CLI 2, nell'output verrà visualizzato un avviso. Il rilevamento delle modifiche interrotte si basa sul comando utilizzato, sugli argomenti forniti, sull'ambiente di esecuzione (ad esempio variabili di ambiente, impostazioni di configurazione, ecc.) e, in alcuni casi, sul contenuto o sulla configurazione delle risorse nell' AWS account utilizzato.

Gli avvisi descrivono le azioni per prevenire problemi imprevisti di aggiornamento alla versione 2 della CLI AWS . Dopo aver apportato le modifiche consigliate da un messaggio di avviso, è possibile confermare che il comando è stato aggiornato correttamente eseguendo nuovamente il comando per verificare che l'avviso non venga più visualizzato. Gli avvisi che vengono risolti segnalano che non è più probabile che il comando subisca la modifica definitiva descritta durante l'aggiornamento alla versione CLI 2 AWS .

L'esempio seguente mostra l'aspetto di questi avvisi. Questo comando mostra un esempio di avviso. Tutto il testo di avviso inizia con "AWS CLI V2 UPGRADE WARNING», seguito dal messaggio di avviso specifico. In questo caso, viene emesso un avviso perché il comando si basa sulla AWS CLI per recuperare il contenuto dell'URL e utilizzare il contenuto come valore `--template-body` del parametro, una funzionalità rimossa nella versione AWS CLI 2.

```
$ aws cloudformation create-stack \
  --stack-name "stack012345" \
  --template-body "https://s3.amazonaws.com/amzn-s3-demo-bucket/template.json"

AWS CLI v2 UPGRADE WARNING: For input parameters that have a prefix of http:// or
https://, AWS CLI v2 will not automatically request the content of the URL for
the parameter, and the `cli_follow_urlparam` option has been removed. See
https://docs.aws.amazon.com/cli/latest/userguide/cliv2-migration-changes.html#cliv2-
migration-paramfile.
```

La seguente tabella di tutte le modifiche sostanziali e di come evitare di dover affrontare modifiche sostanziali nella versione AWS CLI 2. Le azioni correttive che risolvono gli avvisi in modalità debug sono in grassetto.

Cambiamento decisivo	Sulla v1, esegui la migrazione al comportamento v2	Sulla v2, mantieni il comportamento v1
Aggiunta di una variabile di ambiente per impostare la codifica dei file di testo	Annulla l'impostazione delle variabili di **`PYTHONUTF8`** ambiente. **`PYTHONIOENCODING`**	Imposta la variabile di **`AWS_CLI_FILE_ENCODING`** ambiente sulla codifica specificata in v1.

Cambiamento decisivo	Sulla v1, esegui la migrazione al comportamento v2	Sulla v2, mantieni il comportamento v1
Per impostazione predefinita, i parametri binari vengono passati come stringhe con codifica Base64.	Codifica il valore del parametro con base64. Se il parametro viene caricato da un file, codificate il contenuto del file con base64.	Imposta l'impostazione su. **`cli_binary_format raw-in-base64-out`**
Gestione migliorata delle proprietà e dei tag dei file per copie in più parti in Amazon S3	N/A. Impossibile migrare al comportamento della versione 2 nella versione 1.	Utilizzo del parametro `--copy-props none` .
Nessun recupero automatico di `http://` o per i parametri `https://` URLs	Usa curl (o strumenti alternativi) per scaricare il contenuto di in un file locale URLs . Quindi, carica il contenuto del file nel parametro con file://. In alternativa, configurate su false **`cli_follow_urlparam`** per specificare l'URL non elaborato come valore del parametro.	N/A. Impossibile mantenere il comportamento della versione 1 nella versione 2.
Pager utilizzato per impostazione predefinita per tutto l'output	N/A. Impossibile migrare al comportamento della versione 2 nella versione 1.	Imposta l'**`cli_pager`** impostazione o la **`AWS_PAGER`** variabile su una stringa vuota.
I valori di output del timestamp sono standardizzati sul formato ISO 8601	Imposta l'impostazione **`cli_timestamp`** _format su**`iso8601`**.	Imposta l'**`cli_timestamp_format`** impostazione su wire.
Migliore gestione delle CloudFormation distribuzioni che non comportano modifiche	Usa il **`--no-fail-on-empty-changeset`** parametro.	Utilizzo del parametro `--fail-on-empty-changeset` .

Cambiamento decisivo	Sulla v1, esegui la migrazione al comportamento v2	Sulla v2, mantieni il comportamento v1
Modifica del comportamento predefinito per l'endpoint Amazon S3 regionale per la Regione `us-east-1`	Imposta la variabile di **`AWS_ENDPOINT_URL_S3`** ambiente o l'opzione della riga di **`--endpoint-url`** comando sull'URL regionale us-east-1.	Usa l'opzione della **`--region aws-global`** riga di comando.
`ecr get-login` rimosso e sostituito con `ecr get-login-password`	(1.17.10 o successivo) Usa **`ecr get-login`** rimosso e sostituito con **`ecr get-login-password`** e reindirizza l'output a un comando docker.	N/A. Impossibile mantenere il comportamento della versione 1 nella versione 2.
AWS CLI il supporto della versione 2 per i plugin sta cambiando	N/A. Impossibile migrare al comportamento della versione 2 nella versione 1.	Configura cli_legacy_plugin_path nella sezione del file di configurazione. `[plugins]` Prova i plugin nella versione 2, blocca la versione 2 e testa i plugin ogni volta che esegui l'aggiornamento.
Rimozione del supporto per gli alias nascosti	Passa dall'uso dell'alias nascosto obsoleto a Rimozione del supporto per gli alias nascosti quello che funziona in tutte le versioni.	N/A. Impossibile mantenere il comportamento della versione 1 nella versione 2.
L'impostazione del file di configurazione `api_versions` non è supportata	Migra e testa l'utilizzo delle versioni API precedenti alla versione API più recente e rimuovi api_versions dalle impostazioni di configurazione.	N/A. Impossibile mantenere il comportamento della versione 1 nella versione 2.

Cambiamento decisivo	Sulla v1, esegui la migrazione al comportamento v2	Sulla v2, mantieni il comportamento v1
AWS CLI la versione 2 utilizza solo Signature v4 per autenticare le richieste Amazon S3	Specificare la versione della firma nella versione 4 (vedere Specificazione della versione della firma nella richiesta di autenticazione).	N/A. Impossibile mantenere il comportamento della versione 1 nella versione 2.
AWS CLI la versione 2 è più coerente con i parametri di paging	Sposta i parametri di impaginazione nel parametro JSON di input nel comando stesso.	Rimuovi i parametri di impaginazione dal parametro JSON di input.
AWS CLI la versione 2 fornisce codici di ritorno più coerenti per tutti i comandi	N/A. Impossibile migrare al comportamento della versione 2 nella versione 1.	N/A. Impossibile mantenere il comportamento della versione 1 nella versione 2.

Limitazioni

Consigliamo vivamente ai clienti di recensire il nostro. Interruzione delle modifiche tra AWS CLI la versione 1 e la AWS CLI versione 2

Rilevamento di Breaking Change non supportato

La funzionalità della modalità di debug di aggiornamento supporta tutte le modifiche sostanziali tranne quella. AWS CLI la versione 2 fornisce codici di ritorno più coerenti per tutti i comandi Questa modalità non può tenere conto del modo in cui utilizzi i codici di errore restituiti dalla AWS CLI downstream.

Rilevamento condizionale delle modifiche di interruzione

Il rilevamento di I valori di output del timestamp sono standardizzati sul formato ISO 8601 è l'unico caso in cui il rilevamento dipende dallo stato dell' AWS account e può essere soggetto a modifiche sostanziali se le risorse dell'account vengono aggiornate in un secondo momento. Se nella risposta API del servizio non è incluso alcun timestamp, non verrà effettuata alcuna rilevazione di questa modifica sostanziale.

Se ti affidi al formato timestamp restituito dai comandi AWS CLI e non hai già configurato la AWS CLI per utilizzare ISO 8601, prendi ulteriori precauzioni per assicurarti che l'elaborazione del timestamp non si interrompa dopo l'aggiornamento alla versione 2.

Rilevamento di modifiche irrisolvibili

Alcuni avvisi emessi dalla modalità di debug dell'aggiornamento non possono essere risolti modificando il comando o l'ambiente. I casi seguenti sono casi in cui gli avvisi verrebbero sempre emessi dalla modalità di debug dell'aggiornamento purché si utilizzi la funzionalità corrispondente:

- AWS CLI il supporto della versione 2 per i plugin sta cambiando- se ci si affida ai plugin nel file di configurazione, la modalità di debug dell'aggiornamento emetterà sempre un avviso che non può essere risolto. Questa modalità non può garantire che uno dei tuoi plugin funzioni nella versione AWS CLI 2.
- Gestione migliorata delle proprietà e dei tag dei file per copie in più parti in Amazon S3- se `aws s3` viene utilizzato per eseguire una copia di bucket-to-bucket Amazon Amazon S3, la modalità di debug dell'upgrade emetterà sempre un avviso che non può essere risolto.

Falsi rilevamenti

Gli avvisi emessi dalla modalità di debug dell'aggiornamento non garantiscono che vengano apportate modifiche sostanziali dopo l'aggiornamento a AWS CLI v2. Di seguito sono riportati i casi in cui gli avvisi verrebbero emessi dalla modalità di debug di aggiornamento, nonostante non sia stata introdotta alcuna modifica sostanziale in CLI v2 AWS :

- Aggiunta di una variabile di ambiente per impostare la codifica dei file di testo- se le variabili `PYTHONUTF8` o di `PYTHONIOENCODING` ambiente sono specificate per impostare la codifica del file di testo e la codifica specificata corrisponde già alla locale installata, l'avviso potrebbe essere un falso rilevamento poiché la modalità di debug non verifica se la codifica corrisponde alla locale installata.
- Gestione migliorata delle proprietà e dei tag dei file per copie in più parti in Amazon S3- se `aws s3` viene utilizzato per eseguire una copia di bucket-to-bucket Amazon Amazon S3 e la copia non è multiparte perché l'oggetto di origine è al di sotto della soglia multiparte, verrà emesso un falso rilevamento.
- I valori di output del timestamp sono standardizzati sul formato ISO 8601- se l'impostazione di configurazione cli_timestamp_format è impostata su wire (impostazione predefinita) e il servizio restituisce un timestamp in formato ISO 8601.

- Migliore gestione delle CloudFormation distribuzioni che non comportano modifiche- se il `--fail-on-empty-changeset` flag viene utilizzato con il `aws cloudformation deploy` comando e il changeset risultante è vuoto, verrà emesso un falso rilevamento. Inoltre, se il changeset non è vuoto e non viene utilizzato, `--no-fail-on-empty-changeset` verrà emesso un falso rilevamento.
- Modifica del comportamento predefinito per l'endpoint Amazon S3 regionale per la Regione `us-east-1`- se `aws s3` o `aws s3api` viene utilizzato per eseguire un'operazione Amazon S3 e la regione è configurata a livello regionale e la `s3.us_east_1_regional_endpoint` configurazione non è configurata a `us-east-1` livello regionale e l'endpoint Amazon S3 globale viene utilizzato per la richiesta a causa delle impostazioni di configurazione dell'endpoint, l'avviso potrebbe essere un falso rilevamento, poiché la modalità di debug non verifica le impostazioni degli endpoint configurati.

Configura la modalità di debug per l'aggiornamento

È possibile abilitare o disabilitare la modalità di debug dell'aggiornamento utilizzando i seguenti metodi, elencati in ordine di precedenza:

- Le opzioni della riga di comando abilitano o disabilitano la modalità di debug dell'aggiornamento per un singolo comando. Usa --v2-debug per usare la modalità di debug di aggiornamento.
- Le variabili di ambiente utilizzano la variabile _UPGRADE_DEBUG_MODE. AWS_CLI

Utilizzo dello strumento di migrazione AWS CLI v1-v2 per aggiornare la CLI versione 1 alla AWS CLI versione 2 AWS

Questo argomento descrive lo strumento di migrazione AWS CLI v1-v2.

Consigliamo agli utenti della AWS CLI versione 1 di eseguire l'aggiornamento alla versione AWS CLI 2 per accedere a nuove funzionalità e prestazioni migliorate. Esistono cambiamenti nel comportamento tra la versione AWS CLI 1 e la versione AWS CLI 2 che potrebbero richiedere l'aggiornamento degli script o dei comandi per ottenere lo stesso comportamento. Lo strumento di migrazione AWS CLI v1-v2 analizza gli script di bash contenenti i comandi della AWS CLI versione 1 e rileva l'utilizzo di funzionalità che sono state aggiornate con le modifiche più importanti nella versione CLI 2. AWS Inoltre, lo strumento può modificare automaticamente gli script per risolvere la maggior parte dei problemi rilevati. Questo strumento migliora l'esperienza di aggiornamento

rilevando e modificando automaticamente i comandi AWS CLI versione 1 negli script di bash per prevenire problemi imprevisti durante l'aggiornamento alla versione 2.

Rispetto alla modalità Upgrade Debug, lo strumento di migrazione AWS CLI v1-v2 è uno strumento autonomo e non richiede l'esecuzione di comandi CLI. AWS Per un confronto completo tra la modalità di debug dell'aggiornamento e lo strumento di migrazione AWS CLI da v1 a v2, vedere in. Utilizzo degli strumenti di migrazione AWS CLI per mitigare le interruzioni Guida alla migrazione per la AWS CLI versione 2

Per ulteriori dettagli, vedere in. Interruzione delle modifiche tra AWS CLI la versione 1 e la AWS CLI versione 2 Nuove funzionalità e modifiche nella AWS CLI versione 2

Come funziona

Lo strumento di migrazione AWS CLI v1-to-v2 è un pacchetto Python in grado di lintare script bash che utilizzano la versione 1 della CLI. AWS Essendo un linter statico, non dipende dalla versione 1 della AWS CLI che potresti aver installato sul tuo computer. Il linter viene richiamato tramite la riga di comando, dove il percorso del file locale allo script bash viene fornito come parametro.

Lo strumento di migrazione può anche produrre automaticamente uno script bash modificato che risolve la maggior parte dei risultati rilevati modificando i comandi AWS CLI versione 1 utilizzati nello script. Lo strumento di migrazione modifica i comandi in modo che siano compatibili con la AWS versione 2 e mantiene il comportamento della versione 1.

Per alcuni risultati, lo strumento di migrazione AWS CLI v1-v2 è in grado di rilevarli ma non offre una correzione automatica. In questi casi, lo strumento di migrazione li contrassegna come rilevamenti che richiedono una revisione manuale.

Consulta Limitazioni l'elenco completo delle modifiche più importanti e delle correzioni automatiche supportate dallo strumento di migrazione AWS CLI v1-v2.

Prerequisiti

Python

Per utilizzare questo strumento, è necessario che Python 3.9 o versione successiva sia installato.

Per verificare che sia installata una versione di Python corretta, esegui il seguente comando in un terminale e conferma che l'output mostri una versione di Python di almeno 3.9.

```
$ python3 --version
```

Se disponi di una versione precedente di Python, o non hai installato Python, puoi scaricare una versione compatibile dalla pagina ufficiale di download di Python.

pip

Oltre ad avere una versione compatibile di Python installata, devi `pip` averla installata.

Per verificare che `pip` sia installato, esegui il seguente comando:

```
$ python3 -m pip --version
```

Se l'hai `pip` installato, vedrai un output simile al seguente:

```
pip 25.0.1 from ~/.local/lib/python3.13/site-packages (python 3.13)
```

Se non l'hai `pip` installato, vedi Installa pip.

Installazione

Installa lo strumento di migrazione AWS CLI v1-to-v2 in un nuovo ambiente virtuale:

```
$ python3 -m venv .venv
$ source .venv/bin/activate
$ python3 -m pip install aws-cli-migrate
```

Utilizzo

Modalità dry-run (impostazione predefinita)

Con la modalità dry-run, puoi rilevare automaticamente i comandi AWS CLI versione 1 soggetti a modifiche sostanziali senza modificare lo script di input:

```
$ migrate-aws-cli --script upload_s3_files.sh
```

Modalità di correzione automatica

Con la modalità di correzione automatica, puoi rilevare e aggiornare automaticamente i comandi AWS CLI versione 1 nello script di input per mitigare le interruzioni dovute alle modifiche introdotte nella versione AWS CLI 2, ove possibile:

```
$ migrate-aws-cli --script upload_s3_files.sh --fix
```

Facoltativamente, è possibile fornire un percorso di output tramite il `--output` parametro per scrivere lo script aggiornato, anziché aggiornare lo script di input:

```
$ migrate-aws-cli --script upload_s3_files.sh --output upload_s3_files_v2.sh --fix
```

Modalità interattiva

Con la modalità interattiva, puoi rilevare automaticamente i comandi AWS CLI versione 1 soggetti a modifiche sostanziali. La maggior parte dei risultati mostrerà una soluzione suggerita per mitigare le interruzioni nella versione AWS CLI 2. È possibile esaminare le correzioni suggerite e decidere se applicarle. Facoltativamente, fornisci un percorso di output tramite il `--output` parametro per controllare dove scrivere lo script aggiornato:

```
$ migrate-aws-cli --script upload_s3_files.sh --interactive \
--output upload_s3_files_v2.sh
```

Il seguente frammento di output è un esempio di ricerca in modalità interattiva:

```
14 14#
15 15# aws s3 ls s3://mybucket
16 16#
17   #-aws s3 cp s3://amzn-demo-bucket s3://amzn-demo-bucket2 --recursive
   17#+aws s3 cp s3://amzn-demo-bucket s3://amzn-demo-bucket2 --recursive --copy-props
 none
18 18#
19 19# TEMPLATE_KEY="cloudformation/$(basename "$TEMPLATE_FILE")"
20 20#

examples/upload_s3_files.sh:17 [s3-copy] In AWS CLI v2, object properties will be
 copied
from the source in multipart copies between S3 buckets. If a copy is or becomes
 multipart
after upgrading to AWS CLI v2, extra API calls will be made. See
https://docs.aws.amazon.com/cli/latest/userguide/cliv2-migration-changes.html#cliv2-
migration-s3-copy-metadata.

Apply this fix? [y] yes, [n] no, [a] accept all of type, [r] reject all of type, [u]
 update all,
```

```
[s] save and exit, [q] quit:
```

La correzione suggerita viene mostrata in un formato simile a un Git diff. Lo strumento di migrazione suggerisce la rimozione delle righe che iniziano con a `-` e suggerisce l'aggiunta di righe che iniziano con a`+`. Nell'esempio precedente, il suggerimento può essere interpretato come l'aggiunta del `--copy-props none` parametro al comando AWS CLI versione 1 che esegue una copia di Amazon S3.

Per ogni correzione suggerita, puoi inserire uno dei seguenti controlli:

- Inserisci `y` per accettare la correzione suggerita.
- Invio `n` per rifiutare la correzione corrente.
- Invio `a` per accettare tutte le correzioni dello stesso tipo.
- Invio `r` per rifiutare tutte le correzioni dello stesso tipo.
- Inserisci `u` per accettare tutte le correzioni rimanenti.
- Entra `s` per salvare e uscire.
- Entra `q` per uscire senza salvare.

Alcuni risultati potrebbero essere contrassegnati per la revisione manuale senza suggerire una correzione. Dovresti esaminare questi risultati e verificare se le ultime modifiche citate hanno influito su di te. Se siete interessati da questi risultati, dovreste seguire le indicazioni specificate nella descrizione dei risultati per apportare le modifiche necessarie per evitare o prepararvi alle modifiche sostanziali introdotte nella versione AWS CLI 2.

Limitazioni

Lo strumento di migrazione AWS CLI da v1 a v2 attualmente non supporta tutte le modifiche introdotte con la versione AWS CLI 2 e presenta casi di falsi positivi in cui genera rilevamenti per i comandi anche se non si verificherebbero effettivamente modifiche sostanziali.

Consigliamo vivamente ai clienti di comprendere la pubblicazione in. Interruzione delle modifiche tra AWS CLI la versione 1 e la AWS CLI versione 2 Nuove funzionalità e modifiche nella AWS CLI versione 2

Analisi basata su testo

Lo strumento di migrazione analizza lo script senza eseguirlo. Ciò limita il modo in cui rileva i comandi AWS CLI per interrompere le modifiche. Lo strumento di migrazione può esaminare solo il testo di

un comando AWS CLI. Non è in grado di rilevare problemi che si presentano in fase di esecuzione, come la memorizzazione di parametri obsoleti in una variabile invece di passarli direttamente alla CLI. AWS

Rilevamento delle modifiche interrotte non supportato

L'entità del supporto per le modifiche sostanziali nello strumento di migrazione è riepilogata nella tabella seguente.

Cambiamento rivoluzionario	Rilevamento supportato	Correzione automatica supportata
Aggiunta di una variabile di ambiente per impostare la codifica dei file di testo	No	No
Per impostazione predefinita, i parametri binari vengono passati come stringhe con codifica Base64.	Sì	Sì
Gestione migliorata delle proprietà e dei tag dei file per copie in più parti in Amazon S3	Sì	Sì
Nessun recupero automatico di `http://` o per i parametri `https://` URLs	No	No
Pager utilizzato per impostazione predefinita per tutto l'output	Sì	Sì
I valori di output del timestamp sono standardizzati sul formato ISO 8601	No	No

Cambiamento rivoluzionario	Rilevamento supportato	Correzione automatica supportata
Migliore gestione delle CloudFormation distribuzioni che non comportano modifiche	Sì	Sì
Modifica del comportamento predefinito per l'endpoint Amazon S3 regionale per la Regione `us-east-1`	No	No
Comportamento predefinito modificato per gli endpoint regionali AWS STS	No	No
`ecr get-login` rimosso e sostituito con `ecr get-login-password`	Sì	No
AWS CLI il supporto della versione 2 per i plugin sta cambiando	No	No
Rimozione del supporto per gli alias nascosti	Sì	Sì
L'impostazione del file di configurazione `api_versions` non è supportata	No	No
AWS CLI la versione 2 utilizza solo Signature v4 per autenticare le richieste Amazon S3	No	No

Cambiamento rivoluzionario	Rilevamento supportato	Correzione automatica supportata
AWS CLI la versione 2 è più coerente con i parametri di paging	Sì	No
AWS CLI la versione 2 fornisce codici di ritorno più coerenti per tutti i comandi	No	No

Disinstallare la AWS CLI versione 2

In questo argomento viene descritto come installare AWS Command Line Interface versione 2 (AWS CLI versione 2).

Istruzioni per la disinstallazione della AWS CLI versione 2:

Linux

Per disinstallarla, segui lo stesso metodo usato per installare la AWS CLI.

Command line installer

Per disinstallare la AWS CLI versione 2, esegui i seguenti comandi.

1. Individuare il collegamento simbolico e i percorsi di installazione.

   - Usare il comando `which` per individuare il collegamento simbolico. Questo mostra il percorso utilizzato con il parametro `--bin-dir`.

     ```
     $ which aws
     /usr/local/bin/aws
     ```

   - Utilizzare il comando `ls` per individuare la directory a cui punta il collegamento simbolico. Questo fornisce il percorso utilizzato con il parametro `--install-dir`.

     ```
     $ ls -l /usr/local/bin/aws
     lrwxrwxrwx 1 ec2-user ec2-user 49 Oct 22 09:49 /usr/local/bin/aws -> /usr/
     local/aws-cli/v2/current/bin/aws
     ```

2. Eliminare i due collegamenti simbolici nella directory `--bin-dir`. Se l'utente dispone dell'autorizzazione di scrittura in queste directory, non è necessario utilizzare `sudo`.

   ```
   $ sudo rm /usr/local/bin/aws
   $ sudo rm /usr/local/bin/aws_completer
   ```

3. Eliminare la directory `--install-dir`. Se l'utente dispone dell'autorizzazione di scrittura in questa directory, non è necessario utilizzare `sudo`.

   ```
   $ sudo rm -rf /usr/local/aws-cli
   ```

4. (Facoltativo) Rimuovi le informazioni condivise relative all'SDK AWS e alle impostazioni della AWS CLI nella cartella `.aws`.

   > **Warning**
   >
   > Queste impostazioni di configurazione e credenziali sono condivise tra tutti gli AWS SDK e la AWS CLI. Se rimuovi questa cartella, nessun AWS SDK ancora presente sul sistema potrà accedervi.

   La posizione predefinita della cartella `.aws` varia a seconda della piattaforma. Per impostazione predefinita, la cartella si trova in *~/.aws/*. Se l'utente dispone dell'autorizzazione di scrittura in questa directory, non è necessario utilizzare `sudo`.

   ```
   $ sudo rm -rf ~/.aws/
   ```

Snap

Viene fornita una versione ufficiale AWS supportata di AWS CLI su `snap`. Se in passato hai utilizzato `snap` per installare la AWS CLI, procedi nel seguente modo:

1. Eseguire il comando `snap remove` seguente per AWS CLI.

   ```
   $ snap remove aws-cli --classic
   ```

   Se in precedenza hai utilizzato `sudo` per installare la AWS CLI, è necessario aggiungerlo al comando.

   ```
   $ sudo snap remove aws-cli --classic
   ```

2. Potrebbe essere necessario riavviare la finestra del prompt dei comandi o il computer per rimuovere tutti i file.
3. (Facoltativo) Rimuovi le informazioni condivise relative all'SDK AWS e alle impostazioni della AWS CLI nella cartella `.aws`.

> **Warning**
>
> Queste impostazioni di configurazione e credenziali sono condivise tra tutti gli AWS SDK e la AWS CLI. Se rimuovi questa cartella, nessun AWS SDK ancora presente sul sistema potrà accedervi.

La posizione predefinita della cartella `.aws` varia a seconda della piattaforma. Per impostazione predefinita, la cartella si trova in *~/.aws/*. Se disponi dell'autorizzazione di scrittura in questa directory, non è necessario utilizzare `sudo`.

```
$ sudo rm -r ~/.aws/
```

macOS

Per disinstallare la AWS CLI versione 2, esegui i seguenti comandi, sostituendo i percorsi utilizzati per l'installazione. I comandi di esempio utilizzano i percorsi di installazione predefiniti.

1. Trova la cartella contenente i collegamenti simbolici del programma principale e al programma di completamento.

   ```
   $ which aws
   /usr/local/bin/aws
   ```

2. Utilizzando queste informazioni, eseguire il comando seguente per trovare la cartella di installazione a cui puntano i collegamenti simbolici.

   ```
   $ ls -l /usr/local/bin/aws
   lrwxrwxrwx 1 ec2-user ec2-user 49 Oct 22 09:49 /usr/local/bin/aws -> /usr/local/
   aws-cli/aws
   ```

3. Eliminazione dei due collegamenti simbolici nella prima cartella. Se disponi dell'autorizzazione di scrittura in queste cartelle, non è necessario utilizzare `sudo`.

   ```
   $ sudo rm /usr/local/bin/aws
   $ sudo rm /usr/local/bin/aws_completer
   ```

4. Eliminazione della cartella di installazione principale. Utilizza `sudo` per ottenere l'accesso in scrittura alla cartella `/usr/local`.

```
$ sudo rm -rf /usr/local/aws-cli
```

5. (Facoltativo) Rimuovi le informazioni condivise relative all'SDK AWS e alle impostazioni della AWS CLI nella cartella `.aws`.

> **Warning**
>
> Queste impostazioni di configurazione e credenziali sono condivise tra tutti gli AWS SDK e la AWS CLI. Se rimuovi questa cartella, nessun AWS SDK ancora presente sul sistema potrà accedervi.

La posizione predefinita della cartella `.aws` varia a seconda della piattaforma. Per impostazione predefinita, la cartella si trova in *~/.aws/*. Se l'utente dispone dell'autorizzazione di scrittura in questa directory, non è necessario utilizzare `sudo`.

```
$ sudo rm -rf ~/.aws/
```

Windows

1. Aprire Programmi e caratteristiche eseguendo una delle seguenti operazioni:
   - Aprire il Pannello di controllo e scegliere Programmi e caratteristiche.
   - Aprire un prompt dei comandi e digitare quindi il comando seguente.

```
C:\> appwiz.cpl
```

2. Seleziona la voce denominata AWS Command Line Interface e quindi scegli Uninstall (Disinstalla) per avviare il programma di disinstallazione.
3. Confermare che si desidera disinstallare l'AWS CLI.
4. (Facoltativo) Rimuovi le informazioni condivise relative all'SDK AWS e alle impostazioni della AWS CLI nella cartella `.aws`.

> **Warning**
>
> Queste impostazioni di configurazione e credenziali sono condivise tra tutti gli AWS SDK e la AWS CLI. Se rimuovi questa cartella, nessun AWS SDK ancora presente sul sistema potrà accedervi.

La posizione predefinita della cartella `.aws` varia a seconda della piattaforma. Per impostazione predefinita, la cartella si trova in *`%UserProfile%\.aws`*.

```
$ rmdir %UserProfile%\.aws
```

Risoluzione degli errori di installazione e disinstallazione della AWS CLI

Se riscontri problemi dopo l'installazione o la disinstallazione della AWS CLI, consulta la procedura the section called "Risoluzione degli errori" per la risoluzione dei problemi. Per le procedure di risoluzione dei problemi più importanti, consulta the section called "Errori di comando non trovato", the section called "Il comando `aws --version` restituisce una versione diversa da quella installata" e the section called "Il comando "`aws --version`" restituisce una versione dopo la disinstallazione di AWS CLI".

Cronologia dei documenti della Guida per l'utente di AWS CLI

Nella seguente tabella sono descritte le aggiunte significative apportate alla guida per l'utente della AWS Command Line Interface, a partire da gennaio 2019. Per ricevere notifiche sugli aggiornamenti della documentazione, puoi sottoscrivere il feed RSS.

Modifica	Descrizione	Data
Aggiunta del supporto per gli endpoint basati sull'account alla AWS CLI versione 2	Gli endpoint basati sull'account che utilizzano l'ID del tuo account AWS sono ora supportati nella AWS CLI versione 2.	20 marzo 2025
Aggiunta del supporto per gli endpoint basati sull'account alla AWS CLI versione 1	Gli endpoint basati sull'account che utilizzano l'ID del tuo account AWS sono ora supportati nella AWS CLI versione 2.	24 febbraio 2025
Informazioni aggiornate su credenziali e autenticazione.	Istruzioni ed esempi aggiornati sulle credenziali e sui metodi di autenticazione. È incluso l'aggiornamento delle pagine introduttive e delle pagine di configurazione rilevanti. Per far fronte a questo aumento della documentazione, gli argomenti pertinenti sulle credenziali sono stati spostati nella nuova sezione Credenziali di autenticazione e accesso.	31 marzo 2023
Aggiunta della configurazione del provider di token con	Il nuovo processo per configurare la AWS CLI per	7 dicembre 2022

aggiornamento automatico dell'autenticazione per AWS IAM Identity Center	l'autenticazione degli utenti con AWS IAM Identity Center (Centro identità IAM) utilizza la configurazione del provider di token SSO, che può recuperare automaticamente i token di autenticazione aggiornati.	
Rilascio dell'immagine pubblica ufficiale di Amazon ECR per la AWS CLI versione 2	È stata rilasciata l'immagine pubblica ufficiale di Amazon ECR supportata per la AWS CLI versione 2 per Linux, macOS e Windows.	18 novembre 2022
Aggiornamento della guida per la migrazione dalla V1 alla V2 della AWS CLI	La guida alle ultime modifiche è stata ampliata in modo da includere le istruzioni per la migrazione dalla AWS CLI versione 1 alla AWS CLI versione 2. Include aggiornamenti alla pagina Risoluzione dei problemi per risolvere i problemi di installazione.	13 maggio 2022
Nuovo processo per creare un programma di installazione della AWS CLI dall'origine.	Nuovo processo per installare o eseguire l'aggiornamento dall'origine alla versione più recente della AWS CLI sui sistemi operativi supportati.	17 febbraio 2022

Contenuti per la AWS CLI V1 e V2 forniti in guide separate	Per chiarezza e semplicità, i contenuti relativi alla AWS CLI versione 1 e alla AWS CLI versione 2 sono ora forniti separatamente nelle rispettive guide. Per la AWS CLI versione 1, consulta la guida per l'utente di AWS CLI versione 1.	2 novembre 2021
Aggiunta di informazioni sull'alias della AWS CLI	Aggiunta di informazioni sull'alias della AWS CLI Gli alias sono scorciatoie che è possibile creare in AWS Command Line Interface (AWS CLI) per abbreviare i comandi o gli script utilizzati di frequente.	11 marzo 2021
Aggiornamento delle informazioni sull'output dei filtri	Le informazioni per i filtri sono stata aggiornate e spostate nella relativa pagina.	1° febbraio 2021
Aggiunta di informazioni per le procedure guidate	Sono state aggiunte informazioni sulla procedura guidata della AWS CLI versione 2.	20 novembre 2020
Aggiornamento del prompt automatico	Le informazioni relative al prompt automatico della AWS CLI versione 2 sono state aggiornate con le funzionalità correnti.	10 novembre 2020
Aggiunta di un esempio di script Amazon S3	È stato aggiunto un esempio di script del ciclo di vita di Amazon S3.	15 ottobre 2020

Aggiunta di un esempio di script Amazon EC2	È stato aggiunto un esempio di script di tipo di istanza Amazon EC2.	15 ottobre 2020
Aggiunta di informazioni sui nuovi tentativi	È stata aggiunta una pagina relativa alle funzionalità e al comportamento dei nuovi tentativi nella AWS CLI.	17 settembre 2020
Pagina di paginazione lato server e lato client	Le informazioni di paginazione sono state aggiornate e centralizzate su un'unica pagina.	17 agosto 2020
Aggiornamento della pagina dei comandi s3	La pagina dei comandi s3 di alto livello è stata aggiornata con nuovi esempi e risorse.	30 luglio 2020
Informazioni aggiornate sull'installazione	Le informazioni di installazione, aggiornamento e disinstallazione per Linux, macOS e Windows sono state aggiornate.	19 maggio 2020
Aggiunta di informazioni per la codifica dei file di testo nella AWS CLI versione 2	Per impostazione predefinita, la AWS CLI versione 2 utilizza la stessa codifica del file di testo locale. È ora possibile utilizzare le variabili di ambiente per impostare la codifica dei file di testo.	14 maggio 2020
Rilascio dell'immagine Docker ufficiale per la AWS CLI versione 2	L'immagine Docker di supporto ufficiale per la AWS CLI versione 2 è stata rilasciata per Linux, macOS e Windows.	31 marzo 2020

Aggiunta di informazioni relative ai pager lato client per la AWS CLI versione 2	Per impostazione predefinita, la AWS CLI versione 2 utilizza il programma pager `less` per tutti gli output lato client.	19 febbraio 2020
AWS Command Line Interface (AWS CLI) versione 2 è stata ufficialmente rilasciata	La AWS CLI versione 2 è disponibile al pubblico ed è la versione consigliata ai clienti.	10 febbraio 2020
Il programma di installazione macOS per la AWS CLI versione 2 è ora un file `.pkg` presente nel programma di installazione del pacchetto Apple.	Il programma di installazione macOS per la AWS CLI versione 2 è stato aggiornato con un file `.zip` che contiene uno script shell per un pacchetto completo del programma di installazione macOS. Ciò semplifica l'installazione e la rende compatibile con le versioni più recenti di macOS.	3 febbraio 2020
Aggiunta di contenuti per migliorare la gestione predefinita della AWS CLI versione 2 degli endpoint regionali S3 e STS	Per impostazione predefinita, la AWS CLI versione 2 ora indirizza le richieste per i servizi Amazon S3 e AWS STS all'endpoint regionale attualmente configurato anziché all'endpoint globale.	13 gennaio 2020
Rilascio in anteprima della AWS CLI versione 2 per sviluppatori	È stato annunciato il rilascio in anteprima della AWS CLI versione 2. Aggiunte istruzioni sull'installazione della versione 2. Aggiungi argomento di migrazione per discutere le differenze tra le versioni 1 e 2.	7 novembre 2019

Aggiunta del supporto per AWS IAM Identity Center ai profili denominati della AWS CLI	La AWS CLI versione 2 aggiunge il supporto per la creazione di un profilo denominato, che può accedere direttamente al Centro identità IAM e ottenere le credenziali temporanee AWS per l'utilizzo nei comandi AWS CLI successivi.	7 novembre 2019
Nuova sezione MFA	Aggiunta una nuova sezione che descrive come accedere alla CLI utilizzando i ruoli e l'autenticazione a più fattori.	3 maggio 2019
Aggiornamento della sezione "Utilizzo della CLI"	Miglioramenti e aggiunte importanti alle istruzioni e alle procedure di utilizzo.	7 marzo 2019
Aggiornamento della sezione "Installazione della CLI	Miglioramenti e aggiunte importanti alle istruzioni e alle procedure di installazione della AWS CLI.	7 marzo 2019
Aggiornamento della sezione "Configurazione della CLI	Miglioramenti e aggiunte importanti alle istruzioni e alle procedure di configurazione della AWS CLI.	7 marzo 2019